Acta Universitatis Sapientiae

Informatica

Volume 15, Number 2, 2023

Sapientia Hungarian University of Transylvania
Scientia Publishing House

Journal Metrics (2022)

Impact Factor: **0.3**
Five Year Impact Factor: **0.8**
JCI: **0.09**
MEiN: **20**
Google Scholar h5-index: **21**
Google Scholar h5-median: **28**

Acta Universitatis Sapientiae Informatica is covered by the following services:

ACM Digital Library
Baidu Scholar
Cabell's Journalytics
CNKI Scholar
CNPIEC – cnpLINKer Dimensions
DOAJ
EBSCO
Engineering Village
ExLibris
Google Scholar
Inspec
Japan Science and Technology Agency
J-Gate
JournalTOCs
KESLI-NDSL
MyScienceWork
Naver Academic
Naviga (Softweco)
QOAM
ReadCube
SCILIT
Semantic Scholar
Sherpa/RoMEO
TDNet
Ulrich's Periodicals Directory
WanFang Data
Web of Science – ESCII
WorldCat (OCLC)
zbMATH Open
X-MOL

Contents

Acta Univ. Sapientiae Informatica **15,** 2 (2023) 181–204

DOI: 10.2478/ausi-2023-0013

Formal modeling of multi-viewpoint ontology alignment by mappings composition

Manel KOLLI
Applied Mathematics Didactics Laboratory MAD,
Higher Normal School of Constantine,Algeria.
email: kollimanel@yahoo.fr

Abstract. We propose a formal approach based on Bigraphical Reactive Systems (BRS) to provide a formal modeling of multi-viewpoint ontology alignment by composition systems' structure using bigraphs their dynamic behaviors using bigraphical reaction rules. In the first phase of this approach, we address the modeling of the static structure the dynamic behavior of multi-viewpoint ontology alignment systems. We show how bigraphs enable the description of the different multi-view point ontology entities. Furthermore, we define a set of bigraphical reaction rules to model the dynamic nature of the alignment. We introduce composition strategies to describe multi-viewpoint ontology alignment systems' behaviors. Then, we present a case study on which we illustrate the application of our proposed approach. Finally, we combine the logical reflection of Maude language the hierarchical structure of the BRS to provide an executable formal model for multi-viewpoint ontology alignment by composition systems.

1 Introduction

During the last decades, several computing systems methods have been proposed to make the semantic interoperability a reality. The semantic web has

Key words and phrases: MVp ontology alignment, viewpoint, composition, formal methods, bigraphical reactive system

led to the deployment of ontologies on the web connected through various mechanisms, in particular, ontology alignments [12]. Ontology alignment is one of the well-known emerging methods which aim to allow the joint use of several ontologies. The result of this task ensures facilitates the exchange, sharing, merging of data information between systems or communities in the Semantic Web. Generally, it's about constructing matches between elements described in different ontologies. In the literature, several ontology alignment methods have been proposed. They take advantage of the different aspects of ontologies they are interested in the alignment of ontologies described in different ontological languages. Therefore, the majority of alignment methods only detect relationships between classical ontologies that do not take into account the notion of multiple viewpoints. In this work, we are interested in the problem of developing ontologies in a heterogeneous organisation by taking into account different viewpoints, different terminologies of people, groups even diverse communities within this organisation. Such ontology, called a multi-viewpoint ontology, allows both heterogeneity consensus to coexist in a heterogeneous organisation. Unlike a classical ontology, a multi-viewpoint ontology confers on the same universe of discourse several different representations such that each relates to a particular viewpoint [7]. This need to take into account multi-point of view knowledge within the same ontology, essentially results from a multidisciplinary environment where several diverse groups of people coexist collaborate with each other. Each group has its own particular interests differently perceives the particular properties relationships of conceptual entities in the same knowledge universe to be represented. The MVp ontologies can be used to represent the exchanged data by different actors in a given domain. However, in a large organisation, different MVp ontologies can coexist. Partially, they can model the knowledge of the same domain used by several heterogeneous communities. Indeed, the interoperability between heterogeneous MVp ontologies is necessary in many applications. The heterogeneity comes from the fact that these ontologies are generally built collaboratively independently of each other. Thus, like classical ontologies, the MVp ontologies have no reason to have a common unique formalism or vocabulary, this makes it difficult to share, reuse exchange the knowledge represented by the different communities. This problem generates the need of an alignment of MVp ontologies for the purpose of merging, integrating, reusing them in other applications or for the more ambitious reason of having a global MVp ontology. In our context, an MVp ontology is composed of a set of viewpoints, global concepts, local concepts, roles, local roles set of bridge links. Therefore, these ontologies have a particular specificity that the alignment process must

take into account. This particularity is the existence of bridge links, whose role is to represent the consensual links between local concepts from different viewpoints. They play a very crucial role in the construction of an MVp ontology. As a result, it is possible to exploit these different bridge links in an alignment process to identify new correspondences between multi-viewpoint ontologies. This identification can be done through the combination of the bridge links which are already existed can be realized by the composition operation of the alignments. According to [21], the alignment composition operation consists in deducing correspondences between two ontologies that are not yet aligned from a succession of alignments between these ontologies one or more intermediate ontologies.

One of the proposed solutions for specifying modelling these complex alignment systems is to use formal methods that offer unambiguous abstraction mechanisms, a rigour a precision in the specification of the structural behavioural aspects of these systems. In our work, we are interested in two formalisms for the modelling the realisation of multi-viewpoint ontology alignment systems. Namely: the bigraphic reactive systems the Maude formal language. The Bigraphic Reactive Systems (BRS) is a new formalism that is characterised by its graphical aspect its ability to represent both the locality the connectivity of ubiquitous distributed computing systems. Nevertheless, the tools developed around BRS are limited in terms of expressiveness performance. For this purpose, we also opted to use the language Maude language as the most suitable alternative. Maude is a functional language that enables realizable checkable specifications for a broad range of systems. The objective of this article is twofold. First, we clarify how we take on bigraphs to specify model both structural behavioral sides of multi-viewpoint ontology alignment by composition. Thus, an MVp ontology is treated as an ensemble of nodes links clustered in roots. multi-viewpoint ontology alignment by composition is established by different reaction rules. So, we can deduct that each MVp ontology component can have a specific semantic in the BRS formalism. Therefore, the designed bigraphs identify the graphical representation of a multi-viewpoint ontology alignment by composition, as well as its predicted mathematical patterns. Then, we demonstrate how we can use Maude language the hierarchical organization of the BRS to give an implementable model for multi-viewpoint ontology alignment system. The remainder of this article is organized as followings. The next section presents related work. Section 3 presents appropriate definitions for multi-viewpoint ontology its alignment. Section 4 provides a summary of BRS formalism gives a detailed presentation of the proposed approach to specify multi-viewpoint ontology alignment

by composition. In Section 5, the proposed approach is demonstrated by a case study. In section 6, we encode the bigraphical specifications into Maude language. Finally, in section 7, we conclude with the future directions of work.

2 Related work

The ontology alignment has been investigated in different research works like [1], [2], [8], [12], [17], [19] , [16]. While, there are a few works concerning formal ontology alignment in the state of the art. Also, all these works are not able to handle the concept of multi-Viewpoints. For instance, Reference [20] presents Alin, an interactive ontology matching approach which uses expert feedback not only to approve or reject selected mappings, but also to dynamically improve the set of selected mappings. This supplementary exploit for expert answers tries increasing in the benefit brought by each expert answer. To achieve this goal, Alin relies on four mechanisms. The two first mechanisms were used to dynamically choose concept attribute mappings. The Two other mechanisms are established dynamically to choose relationship mappings to refuse inconsistent chosen mappings by anti-patterns. In the process of ontology alignment, the idea of mapping composition is significant has been perfectly considered in [11]. This paper provides a novel technique of the ontology alignment process. Here, the deduction of the relations between the entities is made by aggregating or composing the relations among their subsumers, which are previously deducted according to the semantic distance. The results were validated through the description logics (DLs) techniques. The author in [5] introduces new formal ontology Networks. These later are realized by a set of logic theories, called ontologies, linked by alignments. It demonstrates how belief revision operators, constrained by the structure of networks of ontologies, may be defined. Next, it establishes two revision operators as well as associated consequences two notions of consistency. The authors in [9], proposed an extended semantics to handle separately alignment interpretation of Network on various stage. Where, the focus was on formalisms which consist of specifying reasoning about aligned ontologies. According to [19], “The advantage of the extended semantics lies in the fact that each alignment expressed between a source target ontology is independently treated, as each one possesses its own distinct vocabulary semantics”. In [18] authors introduced a novel solution, SubInterNM, focused on algebraic operations. These operations allow reducing the amount of comparisons necessary to match the networks following the System of systems, which are interconnected systems that bring value

Work	Formalism	Dynamicity & evolution
[20]	No	No
[11]	Description Logics (DLs)	No
[5]	Collection of logic theories,	No
[9]	Distributed Description Logics (DDLs)	No
[18]	Algebraic operations	No
Our approach	BRS	Yes

Table 1: The comparison of ontology alignment work

to different domains. They implemented the subsumed internetwork matching, which reduces the amount of pairs in order to be assessed in the alignment. But, neither of these approaches provide a standard complete way for modeling specifying the feature of dynamicity of ontology networks. In this field, the developers mainly concentrate on the syntactic semantic aspect, which is the most commonly adopted solution for the formalization of networks of ontologies.

Table1 recapitulates this part by the comparison of the proposed approach with the previously presented works. This comparison was made according to a set of criteria such as: the used formalism or formal model the provided dynamicity evolution in the modeling approach. So, we can say that our approach is the first work which considers the dynamicity of alignment systems.

3 Multi-viewpoint ontology alignments by composition

A multi-viewpoint ontology is a multiple description of the same discourse context according to various viewpoints. Where, a viewpoint is a partial description of a discourse context within a particular perception. At a global level, the partial descriptions share ontological elements semantic links constituting a consensus between the different viewpoints. Such links, called bridge links, establish the communications between the viewpoints represent the interdisciplinary collaboration. Indeed, the bridge links are semantic links between local concepts; they define how a local concept in one viewpoint is related to another local concept in a different viewpoint, represent accessibility relationships be-

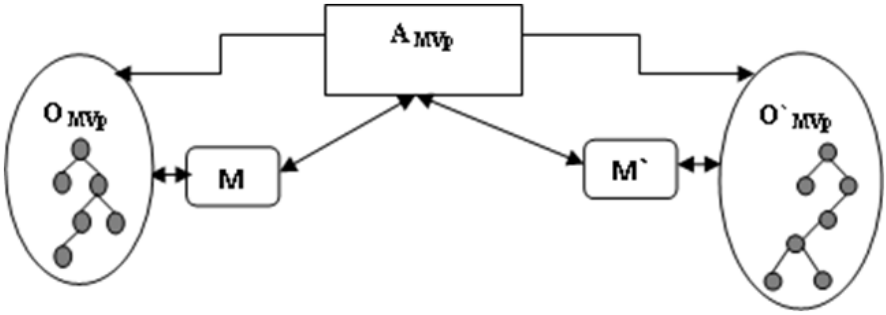


Figure 1: MVp Ontology alignment by composition

tween local concepts in two different viewpoints. These links are in the heart of the construction of an MVp ontology, as they allow the navigation between the different viewpoints. Given their importance, we consider to use them in a composition process to improve the set of identified correspondences. In our context, this operation takes as input the bridge links of two MVp ontologies, gives as output an alignment between these ontologies. Here, we base on MVp ontologies described in DLs, so our composition mechanism consists of using the properties of the relations involved in the different correspondences of the MVp ontologies those involved in the different bridge links. Formal definitions are given in what follows. The Definition 1 is according to [6].

Definition 1 *(MVp Ontology) An MVp ontology is defined as a four-tuple of the form OMVp = (CG, R, VP, M), where: CG is a set of global concepts, R is a set of roles, VP is a viewpoints set, M is a bridge links set. A viewpoint is given by a triple VP= (CG, CL, R), where: CL is a set of local concepts R is a set of roles. In what follows, we provide various essential definitions:*

- *Global concept represents a generic family of the real world. Each global concept can be expressed by different viewpoints.*
- *Local concept is expressed locally by a specific viewpoint.*
- *Role is a connection among two local concepts specified in two distinct viewpoints.*
- *Bridge links signify consensual relationships among two local concepts or roles specified in two varions viewpoints. Viewpoints are not totally disconnected. Bridges are semantic relationships which link local concepts under pairwise disjoint viewpoints to a local concept in another viewpoint.*

Definition 2 *(MVp Ontology alignment) Given two MVp ontologies OMVp O'MVp, the alignment of these two onttologies is defined as a 3-tuple:* $\mathsf{AMVp} = (\mathsf{OMVp}, \mathsf{O'MVp}, \Sigma\mathsf{M})$. *The product of* AMVp *is a set of concepts* $(\mathsf{C_i}, \mathsf{C_j})$ *related by a semantic relations Rel. So,* $\Sigma\mathsf{M}$ *is a set of mappings of concepts* $\mathsf{C_i}$ $\mathsf{C_j}(\mathsf{M}(\mathsf{C_i}, \mathsf{C_j}) = \mathsf{Rel})$. *The semantic relation Rel is belonging to the set:* $\{\equiv, \subseteq, \supseteq, \perp\}$.

Where: the equivalence relation is represented by $\equiv$

This later is represented as the subsumption in both directions $\{\subseteq, \supseteq\}$ *. Finally, the relation of distinction is represented by* $\perp$.

Definition 3 *(MVp Ontology alignment by composition) The process of MVp Ontology alignment by composition consists of connecting the ontologies via their Bridge links (see Figure 1). Indeed, an MVp Ontology alignment by composition A MVp is defined as a semantic composition between the sets of Bridge links* $\mathsf{M_1}$ $\mathsf{M_2}$ *of the ontologies* $\mathsf{OMVp_1}$ $\mathsf{OMVp_2}$ *respectively. The composition operation is defined as a function associating to a pair* $(\mathsf{M_1}, \mathsf{M_2})$ *an alignment A MVp such as A* $\mathsf{MVp} = \mathsf{M_1} \circ \mathsf{M_2}$ *.*

The MVp Ontology alignment is directional because the correspondences start from the first to the second MVp ontology the composition operation is neither commutative nor associative, due to the fact that there are several viewpoints in each MVp ontology to be aligned. Therefore, it is necessary to define the composition operation suggest a rules set to achieve the MVp ontology alignment by composition.

Definition 4 *(The composition of multi-viewpoint) The Semantic Web domain is based on the description logics for the MVp ontology creation. This implies that the concepts are structured by the subsumption relation the bridge links, which permits the inference of semantic relations between the concepts in a simple straight manner. The operation which allows this inference is called composition. The application of this composition is presented in Table 2.*

The proof of these results was done by the interpretation notion of DLs in [10].

$C_1 R C_2$ / $C_2 R C_3$	$\equiv$	$\subseteq$	$\perp$
$\models$	$\equiv$	$\subseteq$	$\perp$
$\subseteq (\supseteq)$	$\subseteq (\supseteq)$	$\subseteq (\supseteq)$	$\perp$
$\perp$	$\perp$	undecidable	undecidable

Table 2: Composition of the semantic relations

4 A BRS model for multi-viewpoint ontology alignment by composition

4.1 Bigraphical reactive systems overview

Bigraphical Reactive Systems (BRS) is a formalism designed to model the temporal spatial evolution of computing. The bigraph theory was recently introduced by Robin Milner, co-workers [13], [14] , [15] to provide an intuitive graphical model for representing the locality connectivity of systems. Therefore, it strongly appropriates with MVp ontology alignment concepts. A reactive bigraph system consists of a set of bigraphs representing the state of the system a set of reaction rules describing its evolution. The theory of bigraphs has two main objectives: (1) to be able to integrate in the same formalism the important aspects of the systems; (2) to provide a unification of existing theories by developing a general theory, which contains the different calculations for concurrency mobility.

Bigraph anatomy graphic form Let us consider the following example, depicted in Figure 2. In the graphical form of bigraphs, the entities components (real or virtual) of a system are expressed as nodes represented as ovals, circles, triangles other graphical shapes. The spatial location of nodes is described in terms of arbitrary nestings between different nodes in a given system. All nodes in a bigraph have an identifier (type), called a control designated by letters (e.g. A; B; etc.). The interactions between different nodes are represented by links, for example, the hyper-arc in Figure 2 connecting node A node C. Each

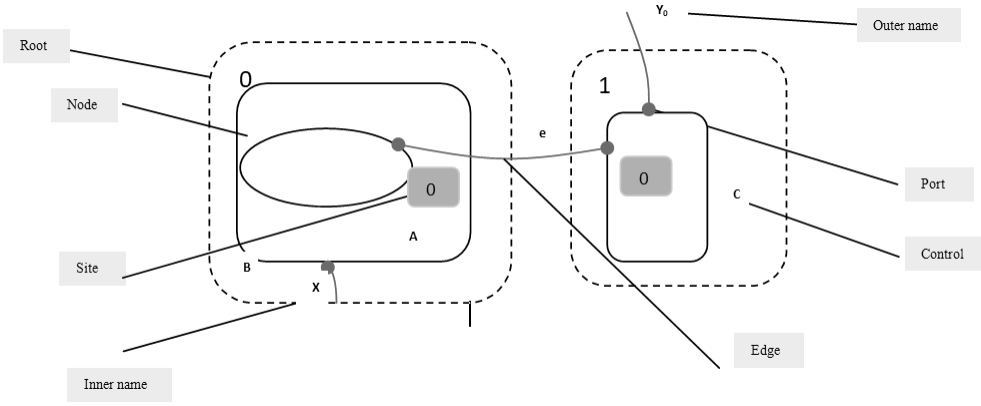


Figure 2: Bigraph example

node can have zero, one or more ports, represented by round points to express possible connections. Ports are represented by bullets. In the example, connections are depicted as links, by curvy lines, which may connect ports names (x z). These links, also called hyperedges, indicate the bigraph's connectivity (e.g., they can be considered as links to other bigraphs). We note that nodes which have the same control also have the same number of ports. The dotted rectangles indicate regions (also, called roots), their role is to describe parts of the system that are not necessarily adjacent. The blue squares represent sites that are abstract parts of the model. The regions sites are indexed by natural numbers from left to right (starting from 0). The nodes, sites regions are called the places of a bigraph. In addition to hyper-arcs, a bigraph can have other types of communication links which are internal external names. In our example (see Figure 2), y is an external name, while x represents an internal name. They express (potential) links to other bigraphs, representing external environments. Besides to their simple generous graphical form, an algebraic term language was provided to represent BRS. Indeed, the bigraphs can be constructed using elementary bigraphs with the help of algebraic operations. For instance, merge product designates the adjacency of bigraphs A B which are located in a one region (noted by $\mathsf{A} \mid \mathsf{B}$) .

Nesting operation (noted by $\mathsf{A}.\mathsf{B}$) allows placing bigraph B inside A parallel product term can be involved to create bigraphs by adjacencing their roots integrating their joint names (noted by $\mathsf{A} \parallel \mathsf{B}$).

For additional information concerning algebraic operations of bigraphs, the lector is invited to consult [15].

Sorting mechanism The sorting discipline for bigraphs was proposed in [15]. This discipline classifies controls links in diverse sorts. The sorting discipline is defined as $\Sigma = \{\Theta, \mathsf{k}, \Phi\}$, where Θ is a non-empty sort set, k is a signature, Φ is a rule set. A rule is a property set that a bigraph must to gratify.

Dynamical aspects: To finalize the description of a dynamic system by specifying its dynamic behavior, the corresponding bigraphs are provided with a set of bigracal reaction rules. This mixture generates the BRS. Formally, a reaction rule takes the form: $\mathsf{R} \longrightarrow \mathsf{R}'$ where the redex R defines the conditions that must be met, the reactum R' depicts the result of this rule.

4.2 Modeling multi-viewpoint ontology alignment structures

An MVp ontology is represented as a node set corresponding to the viewpoints as well as the concepts (global local) which can be determined by the context. Also, the viewpoint is represented by a set the nodes corresponding to the different internal concepts (global local). An MVp ontology alignment by composition system is interpreted by a bigraph $\mathsf{OM_A}$ including all MVp ontology elements. The bigraph OM is made of set of regions marked 0, 1, ... N that depict the viewpoints of a context. Where, each viewpoint is modeled separately by a distinct bigraph. Where, the hierarchical aspect in an MVp ontology is supported by the concept of node nesting. The interactions between the different entities of an MVp ontology are defined by hyper arcs (roles bridge links). The configuration of the Bigraph OM is obtained by the tensor product of viewpoints bigraphs. The bigraph $\mathsf{OM_A}$ is generated by the parallel product of MVp ontology bigraphs. The introduced sorting logic gives mapping rules formulates all constraints formation rules, that $\mathsf{OM_A}$ satisfies in order to guarantee appropriate exact encoding of MVp ontology alignment semantics into BRS concepts. In what follows, formal definitions are given.

Definition 5 *(viewpoint bigraph). The bigraph of a pointview* $\mathsf{View_i}$ *is formally given by:* $\mathsf{View_i} = (\mathsf{V_i}, \mathsf{E_i}, \mathsf{ctrl_i}, \mathsf{OP_i}, \mathsf{OL_i}) : \mathsf{I} \longrightarrow \mathsf{K}$

- $\mathsf{V_i}$ *is a finite set of nodes representing the different local global concepts of the* $\mathsf{View_i}$*.*
- $\mathsf{E_i}$ *is a finite set of roles (internal hyper-arcs).*
- $\mathsf{Ctrl_i} : \mathsf{V_i} \longrightarrow \mathsf{K}$ *is a transformation function which associates with each node* $\nu_\mathsf{i} \in \mathsf{V_i}$ *a control* $\mathsf{k} \in \mathsf{K_i}$ *indicating the number of ports. The signature K is a finite set of controls associated with the elements of ontologies.*

- $OP_i = (V_i, ctr_i, prnt_i) : m_0 \longrightarrow n_0$ *is the place graph associated with Viewid.* m_0 n_0 *are the number of sites regions.* $Prnt_i : m_0 \uplus V_i \longrightarrow V_i \uplus n_0$ *is a parent map that associates each entity with its hierarchical parent (e.g. the parent of a Man node is a human node).*

- $OL_i = (V_i, E_i, ctrl_i, link_i) : X_0 \longrightarrow Y_0$ *is the link graph of* $View_i$*, where* $link_i : X_0 \uplus P_0 \longrightarrow E_i \uplus Y_0$ *is a transformation function that specifies the interactions of each entity of the ontology.* X_0*,* Y_0 P_0 *are respectively, the inner names, the outer names, the port set of* $View_i$*.*

- $I_0 = (m_0, X_0)$ $J_0 = (n_0, Y_0)$ *are respectively, the inner outer interfaces of the bigraph* $View_i$*.*

Definition 6 *(Multi-viewpoint ontology bigraph). A bigraph* OM *modeling a multi-viewpoints ontology of a context id is formally given by:*

$OM \equiv View_1 \otimes View_2 \otimes \ldots \otimes View_n$

Where: $OM = (V, E, ctrl, OP, OL) : I \longrightarrow J$

- $V = V_1 \uplus V_2 \uplus \ldots \uplus V_n$ *is a finite set of nodes (local concepts, global concepts views) in a context i which given by the union of the set of* V_i *nodes of all views.*

- $E = E_1 \uplus E_2 \uplus \ldots . \uplus E_n \uplus$ *M is a finite set of hyper-arcs representing in a context id which given by the union of the set of hyperarcs* E_i *the set of the bridge links M.*

- $K = K_1 \uplus K_2 \uplus \ldots \uplus K_n$ *is an extended signature, defined by a set of controls where,* $Ctrl : V \longrightarrow K$ *is a new transformation which associates with each node* $vi \in V_i$ *a control* $k \in K_i$ *indicating the number of its fixed ports.*

- $OP = OP_1 \otimes OP_2 \otimes \ldots \otimes OP_n : m \longrightarrow n$ *is the places graph associated with* OM *given by the tensor product of the graphs of places* $OP_1 : m_1 \longrightarrow n_1, OP_2 : m_2 \longrightarrow n_2$ *, ...* $OP_n : m_n \longrightarrow n_n$ *, where its parent map is:* $prnt = prnt_1 \uplus prnt_2 \uplus \ldots \uplus prnt_n$

- $OL = OL_1 \otimes OL_2 \otimes \ldots \otimes OL_n$ *: is the links graph of* OM *given by the tensor product of the links graphs* $OL_1 : X_1 \longrightarrow Y_1, OL_2 : X_2 \longrightarrow Y_2, \ldots, OL_n : X_n \longrightarrow Y_n$*. Where,*

  $OL \equiv (V, E, ctrl_1 \uplus ctrl_2 \uplus \ldots \uplus ctrl_n, link_1 \uplus link_2 \uplus \ldots \uplus link_n)$

- *I J are respectively, the inner outer interfaces of the bigraph* OM*.*

Definition 7 *(Multi-viewpoint ontology alignment bigraph) The bigraph* $\mathsf{OM_A}$ *is formally given by:* $\mathsf{OM_A} \equiv \mathsf{OM} \parallel \mathsf{OM}'$ *Where:*
$\mathsf{OM_A} = (\mathsf{VOM_A}, \mathsf{EOM_A}, \mathsf{ctrlOM_A}, \mathsf{OMAP}, \mathsf{OMAL}) : \mathsf{IOM_A} \longrightarrow \mathsf{JOM_A}$

- $\mathsf{VOM_A} = \mathsf{VOM} \uplus \mathsf{VOM}'$ *is a finite set of nodes representing the different entities of ontologies* OM OM‘ *which given by the union of nodes sets* VOM VOM‘.
- $\mathsf{EOM_A} = \mathsf{EOM} \uplus \mathsf{EOM}'$ *is a finite set of hyper-arcs representing the different connections that can link entities together which given by the union of hyper-arcs sets* EOM EOM‘.
- $\mathsf{K} = \mathsf{KOM} \uplus \mathsf{KOM}'$ *is an extended signature, defined by a set of controls. Where,* $\mathsf{ctrlOM_A} : \mathsf{VOM_A} \longrightarrow \mathsf{K}$ *a control map that assigns each node* $\nu_i \in \mathsf{VOM_A}$ *with a control* $\mathsf{k} \in \mathsf{KOM_A}$.
- $\mathsf{OMAP} = \mathsf{OPOM} \parallel \mathsf{OPOM}' : \mathfrak{m} \longrightarrow \mathfrak{n}$ *is the places graph associated with* $\mathsf{OM_A}$ *given by the parallel product of the graphs of places* OPOM OPOM‘ . *Where its parent map is:* $\mathfrak{prnt} = \mathfrak{prnt}\mathsf{OM} \uplus \mathfrak{prnt}\mathsf{OM}'$.
- $\mathsf{OMAL} = \mathsf{OLOM} \parallel \mathsf{OLOM}'$ *: is the links graph associated with* $\mathsf{OM_A}$ *given by the parallel product of the graphs of links* $\mathsf{OLOM} : \mathsf{X_O} \longrightarrow \mathsf{Y_O}$ $\mathsf{OLOM}' : \mathsf{X}'_\mathsf{O} \longrightarrow \mathsf{Y}'_\mathsf{O}$. *Where,*

  $\mathsf{OL} \equiv (\mathsf{V}, \mathsf{E}, \mathsf{ctrlOM} \uplus \mathsf{ctrlOM}', \mathsf{linkOM} \uplus \mathsf{linkOM}')$*:*

  $(\mathsf{XOM} \uplus \mathsf{XOM}') \longrightarrow (\mathsf{YOM} \uplus \mathsf{YOM}')$
- $\mathsf{IOM_A}$ $\mathsf{JOM_A}$ *are respectively, the inner outer interfaces of the bigraph* $\mathsf{OM_A}$.

Definition 8 *(Multi-viewpoint ontology alignment discipline of sorting). The sorting discipline associated with the* $\mathsf{OM_A}$ *bigraph modeling a multi-viewpoint ontology alignment is defined by the triplet* $\Sigma\mathsf{OM_A} = \{\Theta\mathsf{OM_A}, \mathsf{k}, \Phi\ \mathsf{OM_A}\}$.

Where $\Theta\mathsf{OM_A}$ *represents a non-empty set of sorts of* $\mathsf{OM_A}$, $\mathsf{KOM_A}$ *is a* $\Sigma\mathsf{OM_A}$*-typed signature which associates a sort with each control of* $\mathsf{OM_A}$, $\Phi\mathsf{OM_A}$ *is a non-empty set of training rules imposing construction restrictions for* $\mathsf{OM_A}$.

Table 3 grants for each ontology concept, mapping rules for BRS equivalence. This consists of the control associated to the entity, its arity (number of ports) (e.g. view has at least 1 port), its associated sort its graphic notation (e.g. a local concept instance is represented by a circle). Sorts are involed to

Description	Control	Arity	Sorts	Graphical notation	Bigraph
Entity					
Global concept	GC	N	g	Rectangle	OMA, OM and View
Local concept	LC	N	l	Circle	OMA, OM and View
View	$View_i$	≥ 1	V	Rectangle	OMA and OM
MVp ontology	OM	≥ 1	O	Rectangle	OMA

Table 3: Controls sorts of the bigraph OM_A

	Rule Descriptions
Φ_0	All children of a 0-region and 1-region have a sort $x \in \{ g, l, v, O \}$
Φ_1	All children of a 0-node have a sort $x \in \{ g, l, O \}$
Φ_2	All nodes of sort l are atomic
Φ_3	In a v-node, at least one port is linked to other child v-node of the same bigraph
Φ_4	In a g-node, one or more ports can be linked to other child g-nodes of the others v-nodes
Φ_5	In a l-node, one or more ports can be linked to other child l-nodes or g-nodes of the same bigraph
Φ_6	In a O-node, at least one port is linked to other child O-node of the same bigraph

Table 4: Training rules PhiOM for the bigraph OM

differentiate node types for structural goals constraints while controls identify states parameters that a node can have.

Table 4 shows the formation rules $\Phi i, i \in [0 \ldots 6]$ *which provide construction constraints over the BRS specification. Formation rules present structural constraints over the BRS model. The rules* $\Phi 0 - \Phi 2$ *define the constraints on the hierarchical nesting of the different entities while the rules* $\Phi 3 - \Phi 6$ *define the restrictions on their links. For example, the rule* $\Phi 0$ *states that the principal region denoted 0 which represents a multi-viewpoint ontology, can only have children of nodes of sort g, l, v, O. Finally, the rule* $\Phi 3$ *requires that all viewpoints must be related to at least one another multi-viewpoint ontology.*

4.3 Modeling composition behaviors with BRS

In addition to their ability to model the infrastructure of the MVp ontology alignment by composition system, BRS allow the formal specification of this system state evolution thanks to their reaction rules. In this Section, our main contribution is to propose a set of parametric reaction rules that model the behavior of the MVp ontology alignment.

Here, the defined reaction rules describe the different mechanisms applied to the proposed system, in order to manage its alignment. It is about modeling

Reaction rule	Algebraic form
Generate new equivalence link	$R_1 \stackrel{def}{=} c.OM_i \mid c`.OM_i \rightarrow c_{\equiv}.OM_i \mid c`_{\equiv}.OM_i$
Generate new subsumption link	$R_2 \stackrel{def}{=} c.OM_i \mid c`.OM_i \rightarrow c_{\sqsubseteq}.OM_i \mid c`_{\sqsubseteq}.OM_i$
Generate new disjunction link	$R_3 \stackrel{def}{=} c.OM_i \mid c`.OM_i \rightarrow c_{\perp}.OM_i \mid c`_{\perp}.OM_i$
Update bridge link	$R_4 \stackrel{def}{=} c_R.OM \mid c_R`.OM_i \rightarrow c_{R'}.OM \mid c`_{R'}.OM_i$

Table 5: Reaction rules modeling alignment by composition actions in MVp ontology bigraph

the actions related to the creation of new mappings through the composition between the bridge links already existing in the system. Table 5 gives the defined reaction rules Ri expressing a set of possible actions that can be applied over an alignment system. These rules take the form Ri = R →R‘, where i is the index of the rule, R is the redex part of the reaction R‘ is its reactum part. A reaction is applied by replacing the redex bigraph (left-hand side) with the reactum bigraph (right-hand side of the reaction). As both redex reactum bigraphs respect the formation rules. Concretely, the specified rules describe the different actions related to the generation of new mappings (R1 - R3) at the MVp ontology level. Composition strategies:

In this section, we explain how we can use the reaction rules previously defined to formalize the overall behavior of an alignment system in terms of composition strategies. Indeed, the presented reaction rules can be used to simulate different evolution strategies of MVp ontology alignment by composition systems. Each strategy consists of a sequence of applications of reaction rules. This leads to restricting the application of rules via conditions, so that a reaction rule is only applied when desired (i.e. when the conditions for triggering that action are satisfied). Here, we introduce reactive composition strategies of the form: IF (s) then (s).

In the context of bigraphic semantics, a condition takes the form ($OM_A \models \Phi i$). This condition is satisfied iff ∃ a bigraph OM_A‘ ($OM_A \Phi i$), encoding the predicate Φi, which occurs in the context of OM_A. A predicate Φi, often expressed in first-order logic, is used to define a state of the MVp ontology alignment systems OM_A Φi defines a bigraphic model encoding this state. So, a strategy that reacts to a condition ($OM_A \models \Phi i$) is expressed as: START; IF $OM_A \models \Phi i$ THEN Ri. The actions Ri are modeled as bigraphical reaction rules.

Level	Condition
MVp ontology alignement system	Existance of equivalence bridge link between two concepts C_i and C_j of two MVp ontology OM and OM` $\varphi 1 \stackrel{def}{=} \exists C_i \in OM$ and $\exists C_j \in OM`$ $link_{OMA}(C_i, C_j) = \equiv$
	Existance of equivalence bridge link between two concepts C_i and C_k of a same MVp ontology OM $\varphi 2 \stackrel{def}{=} \exists C_i$ and $\exists C_k \in OM`$ $link_{OMA}(C_i, C_k) = \equiv$
	Existance of disjction bridge link between two concepts C_i and C_j of two MVp ontology OM and OM` $\varphi 3 \stackrel{def}{=} \exists C_i \in OM$ and $\exists C_j \in OM`$ $link_{OMA}(C_i, C_j) = \perp$
	Existance of bridge link between two concepts C_i and C_k of a same MVp ontology OMA $\varphi 4 \stackrel{def}{=} \exists C_i$ and $\exists C_k \in OMA$ $link_{OMA}(C_i, C_k) = \perp$
	Existance of subsumption bridge link between two concepts C_i and C_j of two MVp ontology OMA and OMA` $\varphi 5 \stackrel{def}{=} \exists C_i \in OMA$ and $\exists C_j \in OMA`$ $link_{OMA}(C_i, C_j) = \subseteq (\supseteq)$
	Existance of subsumption bridge link between two concepts C_i and C_k of a same MVp ontology OMA $\varphi 6 \stackrel{def}{=} \exists C_i$ and $\exists C_k \in OMA$ $link_{OMA}(C_i, C_k) = \subseteq (\supseteq)$

Table 6: Definitions of conditions

Level	Conditions	Action
MVp ontology alignement system	$\varphi 1$ and $\varphi 2$	R1
	$\varphi 3$ and $\varphi 4$	R2
	$\varphi 5$ and $\varphi 6$	R3
	$\varphi 1$ and $\varphi 4$	R2
	$\varphi 1$ and $\varphi 6$	R3
	$\varphi 3$ and $\varphi 4$	R2
	$\varphi 5$ and $\varphi 4$	R3

Table 7: Composition strategies

These strategies consist of creating new mappings which used to ensure correspondences between the different concepts of the different MVp ontologies. This involves authorizing the addition of new mapping links by applying the associated reaction rules according to the composition table cited in Definition 8. These strategies can be applied at the level of the MVp ontology alignment system as following:

It reacts to conditions: "if there exists an equivalence bridge link among concepts C_i of MVp ontology OM C_j of MVp ontology OM'", "if there exists one more equivalence bridge link among concepts C_j C_k of the same MVp ontology (OM')", respectively expressed through the predicates $\Phi 1$ $\Phi 2$. The verification of existence consists of checking if there exists a node node1, which belongs to the node set VOM_1 of OM_1 another node node2, belonging to the node set VOM_2 of OM_2, whose control is $linkOM_A(node_1, node_2)$ is equivalence. The definitions of all possible conditions are introduced in Table 6. If the predicates $\Phi 1$ $\Phi 2$ are satisfied, the rule R1 is applied to generate a novel mapping link. In the same way, the reaction rules R3 R2 are applied, according to table 7.

5 Case study

We introduce in this section a simplified application case of an MVp ontology alignment by composition system. In the beginning, we apply the proposed approach to realize this application case. After, we identify the related Maude descriptions of the resulted BRS models, as presented in section 6, to carry out them by the Maude program. We considered two MVp ontologies: OM_1 OM_2 describing the staff domain. The first MPV ontology is composed of two viewpoints: personnel affairs civil status while the second MVP ontology is composed of the two viewpoints: family membership nationality (see Figure3 Figure 4).

The parallel product of the graphs OM_1 OM_2 generates a new bigraph: $OM_A \equiv OM_1 \parallel OM_2$. OM_A models a composite ontology for alignment (see Figure 6) by :

- $VOM_A = \{Familymembership, Nationality, Personnelaffairs, \ldots\}$ (set of internal nodes).
- $EOM_A = \{sexe, is, has, \equiv, \supseteq, \perp\}$
- $CtrlOM_A = \{human : 3, man : 1, boy : 1, father : 1, person : 3, woman : 1 \ldots.\}$ indicates the number of fixed ports for each node.

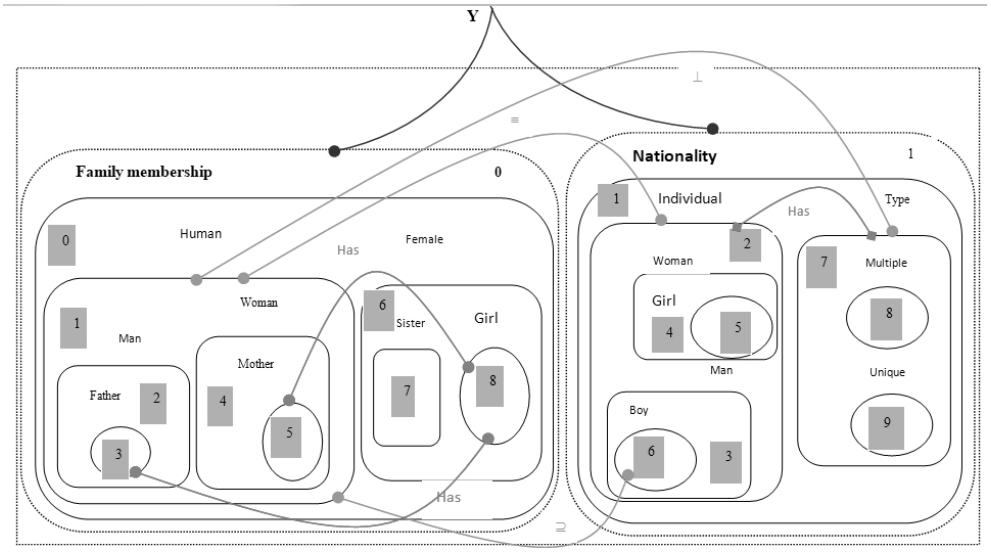


Figure 3: Example of a multi-viewpoint ontology bigraph OM_1.

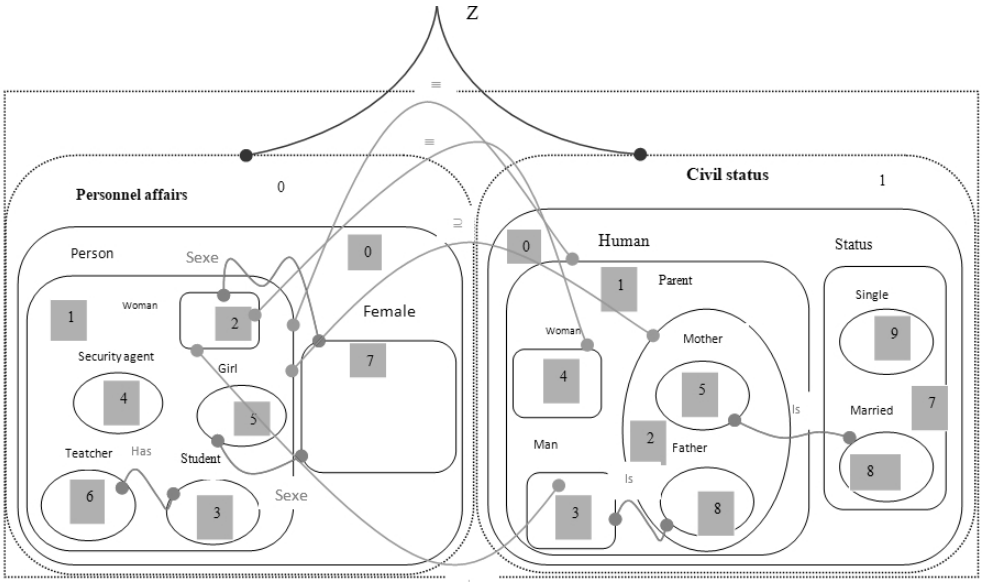


Figure 4: Example of a multi-viewpoint ontology bigraph OM_2.

y/Family membership (human (man(father. d_1) | d_2) | d_3) | (woman(mother. d_5) … | d_0‖ | y/ Nationality (individual (man (boy. d_6) | d_3) | d_1| woman ...) |d_0 ‖ z/ Personal affairs (person (woman . d_2) ... | d_1 |d_0 ‖ z/ Civil status (human (parent (mother . d_5) | d_2) | d_1 ...) | d_0 → y/Family membership (human (man(father. d_1) | d_2 | d_3 | (woman(mother. d_5) … | d_0‖ | y/ Nationality (individual$_{\equiv}$ (man (boy. d_6) | d_3) | d_1| woman ...) |d_0 ‖ z/ Personal affairs (person$_{\equiv}$ (woman . d_2) ... | d_1 |d_0 ‖ z/ Civil status (human (parent (mother . d_5) | d_2) | d_1 ...) | d_0

Figure 5: The rule R1 application

- $\mathsf{Prnt} = \{\mathsf{human} : \mathsf{man}, \mathsf{man} : \mathsf{boy}, \mathsf{man} : \mathsf{father}, \mathsf{person} : \mathsf{man}, \mathsf{person} : \mathsf{boy}, \mathsf{person} : \mathsf{woman}...\}$ indicates the hierarchical parent of each internal node.

- $\mathsf{IOM_A} = (34, \phi)$ is the internal interface of the bigraph $\mathsf{OM_A}$. m = 34 represents the number of regions. Where 2 regions containing nodes that can be hosted, $\mathsf{X} = \mathsf{phi}$ represents the set of internal names.

- $\mathsf{JOM_A} = (4, \{\mathsf{y}, \mathsf{z}\})$ is the external interface of the bigraph $\mathsf{OM_A}$. Where n = 4 Y represents the number of sites the set of external names, respectively.

- Finally, the places graph $\mathsf{OMAP} : 34 \longrightarrow 4$ is the result of the parallel product of the graphs of places $\mathsf{OMP_1}$ $\mathsf{OMP_2}$, the links graph OMAL $\{\} \longrightarrow \{\mathsf{y}, \mathsf{z}\}$ is the result of the parallel product of the graphs of links $\mathsf{OML_1}$ $\mathsf{OML_2}$.

In the ontology $\mathsf{OM_A}$, the bridge links among the concepts: person, human individual are as follows: (human ≡ individual) which means that the condition: $\mathsf{OM_A} \models \Phi 1$ is satisfied ($\mathsf{human} \equiv \mathsf{person}$) which means that the condition: $\mathsf{OM_A} \models \Phi 2$ is satisfied. According to the table 7, we can deduct the bridge link: ($\mathsf{person} \equiv \mathsf{individual}$). After applying R1, this link is easily added to the bigraph $\mathsf{OM_A}$ as shown in Figure 5. In the same way, we apply the other possible rules. Figure 6 shows the result of applying the composition strategies.

In the next section, we show how we apply the Maude program to implement simulate our models. Furthermore, we demonstrate how Maude's mecanism can be used to attain the composition process at the MVp ontology alignment system level.

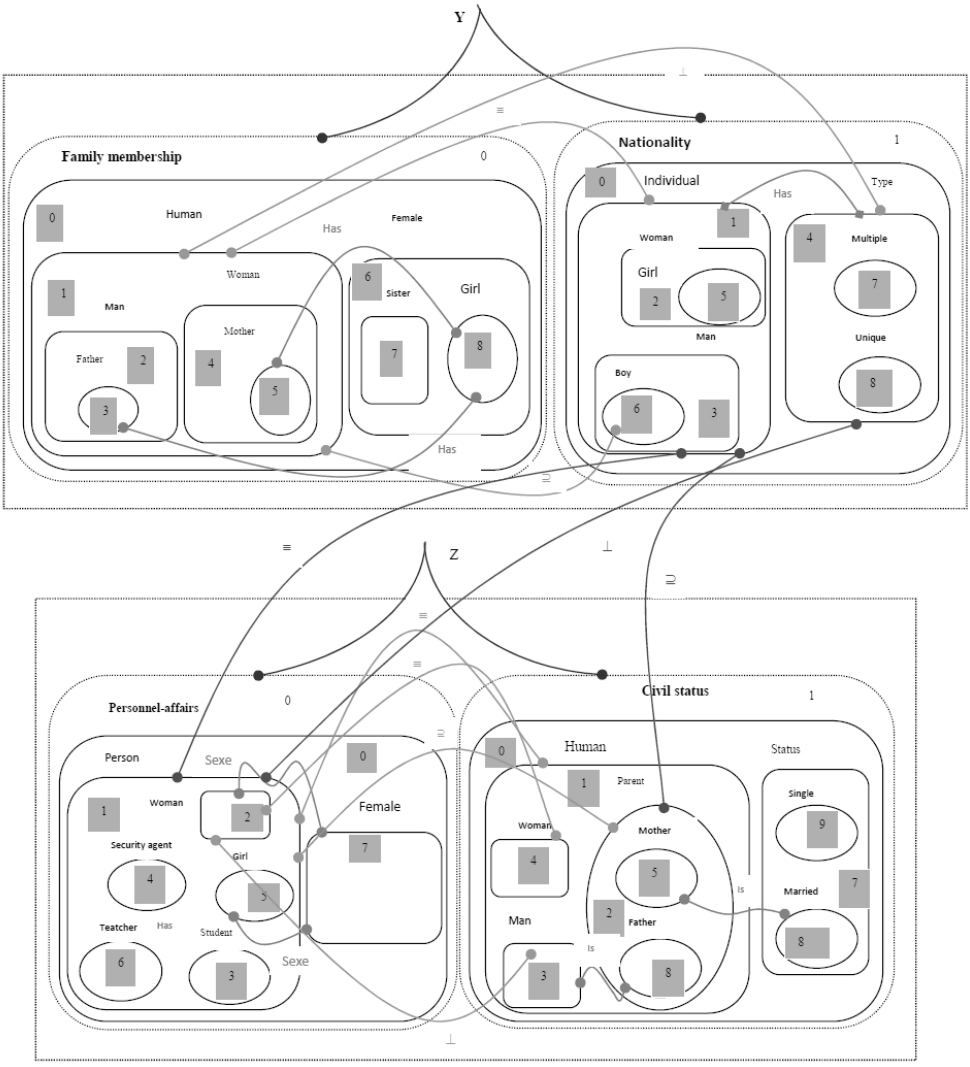


Figure 6: The result of applying the composition strategies on the bigraph OM_A

MVp ontology alignment by composition based model	Maude language
Syntax	
Multi-viewpoint ontology alignment bigraph	**Sort bigraph** op -|- ||-|- MVp-ontology1 site MVp-ontology2 → Bigraph
MVp-ontology roots	Sorts MVp-ontology Viewpoint Concept Subsort global-concept < Concept Subsort local-concept < Concept Op MVp-ontology -, (-|-) Nat Viewpoint Viewpoint → MVp-ontology [ctor] Op Family-membership [-] Link → Viewpoint [ctor] Op Personnel-affairs [-] Link → Viewpoint [ctor] Op Human [-, -, -] Link Link Link → global-concept [ctor] Op Father [-] Link → local-concept [ctor] .....
Types of links	Sort Link Subsorts Inner Outer Role < Link Subsorts equivalent subsume disjoint < Inner Subsorts sexe is has < Role Op Y - : Nat → Outer [ctor]
Concepts relationship predicates	Op Isequivalent (-, -): Concept Concept → Bool Op Issubsume (-, -): Concept Concept → Bool Op Isdisjoint (-, -): Concept Concept → Bool
Site	Sort Site Op $- : Nat → Site [ctor]
Dynamic	
Reaction rules	**Conditional rewrite rules of the form:** Crl [rewrite rule name] : bigraph ⇒ bigraph` if conditions

Table 8: Mapping MVp-ontology BRS-based model to Maude.

6 Executing MVp ontology alignment by composition model

BRS represents an ideal formalism for specifying the structural behavioral issues of MVp ontology alignment. However, existing tools based on this formalism are restricted fixed to some application fields. Here, we have chosen the Maude program [3] for realizing the BRS-based MVp ontology alignment model. The selection of the Maude program is justified by its capacity to specify at a high formal level.

In this section, we explain how to interpret our models to Maude codifications. So, two fundamental modules are given: the Syntax Dynamic modules. Table 8 summarizes the mapping rules among BRS concepts Maude language.

```
Vars d0 d1 d2 d3 d4 d5 s0 s1 : Nat.

rl [ Add-Equivalence-Link]:

… Nationality (individual[Equivalent, Has] d0. (man .d3 … | $s1) Personal-
affairs | person[Equivalent, subsume]d0 . (woman . d2) … | $s0) ||…  => … ||
Nationality (individual[Equivalent, Has, Equivalent] d0. (man .d3 … | $s1)
Personal-affairs | person [Equivalent, subsume, Equivalent] d0 . (woman .
d2) … | $s0) ||…
If (Isequivalent (human, individual) and Isequivalent (human, person)
```

Figure 7: Generating new equivalence link rewrite rule.

Encoding MVp ontology alignment compositions: For the Syntax module, the sorts g, l, v o (global concept, local concept, viewpoint MVp ontlogy) are generated depending to their related Maude constructors (ctor). We design the sorts like: global-concept, local-concept, view MVp ontology we propose sort Link Site for identifying the different links sites of an MVp ontology alignment system. Finally, a sub-sort relationship is established between some mentioned sorts. The main operator of this part is: (op− | − || − |- MVp ontology1 site MVp ontology2 → Bigraph). It consists of declaring of the static composition of an MVp ontology alignment bigraphical model, that is established by two diverse roots performing the MVp ontology1 the MVp ontology2 of an alignment process. The term of juxtaposition (||) is used to split MVp ontology1 the MVp ontology2.

Encoding Alignment Predicates: the syntax part identify a predicate set that reflects a relation among two concepts. For example, Isequivalent() means that "the concepts are equivalents". Issubsume()means that "there exists a subsumption relation among concepts". Finally, Isdisjoint () is a predicate that stands for "the concepts are disjoints".

Encoding Alignment Strategies: The dynamic part implements the alignment strategy in the form of conditional rewrite rules. Figure 7 illustrates an example of rule; that is in charge of generating the new bridge link: (person ≡ individual).

6.1 Evaluation

In the following sections, we achieve a sequence of experiments to evaluate our approach on different aligned ontologies of different domains (including smart homes, education, healthcare) with different ontology sizes (number of concepts rules). These experiments are conducted on the precision, recall, F-

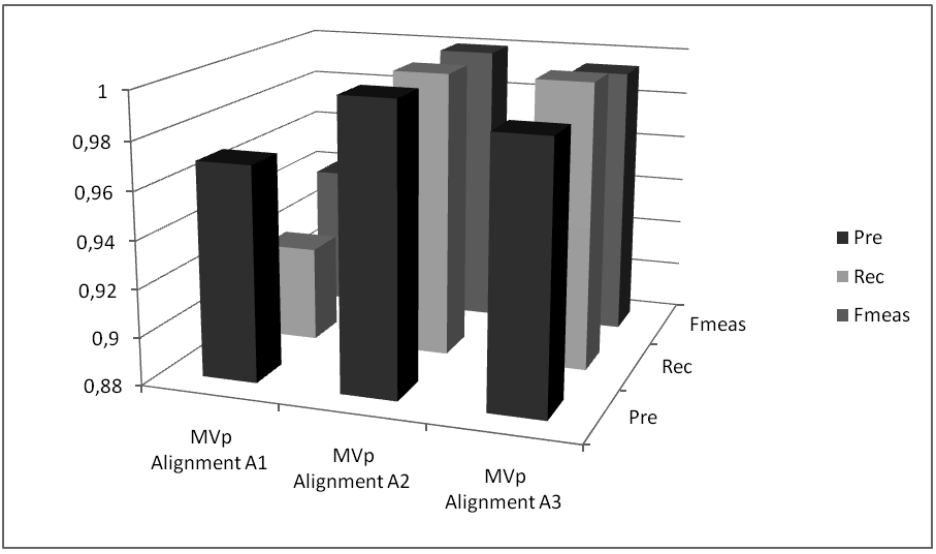


Figure 8: Figure 8. MVp ontology alignment results comparison

measure suggested by [4] as follows: Given a reference alignment $R-AMVp$, the precision Pre, the recall Rec $F-measure$ $Fmeas$ of an MVp Ontology alignment $A-MVp$ is considered such that:

$Pre(A-MVp, RA-MVp) =\mid RA-MVp \cap A-MVp \mid /A-MVp$

$Rec(A-MVp, RA-MVp) =\mid RA-MVp \cap A-MVp \mid /RA-MVp$

F-measure merges precision recall such that: $Fmeas(A-MVp, RA-MVp) = [2*Pre(A-MVp, RA-MVp)*Rec(A-MVp, RA-MVp)]/[Pre(A-MVp, RA-MVp) + Rec(A-MVp, RA-MVp)]$

The alignments generated by the proposed approach, are assimilated to the reference alignment (realized manually by experts in domains). Indeed, the values of the alignment quality measures (precision, recall F-measure) are calculated.

The results are provided in Figure 8. It can be seen that the results of the alignments generated by the experts the proposed approach are very similar. So in general, we can say that our approach proves good performance.

7 Conclusion

For the Semantic Web, ontology-based semantic interoperability is considered as a major defy. Ontology alignment is a key process that plays a significant

role in improving this interoperability resolve ontological heterogeneity issue. The main goal of this work is to deal with this problem by adopting the BRS as formalism to Specify MVp ontology alignment system. Especially, we have proposed a novel approach, relied on BRS with their sorting controlling function. Concerning the static structural side, we have established the definitions of all MVp ontology alignment entities such as viewpoints, concepts roles. As for the behavioral side, is established by a generic reaction rule set that depicts the alignment system in terms of composition strategies. After, we have explained how to merge Maude program BRS to carry out the MVp ontology alignment. Finally, the possibility of realizing the proposed approach is proven by a case study. As far as we know, this is the first paper to deal with the MVp ontology alignment using BRS. As part of the forthcoming works, our goal is to more elaborating expanding our bigraphical model of multi-viewpoint ontology alignment by composition systems, in order to handle all kinds of alignment.

References

[1] A. Algergawy, S. Massmann, E. Rahm, A clustering-based approach for large-scale ontology matching. *ADBIS 2011*, Vienna, Austria, 2011, pp. 415–428. ⇒ 184

[2] J. Chakraborty, H.M. Zahera, M.A. Sherif, S.K. Bansal, Ontoconnect: domain-agnostic ontology alignment using graph embedding with negative sampling. *ICMLA 2021*, Pasadena, USA, 2021, pp. 942–945. ⇒184

[3] M. Clavel, F. Duran, S. Eker, P. Lincoln, N. J. Martf-Oliet Meseguer, CL. Talcott, All about Maude, *A High- Performance Logical Framework*. Lecture Notes in Computer Science. Springer. 4350 (2007). ⇒200

[4] J. Euzenat P. Shvaiko, *Ontology Matching*. Springer 2007, Heidelberg. (2007) 1–333. ⇒202

[5] J. Euzenat, Revision in networks of ontologies. Artificial Intelligence, *Elsevier*. 228, (2015) 195–216. ⇒184

[6] M.,Hemam, M. Djezzar, Z. Boufaida, Multi-viewpoint ontological representation of composite concepts: a description logics-based approach. *Int. J. Intelligen Information Database Systems* l0, 1-2, (2017) 51–68. ⇒186

[7] M. Hemam, M. Djezzar, Z.A. Seghir, Multi-viewpoints ontological knowledge representation: a fuzzy description logics based approach. *Proceedings of the Fourth International Conference on Engineering & MIS 2018*, Istanbul, Turky, (2018) pp. 1–6. ⇒182

[8] E. Jimenez-Ruiz, A., Agibetov, M. Samwald, V. Cross. Breaking-down the ontology alignment task with a lexical index neural embeddings. CoRR, abs/1805.12402, (2018) ⇒184

[9] S. Klai, A. Zimmermann M.T. Khadir, Netw-orked ontologies with contextual alignments. *Computing Informatics*, 38, 1, (2019) 115–150. ⇒184
[10] M. Kolli, Z. Boufaida, A description logics formalization for the ontology matching. *Proc. of Computer Sci. Journal, Elsevier,* 3, 5, (2010) 29–35. ⇒187
[11] M. Kolli, Z. Boufaida, Composing semantic relations among ontologies with description logics. *Information Technology Journal*, 10, 6, (2011) 1106–1112. ⇒184
[12] M. Kolli, Formalising repairing semantic networks of ontologies with linear temporal logics. *International Journal of Metadata, Semantics Ontologies*, 11, 4, (2016) 264–272. ⇒182, 184
[13] R. Milner, Pure bigraphs: Structure dynamics, *Information Computation*, 204, 1, (2006) 60–122. ⇒188
[14] R. Milner, Bigraphs their algebra. *Electronic Notes in Theoretical Computer Science*, 209, (2008) 5–19. ⇒188
[15] R. Milner, *The Space motion of communicating agents.* Cambridge. New York, USA, (2009). ⇒188, 189, 190
[16] S. Pereira, V. Cross, E. Jimenez-Ruiz, On partitioning for ontology alignment, in Int'l Sem. Web Conf, Vienna, Austria, 2017 pp. 1–4. ⇒184
[17] J. Portisch, G. Costa, K. Stefani, K. Kreplin, M. Hladik, H. Paulheim,. Ontology Matching Through Absolute Orientation of Embedding Spaces. *ESWC*, Satellite Events, 2022 pp. 153–157. ⇒184
[18] F. Santos, K. Revoredo, F. Baiao,. SUBINTERNM: Optimizing the matching of networks of ontologies. *OM@ISWC 2020*, Athens, Greece, 2020 pp. 77–81. ⇒184
[19] G.F. Schneider. Automated ontology matching in the architecture, engineering construction domain a case study, *7th Linked Data in Architecture Construction Workshop (LDAC)*, Lisbon, Portugal, 2019 pp. 35–49. ⇒184
[20] J. Silva, K. Revoredo, Araujo, F. Baião, J. Euzenat, Alin: improving interactive ontology matching by interactively revising mapping suggestions. *Knowl. Eng. Rev.*, 35, e1, (2020) 1–22. ⇒184
[21] A. Zimmermann, Sémantique des réseaux de connaissances. Université Joseph Fourier Grenoble 1, (2008), http://hal.inria.fr/tel-00341525/PDF/thesezimmermann.pdf ⇒183

Received: May 6, 2023 • Revised: September 30, 2023

Acta Univ. Sapientiae Informatica **15,** 2 (2023) 205–220

DOI: 10.2478/ausi-2023-0014

Average distance colouring of graphs

Priyanka PANDEY
CHRIST (Deemed to be University)
Bengaluru, India
email:
priyanka.pandey@res.christuniversity.in

Mayamma JOSEPH
CHRIST (Deemed to be University)
Bengaluru, India
email:
mayamma.joseph@christuniversity.in

Abstract. For a graph G with n vertices, average distance $\mu(G)$ is the ratio of sum of the lengths of the shortest paths between all pairs of vertices to the number of edges in a complete graph on n vertices. In this paper, we introduce average distance colouring and find the average distance colouring number of certain classes of graphs.

1 Introduction

For the present study, we consider a graph $G = (V, E)$ with the set of vertices denoted by V and the set of edges E. For terminology and notation not defined here we refer to [1]. The distance between two vertices u and v denoted by $d(u, v)$, is the length of the shortest $u - v$ path, also called a $u - v$ geodesic. The distance between two vertices is considered as the base of the definition of various graph parameters [3]. In this paper, we introduce average distance colouring and obtain average distance colouring number for certain classes of graphs. Note that the distance between two vertices becomes infinite in disconnected graphs therefore, we consider only connected graphs for our study. Although the value of the average distance [6] $\mu(G)$ of any graph G depends on the sum of the distance between every pair of vertices, which generally would keep changing with the change in n, interestingly, we can find

Key words and phrases: graph colouring, average distance, average distance colouring

a constant bound for the value of $\mu(G)$ for various graph classes. We use such classes and study average distance colouring for the same. Before defining *average distance colouring*, we present the definition of average distance of graph G as defined in [2]. For a graph G with n vertices, average distance $\mu(G)$ is the ratio of sum of the lengths of the shortest paths between all pairs of vertices to the number of edges in a complete graph on n vertices. This can also be represented by the following equation.

$$\mu(G) = \frac{1}{\binom{n}{2}} \sum_{u,v \in V} d(u,v).$$

In our study, we focus on finding the exact value of $\mu(G)$ for certain classes not studied before and use them to colour the related graphs. Note that all graphs considered are connected and are of order at least two.

For a graph G with average distance $\mu(G)$, an average distance colouring of $G = (V, E)$ is defined as a function c from V to the set of non-negative integers, such that for any $v \in V$, $|c(v) - c(u)| \geq 1$ for all u such that $d(u,v) \leq \lceil \mu(G) \rceil$. The minimum number of distinct colours required to colour any graph G such that G admits average distance colouring is called average distance colouring number, χ_μ, of G.

Example 1 *Figure 1 shows the average distance colouring of graph G whose $\mu(G) = 1.714$.*

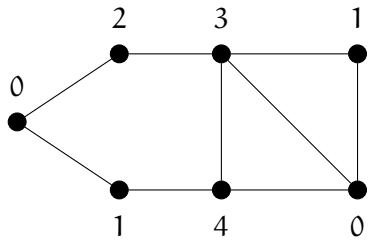


Figure 1: Average distance colouring of graph G with $\mu(G) = 1.714$

From the definition, it follows that all graphs admit average distance colouring. Also, average distance colouring is equivalent to chromatic colouring only for graphs with diameter 1, i.e., *complete graphs*. Further, when $k = \lceil \mu(G) \rceil$, average distance colouring is equivalent to *distance-k colouring* [4, 5]. Before we obtain the bounds and values for average distance colouring number for

some specific classes of graphs, we obtain the relation between the value of χ_μ of a graph G and its spanning subgraph H.

Observation 2 *For a graph* G *and its spanning subgraph* H*, such that the average distance of* G *is* $\mu(G)$ *and the average distance of its spanning subgraph* H *is* $\mu(H)$*,* $\mu(G) \leq \mu(H)$*.*

The reason is that the spanning subgraph of graph G on n vertices has a larger value of average distance as the number of edges in the spanning subgraph of the graph will be less as compared to the original graph, thus increasing the distance between the pair of vertices joined by the deleted edge thus leading in the increase of the value of numerator which eventually leads to the increase in the value of average distance. From the above argument, we get the following observation.

Observation 3 *For two graphs* G_1 *and* G_2 *such that* $\mu(G_1) \geq \mu(G_2)$*,* $\chi_\mu(G_1) \geq \chi_\mu(G_2)$*.*

Theorem 4 *For a graph* G *on* n *vertices with average distance* $\mu(G)$*,* $\chi_\mu = n$ *if and only if* $d = \lceil \mu(G) \rceil$ *where* d *denotes the diameter of* G*.*

Proof. Consider a graph G with diameter $d = \mu(G)$. This implies that the vertices are either distance 1 or 2 apart. In this case, the definition for average distance colouring implies that the pair of vertices u and v at a distance at most $\lceil \mu(G) \rceil$ should receive distinct colours. Since $d = \mu(G)$, the above statement implies that the vertices at most distance d apart should get distinct colours and each pair of vertices are at a distance at most d, Thus we require distinct colours for each vertex.
Let G be a graph on n vertices with $\chi_\mu = n$ and diameter d. We know that $c(u) \neq c(v)$ for all $u, v \in V(G)$. This implies $d(u, v) \leq \lceil \mu(G) \rceil$ for all $u, v \in V(G)$. Since $d = max\ d(u, v)$ over all pairs of $u, v \in V(G)$, $d \leq \lceil \mu(G) \rceil$. Since every pair of vertex gets distinct colour, this implies that every pair of vertices is at most $\lceil \mu(G) \rceil$ distance apart. We know $d(u, v) \leq d$ for all pairs of $u, v \in V(G)$, thus $d \geq \lceil \mu(G) \rceil$. Thus $d = \lceil \mu(G) \rceil$. □

2 Average distance colouring of some classes of graphs

In this section, we obtain the average distance colouring number of certain classes of graphs and show the procedure to colour the same using average distance colouring.

Note that the average distance of any graph can only be one if and only if it is a complete graph. For any graph G where G is not complete, $\mu(G) > 1$. For a complete graph K_n, all vertices are pairwise adjacent and for each $v \in V(K_n)$, $|c(u) - c(v)| \geq 1$ for all u such that $d(u, v) \leq 1$, it is easy to observe that we require n distinct colours to colour the graph which can easily be attained by colouring each vertex with different colours. This implies $\chi_\mu(K_n) \geq n$. We define function c such that $c(v_i) = i-1$, for $i = 1, 2, 3, ..., n$ giving $\chi_\mu(K_n) \leq n$. Using the function c defined above, we require colours $0, 1, 2, ..., n-1$ to colour any complete graph on n vertices, which leads to the following observation.

Observation 5 *For a complete graph* K_n, *for* $n \geq 2$, $\chi_\mu(K_n) = n$.

Next, we consider paths and cycles and obtain the value for average distance colouring number $\chi_\mu(G)$. Before obtaining the result on average distance colouring, we require the following results.

Theorem 6 [2] *The average distance of paths on* n *vertices*

$$\mu(P_n) = \frac{n+1}{3}.$$

Theorem 7 [2] *The average distance of cycles on* n *vertices*

$$\mu(C_n) = \begin{cases} \dfrac{(n+1)}{4} & \text{if } n \text{ is odd, and} \\ \dfrac{n^2}{4(n-1)} & \text{if } n \text{ is even.} \end{cases}$$

Theorem 8 *For path* P_n *on* n *vertices,* $n \geq 3$, $\chi_\mu(P_n) = \left\lceil \dfrac{n+1}{3} \right\rceil + 1$.

Proof. Consider a path P_n with vertices labelled $v_1, v_2, ..., v_n$. Using Theorem 6, the definition of average distance colouring reduces to the function c from V to a set of non-negative integers such that for any $v \in V$, $|c(u) - c(v)| \geq 1$ for all u such that $d(u, v) \leq \left\lceil \dfrac{n+1}{3} \right\rceil$. We define a colouring c such that

$$c(v_i) = \begin{cases} 0 & \text{if } i \equiv 1 \pmod{\left\lceil \frac{n+1}{3} \right\rceil + 1} \\ 1 & \text{if } i \equiv 2 \pmod{\left\lceil \frac{n+1}{3} \right\rceil + 1} \\ 2 & \text{if } i \equiv 3 \pmod{\left\lceil \frac{n+1}{3} \right\rceil + 1} \\ \vdots & \\ \left\lceil \frac{n+1}{3} \right\rceil - 1 & \text{if } i \equiv \left\lceil \frac{n+1}{3} \right\rceil \pmod{\left\lceil \frac{n+1}{3} \right\rceil + 1} \\ \left\lceil \frac{n+1}{3} \right\rceil & \text{if } i \equiv 0 \pmod{\left\lceil \frac{n+1}{3} \right\rceil + 1} \end{cases}$$

This function gives an average distance colouring with $\chi_\mu(P_n) \leq \left\lceil \frac{n+1}{3} \right\rceil + 1$. Due to average distance colouring constraint the vertices which are at distance $\left\lceil \frac{n+1}{3} \right\rceil$ from v_1 cannot have same colour thus we require $\left\lceil \frac{n+1}{3} \right\rceil + 1$ distinct colour to colour any path of length n giving $\chi_\mu(P_n) \geq \left\lceil \frac{n+1}{3} \right\rceil + 1$. Hence, the result. □

Theorem 9 *For cycles on* n *vertices,*

$$\chi_\mu(C_n) \leq \begin{cases} \left\lceil \frac{(n+1)}{4} \right\rceil + 1 + r & \textit{if } n \textit{ is odd, and} \\ \left\lceil \frac{n^2}{4(n-1)} \right\rceil + 1 + r & \textit{if } n \textit{ is even.} \end{cases}$$

where r is the remainder obtained after dividing n *by* $\lceil \mu(C_n) \rceil$.

Proof. Consider a cycle C_n with n vertices ordered $v_1, v_2, ..., v_n$ such that v_i is adjacent to v_{i+1} for $1 \leq i \leq n-1$ and v_1 adjacent to v_1.
Using Theorem 7, the definition for average distance colouring reduces to a colouring c such that $|c(u) - c(v)| \geq 1$ for all u such that $d(u, v) \leq \lceil \mu(C_n) \rceil$ for every $v \in V$, where $\mu(C_n) = \frac{(n+1)}{4}$ if n is odd, and $\mu(C_n) = \frac{n^2}{4(n-1)}$ if n is even.

To colour the cycle, we will use the following function till a certain value of i which is given in the following cases.

$$c(v_i) = \begin{cases} 0 & \text{if } i \equiv 1 \pmod{\lceil \mu(C_n) \rceil + 1} \\ 1 & \text{if } i \equiv 2 \pmod{\lceil \mu(C_n) \rceil + 1} \\ 2 & \text{if } i \equiv 3 \pmod{\lceil \mu(C_n) \rceil + 1} \\ \vdots & \\ \vdots & \\ \lceil \mu(C_n) \rceil - 1 & \text{if } i \equiv \mu(C_n) \pmod{\lceil \mu(C_n) \rceil + 1} \\ \lceil \mu(C_n) \rceil & \text{if } i \equiv 0 \pmod{\lceil \mu(C_n) \rceil + 1} \end{cases} \tag{1}$$

Further, we will consider the following cases.

Case 1: When n divided by $\lceil \mu(C_n) \rceil + 1$ leaves no remainder.
In this case, the function c defined as in Equation (1) is used for all $1 \leq i \leq n$ thus, giving the average distance colouring of cycle with $\chi_\mu \leq \lceil \mu(C_n) \rceil + 1$.

Case 2: When n divided by $\lceil \mu(C_n) \rceil + 1$ leaves a remainder.
In this case, we obtain the remainder (say r) after dividing n by $\lceil \mu(C_n) \rceil + 1$. For $1 \leq i \leq n - r$, we use function c defined as in Equation (1) to colour the vertices. For the remaining r vertices, we define c given as in Equation (2)

$$c(v_i) = \begin{cases} \lceil \mu(C_n) \rceil + 1 & \text{if } i = n - r + 1 \\ \lceil \mu(C_n) \rceil + 2 & \text{if } i = n - r + 2 \\ \lceil \mu(C_n) \rceil + 3 & \text{if } i = n - r + 3 \\ \vdots & \\ \vdots & \\ \lceil \mu(C_n) \rceil + (r-1) & \text{if } i = n - r + (r-1) \\ \lceil \mu(C_n) \rceil + (r) & \text{if } i = n - r + (r) \end{cases} \tag{2}$$

which gives $\chi_\mu \leq \mu(C_n) + r + 1$.

On substituting the value of $\mu(C_n)$ for odd and even number of vertices we get,

$$\chi_\mu(C_n) \leq \begin{cases} \left\lceil \frac{(n+1)}{4} \right\rceil + 1 + r \text{ if n is odd, and} \\ \left\lceil \frac{n^2}{4(n-1)} \right\rceil + 1 + r \text{ if n is even.} \end{cases}$$

□

For the next result, we consider complete bipartite graphs. The following result gives the χ_μ of complete bipartite graph $K_{r,s}$.

Theorem 10 *For a complete bipartite graph* $K_{r,s}$, $\chi_\mu(K_{r,s}) = r + s$.

Proof. Consider a complete bipartite graph $K_{r,s}$ with two partite sets A and B consisting of r and s vertices respectively. The vertices in A and B can be ordered as $v_1, v_2, v_3, ..., v_r$ and $u_1, u_2, u_3, ..., u_s$ respectively. Vertices in the different partite sets are distance one apart, therefore $\sum_{i=1}^{r} d(v_1, u_i) = s$. Similarly, for r vertices in A we get

$$\sum_{\substack{1\leq i\leq r \\ 1\leq j\leq s}} d(v_i, u_j) = rs. \tag{3}$$

Also, vertices in the same partite set are distance two apart, therefore, on considering partite set A we obtain

$$\left.\begin{aligned} \sum_{i=2}^{r} d(v_1, v_i) &= 2(r-1) \\ \sum_{i=3}^{r} d(v_2, v_i) &= 2(r-2) \\ &\vdots \\ \sum_{i=r-1}^{r} d(v_{r-2}, v_i) &= 2(2) \\ \sum_{i=r}^{r} d(v_{r-1}, v_i) &= 2(1) \end{aligned}\right\} \tag{4}$$

Adding all the equations in Expression (4),

$$\sum_{u,v\in V(A)} d(u, v) = 2\{1 + 2 + ... + (r-1)\} = r(r-1). \tag{5}$$

Similarly, for partite set B

$$\sum_{u,v\in V(B)} d(u,v) = 2\{1+2+...+(s-1)\} = s(s-1). \tag{6}$$

Adding Equations (3), (5), and (6), we obtain

$$\sum_{u,v\in V(G)} d(u,v) = rs + r(r-1) + s(s-1). \tag{7}$$

Given that the total number of vertices in $K_{r,s} = r+s$, and by the definition of average distance of graph, we get

$$\mu(K_{r,s}) = \frac{2(rs + r(r-1) + s(s-1))}{(r+s)(r+s-1)}. \tag{8}$$

Next, we claim that for $r, s \geq 1$, $\mu(K_{r,s}) \leq 2$.
If possible, let $\mu(K_{r,s}) > 2$.

$$\frac{2(rs + r(r-1) + s(s-1))}{(r+s)(r+s-1)} > 2$$

$$\implies \frac{2(r+s)(r+s-1) - 2(rs + r(r-1) + s(s-1))}{(r+s)(r+s-1)} < 0.$$

On simplification, we get $rs < 0$ which is a contradiction since r and s are greater than 0. Therefore, $\mu(K_{r,s}) \leq 2$.

Hence, it reduces the definition of average distance colouring as the function c from V to a set of non-negative integers such that for any $v \in V$, $|c(u) - c(v)| \geq 1$ for all u such that $d(u,v) \leq 2$. For a complete bipartite graph, every pair of vertices is either distance one or two apart which implies that the colour given to each vertex must be unique, giving $\chi_\mu(K_{r,s}) \geq r+s$. Further, this can be attained by using the following colouring c defined by $c(v_i) = i-1$ for $1 \leq i \leq r$ and $c(u_i) = c(v_r) + i$ for $1 \leq i \leq s$ given for graph $K_{r,s}$ with vertices of first and second partite set labelled as $v_1, v_2, ..., v_r$ and $u_1, u_2, ..., u_s$ respectively. The above function c gives $\chi_\mu(K_{r,s}) \leq r+s$,thus proving that $\chi_\mu(K_{r,s}) = r+s$.

□

For the next result, we consider a unicyclic graph $S_k + e$ obtained by adding a single edge between two pendant vertices of the star graph S_k shown in Figure 2.

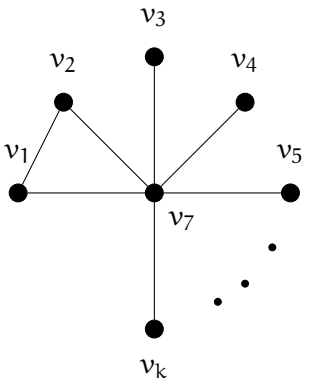


Figure 2: Graph $S_k + e$

Theorem 11 *For a graph obtained by joining the two pendant vertices of the star by an edge,* $S_k + e$, $\chi_\mu(S_k + e) = k + 1$.

Proof. Consider a graph $S_k + e$ with its vertices labelled as follows. Let the central vertex be labelled as v and the pendant vertices be labelled as $v_1, v_2, ..., v_k$ and the edge e is drawn between the vertices labelled v_{k-1} and v_k. The sum of the distance from vertex v_1 to other vertices given by t_1 is given by the following equation.

$$t_1 = 2(k-1) + 1. \tag{9}$$

Similarly, the sum of the distance from vertex v_2 to other vertices given by t_2 is

$$t_2 = 2(k-2) + 1. \tag{10}$$

On generalising Equations (9) and (10), we get the sum of distance from vertices $v_1, v_2, ..., v_{k-2}$ to other vertices denoted by t_m for $1 \leq m \leq k-2$ respectively. Therefore $\sum_{i=1}^{k-2} t_m = 2\left[(k-1) + (k-2) + ... + 2\right] + k - 2$ which can be further simplified to $\left(k^2 - 4\right)$.
Further, the sum of distance between the vertices of the triangle formed by vertices v, v_{k-1} and v_k will be 3. Therefore, the sum of distance between any two pair of vertices is

$$\sum_{u,v \in V} d(u,v) = k^2 - 1$$

Given that the total number of vertices in $S_k + e = k+1$, and by the definition of average distance of graph, we get

$$\mu(S_k + e) = \frac{2(k^2-1)}{k(k+1)}.$$

Next, we claim $\mu(S_k + e) \leq 2$. If possible, let $\mu(S_k + e) > 2$.

$$\implies \frac{2(k^2-1)}{k(k+1)} > 2$$
$$\implies k < k-1.$$

which is impossible. Therefore, $\mu(S_k+e) \leq 2$. Using the above inequation, the definition of average distance colouring reduces to the function c from V to set of non-negative integers such that for any $v \in V$, $|c(u)-c(v)| \geq 1$ for all u such that $d(u,v) \leq 2$. Since the diameter of the graph is 2, we know that each vertex should get a distinct colour to satisfy the given constraint for colouring giving $\chi_\mu(S_k+e) \geq k+1$. This can be obtained by using the function c which assigns integers to the vertices of the graph defined by $c(v_i) = i-1$ for the vertices of the graph labelled v_1 and $v_2, v_3, ..., v_{k+1}$ representing the central vertex and pendant vertices respectively with an edge drawn between each v_i for $2 \leq i \leq k+1$ and v_1. Also, there exists an edge between v_2 and v_3. Since the total number of vertices of the graph is $k+1$, $\chi_\mu(S_k+e) \leq k+1$. Hence the result.

□

On further increasing the number of partitions of vertices from two to r such that no two vertices in the same partition have an edge, we get a complete multipartite graph. In the next result, we obtain the value for χ_μ for a complete multipartite graph.

Theorem 12 *For a complete multipartite graph* $K_{m_1,m_2,...,m_r}$,

$$\chi_\mu(K_{m_1,m_2,...,m_r}) = m_1 + m_2 + ... + m_r.$$

Proof. Consider a complete multipartite graph G with r partite sets namely $A_1, A_2, ..., A_r$ such that $|A_i| = m_i$ for $1 \leq i \leq r$. Vertex in A_i partite set is given by $\{v_{i1}, v_{i2}, ..., v_{im_i}\}$ for $1 \leq i \leq r$.
Using the fact that the vertices in the same partite set are distance 2 apart, for the partite set A_1, we get

$$\sum_{u,v \in V(A_1)} d(u,v) = 2(1+2+...+(m_1-1)).$$

In general, the sum of distances between vertices belonging to the same partite set is given as

$$\sum_{u,v\in V(A_i)} d(u,v) = 2(1+2+...+(m_i-1)),\ \text{for } 1 \le i \le r.$$

Also, vertices belonging to different partite sets are distance 1 apart, therefore the sum of the distance from any vertex of A_1 to vertices of partite set $A_2, A_3, ...A_r$ is given by $m_1(m_2+m_3+m_4+...+m_r)$. Similarly, the sum of the distance from any vertex of A_i to vertices of other partite sets $A_{i+1}, A_{i+2}, ...A_r$ is given by $m_i(m_{i+1}+m_{i+2}+...+m_r)$.
Therefore, the sum of distances between vertices belonging to the same partite set for $K_{m_1,m_2,...,m_r}$ is given by (say s_1)

$$s_1 = m_1(m_1-1)+m_2(m_2-1)+...m_r(m_r-1) \tag{11}$$

and the sum of the distance between vertices taken from a different partite set is given by (say s_2)

$$s_2 = m_1(m_2+m_3+...+m_r)+m_2(m_3+m_4+...+m_r)+...+m_{r-1}m_r. \tag{12}$$

Adding Equations (11) and (12), we get

$$\begin{aligned}\sum_{u,v\in V(G))} d(u,v) = {} & m_1(m_1-1)+m_2(m_2-1)+...+m_r(m_r-1)\\ & + m_1(m_2+m_3+...+m_r)\\ & + m_2(m_3+m_4+...+m_r)+...+m_{r-1}m_r\end{aligned} \tag{13}$$

which can be simplified to

$$\begin{aligned}\sum_{u,v\in V(G))} d(u,v) = {} & (m_1^2+m_2^2+...+m_r^2)-(m_1+m_2+...+m_r)\\ & + \sum_{1\le i\le r-1} m_im_{i+1}+m_im_{i+2}+...+m_im_{i+(r-i)}.\end{aligned} \tag{14}$$

Since the number of vertices in a complete multipartite graph is $m_1+m_2+...+m_r$, we get

$$\begin{aligned}\mu(K_{m_1,m_2,...,m_r}) = {} & 2\frac{(m_1^2+m_2^2+...+m_r^2)-(m_1+m_2+...+m_r)}{(m_1+m_2+...+m_r)(m_1+m_2+....+m_r-1)}\\ & +2\frac{\sum_{i=1}^{r-1}(m_im_{i+1}+m_im_{i+2}+...+m_im_{i+(r-i)})}{(m_1+m_2+...+m_r)(m_1+m_2+....+m_r-1)}.\end{aligned}$$

We claim that for $r \geq 2$ and $n \geq 2$, value of $\mu(K_{m_1,m_2,...,m_r}) \leq 2$.
If possible, let $\mu(K_{m_1,m_2,...,m_r}) > 2$.
$\implies 2 - \mu(K_{m_1,m_2,...,m_r}) < 0$

$$2 - 2\frac{(m_1^2 + m_2^2 + ... + m_r^2) - (m_1 + m_2 + ... + m_r)}{(m_1 + m_2 + ... + m_r)(m_1 + m_2 + ... + m_r - 1)} + 2\frac{\sum_{i=1}^{r-1}(m_i m_{i+1} + m_i m_{i+2} + ... + m_i m_{i+(r-i)})}{(m_1 + m_2 + ... + m_r)(m_1 + m_2 + ... + m_r - 1)} < 0.$$

Since for all $1 \leq i \leq r$, $m_i \geq 1$ therefore, the denominator is always greater than 0. Therefore, multiplying both side by $(m_1 + m_2 + ... + m_r)(m_1 + m_2 + ... + m_r - 1)$, and expanding the summation we get
$2(m_1+m_2+...+m_r)(m_1+m_2+...+m_r-1)-2(m_1^2+m_2^2+...+m_r^2)+2(m_1+m_2+...+m_r)-2m_1(m_2+m_3+...+m_r)-2m_2(m_3+m_4+...m_r)-...-2m_{r-1}(m_r) < 0$.
On solving the above inequation, we get
$2(m_1+m_2+...+m_r)+m_2m_1+m_3m_1+m_3m_2+...+m_rm_1+m_rm_2+..+m_rm_{r-2} < 0$. Since each of m_i for $1 \leq i \leq r$ is greater than zero, the above in equation is not possible. Therefore, $\mu(K_{m_1,m_2,...,m_r}) \leq 2$.

The above arguments reduce the definition of average distance colouring as the function c from V to a set of non-negative integers such that for any $v \in V$, $|c(u) - c(v)| \geq 1$ for all u such that $d(u, v) \leq 2$. For a complete multipartite graph, every pair of vertices is either distance one or two apart, which implies that the colour given to each vertex must be unique giving $\chi_\mu(K_{m_1,m_2,...,m_r}) \geq m_1+m_2+...+m_r$. This bound can be achieved by colouring the vertices of the graph using the function c defined as follows $c(v_i) = i - 1$ for $1 \leq i \leq m_1 + m_2 + ... + m_r$ where the vertices of m_j^{th} partite set are labelled $v_{m_1+m_2+..+m_{j-1}+1}, v_{m_1+m_2+..+m_{j-1}+2}, ..., v_{m_1+m_2+..+m_{j-1}+m_j}$ for $2 \leq j \leq r$ and vertices of the first partite set are labelled $v_1, v_2, ..., v_{m_1}$ which gives $\chi_\mu(K_{m_1,m_2,...,m_r}) \leq m_1 + m_2 + ... + m_r$. Hence the result. □

For the next result, we consider double star $B_{s,t}$ such that $s \leq t$ with s and t number of pendant vertices as shown in Figure 3. In this case, the diameter of the graph considered for study has a diameter of three.

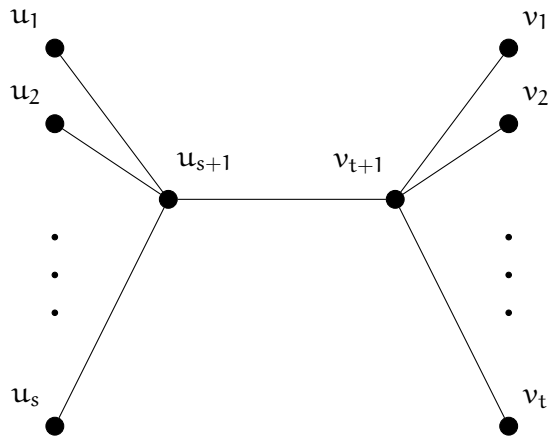


Figure 3: Double star $B_{s,t}$

Theorem 13 *For a double star* $B_{s,t}$*, with* s *and* t *number of pendant number of vertices,*

$$\chi_\mu(B_{s,t}) = \begin{cases} (t+2) \text{ for } \begin{cases} s = 1 \text{ and } t \geq 1 \text{ or } t = 1 \text{ and } s \geq 1 \\ s = 2 \text{ and } t = 3 \end{cases} \\ s + t + 2 \quad \text{otherwise.} \end{cases}$$

Proof. Consider a double star $B_{s,t}$ with vertices labelled as shown in Figure 3. Note that the structure of the graph is symmetric, we consider only one of the cases to prove the result by assuming $s \leq t$. Using the result of complete bipartite graph, we get the sum of distances between vertices $\{u_1, u_2, ..., u_s, u_{s+1}\}$ (say sum_1).

$$sum_1 = (s + s(s-1)) \tag{15}$$

Similarly, sum of distances between the vertices $\{v_1, v_2, ..., v_t, v_{t+1}\}$ (say sum_2) is given by the following equation.

$$sum_2 = (t + t(t-1)). \tag{16}$$

Also,

$$\sum_{1 \leq i \leq s} d(u_i, v_{t+1}) = 2s. \tag{17}$$

Similarly,

$$\sum_{1 \leq i \leq t} d(u_{s+1}, v_i) = 2t. \tag{18}$$

Also, note that

$$d(u_{s+1}, u_{t+1}) = 1 \tag{19}$$

.

Now, the distance between any two vertices each taken from set $\{u_1, u_2, ..., u_s\}$ and $\{v_1, v_2, ..., v_t\}$ is 3. This gives the following set of equations.

$$\left.\begin{aligned} \sum_{1\le i\le t} d(u_1, v_i) &= 3t \\ \sum_{1\le i\le t} d(u_2, v_i) &= 3t \\ &\vdots \\ \sum_{1\le i\le t} d(u_s, v_i) &= 3t \end{aligned}\right\} \tag{20}$$

Adding equations from (15) to (20) we get,

$$\sum_{u,v\in V(B_{s,t})} d(u, v) = s + s(s-1) + t + t(t-1) + 2(s+t) + 3st + 1.$$

Since the total number of vertices in double star $B_{s,t}$ is $s + t + 2$, we obtain

$$\mu(B_{s,t}) = \frac{2(s^2 + t^2 + 3st + 2s + 2t + 1)}{(s+t+2)(s+t+1)}.$$

Next, We examine the value of s and t for which $\mu(B_{s,t}) \le 2$.

$$\frac{2(s^2 + t^2 + 3st + 2s + 2t + 1)}{(s+t+2)(s+t+1)} \le 2$$

On simplification we get,

$$s + t + 1 \ge st$$

It is easy to verify that the above inequation holds true in the following two cases.

Case 1: When either s or t is equal to 1 and the other variable assumes any value greater than or equal to 1.

Case 2: When one of the variables is equal to two and the other is equal to three. To colour the double star for the above cases, we consider a double

star with vertices $u_1, u_2, ...u_{s+1}, v_1, v_2, ...v_{t+1}$ as shown in Figure 3. We know $\mu(B_{s,t}) \leq 2$ reducing the definition of average distance colouring as the function c from V to set of non-negative integers such that for any $v \in V$, $|c(u)-c(v)| \geq 1$ for all u such that $d(u,v) \leq 2$.

Subcase 1: When $s < t$.

We define a colouring c such that $c(v_{t+1}) = 0, c(v_i) = i$ for $1 \leq i \leq t$. Further, $c(u_{s+1}) = (t+1)$ given that $d(u_{s+1}, v_t) = 2$ and $c(u_i) = i$ for $1 \leq i \leq s$. This function gives the same set of colours to pendant vertices and since we know $s < t$, we have t distinct colours assigned. Also, v_{t+1} and u_{s+1} get different colour by using the above-defined function. Thus, $\chi_\mu \leq t + 2$.
Since double star consists of two stars with two non-pendant vertices joined by an edge, using Theorem 10, we require minimum $t + 2$ colours knowing $s < t$ giving $\chi_\mu \geq t + 2$. Thus, $\chi_\mu = t + 2$.

Subcase 2: When $s = t = 1$.

In this case, we get a P_4, which requires 3 distinct colours to colour it with average distance colouring protocol. This is attained by colouring four consecutive vertices with colours $1, 0, 2, 1$.

For the remaining values of s and t, $\mu(B_{s,t}) > 2$, which implies that we require $s + t + 2$ colours using Theorem 4. □

Further, we consider a graph obtained by joining k-copies of K_s with one common vertex termed windmill graph.

3 Conclusion

In this paper, we have considered the average distance of the graph which gives the approximate distance between any two vertices in the graph and introduce the concept of average distance colouring of graphs. We study average distance colouring number $\chi_\mu(G)$ for certain networks. We have already worked on the condition where a graph would require n distinct colours for it to admit average distance colouring. It would be interesting to identify the bound for k characterise graphs with $\chi_\mu(G) = k$ where k is any positive integer.

References

[1] F. Harary, *Graph Theory*, Addison Wesley, Reading, Massachusetts, 1969. ⇒ 205

[2] W. Goddard and O. R. Oellermann, *Structural Analysis of Complex Networks*: 49-72, 2011. ⇒206, 208

[3] H. Wiener, Structural determination of paraffin boiling points, *J. Am. Chem. Soc.* (1947), 69, 1, 17–20. ⇒205

[4] F. Kramer and H. Kramer, A survey on the distance-colouring of graphs, *Discrete Mathematics*, vol. 308, no. 2, pp. 422–426, 2008. ⇒206

[5] P. K. Niranjan and S. R. Kola, The k-distance chromatic number of trees and cycles, *AKCE International Journal of Graphs and Combinatorics*, vol. 16(2), 2019 ⇒206

[6] L. March and P. Steadman, *The Geometry of Environment: An introduction to spatial organization in design*, M.I.T. Press, Cambridge, Mass., 1974. ⇒205

Received: July 24, 2023 • Revised: October 4, 2023

Acta Univ. Sapientiae Informatica **15,** 2 (2023) 221–238

DOI: 10.2478/ausi-2023-0015

Computing closeness for some graphs

Hande TUNÇEL GÖLPEK
Dokuz Eylul University Maritime Faculty
Buca Izmir, Turkey
email: hande.tuncel@deu.edu.tr

Aysun AYTAÇ
Ege University Department of Mathematics
Bornova Izmir, Turkey
email: aysun.aytac@ege.edu.tr

Abstract. The analysis of networks involves several crucial parameters. In this paper, we consider the closeness parameter, which is based on the total distance between every pair of vertices. Initially, we delve into a discussion about the applicability of the closeness parameter to Mycielski graphs. Our findings are categorized based on the underlying graph's diameter. The formula for calculating the closeness of a Mycielski graph is derived for graphs with a diameter of less than or equal to 4. Furthermore, we establish a sharp lower bound for the closeness of a Mycielski graph when the diameter of the underlying graph is greater than 4. To achieve this, the closeness of the Mycielski transformation of a path graph plays an important role. Additionally, leveraging the obtained results, we examine the closeness of a special planar construction composed of path and cycle graphs, as well as its Mycielski transformation.

1 Introduction

Network science has evolved greatly over the past decade and is now the leading scientific field in the description of complex networks. Therefore, the complex network is a significant research area of complexity science.
Recently, due to the construction of smart cities, complex network applications have been gaining popularity. Complex networks such as traffic networks,

Key words and phrases: graph vulnerability, closeness, Mycielski graph, Tadpole graph

power grids, social networks, and others can now be observed ubiquitously. These networks bring significant simplicity to our lives. As a result, complex networks, as a novel and dynamic field of scientific research, are increasingly capturing people's attention. They draw substantial inspiration from experimental studies conducted on real-world networks.

Graph theory emerges as an invaluable instrument for deciphering complex networks. By translating network structures into graphs, this theory offers an intuitive and streamlined representation. This renders graph theory a widely adopted tool across contemporary sciences, facilitating the modeling and resolution of real-life quandaries. [13, 20, 24–26, 28].

In a complex network that composed of processing nodes and communication links, it is very important for a network designer to determine which vertices or edges are important. Hereby, centrality is a critical metric because it indicates which vertex is in a sensitive location in an entire network. It has also been widely used in complex network analysis. If we think of a graph as modeling a network, there are many centrality parameters such as closeness centrality, degree centrality, vertex and edge betweenness centrality, residual closeness and etc. which are used to determine the importance of a vertex or an edge in the network including.

The purpose of centrality measures, as closeness or betweenness, determines how centrally a vertex is in a network. There are many studies in the literature on the rapid calculation about centrality index especially on issues related to the solution and calculation of application problems such as social networks, network analysis and determining the best location [14, 16, 18, 19].

Closeness centrality, one of the most studied parameter of complex network through centrality indexes, is applied much from many researchers. The closeness of a vertex is the sum the distances from all other vertices, where the distance from a vertex to another is defined as the length of the shortest path between them. The closeness centrality based on the shortest paths among vertices in the network and it relates how quickly information can spread across the network. If spread of information from one vertex to other vertices can occur rapidly, that node is based on the intuition that it is in an important position.

The closeness centrality takes value between 0 and 1. If closeness value of a vertex approaches to 0 this means indicated vertex far from others. While closeness value of a vertex approaches to 1 this means addressed node is in close proximity to all other vertices.

Closeness centrality concept was first defined in 1948 by Bavelas [5]. Then, a notable definition for closeness defined by Freeman yet it can be utilized solely for connected graphs [12]. After that, Latora and Marchiori [15] provided new definition for point closeness even it can be applied to disconnected graphs. Later, Danglachev introduced a modified closeness definition due to ease of calculation and formulation [8]. Furthermore, Danglachev defined another measure of vulnerability parameter, called as residual closeness. We refer the readers to references about closeness and its varieties in order to get detailed knowledge [1–3, 9, 27].

In this work, we will use Danglachev's closeness parameter. In this definition, the closeness of a graph is defined as: Dangalchev introduced closeness of a vertex definition as $C(u_i) = \sum_{j \neq i} \frac{1}{2^{d(u_i,u_j)}}$ and closeness of the graph is defined as $C = \sum_i C(u_i)$ where $d(u_i, u_j)$ denotes *the distance between two vertices* u_i and u_j is shortest path between them.

In this paper, let G be simple, finite and undirected graph with vertex set $V(G)$ and edge set $E(G)$. *The open neighborhood*of any vertex in $V(G)$, denoted by $N_G(v) = \{u \in V(G) : (uv) \in E(G)\}$. Also, $deg(u_i)$ denotes *the degree of a vertex* u_i that is cardinality of its neighborhood. *The diameter of* G is largest distance between two vertices in $V(G)$ and represented by $diam(G)$. The complement $\bar{G}$ of a graph G has $V(G)$ as its vertex sets, but two vertex are adjacent in $\bar{G}$ if only if they are not adjacent in G [6, 7].

The goal of this paper is provide exact formula and sharp lower bound for a Mycielski graph depending on diameter of underlying graph. Mycielski introduced a graph structure that does not contain triangles with large chromatic number. The Mycielski structure, denoted by $\mu(G)$ notation, is defined for a graph $G = (V, E)$ with the vertex set $V(\mu(G)) = V(G) \cup V(G') \cup \{v\}$ where $V(G) = \{v_i : 1 \leq i \leq n\}$ is vertex set of G and $V(G') = \{u_i : 1 \leq i \leq n\}$ is copy of the vertex set $V(G)$ and $E(\mu(G)) = E(G) \cup \{v_i u_j : v_i v_j \in E(G)\} \cup \{u_j v : \forall u_j \in V(G')\}$ [17] (see in Fig. 1). Recently, there has been an increasing interest in studies related to the Mycielski graph and there are many the research papers

in the literature about mycielski structures.

For our study, in order to obtain the lower bound of closeness of Mycielski

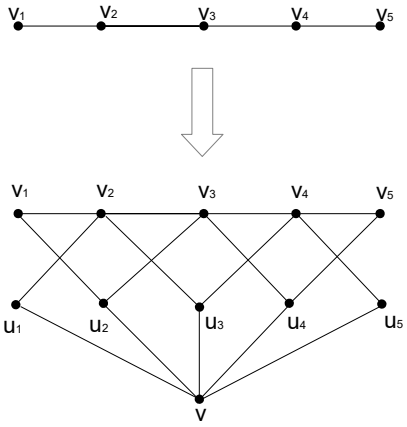


Figure 1: An illustration of a Mycielski graph.

graph, we establish a relationship with the path structure has been the basis. Therefore, first closeness of path mycielskian is provided. Furthermore, we consider Tadpole graph, a construction containing a path. We investigate some results about Tadpole graph and its Mycielski form. In literature, there are some findings about splitting graphs and analogous structure of Mycielski graph [4, 10, 22, 23]. As well as verifying some known basic results with our formula, we also present new general conclusions about closeness of Mycielski graph. Now we state some known lemmas which we use in the proofs of our results.

Theorem 1 [1, 8] *The closeness of*

(a) the complete graph K_n *with* n *vertices is* $C(K_n) = \dfrac{n(n-1)}{2}$;

(b) the star graph S_n *with* n *vertices is* $C(S_n) = \dfrac{(n-1)(n+2)}{4}$;

(c) the path P_n *with* n *vertices is* $C(P_n) = 2n - 4 + \dfrac{1}{2^{n-2}}$;

(d) the cycle C_n *with* n *vertices is* $C(C_n) = \begin{cases} 2n(1 - \frac{1}{2^{[n/2]}}) & \text{if } n \text{ is odd} \\ n(2 - \frac{3}{2^{n/2}}) & \text{if } n \text{ is even} \end{cases}$.

Theorem 2 [22] *The closeness of* $\mu(G)$

(a)For $n \geq 4$*; the star graph* $G = S_n$ *is* $C(\mu(G)) = (2(n)^2 + 5n - 3)/2$.

(b)For $n \geq 3$*; the complete graph* $G = K_n$ *is* $C(\mu(G)) = (7n^2 + n)/4$.
(c)For $n \geq 8$*; the cycle graph* $G = C_n$ *is* $C(\mu(G)) = (9n^2 + 77n)/16$.

2 Results about Closeness of Mycielski Graph

In [10] Dangalchev has expressed closeness of splitting graph of G in terms of closeness of G. Analogously, we can apply this process to obtain results about closeness of Mycielski graph depending on diameter of G.

Theorem 3 *Let* G *be* n *order graph and* $\mathrm{diam}(G) \leq 4$. *Then,*

$$C(\mu(G)) = 3C(G) + \frac{n^2 + 7n}{4}.$$

Proof. *To derive the closeness formula of a Mycielski graph with a diameter less than 4, the vertices of the graph can be partitioned into five distinct parts:*

$$\begin{aligned} C(\mu(G)) &= \sum_{i=1}^{n} \sum_{j \neq i} 2^{-d(v_i, v_j)} + 2\sum_{i=1}^{n} \sum_{j=1}^{n} 2^{-d(u_i, v_j)} \\ &+ \sum_{i=1}^{n} \sum_{j \neq i} 2^{-d(u_i, u_j)} + 2\sum_{i=1}^{n} 2^{-d(v, u_i)} + 2\sum_{i=1}^{n} 2^{-d(v, v_i)} \\ &= C(G) + 2\sum_{i=1}^{n} 2^{-d(u_i, v_i)} + 2\sum_{i=1}^{n} \sum_{j \neq i} 2^{-d(u_i, v_j)} + \frac{n(n-1)}{4} + 2 \cdot \frac{n}{2} + 2 \cdot \frac{n}{4} \end{aligned}$$

Since, $d(u_i, u_j) = 2$, $d(v, u_i) = 1$ *and* $d(v, v_j) = 2$. *Also,* $d(u_i, u_j) = 2$. $d(u_i, v_j) = d(v_i, v_j)$ *then* $\sum_{i=1}^{n} \sum_{j \neq i} 2^{-d(u_i, v_j)} = \sum_{i=1}^{n} \sum_{j \neq i} 2^{-d(v_i, v_j)} = C(G)$

$$\begin{aligned} &= 3C(G) + 2\frac{n}{4} + \frac{n(n-1)}{4} + n + \frac{n}{2} \\ &= 3C(G) + \frac{n^2 + 7n}{4}. \end{aligned}$$

□

Utilizing previous result we can express the closeness for some special graphs. Also, the results can be compared by formulae in [22] and it can be validated.

Corollary 4 *The Mycielski graph* $\mu(S_n)$ *of star graph* S_n *has closeness* $C(\mu(S_n)) = \frac{2n^2+5n-3}{2}$ *which is proven in* [22].

Proof. *It is known that closeness of star graph*

$$C(S_n) = \frac{(n-1)(n+2)}{4}$$

from [8], *we have*

$$\begin{aligned} C(\mu(S_n)) &= 3C(S_n) + \frac{n^2+7n}{4} \\ &= 3 \cdot \frac{(n-1)(n+2)}{4} + \frac{n^2+7n}{4} \\ &= \frac{2n^2+5n-3}{2}. \end{aligned}$$

□

Corollary 5 *The closeness of Mycielski complete graph which is proved in* [22] *is*

$$C(\mu(K_n)) = \frac{7n^2+n}{4}.$$

Proof. *The closeness of complete graph* K_n *is known from* [8]

$$C(K_n) = \frac{n(n-1)}{2}$$

Then, we get

$$\begin{aligned} C(\mu(K_n)) &= 3C(K_n) + \frac{n^2+7n}{4} \\ &= 3 \cdot \frac{n(n-1)}{2} + \frac{n^2+7n}{4} \\ &= \frac{7n^2+n}{4}. \end{aligned}$$

□

Corollary 6 *The closeness of double star* $S_{m,n}$ *is*

$$C(S_{m,n}) = \frac{n^2+5n+m^2+5m+mn+4}{4}.$$

Proof. *A double star, $S_{m,n}$, can be obtained by joining two star graphs $K_{1,m}$ and $K_{1,n}$ with an edge. Let, v and w be two non pendant vertices whose degrees are $\deg(v) = m+1$ and $\deg(w) = n+1$, respectively. Then, v adjacent to m pendant vertices and w adjacent to n pendant vertices also w and v are adjacent. In addition, pendant vertices in $K_{1,m}$ and $K_{1,n}$ are three distances away also those are 2 distances in themselves. Therefore,*

$$\begin{aligned}
C(S_{m,n}) &= 2\sum_{i=1}^{m}\frac{1}{2} + 2\sum_{i=1}^{n}\frac{1}{2} + 2\sum_{i=1}^{n}\frac{1}{2^2} + 2\sum_{i=1}^{m}\frac{1}{2^2} + \sum_{i=1}^{n}\sum_{j=1}^{n-1}\frac{1}{2^2} + \sum_{i=1}^{m}\sum_{j=1}^{m-1}\frac{1}{2^2} \\
&\quad + 2\cdot\frac{1}{2} + 2\sum_{i=1}^{n}\sum_{j=1}^{m}\frac{1}{2^3} \\
&= m+n+\frac{m}{2}+\frac{n}{2}+\frac{n(n-1)}{4}+\frac{m(m-1)}{4}+1+\frac{mn}{4} \\
&= \frac{n^2+5n+m^2+5m+mn+4}{4}.
\end{aligned}$$

□

Since, diameter of double graph is 3, closeness of Mycielski graph of double star can be constructed using Theorem 1.

Corollary 7 *Let W_n be wheel graph with n vertices. The closeness value of W_n is $C(W_n) = \dfrac{(n-1)(n+4)}{4}$.*

Proof. *Let $V(W_n) = \{1, ..., n\}$ be vertex set and 1 be center vertex with $\deg(1) = n-1$:*

$$\begin{aligned}
C(W_n) &= 2\sum_{i=1}^{n-1}\frac{1}{2^{-d(1,i)}} + \sum_{i=2}^{n-1}\sum_{\substack{i\sim j,\\ j\neq 1}}\frac{1}{2^{-d(i,j)}} + 2\sum_{\substack{i\nsim j\\ i,j\neq 1}}\frac{1}{2^{-d(i,j)}} \\
&= 2(1(n-1)\frac{1}{2}) + 2\cdot\frac{1}{2}(n-1) + 1\cdot(n-4)\frac{1}{4}(n-1) \\
&= \frac{(n-1)(n+4)}{4}
\end{aligned}$$

where $d(1,i) = 1$, the notation $i \sim j$ refers that i is adjacent to j. □

Corollary 8 *Let $K_{m,n}$ be complete bipartite graph. The closeness value of K_n is $C(K_{m,n}) = \frac{1}{4}((m+n)^2 - (m+n) + 2mn)$.*

Proof. *Let* $V(K_{m,n}) = \{1, 2, ..., m, ...m + n\}$ *be vertex labeling and* $|V_1| = m$ *and* $|V_2| = n$ *be two subset of vertices such that no edge has both endpoints in the same subset:*

$$\begin{aligned} C(K_{m,n}) &= 2\sum_{i=1}^{m}\sum_{j=m+1}^{m+n}\frac{1}{2^{d(i,j)}} + \sum_{i=1}^{m}\sum_{\substack{j=1\\ i\neq j}}^{m}\frac{1}{2^{d(i,j)}} + \sum_{i=m+1}^{m+n}\sum_{\substack{j=m+1\\ i\neq j}}^{m+n}\frac{1}{2^{d(i,j)}} \\ &= mn + \frac{m(m-1)}{4} + \frac{n(n-1)}{4} \\ &= \frac{1}{4}((m+n)^2 - (m+n) + 2mn). \end{aligned}$$

□

Corollary 9 *Closeness of Mycielski Double Star, complete bipartite and wheel graphs are*

$$\begin{aligned} C(\mu(S_{m,n})) &= 3C(S_{m,n}) + \frac{(m+n+2)^2 + 7(m+n+2)}{4} \\ C(\mu(K_{m,n})) &= 3C(K_{m,n}) + \frac{n^2+7n}{4} \\ C(\mu(W_n)) &= \frac{3(n^2+5n-2)}{2}. \end{aligned}$$

Proof. *The results can be obtained from Theorem 3 and previous corollaries about* $C(S_{m,n})$, $C(K_{m,n})$ *and* $C(W_n)$. □

Theorem 10 *Let* G *be* n *order graph and* $\mathrm{diam}(G) = k > 4$. *Then,*

$$C(\mu(P_{k+1})) \leq C(\mu(G)).$$

Proof. *Let* $\mathrm{diam}(G) = k$, *then the lower bound can be found from* P_{k+1}. *Since, a k-diameter graph includes at least one* P_{k+1}.*Therefore, total closeness value of Mycielski form of a k-diameter graph will be more than closeness value of Mycielski of path graph denoted by* $C(\mu(P_{k+1}))$. *Thus, we have* $C(\mu(P_{k+1})) \leq C(\mu(G))$. □

So, it is necessary to formulate $C(\mu(P_{k+1}))$ value in order to supply sharp lower bound for closeness of Mycielski graph of P_n.

Corollary 11 *The closeness of Mycielski graph path graph* $\mu(P_n)$ *for* $\mathtt{diam}(P_n) = k > 4$ *is*

$$C(\mu(P_n)) = \frac{7n^2 + 91n - 96}{16}.$$

Proof. *In order to calculate closeness of Mycielski graph path graph* $\mu(P_n)$ *for* $\mathtt{diam}(P_n) = k > 4$*, relationship between vertices can be divided into five parts:*

$$\begin{aligned} C(\mu(P_n)) &= \sum_{i=1}^{n}\sum_{j\neq i} 2^{-d(v_i,v_j)} + 2\sum_{i=1}^{n}\sum_{j=1}^{n} 2^{-d(u_i,v_j)} + \sum_{i=1}^{n}\sum_{j\neq i} 2^{-d(u_i,u_j)} \\ &\quad + 2\sum_{i=1}^{n} 2^{-d(v,u_i)} + 2\sum_{i=1}^{n} 2^{-d(v,v_i)} \\ &= 3\sum_{i=1}^{n}\sum_{j\neq i} 2^{-d(v_i,v_j)} + \frac{n^2+7n}{4} \end{aligned}$$

In the Mycielski Graph for G *whose diameter is greater than 4, the value of* $3\sum_{i=1}^{n}\sum_{j\neq i} 2^{-d(v_i,v_j)}$ *is greater than* $3C(G)$*. Since,* $\mathtt{diam}(\mu(G)) = 4$*, and the value of* $2^{-d(v_i,v_j)}$ *in* $C(G)$ *is less than* 2^{-4} *for some pair of vertices. In order to form* $3\sum_{i=1}^{n}\sum_{j\neq i} 2^{-d(v_i,v_j)}$*, let define a set for* P_n *that contains pair of vertices whose distance greater than 4 and the set denoted by* E_{5^+}*.*

$$E_{5^+} = \{(v_i, v_j) : \ |v_i - v_j| \geq 5, \ v_i, v_j \in V(P_n)\}$$

Then $|E_{5^+}| = (n-5)(n-4)$*. The value of* $\sum_{i=1}^{n}\sum_{j\neq i} 2^{-d(v_i,v_j)}$ *increases in the summation of* $C(\mu(P_n))$*, due to the diameter of Mycielski graph. In* $C(P_n)$*, the value* $2\sum_{i=1}^{n-5} \frac{i}{2^{n-i}}$ *that comes from vertices of* E_{5^+} *will be turn into* $\frac{|E_{5^+}|}{16}$*. Therefore, we get*

$$C(\mu(P_n)) = 3(C(P_n) - 2\sum_{i=1}^{n-5} \frac{i}{2^{n-i}} + \frac{(n-5)(n-4)}{16}) + \frac{n^2+7n}{4}.$$

To calculate the summation, we are going to use geometric summation formula as below:

$$\sum_{i=1}^{n} X^{i-1} = 1 + X + X^2 + ... + X^{n-1} = \frac{X^n - 1}{X - 1}$$

and also differentiating both side of geometric sum, we have

$$\sum_{i=1}^{n}(i-1)X^{i-2} = 1 + 2X + ... + (n-1)X^{n-2} = \frac{nX^{n-1}}{X-1} - \frac{X^n - 1}{(X-1)^2}.$$

Then substitute 2 into the X, we get

$$\sum_{i=1}^{n-5}\frac{i}{2^{n-i}} = \frac{2}{2^n}\sum_{i=1}^{n-5} i \cdot 2^{i-1} = \frac{1}{2^{n-1}}((n-4)2^{n-5} - 2^{n-4} + 1) \tag{1}$$

Using $C(P_n) = 2n - 4 + \frac{1}{2^{n-2}}$ *[8] and the equation 1*

$$C(\mu(P_n)) = \frac{7n^2 + 91n - 96}{16}$$

is obtained. □

Theorem 12 *Let* G *be* n *order graph and* $diam(G) = k > 4$*. Then,*

$$\frac{7(k+1)^2 + 91(k+1) - 96}{16} \leq C(\mu(G)).$$

Proof. *It can be referred from Theorem 10 and Corollary 11.* □

2.1 Results about Tadpole graph

In previous section, we have obtained result for Mycielski graph of G ,whose diameter greater than 4, based on the Mycielski of path graph. In this section, we will investigate results about Tadpole graph and its Mycielski form. Tadpole graph is special planar graph which contains path and cycle graphs as a subgraph. Therefore, results will be benefited from closeness of path and cycle graphs.

Definition 13 *Tadpole graph, denoted by* $T_{n,m}$*, is a graph obtained by identifying a vertex of the cycle graph* C_n *with a pendant vertex of the path graph* P_m*. An example of the illustration of the Tadpole graph can be seen in Figure 2. Truszczynski called these graphs as Dragon [21] and Koh et. al called these forms as Tadpole graphs [11].*

Theorem 14 *Let* $T_{n,m}$ *be a Tadpole graph contains* C_n *and* P_m*. Closeness of Tadpole graph in terms of n is:*

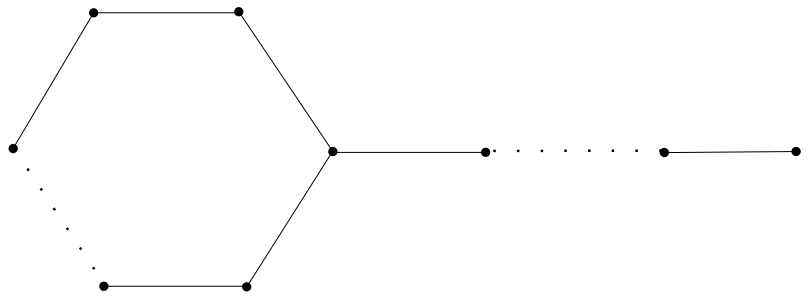


Figure 2: An illustration of a Tadpole graph.

- *if n is odd:*

$$C(T_{n,m}) = 2n(1 - \frac{1}{2^{[n/2]}}) + (2m - 4 + \frac{1}{2^{m-2}}) + 2(2 - 2^{2-\frac{n+1}{2}})(1 - (\frac{1}{2})^{m-1})$$

- *if n is even:*

$$C(T_{n,m}) = 2n(1 - \frac{1}{2^{[n/2]}}) + (2m - 4 + \frac{1}{2^{m-2}}) + 2(2 - 2^{2-\frac{n}{2}} + 2^{-n/2})(1 - (\frac{1}{2})^{m-1}).$$

Proof. *Closeness of* $T_{n,m}$ *can be think as three parts. Closeness of* C_n *and closeness of* P_m *and closeness value which comes from relationship between vertices in* C_n *and vertices in* P_m*, let it be denoted by* $C(C_n - P_m)$

$$\begin{aligned} C(T_{n,m}) &= C(C_n) + C(P_m) + 2C(C_n - P_m) \\ &= 2n(1 - \frac{1}{2^{[n/2]}}) + (2m - 4 + \frac{1}{2^{m-2}}) + 2C(Cn - P_m). \end{aligned}$$

Closeness of P_m *and* C_n *are known [8]. It is need to find* $C(C_n - P_m)$*. Assume that,* v_1 *is a vertex as intersection point of* C_n *and* P_m*. Let divide* C_n *into exactly two pieces. However, form of division depends on whether* n *is odd or even.*

Case 1: Let n *be even and labeling of* C_n *be* $\{v_1, v_2, ..., v_n\}$*. Therefore, the closeness of* v_1 *in* C_n *can be calculated as*

$$2\sum_{i=2}^{n/2} \frac{1}{2^{i-1}} + \frac{1}{2^{n/2}}.$$

Since, there are $(n-2)/2$ *symmetric vertices in* C_n *whose distance from* v_1 *to* v_i *can be calculated as* $(i-1)$ *and there is one vertex whose distance from* v_1 *is* $n/2$.

Also, let vertices of P_m *be labeled as* $\{v_1, v_{\bar{2}}, ..., v_{\bar{m}}\}$. *Distance of* $v_{\bar{j}}$, $j = 2, ...m$, *to* v_1 *equal to*

$$\frac{1}{2^{j-1}}(2\sum_{i=2}^{n/2}\frac{1}{2^{i-1}} + \frac{1}{2^{n/2}}).$$

In general

$$\sum_{j=1}^{m-1}\frac{1}{2^j}((2\sum_{i=2}^{n/2}\frac{1}{2^{i-1}} + \frac{1}{2^{n/2}})) = 2(2 - 2^{2-(n/2)} + 2^{-n/2})(1 - \frac{1}{2^{m-1}}).$$

Case 2: Similarly it can be done for odd n *values. Using same labeling as in Case 1, the closeness of* v_1 *in* $C_n$is :

$$2\sum_{i=2}^{n+1/2}\frac{1}{2^{i-1}}.$$

Because of this, C_n *can be divided into exact two equal part and distance from* v_1 *to* $v_i \in V(C_n)$ *can be calculated as* $(i-1)$.

Also, distance of $v_{\bar{j}}$, $j = 2, ...m$, *to* v_1 *equal to*

$$\sum_{j=1}^{m-1}\frac{1}{2^j}.(2\sum_{i=2}^{n+1/2}\frac{1}{2^{i-1}}) = 2(2 - 2^{2-\frac{n+1}{2}})(1 - (\frac{1}{2})^{m-1}).$$

Therefore, we have

- *if* n *is odd:*

$$C(T_{n,m}) = 2n(1 - \frac{1}{2^{[n/2]}}) + (2m - 4 + \frac{1}{2^{m-2}}) + 2(2 - 2^{2-\frac{n+1}{2}})(1 - (\frac{1}{2})^{m-1})$$

- *if* n *is even:*

$$C(T_{n,m}) = 2n(1 - \frac{1}{2^{[n/2]}}) + (2m - 4 + \frac{1}{2^{m-2}}) + 2(2 - 2^{2-\frac{n}{2}} + 2^{-n/2})(1 - (\frac{1}{2})^{m-1}).$$

□

In previous results, we had talked about closeness value of $\mu(P_n)$ and special variant of P_n and C_n, named Tadpole graph. Similarly, we can ready to investigate Mycielski of Tadpole graph using the closeness result of $\mu(P_n)$ and $T_{n,m}$. Mycielski of Tadpole graph $T_{n,m}$ has $2(n+m)+1$ vertices and its diameter always equal to 4 regardless from $\mathtt{diam}(T_{n,m})$. However, $C(\mu(T_{n,m}))$ should be taken hand according to diameter of $T_{n,m}$.

Let $V(\mu(T_{n,m}))$ be vertex set of $T_{n,m}$ including $V(T_{n,m})$, $V(T'_{n,m})$ and w.

Theorem 15 *Let $T_{n,m}$ be a Tadpole graph contains C_n and P_m and $\mathtt{diam}(T_{n,m}) < 4$. Closeness of Mycielski of Tadpole graph is*

$$C(\mu(T_{n,m})) = 3C(T_{n,m}) + \frac{(n+m-1)^2 + 7(n+m-1)}{4}.$$

Proof. *It can be acquired from Theorem 3.* □

Before giving result about $C(\mu(T_{n,m}))$ when $\mathtt{diam}(T_{n,m}) > 4$, some useful findings will be investigated in order to get rid of expressional burden of $C(\mu(T_{n,m}))$. In Mycielski graph, closeness value of some vertex pairs turns to $\frac{1}{2^4}$ due to form of it. In order to calculate the value of $C(\mu(G))$ when $\mathtt{diam}(G) > 4$, the closeness value of vertex pair with distance 5 or more should be removed from $C(G)$ and $\frac{1}{2^4}$ should be added as the number of subtracted value instead.

Let total closeness value of the pair of vertices in $T_{n,m}$ whose distance between them greater than 4 be excess closeness, denoted by $C_{ex}(T_{n,m})$ and the number of the vertex pair that has closeness value smaller than $\frac{1}{2^4}$, denoted by $|V_{ex}(T_{n,m})|$.

Once it comes to calculating $C_{ex}(T_{n,m})$ and $|V_{ex}(T_{n,m})|$, $T_{n,m}$ can be thought as being divided into two parts as upper and lower as illustrated in the Figure 3. Then, it can be examine in two cases.

Case 1: If n is odd then there is a path having $m + \frac{n-1}{2}$ vertices on upper side. Thus, $(m+\frac{n-1}{2}-5)(m+\frac{n-1}{2}-4)$ pair of vertices with distance 5 or more comes from the upper part. Because of repeated pair of vertices, the lower part can be evaluated as

$$2.\sum_{i=1}^{(n-1)/2}(m+i-5) = (m-5)(n-1) + (\frac{n^2-1}{2}).$$

Because of this, there are $(n-1)/2$ vertices in lower part of cycle whose distance can de greater or equal to 5 to vertices in P_m. Hence, if n is odd;

$$|V_{ex}(T_{n,m})| = (m+\frac{n-1}{2}-5)(m+\frac{n-1}{2}-4) + (m-5)(n-1) + (\frac{n^2-1}{2}).$$

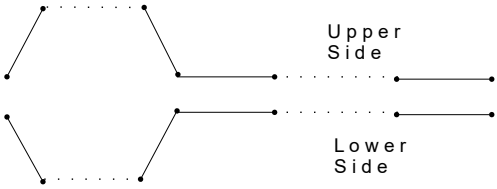


Figure 3: A Tadpole graph divided into two parts

According to proof of Corollary 11, total closeness value of the pair of vertices in P_k whose distance greater or equal than 5 had been calculated as

$$2\sum_{i=1}^{k-5} \frac{i}{2^{k-i}} = \frac{2}{2^{k-1}}((k-4)2^{k-5} - 2^{k-4} + 1).$$

Then, substitute $m + \frac{n-1}{2}$ into k:

$$\frac{2}{2^{m+\frac{n-1}{2}-1}}((m + \frac{n-1}{2} - 4)2^{m+\frac{n-1}{2}-5} - 2^{m+\frac{n-1}{2}-4} + 1) \tag{2}$$

obtained from upper side. In order to hinder repeated value coming from lower part, it should be subtracted the value of $C_{ex}(P_m)$ from the value in equation 2

$$C_{ex}(P_m) = \frac{1}{2^{m-2}}((m-4)2^{m-5} - 2^{m-4} + 1).$$

Therefore, we have

$$C_{ex}(T_{n,m}) = \frac{1}{2^{m+\frac{n-1}{2}-3}}((m + \frac{n-1}{2} - 4)2^{m+\frac{n-1}{2}-5} - 2^{m+\frac{n-1}{2}-4} + 1).$$

$$-\frac{1}{2^{m-2}}((m-4)2^{m-5} - 2^{m-4} + 1)$$

Case 2: If n is even then there is a path having $m + \frac{n}{2}$ vertices. Thus, $(m + \frac{n}{2} - 5)(m + \frac{n}{2} - 4)$ pair of vertices with distance 5 or more comes from

the upper part. As in case 1, if we consider the repetitions as:

$$|V_{ex}(T_{n,m})| = (m + \frac{n}{2} - 5)(m + \frac{n}{2} - 4) + (m-5)(n-2) + (\frac{n(n-2)}{2}).$$

Since n is even, we cannot divide C_n into two equal parts. Thus, we get

$$\begin{aligned} C_{ex}(T_{n,m}) &= C_{ex}(P_{m+\frac{n}{2}}) + C_{ex}(P_{m+\frac{n-2}{2}}) - C_{ex}(P_m) \\ &= \frac{1}{2^{m+\frac{n}{2}-2}}((m + \frac{n}{2} - 4)2^{m+\frac{n}{2}-5} - 2^{m+\frac{n}{2}-4} + 1) \\ &+ \frac{1}{2^{m+\frac{n-2}{2}-2}}((m + \frac{n-2}{2} - 4)2^{m+\frac{n-2}{2}-5} - 2^{m+\frac{n-2}{2}-4} + 1) \\ &- \frac{1}{2^{m-2}}((m-4)2^{m-5} - 2^{m-4} + 1). \end{aligned}$$

Theorem 16 *Let $T_{n,m}$ be a Tadpole graph contains C_n and P_m and $\mathtt{diam}(T_{n,m}) > 4$. Closeness of Mycielski of Tadpole graph is*

$$C(\mu(T_{n,m})) = 3(C(T_{n,m}) - C_{ex}(T_{n,m}) + \frac{|V_{ex}(T_{n,m})|}{2^4}) + \frac{(n+m-1)^2 + 7(n+m-1)}{4}.$$

Proof. *Let $V(\mu(T_{n,m})) = \{V(T_{n,m}), V(T'_{n,m}), w\}$ where copy of Tadpole graph denoted by $T'_{n,m}$. According to form of Mycielski graph, it is known that $\mathtt{diam}(\mu(T_{n,m})) = 4$. However, the diameter of $T_{n,m}$ is greater than 4 in this case. Thus, we have*

$$3C(T_{n,m}) + \frac{(n+m-1)^2 + 7(n+m-1)}{4} < C(\mu(T_{n,m})).$$

Let v_i, v_j be vertices in $T_{n,m}$ provided that distance between them greater than 4 in $T_{n,m}$. Therefore, the value of $2^{-d(v_i,v_j)}$ turns into 2^{-4} in the $\mu(T_{n,m})$. It is also valid for copy vertices u_i, u_j in $T'_{n,m}$.

$$\begin{aligned} C(\mu(T_{n,m})) &= \sum_{i=1}^{m+n-1} \sum_{j\neq i} 2^{-d(v_i,v_j)} + 2\sum_{i=1}^{m+n-1} \sum_{j=1}^{m+n-1} 2^{-d(u_i,v_j)} \\ &+ \sum_{i=1}^{m+n-1} \sum_{j\neq i} 2^{-d(u_i,u_j)} + 2\sum_{i=1}^{m+n-1} 2^{-d(w,u_i)} + 2\sum_{i=1}^{m+n-1} 2^{-d(w,v_i)} \\ &= 3\sum_{i=1}^{m+n-1} \sum_{j\neq i} 2^{-d(v_i,v_j)} + \frac{(n+m-1)^2 + 7(n+m-1)}{4} \end{aligned}$$

Since, the distance $d(v_i, v_j) = d(u_i, v_j)$ *whenever* $i \neq j$ *and* $d(u_i, v_i) = 2 = d(u_i, u_j)$ *and also* $d(w, u_i) = 1$, $d(w, v_i) = 2$. *Whereas* $\sum_{i=1}^{m+n-1} \sum_{j \neq i} 2^{-d(v_i, v_j)}$ *is equal to* $C(T_{n,m})$*, in Mycielski graph this value will be increased. Even so, the value of* $\sum_{i=1}^{m+n-1} \sum_{j \neq i} 2^{-d(v_i, v_j)}$ *can be expressed in terms of* $C(T_{n,m})$.

$$= 3(C(T_{n,m}) - C_{ex}(T_{n,m}) + \frac{|V_{ex}(T_{n,m})|}{2^4}) + \frac{(n+m-1)^2 + 7(n+m-1)}{4}.$$

□

3 Conclusion

In this article, closeness of Mycielski graph has taken into consideration depending on diameter of original graph. For the case of diameter less than 4, the outcome is expressed in terms of closeness of original graph. For special graphs whose diameter less than 4, results calculated in [22] verified with our expression. Furthermore, a sharp lower bound has provided case of diameter greater than 4. This lower bound equal to closeness of Mycielskian of path, calculated by us. In addition, closeness of Tadpole graph and its Mycielski form is evaluated by utilizing closeness of path graph P_n, considering whether n is even or odd.

References

[1] A. Aytac, Z. N. Odabas, Residual closeness of wheels and related networks, *International Journal of Foundations of Computer Science*, **22,** 5 (2011) 1229–1240. doi:10.1142/S0129054111008660. ⇒223, 224

[2] A. Aytac, H. Aksu Öztürk, Graph theory for big data analytics, *Journal of the International Mathematical Virtual Institute*, **10**, 2 (2020) 325–3390. doi:10.7251/JIMVI2002325A. ⇒223

[3] A. Aytac, Relevant graph concepts for big data, *Journal of Modern Technology and Engineering*, **5,** 3 (2020) 255–263. ⇒223

[4] V. Aytac, T. Turaci, Closeness centrality in some splitting networks. *Computer Science Journal of Moldova*, **26,** 3 (2018) 251–269. ID: 57760763. ⇒224

[5] A. Bavelas, A mathematical model for small group structures, *Human Organization*, **7,** (1948) 16–30. ⇒223

[6] F. Buckley, F. Harary, *Distance in Graphs, Addison Wesley Publishing Company*, Redwood City, CA 1990. ⇒223

[7] G. Chartrand, L. Lesniak, *Graphs and Digraphs:* (4th Edition), Chapman and Hall/CRC Inc., Boca Raton, Fl., 2005. ⇒223

[8] Ch. Dangalchev, Residual closeness in networks, *Physica A Statistical Mechanics and Its Applications*, **365,** 2006 556–564. ⇒223, 224, 226, 230, 231

[9] Ch. Dangalchev, Residual Closeness and Generalized Closeness, *International Journal of Foundations of Computer Science*, **22,** 8 (2011) 1939–1948. ⇒223

[10] Ch. Dangalchev, Closeness of splitting graphs, *C. R. Acad. Bulg. Sci.* **73,** 4 2020 461–466. ⇒224, 225

[11] K. M. Koh, D. G. Rogers, H. K. Teo, K. Y. Yap, Graceful graphs: some further results and problems, *Congress. Numer.*, **29** (1980) 559–571. ⇒230

[12] L.C. Freeman, Centrality in social networks: conceptual clarification, *Social Networks* **1,** (1979) 215–239. ⇒223

[13] S. Havlin, D.Y. Kenett, E. Ben-Jacob, A. Bunde, R. Cohen, H. Hermann, J.W. Kantelhardt, J. Kertész, S. Kirkpatrick, J. Kurths, J. Portugali, S. Solomon, Challenges in network science: Applications to infrastructures, climate, social systems and economics, *Eur. Phys. J. Spec. Top.* **214,** (2012) 273–293. ⇒222

[14] S. Jin, Z. Huang, Y. Chen, D. G. Chavarría-Miranda, J. Feo, P. C. Wong, A novel application of parallel betweenness centrality to power grid contingency analysis, *2010 IEEE International Symposium on Parallel and Distributed Processing (IPDPS)*, Atlanta, GA, USA, 2010, 1–7. doi:10.1109/IPDPS.2010.5470400. ⇒222

[15] V. Latora, M. Marchiori, Efficient behavior of small-world networks, *Phys. Rev. Lett.* **87,** 19 (2001) 198701. ⇒223

[16] E. L. Merrer, G. Trédan., Centralities: Capturing the fuzzy notion of importance in social graphs, *SNS '09: Proceedings of the Second ACM EuroSys Workshop on Social Network Systems*, 2009, pp. 33–38, doi:/10.1145/1578002.1578008. ⇒222

[17] J. Mycielski, Sur le coloriage des graphs, *Colloq. Math.*, **3** (1955), 161–162. ⇒223

[18] M. C. Pham and R. Klamma, The structure of the computer science knowledge network, *ASONAM'10: Proceedings of the 2010 International Conference on Advances in Social Networks Analysis and Mining*, 2010, doi:10.1109/ASONAM.2010.58. ⇒222

[19] S. Porta, V. Latora, F. Wang, E. Strano, A. Cardillo, S. Scellato, V. Iacoviello, Messora, Street centrality and densities of retail and services in Bologna, Italy, *Environment and Planning B: Planning and Design*, **36,** 3 (2009), 450–465. ⇒222

[20] S. H. Strogatz, Exploring complex networks, *Nature*, **410** (2001), 268–276. ⇒222

[21] M. Truszczyński, Graceful Unicyclic Graphs, *Demonstatio Math.*, **17** (1984) 377-387. ⇒230

[22] T. Turaci , M. Okten, Vulnerability of Mycielski graphs via residual closeness *Ars Combinatoria* **118** (2015). 419–427. ⇒224, 225, 226, 236

[23] T. Turaci, V. Aytac, Residual closeness of splitting networks, *Ars Combinatoria* **130** (2017) 17–27. ⇒224
[24] X. F. Wang, X. Li, G. R. Chen, Complex Networks Theory and its Application, *Tsinghua University Press*, Beijing, 2006. ⇒222
[25] J. S. Wang, X. P. Wu, B. Yan, J.W. Guo, Improved method of node importance evaluation based on node contraction in complex networks, *Procedia Eng.*, **15** (2011) 1600–1604. ⇒222
[26] D.J. Watts and SH. Strogatz, Collective dynamics of 'small-world' networks, *Nature*, **393** (1998) 440–442. ⇒222
[27] Z.N. Odabas , A. Aytac, Residual closeness in cycles and related networks, *Fundamenta Informaticae*, **124,** 3 (2013) 297–307. doi:10.3233/FI-2013-835. ⇒223
[28] Y.C. Zhang, Y. Liu, H.F. Zhang, H. Cheng, F. Xiong, The research of information dissemination model on online social network, *Acta Phys. Sinica*, **60** (2011) 050501. ⇒222

Received: August 14, 2023 • Revised: October 4, 2023

Acta Univ. Sapientiae Informatica **15,** 2 (2023) 239–266

DOI: 10.2478/ausi-2023-0016

Parallelising semantic checking in an IDE: A way toward improving profits and sustainability, while maintaining high-quality software development

Kristóf SZABADOS
Eötvös Loránd University,
Budapest, Hungary
email: SzabadosKristf@gmail.com

Abstract. After recent improvements brought the incremental compilation of large industrial test suites down to a few seconds, the first semantic checking of a project became one of the longest-running processes. As multi-core systems are now the standard, we derived a parallelisation using software engineering laws to improve the performance of semantic checking.

Our measurements show that even an outdated laptop is fast enough for daily use. The performance improvements came without performance regressions, and we can't expect additional massive benefits even from infinitely scaling Cloud resources.

Companies should utilise cheaper machines that still offer enough performance for longer. This approach can help businesses increase profits, reduce electronic waste and promote sustainability while maintaining high-quality software development practices.

Key words and phrases: parallel computing, cloud computing, semantic checking, integrated development environment, software development tools, software engineering laws, TTCN-3, performance improvement, sustainability, cost reduction, profit increase

1 Introduction

Software is ubiquitous in modern society. It helps us navigate, communicate, and manage energy resources. It drives companies, trades on markets, and supports healthcare.

As software products grow, so do their test systems. Some industrial test systems contain millions of lines of code [1, 63]. For a long time, compiling such codes for several minutes was the most time-consuming part of developers' daily work. Companies used clusters of remote servers or Cloud solutions to make the required performance available.

But, recent improvements [11] brought the incremental compilation of such systems down to a few seconds, leaving the first single-threaded semantic checking of the IDE as one of the longest-running processes. This process can still take several seconds on our industry partner's codes. Too long for an IDE that should be interactive.

However, nowadays, single-threaded execution is an unnecessary constraint, with multi-core and multi-CPU hardware readily and commercially available.

In this paper, we report on how we improved the IDE of our industry partner with parallel processing of the semantic analysis, making better use of available processing power. Our industry partner might no longer need to use Cloud or remote servers, as the laptop their employees would use to reach those services, might already be powerful enough.

We organised this paper as follows. Section 2 presents related works. Section 3 shows a technical description, and 4 is our proposal for the opportunity. Section 5 presents our measurement method, and 6 shows our measurements and observations. Section 7 their validity. Finally, 8 shows our summary, and 9 offers ideas for further research.

2 Related works

In this section, we present earlier related works. In the first group of sections, factors that serve as general requirements for our chosen solution and its general applicability: organisations intentionally design and govern the overall architecture of their products to achieve their business targets (Section 2.1), the generality and inevitability of the internal structure of large software systems (Section 2.3), unavoidability of dependency cycles (Section 2.5), all software systems evolving similar size distributions (Section 2.4).

In the second group of sections, we present how organisations use Project Management to predictably deliver the right products at the right time [21]

(Section 2.2), how all software systems evolve in a similarly predictable way (Section 2.6) to show that our chosen method permanently solves the problem.

2.1 Previous work on the impact of organisational factors on software systems

Empirical observations have identified an isomorphic relationship [71] between an organisation's communication structure and product structure, known as Conway's law [7] or Mirroring law [39]. Nagappan et al. [46] showed that organisational metrics predict failure-proneness better than code complexity, coverage, internal dependencies, churn and pre-release bug measures. This phenomenon was first recognised [39] for its significant managerial impact in 2012. By 2022 researchers observed it as likely the superior strategy used by 92% of investigated firms [6] and that alignments and "mirror breaking" in organisations are strategic to maximise business benefits [38].

Following these laws, contemporary System Architects take several environmental factors (among others: Taxation [10], Export Control [5], Geopolitics [50] and Standardisation[1]) into consideration when planning software architecture and the organisation developing it. Contemporary Project Management recommends [22] tailoring a selected development approach first for the organisation and second to the project. To ensure that individual project decisions do not threaten larger strategic goals.

2.2 Previous work on the impact of project management on software projects

Researchers have identified [68, 55, 43] that Project Management[2] techniques and processes are the critical factors contributing to project success.

Empirical observers have noted [8] that 94% of troubles and possibilities for improvement are the responsibility of management. Instead of rewarding managers for solving crises and heroes for putting out fires, companies should reward managers for preventing problems with systemic problem-solving performed with scientific rigour [17, 59, 4].

Project Management has a long history of evolving systematic practices [70] since the first use of Test Driven Development [3] by the first programmers

[1] As a form of Test-First Development, companies join to create and standardise automated tests, precisely determining required capabilities and interfaces for future products [29].

[2] Defined as "the application of knowledge, skills, tools, and techniques to project activities to meet the project requirements" [21].

in the 1940s [26, 2], Iterative and Incremental Development practices in the 1950s.

Nowadays, Professional Managers can use Agile and Lean methodologies to detect and reduce unnecessary activities before they happen. They can delay their decisions until the last responsible minute. They can direct developers to follow test-first practices to ensure the quality of new developments. Managers can use Continuous Integration to detect unapproved changes to supported requirements. Understanding, that writing programs "is only a small part of Software Engineering" ([51]).

2.3 Previous work on the dependency networks of software systems

Empirical researchers have shown that several architectural properties of software systems are scale-free like many real-life networks. Class collaboration graphs of the C++ language [45], Class, method, and package collaboration graphs of Java [20, 69], connections between the modules in TTCN-3[3] [61, 63], file inclusion graphs in C [44], and the object graph (the objects instances created at runtime) of most of the Object Oriented Programming languages in general [53], the relationships of distributed software packages [31, 28] show scale-free properties.

Taube-Schock et al. [65] showed, that approximately scale-free structures should arise for both between-module and overall connectivity from the preferential attachment-based models (like software), not as the result of poor design. Concluding, that high coupling is not avoidable, and might even be necessary for good design, contradicting previous ideas about software structure, in particular the "high cohesion/low coupling" maxim.

2.4 Previous work on the size distribution of software systems

Empirical studies have revealed that several metrics correlate to the point of redundancy[4] [40]. Measuring SLOC would be enough to obtain a landscape of the evolution of the size and complexity of FreeBSD [19]. Stating that "whatever is measured in a large scale system" the graph shows similar logarithmic distribution in most cases [1].

[3] Testing and Test Control Notation Version 3

[4] Cyclomatic Complexity, the Number of Lines of Code, Statements, Classes, Files, public APIs, and public undocumented APIs are redundant metrics, with Cyclomatic Complexity in classes and functions measuring the same subject

Empirical researchers have shown module length distribution of IBM 360/370 and PL/S code forming logarithm shape [58], Java class sizes following lognormal distribution [73], token distribution in Java code following Zipf's law [72, 74], all metrics measured on FreeBSD following lognormal and power-law distributions [19], double-Pareto distribution for five (C, C++, Java, Python, Lisp) of the top seven programming languages used in the Linux code [18] and LOC following lognormal distribution in Smart Contracts written in Solidity deployed on the Ethereum blockchain [66].

Hatton proposed [15] that the Conservation of Hartley-Shannon Information might play the same role in discrete systems as the Conservation of Energy does in physical systems, proving [16] Zipf's Law in the case of homogeneous systems and showing strong evidence for unusually long components being an inevitable by-product of the total size of the system. He validated the claims on 100 million lines of code in 7 programming languages and 24 Fortran 90 packages.

Hatton also highlighted the importance of changing software design techniques, from attempting to avoid the essentially unavoidable to mitigating its damaging effects.

2.5 Previous work on circular dependencies in software systems

Empirical studies have shown the existence of circular dependencies in several successful and known programs written in Java [49, 42, 9], C# [49], TTCN-3 [64], even in the binaries of Windows Server 2003 [75]. These studies offer empirical evidence for the understanding that if a program has enough components to support them, it is likely to have dependency cycles (with cycle sizes of 1000 classes [42] or involving a substantial part of all classes [49]).

2.6 Previous work on the evolution of software systems

Since Lehman started his work on software evolution [36], showed that commercial systems have a clear linear growth [37] and published the laws of software evolution [34], this phenomenon has been attracting researchers.

There is plenty of empirical research [37, 35, 33, 24, 27, 23, 25, 54, 30, 76] in which the authors show that the laws seem to be supported by solid evidence.

Turski even showed [67] that the gross growth trends can be predicted by a mean absolute error of order 6%. Also observed by others [13, 30, 76].

Looking at the impact of outside effects on software growth, empirical researchers observed [30] that "the introduction of continuous integration, the existence of tool support for quality improvements in itself, changing the development methodologies (from waterfall to agile), changing technical and line management structure and personnel caused no measurable change in the trends of the observed Code Smells", on industrial Java and C++ projects [76], that changing architects, going open source or the organisation moving to a different building had no easily discernible effect on development.

The works performed on large open source systems [24, 13, 35, 27, 23, 25, 76] serve as observations supporting the understanding that there was no hardware, software, tooling, methodological, social, or other change at least since 1995, that would have significantly changed the development speed of large software systems already started.

3 Opportunity description

In this section, we present the opportunity in more technical terms. We describe what is already available in Titan [81], and what we were working with. We show what we can expect from large-scale scale-free systems, and how circular dependencies constrain our work.

3.1 What was already available and what we worked with

The tool of our industry partner performed syntactical analysis of source files in parallel. The complete semantic checking of the parsed modules was available, done sequentially. Parsing [62] and semantic checking [48] could be incremental.

Semantic analysis happens recursively, starting with the components of a semantic entity (including entities contained within or reachable via references), followed by itself, checking each semantic entity at most once per semantic checking.

Although the optimal order is unknown before the first semantic checking of a project, executing the process for each module in the project, in any order, will check every semantic entity exactly once.

As semantic checking was single-threaded, the implementation does not have any locks or guards against parallel access/modification on the level of the entities. Locks only protect against overlapping semantic checks on a project.

3.2 Constraints on parallel processing

Previous works have already revealed properties of software systems that can be constraints for reaching optimal parallelization:

- Large differences in module sizes (Section 2.4):
  - Uneven size distribution means uneven processing time.
  - Processing order of independent modules matters for performance.
- Scale-free distribution of dependencies (Section 2.3):
  - The maximum number of imports in a module being $\geq$ 10% of the modules, semantic checking such a module might also check all reachable modules in the same thread, sub-optimal parallelization.
  - Many modules depending on the same module creates contention waiting for its processing to finish. Large amounts of modules become available when that happens.
- Dependency cycles (Section 2.5):
  - The modules of dependency cycles can't be processed in parallel safely without the risk of introducing deadlocks, but processing a cycle as 1 unit by the same thread is sub-optimal parallelism.
  - Before checking any module from a cycle, all depend on a not yet checked module. Without deep pre-processing, it is not possible to break cycles optimally.
- Gathering more information also costs time [48]:
  - Sequential pre-processing for optimal parallelism might make the entire process last longer.
  - Only the import relations and the number of definitions/assignments in a module are known without semantic checking.

As software evolves predictably (Section 2.6) towards long-standing business goals of an infrequently changing organization (Section 2.1) and governed by people dedicated to ensuring smooth progress (Section 2.2), once some hardware becomes fast enough it will stay that way for a long time. Moore's law and the linear growth trend of software systems can ensure a practically permanent solution.

4 Our method

Our method structures the semantic checking of the list of modules into an initial sequential part, a parallel part, and a final synchronisation point.

Initial sequential part:

1. The list of modules to be processed is ordered descending according to the number of definitions/assignments inside them.

2. An executor service with a thread pool is created and filled with new runnable tasks for each module without imports.

In the parallel part:

1. When a thread has processed its module, it creates a new runnable task for each module not yet processed but having all their imported modules processed.

2. To break cycles, if the thread is the last running and it does not find any new modules to process, while there are still modules to be processed, it selects the first not yet processed module for processing.

The processing ends when all modules are processed.

Titan developers merged our changes with a bug fixed later[5].

5 Measurement methodology

In this section, we present our measurement methodology and its calibration. At the timing precision required, we had to set up a measurement methodology that let us separate the effects of our method from those of the environment.

First, we explored the limitations of the hardware[6] (5.1). Then, we generated a project that could support ideal scaling (5.2), decided on the default measurement process (5.3) and performed exploratory data analysis on 10.000 measurements to calibrate it (5.4). Finally, we used this methodology on the ideal project to establish a baseline (5.5).

[5] `https://gitlab.eclipse.org/eclipse/titan/titan.EclipsePlug-ins/-/merge_requests/918`, last accessed: 2023.05.15

[6] Measurements were performed on a Lenovo Legion R7000 laptop, with an AMD Ryzen™ 7 4800H 8-core 16-thread CPU (at a base Clock of 2.9GHz that can boost to 4.2GHz), 1*8 GB Kingston 3200 MHz SODIMM RAM, using an SSD, running Windows 10 Home and the 3.0.7-1 version of Cygwin, GCC 11.3.0, Eclipse 4.20.0, Java SDK 16.0.2 .

5.1 The limitations of the hardware

We explored the relevant limitations of the hardware using STREAM [41][7].

The source code was compiled with: gcc -O2 -DSTREAM_ARRAY_SIZE = 80000000 -DNTIMES=100 -fopenmp stream.c -o stream.80M.exe

We performed execution from the command line, controlling the thread count via the "OMP_NUM_THREADS" environmental variable.

Table 1: The maximum and minimum STREAM results compared to the single-threaded case and their thread count location.

	max bandwidth		min bandwidth	
Project vs test	ratio	location	ratio	location
Copy	+18.58%	4	+9.47%	22
Scale	+3.27%	2	-14.93%	18
Add	+4.34%	2	-17.93%	18
Triad	+4.59%	3	-18.18%	16

Our measurements show (Table 1) the most bandwidth available using 2-4 threads. At higher thread counts, for all but the "Copy" function, the measured memory bandwidth is below the single-threaded case. The measured values fell into a limited range (Figure 1).

Figure 1: Memory Bandwith measured by STREAM.

[7] `https://www.cs.virginia.edu/stream/FTP/Code/`, version ("5.10") downloaded in May. 2022., last accessed: 2023.05.15

In practical terms, the hardware does not support increasing thread counts with enough bandwidth for memory-bound operations. The hardware also seems to suffer from some constraints or limitations, as the numbers we measured do not reach the peak transfer rate of the standard. *This laptop is a good sample for an outdated laptop, that was never meant for professional development work and cannot support theoretically ideal scaling.*

5.2 The Ideal project

To explore the parallel limits of the solution in finer detail and to calibrate the measurement process in an ideal setting, we created a synthetic project, "Ideal", by copying the RAWCodingAttributes.ttcn file from [12] 16 times[8].

In this project, all files are standalone and processable in parallel, have the same content, and need identical processing. The files are large enough[9] for the analysis to be practically measurable, and their number matches the CPU thread count.

5.3 The default process

We decided to use the closing and opening of a project for measurement, as this deletes all information the IDE knows about that project and triggers its analysis.

We created a small program to perform measurements in a loop, sleeping at the end of each iteration (eliminating its potential impact).

Settings used for the measurements:

- The measurement Eclipse was started with:

  -Xms4G -Xmx4G -XX: UseG1GC-server

- The laptop was in "Quiet Mode"[10].
- The analysis threads were started with priority 1 (lowest) and after checking each Assignment[11]/Definition[12] a "Thread.yield()" call is executed.

[8] Adding their index to both the file and module name to forego collisions.

[9] 7090 lines.

[10] Description from manufacturer: "Keep quiet by reducing your computer performance and fan speed where possible with low power consumption".

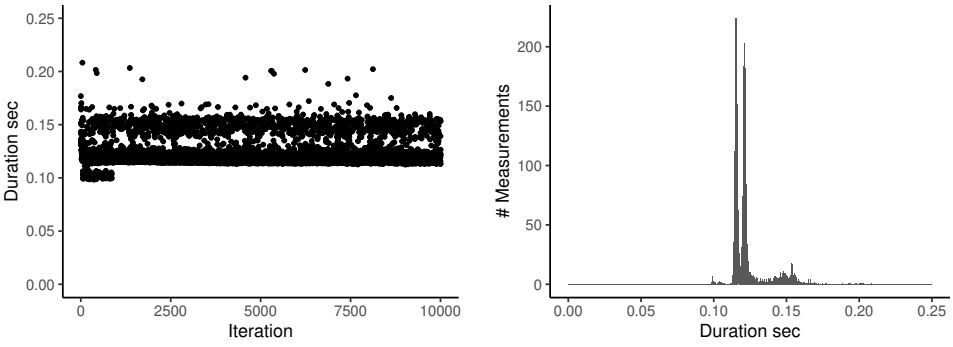
[11] `https://www.itu.int/ITU-T/studygroups/com17/languages/X.680-0207.pdf`, last accessed: 2023.05.15

[12] `https://www.etsi.org/deliver/etsi_es/201800_201899/20187301/04.14.01_60/es_20187301v041401p.pdf`, last accessed: 2023.05.15

5.4 Calibration of the measurement process

During our measurements, we had to account for the Just-in-time compilation and optimisation of Java, the language of the IDE.

We set the helper program to perform 10.000 measurements.



(a) Direct measurements

(b) Histogram

Figure 2: The execution durations for the iterations (2(a)) and their histogram (2(b)) for a 10.000 iteration execution. For visualisation purposes, the first execution (taking 1.6 seconds) is not shown.

Our observations showed (Figure 2) that the first measurement is an outlier. The following measurements form 4 clusters using K-means clustering (means: 0.1021, 0.1158, 0.1225, 0.1536).

At this point, we could devise the routine for measurements:

1. To determine a sufficient sleep duration, perform three iterations manually with the new settings set.
2. Turn off all potential interference and restart the hardware.
3. Do three iterations manually.
4. Execute the automation code for 200 iterations and extract the last 50 measurement points ([14]).

5.5 Measurements on the Ideal project

The overall duration of analysing the project (Figure 3) drops from an average of 0.39s at 1 to 0.23s at 2, 0.147s at 4, 0.131s at 8, and 0.13s at unlimited thread counts. The maximum speed-up of 3.18 is at 10 threads. In practical terms, most performance gains happen till 4-thread parallelism. Higher thread counts

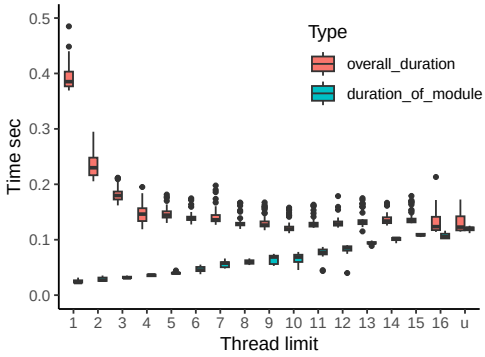


Figure 3: The overall duration for the semantic checking on the ideal project, and the single module duration from the closest to average execution, for each thread limit (u meaning unlimited).

seem to produce only marginally better results. At the same time, there is also no reason to use any smaller setting in practice.

The individual module analysis times from the closest to the average execution of that thread count revealed that the analysis duration of a given module approximately linearly depends on the thread count, indicating that the hardware is memory bandwidth limited.

6 Measurements

In this section, we present our measurements.

Section 6.1 presents that our chosen method solves the problem on large code bases and that further parallelisation is limited by the structure of the problem, not the hardware. Section 6.2 shows that there are no noticeable performance degradations in existing features.

6.1 Standardized test suites

We have analysed the behaviour of our method on all test suites created by 3GPP[13] and publicly available at `www.ttcn3.org`[14].

Used test suites:

- 3GPP UTRA UE Test Suites [80]: SSNITZ, UTRAN.

[13] 3rd Generation Partnership Project

[14] We downloaded all packages in July 2021.

- 3GPP IMS UE Test Suites [78]: IMS_EUTRA, IMS_IRAT, IMS_NR5GC, IMS_UTRAN, IMS_WLAN.
- 3GPP LTE UE Test Suites [77]: LTE, LTE_A_IRAT, LTE_IRAT.
- 3GPP 5G UE Test Suites [79]: ENDC, NR5GC, NR5GC_IRAT.

To process these test suites, we created a new "TITAN Project (Java)" with the name of the TTCN-3 project, copied the source code from the downloaded compressed files into its "src" folder, and converted all XSD files[15]. We manually reviewed all problems detected by Titan on these projects and reported the incorrect ones to the development team[16].

Table 2 presents the most important properties of these projects and table 3 shows how well the logarithmic and power-law trend lines fit the measured data for each project. Figure 4 presents the analysis durations measured for each project.

Table 2: Importation data: number of modules, layers, maximum number of imports, maximum number of being imported and lines of code

Project vs test	Nof modules	Nof layers[17]	I_{max}(project)	O_{max}(project)	LOC
SSNITZ	86	16	30	50	105.191
UTRAN	175	21	46	115	158.392
IMS_EUTRA	207	20	40	131	106.429
IMS_IRAT	207	19	40	140	182.561
IMS_NR5GC	164	20	43	116	83.858
IMS_UTRAN	128	17	35	72	108.732
IMS_WLAN	103	16	24	57	41.297
LTE	230	23	50	171	249.161
LTE_A_IRAT	266	20	42	187	219.496
LTE_IRAT	289	21	78	211	257.719
ENDC	250	22	49	190	152.194
NR5GC	201	19	47	161	130.907
NR5GC_IRAT	242	20	47	178	141.135

6.1.1 Measurement: LTE IRAT

In this section, we present our observations for the largest project processed.

[15] Using the "xsd2ttcn" utility of Titan, using the -N flag.

[16] `https://gitlab.eclipse.org/eclipse/titan/titan.EclipsePlug-ins/-/issues/456`, last accessed: 2023.05.15

Table 3: Trend fitting

	log r^2		power law r^2	
Project vs test	I(module)	O(module)	I(module)	O(module)
SSNITZ	0.92	0.96	0.81	0.83
UTRAN	0.97	0.82	0.84	0.92
IMS_EUTRA	0.97	0.81	0.80	0.91
IMS_IRAT	0.98	0.69	0.85	0.90
IMS_NR5GC	0.97	0.68	0.85	0.89
IMS_UTRAN	0.97	0.81	0.85	0.92
IMS_WLAN	0.95	0.90	0.84	0.90
LTE	0.94	0.87	0.78	0.87
LTE_A_IRAT	0.98	0.73	0.80	0.87
LTE_IRAT	0.97	0.78	0.78	0.87
ENDC	0.98	0.72	0.82	0.90
NR5GC	0.95	0.84	0.76	0.89
NR5GC_IRAT	0.98	0.70	0.83	0.88

The average overall duration of the analysis (Figure 5) takes 25.16s using 1 thread, 14.25s using 2 threads, 11.92s using 3 threads, 11.01s using 4 threads, 9.49s using 8 threads, 10.30s using 16 threads and 10.10s without thread limits. The overall speed-up is 2.48 without thread limits (2.64 using 8 threads).

Module "RRC_MeasurementUG" takes the longest to analyse, with 6.78s using 1 thread, 7.76s using 2 threads, 8.07s using 3 threads, 8.39s using 4 threads, 8.24s using 8 threads, 8.78s using 16 threads and 8.35s without thread limits, a slowdown of approx. 23%.

With unlimited threads, the longest to analyse path contains the modules "LTE_IRAT_Testsuite" with 0.03s, "RRC_MeasurementUG" with 8.35s, "EUTRA_Measurement_Templates" with 0.90s and "EUTRA_RRC_Templates" with 0.13s to analyse. 9.41s of the 10.10s, or approx. 93% of analysing the project with unlimited threads. *Although this is not the critical path, it still shows that no amount of increase in parallel processing threads/CPU cores alone would be able to decrease the processing time further significantly.*

The duration values for analysing a given module in descending order (Figure 6) are similar to the length values of modules plotted in descending order. With the exponential trend lines fitting to all thread limit cases with at least r^2 of 0.97 and power trend lines between 0.84 and 0.88.

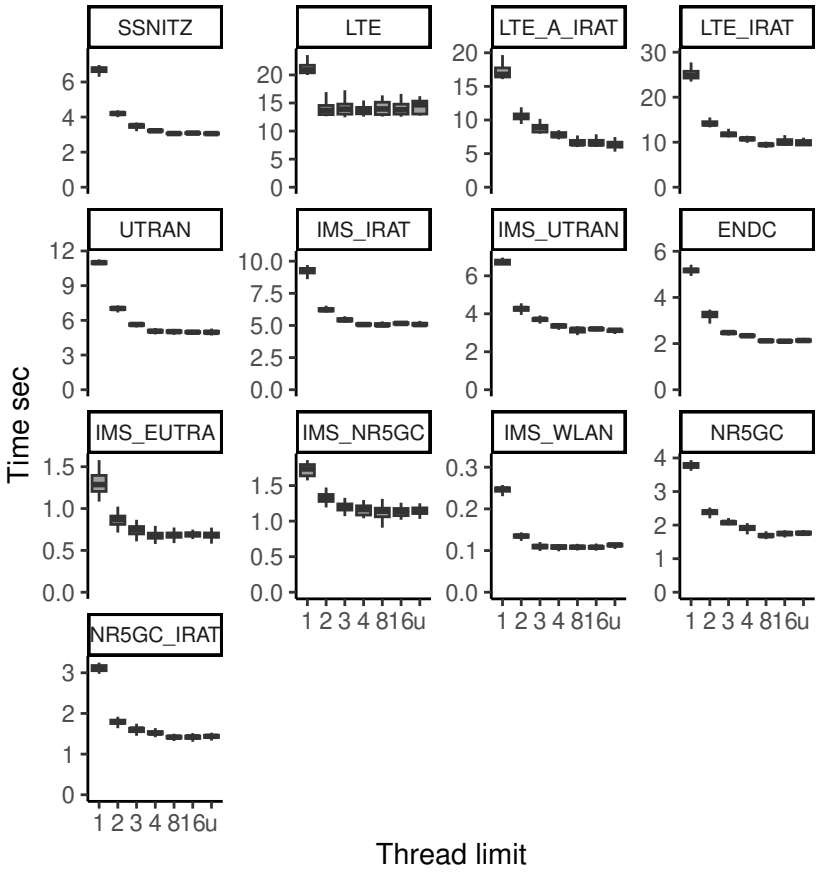
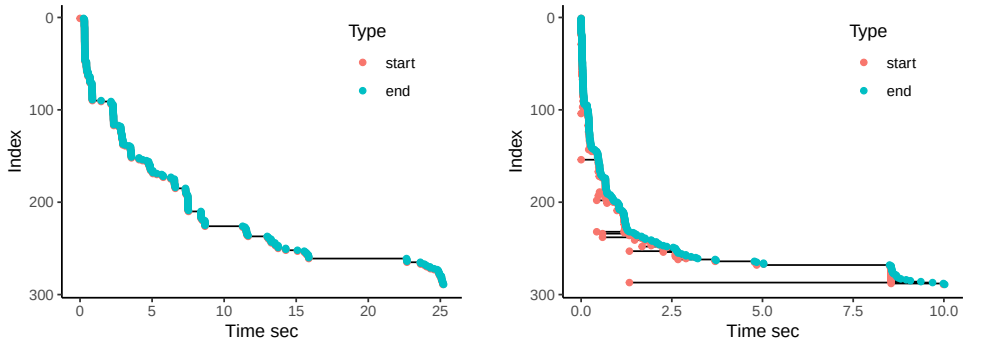


Figure 4: The overall duration, for each thread limit (u meaning unlimited) and each measured standardised test suite.

Without thread limits (Figure 7(b)), the maximum number of active threads[18] is 23. *Indicating that at most 23 threads can work in parallel, additional hardware capacity can not be utilised.*

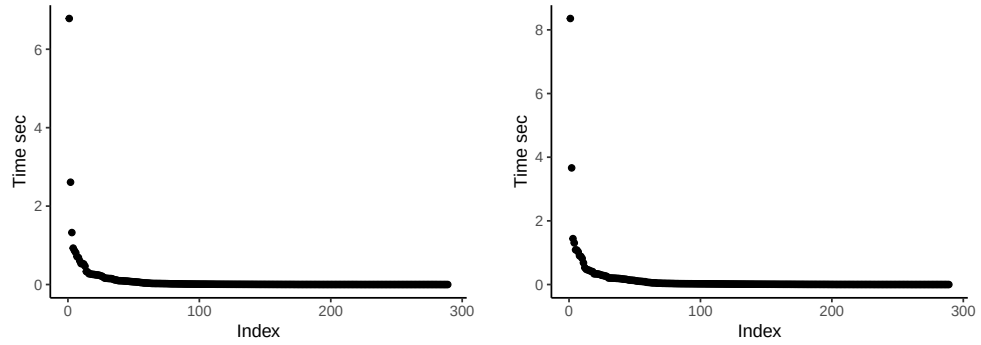
With thread limits set (Figure 7(a)) the number of active threads below the actual thread limit is 0 for 1, 2 for 2, 7 for 3, 14 for 4, 121 for 8 and 270 for 16 thread limit. *At small thread limits, parallel processing operates at or near*

[18]In our measurements we call active threads, the software threads that are actively running at the time of measurement. Where the measuring is done in the "Runnable" object's "run()" function is called, right before it starts to analyse its module.



(a) 1 thread limit (b) Unlimited thread count

Figure 5: Execution trace for the LTE IRAT modules, when the analysis is limited to 1 threaded processing 5(a) and when there is no thread limit (5(b)).

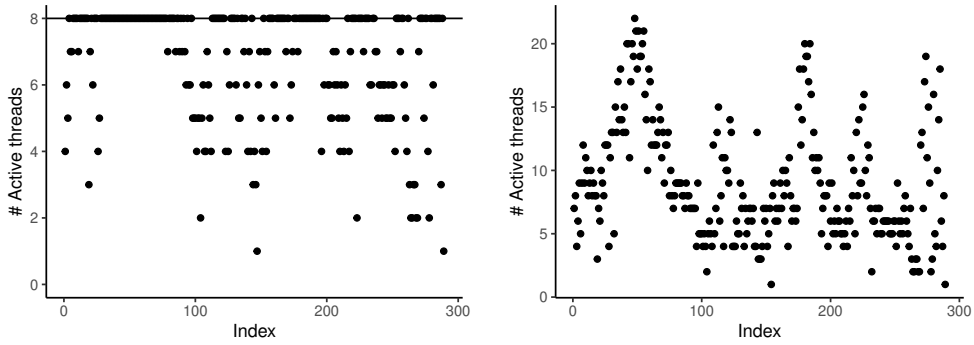


(a) 1 thread limit (b) Unlimited thread count

Figure 6: The time it takes to analyse a given module, in descending order, when there is a 1 thread limit (6(a)) and when the number of threads is not limited (6(b)), while checking LTE IRAT.

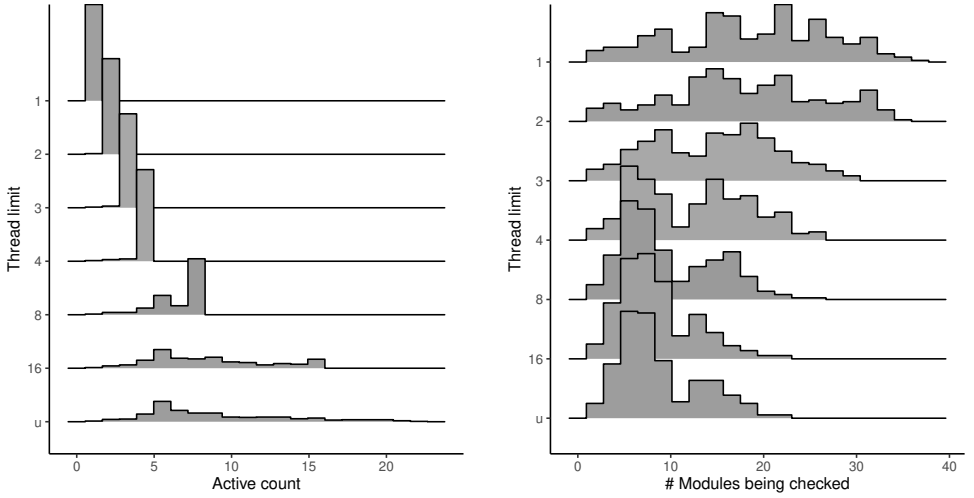
maximum capacity. At high thread limits (8 and 16 measured), the structure of the dependency graph and the runtime processing is the stronger restriction.

The system was underutilised in 41.9% of the 8 thread limit and in 93% of the 16 thread limit measurement points. The increase in the thread count limit comes with a smoother spread for the number of active threads (Figure 8(a)).



(a) 8 thread limit (b) Unlimited thread count

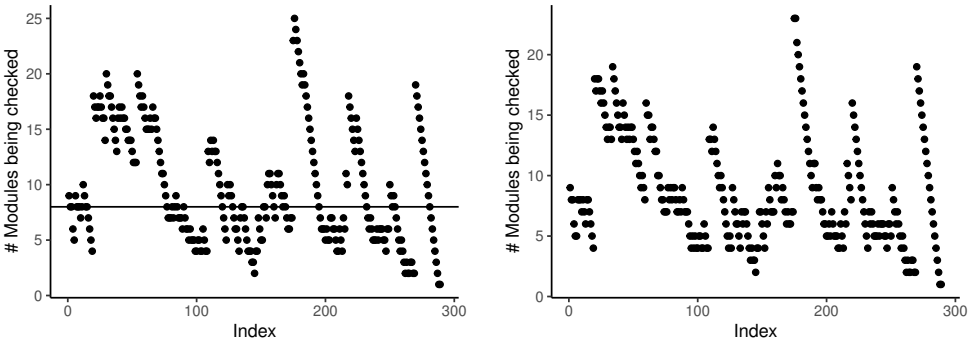
Figure 7: The number of active threads at the time of measurement, when there is an 8 thread limit (7(a)) and when the number of threads is not limited (7(b)) while checking LTE IRAT.



(a) Active thread count (b) Available for checking

Figure 8: The histograms for the active thread count (8(a)) and the number of modules available for checking (8(b)) numbers measured, for each thread limit, while checking LTE IRAT.

We also measured the number of modules being checked or available for checking (Figure 9). The maximum values are 36 for 1, 35 for 2, 29 for 3, 26 for 4, 25 for 8 and 23 for 16 thread limit and 23 when there is no thread limit.



(a) 8 thread limit

(b) Unlimited thread count

Figure 9: The number of modules available for checking at the time of measurement, when there is an 8 thread limit (9(a)) and when the number of threads is not limited (9(b)) while checking LTE IRAT.

A higher thread count limit means more modules processable in parallel and a shorter spread of modules available for processing at any time (Figure 8(b)).

6.2 Impact on existing features

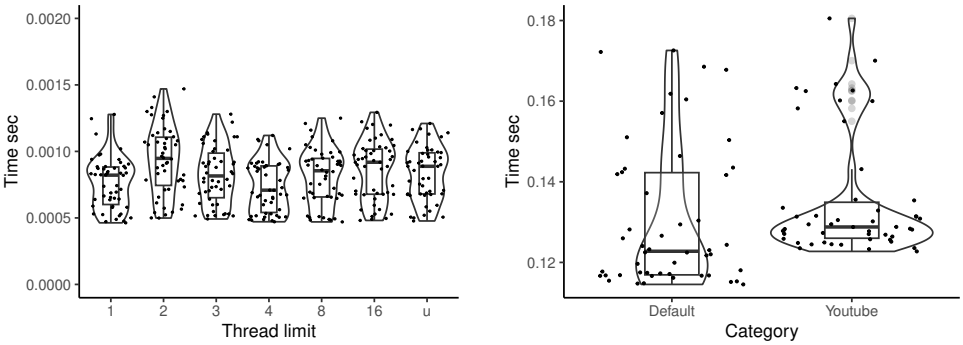
6.2.1 Impact on Titan's existing tests

The Eclipse Plug-ins of Titan we extended have tests for approx. 20.000 syntactic and semantic markers (warnings and errors together). We frequently executed these tests to ensure that existing detections were not changed. *The tests found the exact same issues at the expected locations, texts and severities.*

6.2.2 Impact on incremental parsing

We repeated the measurement described in [62] to prove that the incremental syntax checking is not affected negatively, inserting 203 spaces at the end of a line in a file of the Ideal project and using the last 50 measurements.

Every measured execution time of the syntax analysis (figure 10(a)) was below $1.47 * 10^{-3}$ seconds, going as low as approx. $5 * 10^{-4}$ seconds. *These values, for all thread counts, are similar to the published values indicating that the performance of incremental syntax checking has not regressed.*



(a) Incremental parsing comparisons

(b) YouTube test comparisons

Figure 10: Comparisons of incremental syntax checking duration, for different thread limits (10(a)) and the overall duration in Default mode, unlimited threads, with and without YouTube running in the background (10(b)).

6.2.3 YouTube test

IDEs have an upper limit on the hardware capacity they are allowed to use. For a good user experience, IDEs must not overload the developer's machine to the point where background music playback is negatively affected.

To prove that the new version still respects this upper boundary, we have re-run the measurements on the Ideal project, without thread limits, without blocking external interfaces (WiFi) and playing a video[19] in 1080p60HD quality via the Internet in the background.

During the measurement memory usage was approx. 95%, GPU utilisation approx. 30%. During the analysis phases the CPU was utilised at 100%[20], increasing speed to 3.50 - 3.88 GHz (falling back to approx. 2.15 GHz in-between these phases).

Playing a video in the background created (Figure 10(b)) a difference of 0.0068s in the means. Subjectively, we have not noticed any difference in the music during measurements.

While the analysis shows a statistically meaningful difference, in practical terms, we did not see a significant, real-world performance difference.

[19] `https://www.youtube.com/watch?v=A-aSaw7JfB8`, last accessed: 2023.05.15

[20] The built-in "Task Manager" application

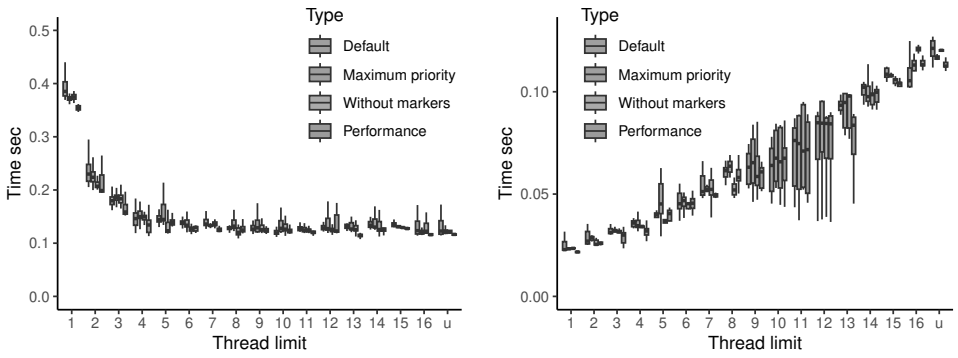
6.3 Additional research directions covered

In this section, we present additional research directions we have covered.

"Maximum priority": We maximised the priority of our application at the cost of other applications. The analysis threads executed at maximum priority (10) "Thread.yield()" was not called after checking each Assignment/Definition.

"Without markers": We removed the crucial functionality of reporting errors to eliminate a potential lock contention, where the Eclipse platform stores markers in an internal database. Here we have commented out the body of the "Location.reportProblem" and all functions in the "ParserMarkerSupport" class with "createOnTheFly" in their name.

"Performance": We overclocked the hardware at the cost of higher noise and power consumption. We have set the laptop in "Performance Mode"[21].



(a) Overall duration

(b) Single module duration

Figure 11: The overall duration (11(a)) and the single module duration (11(b)) for each experiment, for each thread limit (u meaning unlimited).

Our analysis shows (Table 4):

1. "Maximum priority" is mostly faster with a small margin.
2. "Without markers" is clearly faster, but would lose core functionality.
3. "Performance" is the fastest, but to handle the increased heat, the laptop became too loud for office usage.

[21]Description from manufacturer: "Boost your computer performance with higher fan speed and power consumption".

Threads	Max. Priority	Without Markers	Performance
1	0.01577	0.01373	0.03663
2	0.00686	0.02121	0.02332
3	-0.003821	-0.00304	0.018593
4	-0.00654	-0.002403	0.01335
5	-0.00817	0.02175	0.00702
6	0.00300	0.011101	0.009106
7	0.00237	0.000804	0.01143
8	-0.00286	0.00909	0.00364
9	-0.00121	0.00206	0.00413
10	-0.00776	-0.00702	-0.002307
11	0.00317	0.00312	0.00614
12	-0.000437	0.004698	0.005903
13	0.00452	0.005544	0.01673
14	0.002477	0.00746	0.006955
15	0.00262	0.004450	0.006402
16	0.00361	0.00149	0.00664
u	0.000897	0.000424	0.00646

(a) muDiff

Threads	Max.Priority	Without Markers	Performance
1	100	100	100
2	93.8	100	100
3	7.67	15.3	100
4	4.09	23.2	100
5	3.667	100	100
6	95.93	100	100
7	99.7	78.3	100
8	1.151	100	99.94
9	31.19	96.0	100
10	0	0.002	0.188
11	99.9	100	100
12	46.08	100	100
13	100	100	100
14	98.1	100	100
15	99.9	100	100
16	98.5	76.5	100
u	67.0	57.5	100

(b) $>compval

Table 4: Results of comparing the default measurements to a direction, showing the difference in the means (4(a)) and the percentage of the posterior probability mass above the comparison value 0 (4(b)).

While the analysis showed statistically meaningful improvements, no experiment offered practical real-world performance improvements (Figure 11). We consider these research directions closed.

7 Threats to validity

This study might suffer from the usual threats to external validity. There might be limits to generalising our results beyond our settings (the programming language used and possible industry-specific effects). We can only claim the validity of our results for programming languages and code bases which demonstrate the properties discussed in Section 2. Further research could investigate these properties for other languages and validate if our claim also holds for them.

One specific threat to generalization might come from our measurement performed only on a single laptop. To address this threat, we point out that this laptop was already outdated during the measurements and never meant for professional development work. In the paper, we showed how this laptop is already fast enough for daily work and how the structure of the problem would not benefit from more parallel resources, demonstrating how companies could already save on hardware costs.

8 Summary

IDE performance was a pain point for developers for a long time [22]. In this paper, we present our work on improving this situation by parallelizing the semantic checking phase of an industrial IDE.

We have presented earlier works on the structure of software systems, which serve as general requirements and prove the general applicability of our chosen method. We also presented works on the evolution tendencies of software systems to show that our chosen method permanently solves the problem.

We have shown that the new version improves performance on real-life projects, utilising contemporary hardware better without performance regression in other parts of the system. We showed that the structure of the problem limits better utilising all parallel hardware resources. We can not expect additional benefits even from infinitely scaling Cloud resources.

Our measurements showed that even outdated laptop hardware, not aimed at professional development work, is now strong enough to support working on large open-source test systems. From the perspective of performance only, our results make it hard to justify investing in Cloud resources or remote servers to provide developers with a performing IDE. Companies should optimise their development costs and sustainability efforts [52] by utilising weaker/cheaper machines that still offer enough performance.

Our results could also inspire simplification in future IDEs, making pre-built indices obsolete and user interfaces simpler.

The functionality we developed is available in open source [81] as part of the `Titan` toolset.

9 Further work

Further research in improving performance could target: finer-grain locking, keeping the work on "warm cores" [32], running our Java code directly on bare metal [56, 57], and further optimisation of a Java Just-In-Time compiler [60].

Other research could target determining the most cost-efficient hardware configuration for a given project, recommend code management and coding styles that reduce processing times [47].

[22] `https://blog.jetbrains.com/kotlin/2021/06/kotlin-ide-performance/#Reworkedplatform,plugin,andcompilerAPIforcodehighlighting`

10 Acknowledgements

The authors would like to thank the DUCN Software Technology unit of Ericsson AB, Sweden for the financial support of this research and the Test Competence Center of Ericsson Hungary for providing access to their in-house tools. We would also like to thank Izabella Ingrid Farkas for her help invaluable to our measurements, and Peter Verhas for his feedback on this article.

References

[1] N. Bartha, Scalability on it projects, *Master's thesis*, Eötvös Loránd University, 2016. ⇒240, 242

[2] K. Beck, Why does Kent Beck refer to the "rediscovery" of test-driven development? what's the history of test-driven development before Kent Beck's rediscovery?, https://www.quora.com/Why-does-Kent-Beck-refer-to-the-rediscovery-of-test-driven-development-Whats-the-history-of-test-driven-development-before-Kent-Becks-rediscovery, Last acc.: 2023.05.15. ⇒242

[3] K. Beck, *Test Driven Development. By Example (Addison-Wesley Signature)*, Addison-Wesley Longman, Amsterdam, 2002. ⇒241

[4] R. Bohn, Stop fighting fires, *HBR*, 78:83–91, 07 2000. ⇒241

[5] M. Choudaray and M. Cheng, *Export Control. In Open Source Law, Policy and Practice*, Oxford University Press, 10 2022. ⇒241

[6] L. J. Colfer and C. Y. Baldwin, The mirroring hypothesis: Theory, evidence and exceptions, *IRPN: Innovation & Organizational Behavior (Topic)*, 2016. ⇒241

[7] M. E. Conway, How do committees invent, *Datamation*, 1967. ⇒241

[8] W. E. Deming, Out of the Crisis, *volume 1 of MIT Press Books*. The MIT Press, 12.2000. ⇒241

[9] J. Dietrich, C. McCartin, E. Tempero, and S. M. A. Shah, Barriers to modularity - an empirical study to assess the potential for modularisation of Java programs. *In Research into Practice – Reality and Gaps*, pages 135–150, Berlin, Heidelberg, 2010. Springer Berlin Heidelberg. ⇒243

[10] M. Dorner, M. Capraro, O. Treidler, T.-E. Kunz, D. Šmite, E. Zabardast, D. Mendez, and K. Wnuk, Taxing collaborative software engineering, 2023. ⇒241

[11] I. I. Farkas, K. Szabados, and A. Kovács, Improving productivity in large scale testing at the compiler level by changing the intermediate language from C++ to Java, *Acta Univ. Sapientiae Informatica*, **13,** 1 (2021) 134–179. ⇒240

[12] I. I. Farkas, K. Szabados, and A. Kovács, *Regression test data*, http://compalg.inf.elte.hu/ attila/materials/RegressionTestSmall_20190724.zip, 2019. ⇒248

[13] J. Fernandez-Ramil, D. Izquierdo-Cortazar, and T. Mens, What does it take to develop a million lines of open source code?, *In Open Source Ecosystems: Diverse Communities Interacting*, volume 299, pages 170–184, 06 2009. ⇒243, 244

[14] A. Georges, D. Buytaert, L. Eeckhout, Statistically Rigorous Java Performance Evaluation, *In Proceedings of the 22nd Annual ACM SIGPLAN Conference on Object-oriented Programming Systems and Applications*, OOPSLA '07, pages 57–76, New York, NY, USA, 2007. ACM. ⇒249

[15] L. Hatton, Conservation of information: Software's hidden clockwork?, *IEEE Trans. Softw. Eng.*, **40,** 5 (2014) 450–460. ⇒243

[16] L. Hatton and G. Warr, Strong evidence of an information-theoretical conservation principle linking all discrete systems,*R. Soc. O. Sci.*, **6,** 10 (2019) 191101 ⇒243

[17] R. Hayes, Why Japanese factories work, *HBR*, **59,** 1 (1981) 56–66. ⇒241

[18] I. Herraiz, D. Germán, and A. E. Hassan, On the distribution of source code file sizes, *In International Conference on Software and Data Technologies*, v. 2, p. 5–14, 01 2011. ⇒243

[19] I. Herraiz, J. M. Gonzalez-Barahona, and G. Robles, Towards a theoretical model for software growth, *In Fourth International Workshop on Mining Software Repositories* (MSR'07:ICSE Workshops 2007), p. 21–21, 2007. ⇒242, 243

[20] D. Hyland-Wood, D. Carrington, and S. Kaplan, Scale-free nature of java software package, class and method collaboration graphs, *In Proceedings of the 5th International Symposium on Empirical Software Engineering*, 2006. ⇒242

[21] P. M. Institute, *A guide to the Project Management Body of Knowledge* (PMBOK guide), PMI, Newton Square, PA, 6th edition, 2017. ⇒240, 241

[22] P. M. Institute, *A guide to the Project Management Body of Knowledge* (PMBOK guide), PMI, Newton Square, PA, 7th edition, 2021. ⇒241

[23] A. Israeli and D. Feitelson, The Linux kernel as a case study in software evolution, *J. Syst. Softw.*, 83:485–501, 03 2010. ⇒243, 244

[24] C. Izurieta and J. Bieman, The evolution of FreeBSD and Linux, *In Proceedings of the 2006 ACM/IEEE International Symposium on Empirical Software Engineering*, ISESE '06, p. 204–211, NY, USA, 2006. ACM. ⇒243, 244

[25] K. Johari and A. Kaur, Effect of Software Evolution on Software Metrics: An Open Source Case Study, *SIGSOFT Softw. Eng. N.*, 36(5):1–8, 09.2011. ⇒243, 244

[26] T. Joosse, December 1945: The ENIAC Computer Runs its First, Top-Secret Program, https://www.aps.org/publications/apsnews/202212/history.cfm, 2022. Last accessed: 2023.05.15. ⇒242

[27] C. Kemerer and S. Slaughter, An empirical approach to studying software evolution, *IEEE Trans. Softw. Eng.*, 25(4):493–509, 1999. ⇒243, 244

[28] G. Kohring, Complex dependencies in large software systems, *Advances in Complex Systems*, 12, 11 2011. ⇒242

[29] A. Kovács and K. Szabados, Advanced TTCN-3 Test Suite validation with Titan, *In 9th International Conference on Applied Informatics*, p. 273–281, 2015. ⇒241

[30] A. Kovács and K. Szabados, Internal quality evolution of a large test system–an industrial study, *Acta Univ. Sapientiae*, 8(2):216–240, 2016. ⇒243, 244

[31] N. LaBelle and E. Wallingford, Inter-package dependency networks in open-source software, *CoRR*, cs.SE/0411096, 2004. ⇒242

[32] J. Lawall, H. Chhaya-Shailesh, J.-P. Lozi, B. Lepers, W. Zwaenepoel, and G. Muller, Os scheduling with nest: Keeping tasks close together on warm cores, *In Proceedings of the Seventeenth European Conference on Computer Systems*, EuroSys '22, page 368–383, New York, NY, USA, 2022. ACM. ⇒260

[33] M. J. Lawrence, An examination of evolution dynamics, *In Proceedings of the 6th International Conference on Software Engineering*, ICSE '82, page 188–196, Washington, DC, USA, 1982. IEEE CS Press. ⇒243

[34] M. Lehman and J. Fernandez-Ramil, Rules and tools for software evolution planning and management, *ASE*, 11:15–44, 01 2001. ⇒243

[35] M. Lehman, D. Perry, and J. Ramil, On evidence supporting the feast hypothesis and the laws of software evolution, *In Proc Fifth Int. Software Metrics Symposium. Metrics (Cat. No.98TB100262)*, pages 84–88, 1998. ⇒243, 244

[36] M. Lehman and J. Ramil, Towards a theory of software evolution - and its practical impact, *In Proc. Int. Symposium on Principles of Software Evolution*, pages 2–11, 2000. ⇒243

[37] M. M. Lehman and J. F. Ramil, Evolution in software and related areas, *In Proceedings of the 4th International Workshop on Principles of Software Evolution*, IWPSE '01, page 1–16, New York, NY, USA, 2001. ACM. ⇒243

[38] E. Leo, Breaking mirror for the customers: The demand-side contingencies of the mirroring hypothesis, *Cont. Man. Res.*, 18:35–65, Mar. 2022. ⇒241

[39] A. MacCormack, C. Baldwin, and J. Rusnak, Exploring the duality between product and organizational architectures: A test of the "mirroring" hypothesis, *Research Policy*, 41(8):1309–1324, 2012. ⇒241

[40] M. A. Mamun, C. Berger, and J. Hansson, Effects of measurements on correlations of software code metrics, *Empir. Softw. Eng.*, 24, 08 2019. ⇒242

[41] J. McCalpin, Memory bandwidth and machine balance in high performance computers, *IEEE Technical Committee on Computer Architecture Newsletter*, pages 19–25, 12 1995. ⇒247

[42] H. Melton and E. Tempero, An empirical study of cycles among classes in Java, *Empir. Softw. Eng.*, 12(4):389–415, Aug. 2007. ⇒243

[43] S. Moradi, K. Kähkönen, and K. Aaltonen, From past to present- the development of project success research, *J. Mod. Proj.*, 8(1), Apr. 2022. ⇒241

[44] A. Moura, Y. Lai, and A. Motter, Signatures of small-world and scale-free properties in large computer programs, *Phys. Rev. E*, 68(2), 2003. ⇒242

[45] C. R. Myers, Software systems as complex networks: Structure, function, and evolvability of software collaboration graphs, *Phys. Rev. E*, 68(4), 2003. ⇒242

[46] N. Nagappan, B. Murphy, V. Basili, and N. Nagappan, The influence of organizational structure on software quality: An empirical case study, *Technical Report MSR-TR-2008-11*, Microsoft Research, January 2008. ⇒241

[47] G. Nagy and Z. Porkoláb, Performance issues with implicit resolution in scala, *In Proceedings of the 10th International Conference on Applied Informatics*, pages 211–223, 01 2018. ⇒260

[48] P. Olah, Szemantikus elemzés gyorsítása TTCN-3 környezetben, *Master's thesis*, Eötvös Loránd University, 2016. ⇒244, 245

[49] T. D. Oyetoyan, R. Conradi, and D. S. Cruzes, Criticality of defects in cyclic dependent components, *In 2013 IEEE 13th International Working Conference on Source Code Analysis and Manipulation* (SCAM), pages 21–30, 2013. ⇒243

[50] A. Pannier, *Software power: The economic and geopolitical implications of open source software*, 2022. ⇒241

[51] D. L. Parnas, Structured programming: A minor part of software engineering, *Information Processing Letters*, 88(1):53–58, 2003. ⇒242

[52] Z. Porkoláb, Save the Earth, Program in C++!, *In 2022 IEEE 16th Int. Scientific Conf. on Informatics* (Informatics), p. 11–12. IEEE, 2022. ⇒260

[53] A. Potanin, J. Noble, M. Frean, and R. Biddle, Scale-free geometry in OO programs, *Commun. ACM*, 48(5):99–103, May 2005. ⇒242

[54] R. Potvin and J. Levenberg, Why Google stores billions of lines of code in a single repository, *Commun. ACM*, 59(7):78–87, Jun 2016. ⇒243

[55] S. Pretorius, H. Steyn, and T. Bond-Barnard, The relationship between project management maturity and project success,*J. Mod. Proj.*,10:219–231, 2023. ⇒241

[56] W. Puffitsch and M. Schoeberl, PicoJava-II in an FPGA, *In Proceedings of the 5th International Workshop on Java Technologies for Real-Time and Embedded Systems*, JTRES '07, page 213–221, New York, NY, USA, 2007. ACM. ⇒260

[57] D. Simon, C. Cifuentes, D. Cleal, J. Daniels, and D. White, Java™ on the bare metal of wireless sensor devices: the squawk Java virtual machine, *In Proceedings of the 2nd international conference on Virtual execution environments*, pp. 78–88, 2006. ⇒260

[58] C. P. Smith, A software science analysis of programming size, *In Proceedings of the ACM 1980 Annual Conference*, page 179–185, 1980. ACM. ⇒243

[59] S. Spear and H. Bowen, Decoding the DNA of the Toyota Production System, *HBR*, 77, 09 1999. ⇒241

[60] T. Suganuma, T. Yasue, M. Kawahito, H. Komatsu, and T. Nakatani, Design and Evaluation of Dynamic Optimizations for a Java Just-in-Time Compiler, *ACM Trans. Program. Lang. Syst.*, 27(4):732–785, Jul 2005. ⇒260

[61] K. Szabados, Structural Analysis of Large TTCN-3 Projects, *In Proceedings of the 21st IFIP WG 6.1 International Conference on Testing of Software and Communication Systems and 9th International FATES Workshop*, TESTCOM '09/FATES '09, page 241–246, Berlin, Heidelberg, 2009. Springer-Verlag. ⇒242

[62] K. Szabados, Creating an efficient and incremental IDE for TTCN-3, *In 10th Joint Conference on Mathematics and Computer Science, Cluj-Napoca*, In Studia Universitatis Babes-Bolyai, Informatica, volume 60, pages 5–18, 2015. ⇒244, 256

[63] K. Szabados, Quality Aspects of TTCN-3 Based Test Systems, *PhD thesis*, Eötvös Loránd University, 11 2017. ⇒240, 242

[64] K. Szabados, A. Kovács, G. Jenei, and D. Góbor, Titanium: Visualization of TTCN-3 system architecture, *In 2016 IEEE International Conference on Automation, Quality and Testing, Robotics* (AQTR), pages 1–5, 2016. ⇒243

[65] C. Taube-Schock, R. J. Walker, and I. H. Witten, Can we avoid high coupling?, *In Proceedings of the 25th European Conference on Object-Oriented Programming*, ECOOP'11, page 204–228, Berlin, Heidelberg, 2011. Springer-Verlag. ⇒242

[66] R. Tonelli, G. A. Pierro, M. Ortu, and G. Destefanis, Smart contracts software metrics: A first study, *PLoS ONE*, 18, 01 2023. ⇒243

[67] W. Turski, The reference model for smooth growth of software systems revisited, *IEEE Trans. Softw. Eng.*, 28(8):814–815, 2002. ⇒243

[68] J. Varajão, R. P. Marques, and A. Trigo, Project management processes – impact on the success of information systems projects, *Inf.*, 33(2):421–436, 2022. ⇒241

[69] L. Šubelj and M. Bajec, Software systems through complex networks science: Review, analysis and applications, *In Proceedings of the First International Workshop on Software Mining*, p. 9–16, NY, USA, 2012. ACM. ⇒242

[70] A. Whiteley, J. Pollack, and P. Matous, The origins of agile and iterative methods, *J. Mod. Proj.*, pages 20–29, 02 2021. ⇒241

[71] E. Yourdon and L. L. Constantine, *Structured Design: Fundamentals of a Discipline of Computer Program and Systems Design*, Yourdon, 1978. ⇒241

[72] H. Zhang, Exploring Regularity in Source Code: Software Science and Zipf's Law, *In 15th Working Conference on Reverse Engineering*, 101–110, 2008. ⇒243

[73] H. Zhang and H. B. K. Tan, An Empirical Study of Class Sizes for Large Java Systems, *In 14th Asia-Pacific Software Engineering Conference* (APSEC'07), pages 230–237, 2007. ⇒243

[74] H. Zhang, H. B. K. Tan, and M. Marchesi, The Distribution of Program Sizes and Its Implications: An Eclipse Case Study, *CoRR*, abs/0905.2288, 2009. ⇒243

[75] T. Zimmermann and N. Nagappan, Predicting Subsystem Failures using Dependency Graph Complexities, *In The 18th IEEE Int. Symp. on Soft. Rel.* (ISSRE '07), pages 227–236, 2007. ⇒243

[76] A. Zsiga, Termelékenységi trendek, minták elemzése szoftverfejlesztési projektekben, *Master's thesis*, Eötvös Loránd University, 2019. ⇒243, 244

[77] ∗ ∗ ∗, Evolved universal terrestrial radio access (e-utra) and evolved packet core (epc); user equipment (ue) conformance specification; part 3: Abstract test suite (ats), ftp://ftp.3gpp.org/Specs/archive/36_series/36.523-3/36523-3-g90.zip. Last accessed: 2023.05.15. ⇒251

[78] ∗ ∗ ∗, Internet protocol (ip) multimedia call control protocol based on session initiation protocol (sip) and session description protocol (sdp); user equipment (ue) conformance specification; part 3: Abstract test suites (ats), ftp://ftp.3gpp.org/Specs/archive/34_series/34.229-3/34229-3-g20.zip. Last accessed: 2023.05.15. ⇒251

[79] ∗ ∗ ∗, Technical specification group radio access network; 5gs; user equipment (ue) conformance specification; part 3: Protocol test suites, fftp://ftp.3gpp.org/Specs/archive/38_series/38.523-3/38523-3-g20.zip. Last accessed: 2023.05.15. ⇒251

[80] ∗∗∗, User equipment (ue) conformance specification; part 3: Abstract test suites, ftp://ftp.3gpp.org/Specs/archive/34_series/34.123-3/34123-3-g20.zip. Last accessed: 2023.05.15. ⇒250

[81] ∗∗∗, Titan, https://projects.eclipse.org/projects/tools.titan, 2020. Last accessed: January 2020. ⇒244, 260

Received: September 19, 2023 • Revised: October 25, 2023

Acta Univ. Sapientiae Informatica **15,** 2 (2023) 267–293

DOI: 10.2478/ausi-2023-0017

Methods for the graph realization problem

Zoltán KÁSA
Sapientia Hungarian University of Transylvania, Cluj-Napoca, Romania
Dept. of Mathematics and Informatics
email: kasa@ms.sapientia.ro
ORCID:0000-0002-2697-1599

Pál A. KUPÁN
Sapientia Hungarian University of Transylvania, Cluj-Napoca, Romania
Dept. of Mathematics and Informatics
email: kupanp@ms.sapientia.ro
ORCID:0000-0002-9290-3121

Csaba György PĂTCAȘ
Babeș–Bolyai University, Cluj-Napoca, Romania
Faculty of Mathematics and Informatics
email: csaba.patcas@ubbcluj.ro
ORCID:0009-0001-6485-8765

Abstract. The graph realization problem seeks an answer to how and under what conditions a graph can be constructed if we know the degrees of its vertices. The problem was widely studied by many authors and in many ways, but there are still new ideas and solutions. In this sense, the paper presents the known necessary and sufficient conditions for realization with the description in pseudocode of the corresponding algorithms. Two cases to solve the realization problem are treated: finding one solution, and finding all solutions. In this latter case a parallel approach is presented too, and how to exclude isomorphic graphs from solutions. We are also discussing algorithms using binary integer programming and flow networks.

In the case of a bigraphical list with equal out- and in-degree sequences a modified Edmonds–Karp algorithm is presented such that the resulting graph will be always symmetric without containing loops. This algorithm solves the problem of graph realization in the case of undirected graphs using flow networks.

Key words and phrases: degree sequences, graph realization algorithms, flow networks, graphical lists, bigraphical lists, modified Edmonds–Karp algorithm

1 Introduction

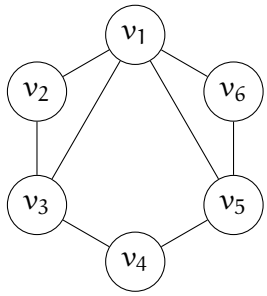
A graph realization or graph construction problem asks if for a given finite sequence $(d_1, d_2, \ldots, d_n)$ of natural numbers there exists a finite simple graph such that $d_1, d_2, \ldots, d_n$ represent the degrees of vertices of this graph. The problem has been widely studied mostly from a theoretical point of view, giving necessary and sufficient conditions for the existence of the solution [8, 7, 5, 9]. The problem can of course also be formulated for directed graphs, if we give two sets of natural numbers for the in-degrees and out-degrees [13, 11, 12].

Two cases to solve the realization problem can arise:

- finding a graph which satisfies the conditions,
- finding all graphs which satisfy the conditions.

A sequence of non-negative integers is called *graphical* if it is the degree sequence of some graph. A list $(a_1, b_1), (a_2, b_2), \ldots, (a_n, b_n)$ of pair of non-negative integers is *bigraphical*, if a_i are the out-degrees, b_i the in-degrees of the vertices of some directed graph.

For example: 4, 3, 3, 2, 2, 2 is a graphical sequence. For the corresponding graph and adjacency matrix see Fig. 1.

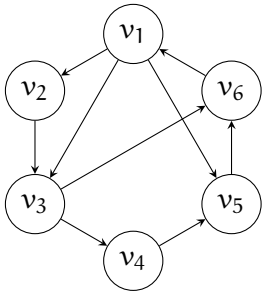


$$\begin{pmatrix} 0 & 1 & 1 & 0 & 1 & 1 \\ 1 & 0 & 1 & 0 & 0 & 0 \\ 1 & 1 & 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 & 1 & 0 \\ 1 & 0 & 0 & 1 & 0 & 1 \\ 1 & 0 & 0 & 0 & 1 & 0 \end{pmatrix}$$

Figure 1: Example of a graph with the degrees $4, 3, 3, 2, 2, 2$ and its adjacency matrix

In this article, we will use the following algorithms:

- Finding a solution
  - algorithm based on the Havel–Hakimi theorem for undirected graphs,
  - algorithm based on the Kleitman–Wang theorem for directed graphs,
  - algorithm using flow network for directed graph,
  - using binary integer programming algorithm.
- Finding all solutions
  - by parallel testing the solutions.



$$\begin{pmatrix} 0 & 1 & 1 & 0 & 1 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 1 \\ 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 \\ 1 & 0 & 0 & 0 & 0 & 0 \end{pmatrix}$$

Figure 2: Example of a digraph with the out-degrees $3,1,2,1,1,1$ and in-degrees $1,1,2,1,2,2$, so with the bigraphical list $(3,1),(1,1),(2,2),(1,1),(1,2),(1,2)$, and its adjacency matrix

2 Necessary and sufficient conditions

The first characterization of graphic sequences, an algorithmic one, was published by Havel [8] in 1955, completed by Hakami in 1962 [7]. Erdős and Gallai gave a completely different type of characterization in 1960 [5]. In 2008 Tripathi and Tyagi presented two new characterizations [19].

The running time of these algorithms is $\Omega(n^2)$ in worst case. Iványi et al. [9] have proposed a faster algorithm called EGL (Erdős-Gallai Linear algorithm), whose worst running time is $\Theta(n)$. Other characterizations can be found in [18] (1994), [1] (1997), [14] (2004), [3], and [20] (2010). In [2] and [6] related problems are discussed.

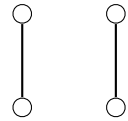
We will present here the first three characterizations which provide a necessary and sufficient condition for a sequence of natural numbers to be graphical.

Theorem 1 (Havel–Hakimi) [8, 7] *A sequence* $d_1, d_2, \ldots, d_n$ *of non-negative integers, with* $d_1 \geq d_2 \geq \cdots \geq d_n$*, where* $n \geq 2$, $d_1 \geq 1$*, is graphical if and only if the sequence*

$$d_2 - 1, d_3 - 1, \ldots, d_{d_1+1} - 1, d_{d_1+2}, d_{d_1+3}, \ldots, d_n$$

is graphical too.

Example:
4 3 3 2 2 2
4 $\boxed{3\ 3\ 2\ 2}$ 2
2 2 1 1 2
2 $\boxed{2\ 2}$ 1 1
1 1 1 1 which is graphical representing the graph:



Because the last sequence is graphical, as is illustrated in the attached figure, the first one is graphical too.

For the next theorem let us consider a sequence of non-negative integers $d_1 \geq \cdots \geq d_n$, and let us denote:

$$H_i = \sum_{k=1}^{i} d_k, \quad K_i = \sum_{k=i+1}^{n} \min(d_k, i).$$

Theorem 2 (Erdős–Gallai) [5] *A sequence of non-negative integers* $d_1 \geq \cdots \geq d_n$ *is graphical if and only if*

- H_n *is even and*
- $H_i \leq i(i-1) + K_i$ *holds for every* i, $1 \leq i \leq n-1$.

Example. The sequence 3, 3, 3, 1 is not graphical, because for $i = 2$ we have $3 + 3 > 2(2-1) + 2 + 1$.

Theorem 4 (to be discussed later) is a better applicable variant of the presented Erdős–Gallai theorem.

The following theorem applies to directed graphs.

Theorem 3 (Kleitman–Wang) [13] *Let*

$$(a_1, b_1), (a_2, b_2), \ldots, (a_n, b_n)$$

be a list of pairs of non-negative integers in non-increasing lexicographic order and a pair (a_i, b_i) *with* $b_i > 0$. *The above list is bigraphical if and only if the list obtained by the following rules is bigraphical too.*

1. *Change* (a_i, b_i) *to* $(a_i, 0)$.
2. *Let's note by* (a_k, b_k) *each of the first* b_i *pairs from the beginning of the sorted list such that* $i \neq k$. *Change all to* $(a_k - 1, b_k)$.
3. *Leave the remaining pairs as they were.*

Example. $(3,1)$, $(2,2)$, $(2,2)$, $(1,3)$ is bigraphical because:

$(3,1)$, $(2,2)$, $(2,2)$, $\boxed{(1,3)}$
$(2,1)$, $(1,2)$, $(1,2)$, $(1,0)$
$(2,1)$, $(1,2)$, $\boxed{(1,2)}$, $(1,0)$
$(1,1)$, $(0,2)$, $(1,0)$, $(1,0)$

$(1,1)$, $(1,0)$, $(1,0)$, $\boxed{(0,2)}$
$(0,1)$, $(0,0)$, $(1,0)$, $(0,0)$
$\boxed{(1,0)}$, $(0,1)$, $(0,0)$, $(0,0)$,
$(0,0)$, $(0,0)$, $(0,0)$, $(0,0)$, which is obviously bigraphical.

Number of degree sequences Antal Iványi et al. [9] have counted the n-element graphical sequences for $n \leq 32$ with a parallel approach (server and client programs) using 350 university laboratory computers operated continuously for two summer months.

3 Algorithms

To see how the question of the graph realization problem arises, let

$$d = (4, 3, 3, 2, 2, 2)$$

be a graphical sequence whose length is $n = 6$, the number of vertices of the graph. A solution graph and the corresponding adjacency matrix appear in Fig. 1.

3.1 Testing possible solutions

To solve the problem of obtaining a graph with the given degree sequence we start from the symmetric adjacency matrix

$$A = \begin{pmatrix} 0 & x_{12} & x_{13} & x_{14} & x_{15} & x_{16} \\ x_{12} & 0 & x_{23} & x_{24} & x_{25} & x_{26} \\ x_{13} & x_{23} & 0 & x_{34} & x_{35} & x_{36} \\ x_{14} & x_{24} & x_{34} & 0 & x_{45} & x_{46} \\ x_{15} & x_{25} & x_{35} & x_{45} & 0 & x_{56} \\ x_{16} & x_{26} & x_{36} & x_{46} & x_{56} & 0 \end{pmatrix},$$

where x_{ij}, $i = 1, \ldots, n-1, i < j$ denotes the edge between the nodes i and j, i.e. $x_{ij} = 1$ if there exists an edge, and $x_{ij} = 0$ otherwise. The $x_{ij} \in \{0, 1\}$ meet

the following conditions:

$$\begin{cases} x_{12}+x_{13}+x_{14}+x_{15}+x_{16} &= 4\\ x_{12}+x_{23}+x_{24}+x_{25}+x_{26} &= 3\\ x_{13}+x_{23}+x_{34}+x_{35}+x_{36} &= 3\\ x_{14}+x_{24}+x_{34}+x_{45}+x_{46} &= 2\\ x_{15}+x_{25}+x_{35}+x_{45}+x_{56} &= 2\\ x_{16}+x_{26}+x_{36}+x_{46}+x_{56} &= 2 \end{cases} \tag{1}$$

Each solution of this system of equations is a solution of the graph realization problem.
Reordering the equations in (1) we obtain the following system of linear equations:

$$\begin{cases} x_{12}+x_{13}+x_{14}+x_{15}+x_{16} & & & & & & & & = 4\\ x_{12}+ & & & x_{23}+x_{24}+x_{25}+x_{26} & & & & & = 3\\ & & & & & & & & = 3\\ & & & & & & \ddots & & = 2\\ & & & & & & & & = 2\\ & & x_{16}+ & & & x_{26}+ & & x_{36}+x_{46}+x_{56} & = 2 \end{cases} \tag{2}$$

with $x_{ij} \in \{0,1\}$, $i < j$,
or

$$Bx = d, \tag{3}$$

where $x = (x_{ij}) \in \{0,1\}^N$, $i < j$, and $B \in \{0,1\}^{n\times N}$, $N = \frac{n(n-1)}{2}$, $n > 2$,

$$B = \begin{pmatrix} 1 & 1 & 1 & 1 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0\\ 1 & 0 & 0 & 0 & 0 & 1 & 1 & 1 & 1 & 0 & 0 & 0 & 0 & 0 & 0\\ 0 & 1 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 1 & 1 & 1 & 0 & 0 & 0\\ 0 & 0 & 1 & 0 & 0 & 0 & 1 & 0 & 0 & 1 & 0 & 0 & 1 & 1 & 0\\ 0 & 0 & 0 & 1 & 0 & 0 & 0 & 1 & 0 & 0 & 1 & 0 & 1 & 0 & 1\\ 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 1 & 0 & 0 & 1 & 0 & 1 & 1 \end{pmatrix}.$$

As we see in each column there are two 1's, and in each row there are $(n-1)$ 1's. The matrix B can be decomposed as

$$B = \begin{pmatrix} E_5 & 0 & & 0 & 0 \\ & E_4 & 0 & \ddots & 0 \\ I_5 & & E_3 & 0 & \\ & I_4 & & E_2 & 0 \\ & & I_3 & I_2 & E_1 \\ & & & & I_1 \end{pmatrix}, \tag{4}$$

or in the general case

$$B = \begin{pmatrix} E_{n-1} & 0 & & 0 & 0 \\ & E_{n-2} & & \ddots & 0 \\ I_{n-1} & & & 0 & \\ & I_{n-2} & \ddots & E_2 & 0 \\ & & & I_2 & E_1 \\ & & & & I_1 \end{pmatrix}, \tag{5}$$

where $E_k = \underbrace{(\ 1\ \ 1\ \ \ldots\ \ 1\ )}_{k}$, and I_k is the identity matrix of order k.

Because $\triangle$, the minor formed with the first $(n-1)$ and the last column, differs from zero:

$$\det \triangle = \det \begin{pmatrix} E_{n-1} & 0 \\ I_{n-1} & \vdots \\ & E_1 \\ & I_1 \end{pmatrix} = 2 \cdot (-1)^{n-1},$$

it follows that the matrix B is of full rank: $\operatorname{rank}(B) = n$. For $n > 3$ the system is underdetermined, so the existence of a solution in $\{0,1\}^N$ depends on the fulfillment of the Havel-Hakimi theorem.

From (2) it follows that

$$\sum_{i<j} x_{ij} = \frac{1}{2} \sum_{k=1}^{n} d_k = m,$$

where m is the number of the edges. So, the number of 1's in each solution is m.

A solution of the system (3) can be obtained by testing binary sequences. From all 2^N, N length binary vectors, we need to test "only" $\binom{N}{m}$ vectors. In our example there are $\binom{15}{8} = 6435$ possibilities from which 27 are solutions (see Table 1). But from these solutions only 4 are not isomorphic. The isomorphic classes each containing respectively 12, 6, 6, and 3 isomorphic graphs.

One way to test a sequences is to calculate the scalar product of the binary sequence and each row of the matrix B. If all this equals the components of the $d = \begin{pmatrix} d_1 & d_2 & \dots & d_n \end{pmatrix}$ vector, then the binary sequence is a solution. But we can avoid the numerous zero operations in the dot product if we use the special shape of the sparse matrix B. We need only to add the components of the binary sequence corresponding to the 1's from the rows of the matrix B.

x_{12}	x_{13}	x_{14}	x_{15}	x_{16}	x_{23}	x_{24}	x_{25}	x_{26}	x_{34}	x_{35}	x_{36}	x_{45}	x_{46}	x_{56}
1	1	1	1	0	1	1	0	0	0	0	1	0	0	1
1	1	1	1	0	1	0	1	0	0	0	1	0	1	0
1	1	1	1	0	1	0	0	1	1	0	0	0	0	1
1	1	1	1	0	1	0	0	1	0	1	0	0	1	0
1	1	1	1	0	1	0	0	1	0	0	1	1	0	0
1	1	1	1	0	0	1	0	1	0	1	1	0	0	0
1	1	1	1	0	0	0	1	1	1	0	1	0	0	0
1	1	1	0	1	1	1	0	0	0	1	0	0	0	1
1	1	1	0	1	1	0	1	0	1	0	0	0	0	1
1	1	1	0	1	1	0	1	0	0	1	0	0	1	0
1	1	1	0	1	1	0	1	0	0	0	1	1	0	0
1	1	1	0	1	1	0	0	1	0	1	0	1	0	0
1	1	1	0	1	0	1	1	0	0	1	1	0	0	0
1	1	1	0	1	0	0	1	1	1	1	0	0	0	0
1	1	0	1	1	1	1	0	0	1	0	0	0	0	1
1	1	0	1	1	1	1	0	0	0	1	0	0	1	0
1	1	0	1	1	1	1	0	0	0	0	1	1	0	0
1	1	0	1	1	1	0	1	0	1	0	0	0	1	0
1	1	0	1	1	1	0	0	1	1	0	0	1	0	0
1	1	0	1	1	0	1	1	0	1	0	1	0	0	0
1	1	0	1	1	0	1	0	1	1	1	0	0	0	0
1	0	1	1	1	1	1	0	0	0	1	1	0	0	0
1	0	1	1	1	1	0	1	0	1	0	1	0	0	0
1	0	1	1	1	1	0	0	1	1	1	0	0	0	0
0	1	1	1	1	1	1	1	0	0	0	1	0	0	0
0	1	1	1	1	1	1	0	1	0	1	0	0	0	0
0	1	1	1	1	1	0	1	1	1	0	0	0	0	0

Table 1: 27 solutions from 6435 possible sequences

Parallel approach. When we generate the $\binom{N}{m}$ possibilities in fact the positions of the elements equal to 1 are generated. Grouping into a set the possible solutions with the same starting position, these sets can be tested in parallel, which greatly reduces the testing time.

Isomorphism. If A and B are adjacency matrices of two isomorphic graphs, then there is a permutation matrix P (each row and each column contains exactly one 1, the other elements are 0) such that $B = PAP^{-1}$.

Verifying isomorphism. We generate all permutations of $12\cdots n$. Each permutation yields a permutation matrix $P = (p_{ij})$ as follows: if $s_1 s_2 \cdots s_n$ is a permutation of $12\cdots n$, then $p_{is_i} = 1$ for $i = 1, 2, \ldots, n$. For example, the following matrix corresponds to permutation 13425:

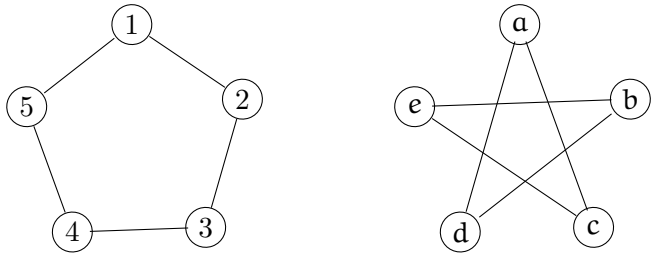
$$P = \begin{pmatrix} 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 \\ 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 \end{pmatrix}$$

Matrix PAP^{-1} can be obtained without multiplying the matrices, only by simply swapping the corresponding rows and columns as follows:

for i =1,2,. . . ,n
  for j =1,2,. . . ,n
    if $p_{ij} = 1$ **then**
      swap row i and row j in A
      swap column i and column j in A

and the obtained matrix will be B, if the graphs represented by the adjacency matrices A and B are isomorphic.

For example the following graphs are isomorphic:



$$A = \begin{pmatrix} 0 & 1 & 0 & 0 & 1 \\ 1 & 0 & 1 & 0 & 0 \\ 0 & 1 & 0 & 1 & 0 \\ 0 & 0 & 1 & 0 & 1 \\ 1 & 0 & 0 & 1 & 0 \end{pmatrix} \quad B = \begin{pmatrix} 0 & 0 & 1 & 1 & 0 \\ 0 & 0 & 0 & 1 & 1 \\ 1 & 0 & 0 & 0 & 1 \\ 1 & 1 & 0 & 0 & 0 \\ 0 & 1 & 1 & 0 & 0 \end{pmatrix} \quad P = \begin{pmatrix} 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 \\ 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 \\ 0 & 0 & 1 & 0 & 0 \end{pmatrix}$$

Based on matrix P the correspondence of the vertices are: $1 \to a$, $2 \to d$, $3 \to b$, $4 \to e$, $5 \to c$.

Parallel approach. As the permutations can be generated in parallel based on the first position, there will be n groups each of $(n-1)!$ permutations. This allows n computers to work in parallel. Thereby the execution time can be reduced from $cn!$ to $c(n-1)!$ This can be continued depending on the number of computers.

3.2 Determining a solution with binary integer programming algorithm

Integer linear programming ([17], [21]) techniques can be also used to solve the graph realization problem.

Problem (3) can be regarded as an integer linear programming problem of the form:

$$\max_x c^t x$$
$$\begin{cases} Bx \leq d \\ x_{ij} \in \{0,1\} \quad i = 1,\ldots,n-1, i < j \end{cases}.$$

Although, the original problem does not contain an objective function, this can be used to obtain different solutions. The components of the sequence c will be set to $0-1$, so they work as weights. It is worth noting that the 1's in the solutions are concentrated at the beginning (see Table (1)). This is due to the fact that the sequence d is non-increasing. In this light, let us set all components of c to zero. In this case we obtain the first solution from the Table 1:

$$sol_1 = (1\,1\,1\,1\,0\,1\,1\,0\,0\,0\,0\,1\,0\,0\,1).$$

If we set the last m components of the vector c to 1:

$$c = (0\,0\,0\,0\,0\,0\,0\,1\,1\,1\,1\,1\,1\,1\,1),$$

we obtain a solution with the most possible 1's in the last m positions (in our example there are only two such solutions):

$$sol_7 = (1\,1\,1\,1\,0\,0\,0\,1\,1\,1\,0\,1\,0\,0\,0).$$

Naturally, different sequences c does not necessarily mean different solutions.

The algorithm uses a branch-and-bound method to divide the problem into a few smaller ones, and a relaxation technique to obtain an optimal integer (binary) solution of the problem. The complexity of the algorithm is polynomial.

Algorithm 1: Linear Erdős–Gallai algorithm

Input: Sequence $d_1 \geq d_2 \geq \cdots \geq d_n > 0$, $n \geq 2$
Output: TRUE if the input sequence is graphical, and FALSE otherwise.

```
H_0 = 0
for i = 1 to n do H_i = H_{i-1} + d_i
if H_n is odd then return FALSE
d_0 = n - 1
for i = 1 to n do
    if d_i < d_{i-1} then
        for j = d_{i-1} downto d_i + 1 do w_j = i - 1
        w_{d_i} = i
    end
end
for j = d_n downto 1 do w_j = n
for i = 1 to n do
    if i ≤ w_i then
        if H_i > i(i-1) + i(w_i - i) + H_n - H_{w_i} then
            return FALSE
        end
    end
    if i > w_i then
        if H_i > i(i-1) + H_n - H_i then return FALSE
    end
end
return TRUE
```

3.3 Testing the graphical sequence property based on Erdős–Gallai theorem

Based on the Erdős–Gallai theorem in [9] a linear time algorithm is presented to check if a sequence is graphical or not. This algorithm follows directly from the next theorem proved in [9].

We need the following: for given sequence $d_1 \geq d_2 \geq \cdots \geq d_n > 0$ let $w = (w_1, \ldots, w_{n-1})$, where w_i gives the index of d_k having the maximal index among such elements of the sequence which are greater or equal to i.

Recall that $H_i = \sum_{k=1}^{i} d_k$.

Theorem 4 [9] *If* $n \geq 1$, *then the sequence* $d_1 \geq d_2 \geq \cdots \geq d_n > 0$, *is graphical if and only if*

$$H_n \quad \textit{is even}$$

and if $i > w_i$, *then*

$$H_i \leq i(i-1) + H_n - H_i,$$

further if $i \leq w_i$, *then*

$$H_i \leq i(i-1) + i(w_i - i) + H_n - H_{w_i}.$$

Algorithm 1 is based on the one described in [9], which will be prerequisite for the Havel–Hakimi graph realization algorithms. It is easy to see that it has a linear time complexity.

3.4 Graph realization problem using the Havel–Hakimi theorem

It is possible to check whether Theorem 1 is satisfied using an $O(n^2 \log n)$ time complexity algorithm by sorting the list after every step. To avoid this we can observe that after decreasing some d_1 values from the list at each step, the values can be sorted by swapping two contiguous subsequences of the list. This can be done in linear time, thus our algorithm has $O(n^2)$ time complexity. A less efficient variant with the same time complexity would be to merge the subsequence containing the decreased elements with the remainder of the list.

The sequence $z_1, z_2, \ldots, z_n$ keeps the original positions of vertices during the algorithm.

Algorithm 2 solves the problem using the Havel–Hakimi theorem.

Example. Using Algorithm 2 for the graphical sequence 4, 3, 3, 2, 2 the solution is given in Fig. 3.

The input of this algorithm must be a graphical degree sequence. A degree sequence can be checked out for this by Algorithm 1. But Algorithm 2 easily can be modified to check also that the input is graphical or not.

3.5 Graph realization problem using the Kleitman–Wang theorem

Let us denote for a vertex v_i in a digraph by a_i its out-degree, and by b_i its in-degree. The list $(a_1, b_1), (a_2, b_2), \ldots, (a_n, b_n)$ of pairs of non-negative integers is a degree sequence of some digraph. In an unusual way, here the

Algorithm 2: Graph realization using the Havel–Hakimi theorem

Input: Graphical sequence $d_1 \geq d_2 \geq \cdots \geq d_n > 0$, $n \geq 2$
Output: Adjacency matrix $A = (a_{ij})$ $(i, j = 1, 2, \ldots n)$ of the solution graph

```
for i := 1 to n do
    z_i := i
    for j := 1 to n do a_ij := 0
end
k := 1; m := n
while (m > k) do
    c := d_k
    s := −1
    for i := k + 1 to k + c do
        d_i := d_i − 1
        a_{z_k,z_i} := 1
        a_{z_i,z_k} := 1
        if s = −1 and k + c < n and d_i < d_{k+c+1} then s := i
    end
    if s > 0 then
        i := s
        j := k + c + 1
        while j ≤ n and d_i < d_j do j := j + 1
        l := k + c + 1 − s
        r := j − k − c − 1
        if l > r then j := k + c + 1
        else j := j − l
        while i ≤ k + c and j ≤ n and d_i < d_j do
            Swap d_i and d_j, respectively z_i and z_j
            i := i + 1
            j := j + 1
        end
    end
    while (d_m = 0) and (m > k) do m := m − 1
    k := k + 1
end
```

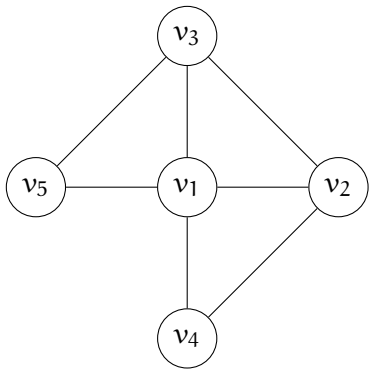


Figure 3: A solution to the graphical sequence 4, 3, 3, 2, 2

first number means the out-degree, the second the in-degree. We will see the benefits of this later.

We recall that a list $(a_1, b_1), (a_2, b_2), \ldots, (a_n, b_n)$ of pairs of nonnegative integers is called *bigraphical* if it is the degree sequence of some digraph.

We denote by $(a_1, b_1) \succeq (a_2, b_2) \succeq, \ldots, \succeq (a_n, b_n)$ a list $(a_1, b_1), (a_2, b_2), \ldots, (a_n, b_n)$ which is in non-increasing lexicographic order. Here, we use a similar idea as in Algorithm 2, sequence $z_1, z_2, \ldots, z_n$ keeps the original positions of vertices during the algorithm, and $y_1, y_2, \ldots, y_n$ denotes the inverse permutation of z, that is y_i stores the current position where the pair originally at position i can be found in the list. Because the sorting order now depends on two parameters, both values of each pair, it is not possible anymore to simply exchange two subsequences, we need the classical merging algorithm this time. We also use a queue (first in first out) called q.

This algorithm works not only for bigraphical sequences, the opposite is indicated by an error message.

Example. Using Algorithm 3 for the bigraphical sequence $(2,0)$, $(1,1)$, $(1,0)$, $(0,3)$ the solution is given in Fig. 4. In the first step of the **while** cycle from
$h = 2$ $(2,0)$, $(1,1)$, $(1,0)$, $(0,3)$, we obtain $(1,0)$, $(1,0)$, $(1,0)$, $(0,3)$,
In the next step from
$h = 4$ $(1,0)$, $(1,0)$, $(1,0)$, $(0,3)$, we obtain $(0,0)$, $(0,0)$, $(0,0)$, $(0,0)$.

3.6 Graph realization problem using flow networks

This method can be applied for directed graphs (see [16]), and for undirected graphs with some modifications. A bigraphical list (a_1, b_1), (a_2, b_2),

Algorithm 3: Digraph realization by the Kleitman–Wang theorem

Input: Bigraphical sequence $(a_1, b_1) \succeq (a_2, b_2) \succeq, \ldots, \succeq (a_n, b_n)$
Output: Adjacency matrix $X = (x_{ij})$ $(i, j = 1, 2, \ldots n)$ of the solution graph

```
for i := 1 to n do
    z_i := i
    y_i := i
    for j := 1 to n do x_ij := 0
    if b_i > 0 then  Push i to the end of q
end
while q is not empty do
    Pop the first element of q into h
    c := b_{y_h}
    if c > n − 1 then
        return Error: The sequence is not bigraphical.
    end
    s := −1
    i := 1
    while i ≤ c do
        if z_i ≠ h then
            if a_i ≤ 0 then
                return Error: The sequence is not bigraphical.
            else
                a_i := a_i − 1
                x_{z_i,h} := 1
            end
            if s = −1 and pair c + 1 ≻ pair i in the list then s := i
        else c := c + 1
        i := i + 1
    end
    if s > 0 then
        Merge a_{s...c} with a_{c+1...n} updating z and y accordingly
    end
    k := y_h
    b_k := 0
    Move pair k to the right in the list if necessary updating z and y
      accordingly
end
if a_1 > 0 then
    return Error: The sequence is not bigraphical.
end
```

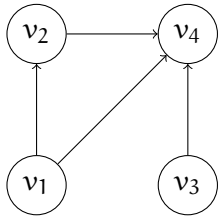


Figure 4: A solution to the bigraphical sequence $(2,0)$, $(1,1)$, $(1,0)$, $(0,3)$

$\ldots$, (a_n, b_n) is given, where as we have already seen $a_1, a_2, \ldots, a_n$ represent the out-degree, and $b_1, b_2, \ldots, b_n$ the in-degree sequences. To find a graph with these out- and in-degrees we will use a flow network of $2n+2$ nodes. Denote the source node by v, the sink node by w, and the internal nodes by $v_1, v_2, \ldots, v_n, w_1, w_2, \ldots, w_n$. The arcs are the following:

$$\begin{array}{ll} (v, v_i) & \text{for} \quad i = 1, 2, \ldots, n, \\ (v_i, w_j) & \text{for} \quad i = 1, 2, \ldots, n, j = 1, 2, \ldots, n, \\ & \quad \text{and} \quad i \neq j, \\ (v_j, w) & \text{for} \quad j = 1, 2, \ldots, n. \end{array} \tag{6}$$

The capacities are:

$$\begin{array}{lll} c(v, v_i) = a_i & \text{for} & i = 1, 2, \ldots, n, \\ c(v_i, v_j) = 1 & \text{for} & i = 1, 2, \ldots, n, j = 1, 2, \ldots, n, \\ & & \text{and } i \neq j, \\ c(v_j, w) = b_j & \text{for} & j = 1, 2, \ldots, n. \end{array} \tag{7}$$

For an example see Fig. 5.

A maximum flow in the above defined flow network can be obtained for example by the Edmonds–Karp algorithm [4]. A maximal flow in this network which saturates all arcs from v, and all arcs to w, will give us a solution to the realization problem. The arcs between the interior vertices with flow equal to 1 yield the edges of the resulting graph. A saturated arc (v_i, w_j) where $i = 1, 2, \ldots, n$, $j = 1, 2, \ldots, n$, and $i \neq j$, gives an arc (v_i, w_j) of the solution. Obviously, the solution is not unique, from any maximum flow with the above conditions, a new graph results.

Algorithm 4 constructs a graph from a bigraphical list using flow networks.

Algorithm 4: Graph realization using flow network

Input: Bigraphical list (a_1, b_1), (a_2, b_2), $\ldots$, (a_n, b_n)
Output: Adjacency matrix $A = (a_{ij})$ $(i, j = 1, 2, \ldots, n)$ of the solution graph
Construct the graph: $G = (V, E)$, where $V = \{v, w, v_1, v_2, \ldots, v_n, w_1, w_2, \ldots, w_n, \}$ and the arcs in E are given by (6) with capacities by (7)
Apply to G the Edmonds–Karp algorithm obtaining a maximum flow f

```
for i := 1 to n do
    for j := 1 to n do
        if i = j then a_ii = 0
        else a_i,j = f(v_i, w_j)
    end
end
```

The complexity of the Algorithm 4 is $O(n^5)$ because of the complexity of the Edmonds-Karp algorithm, but it can be improved to $O(n^3)$ by using the algorithm from [15].

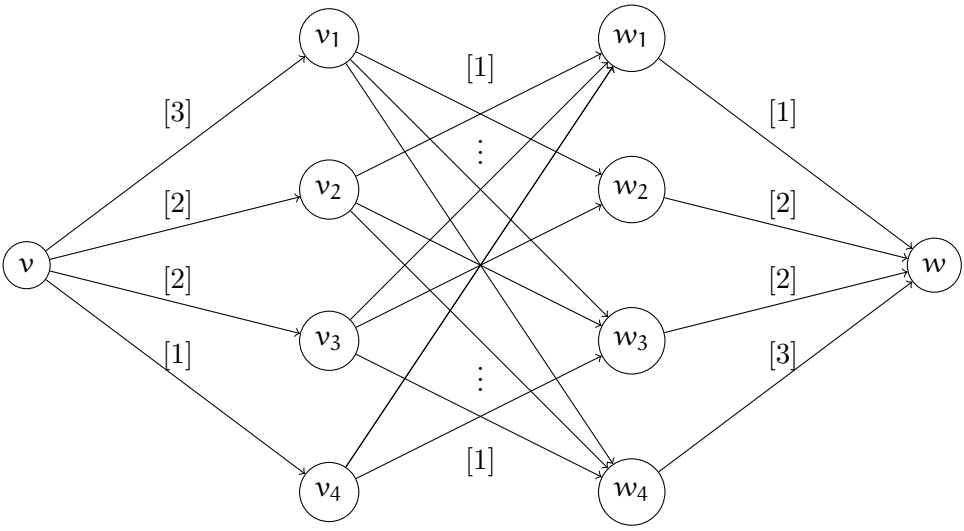


Figure 5: An example of flow network attached to the bigraphical list $(3, 1)$, $(2, 2)$, $(2, 2)$ $(1, 3)$

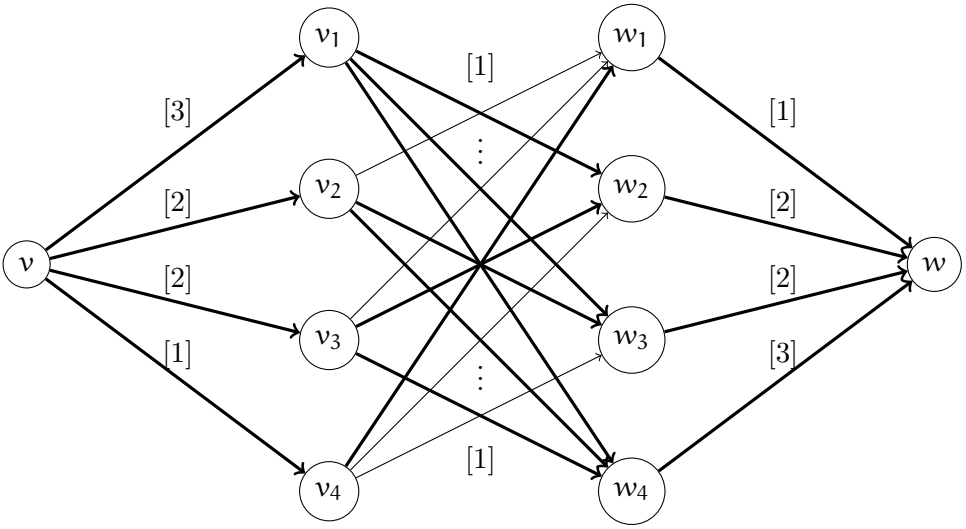


Figure 6: A solution of the example in Fig. 5, the saturated arcs are marked thick. The other arcs have flow equal to 0.

Example. Starting from the bigraphical list $(3,1)$, $(2,2)$, $(2,2)$, $(1,3)$ we obtain the flow network in Fig. 5. By the well-known Edmonds–Karp algorithm we find the maximum flow (see Fig. 6, where the thick arrows are saturated) which corresponds to the following solution, where $f(i,j)$ is the flow on the arc from vertex v_i to vertex w_j:

$f(i,j)$	w_1	w_2	w_3	w_4
v_1	0	1	1	1
v_2	0	0	1	1
v_3	0	1	0	1
v_4	1	0	0	0

and which is the adjacency matrix of the resulting graph (Fig. 7.a).

In the case of a bigraphical list with the equal out- and in-degree sequences (i.e. $a_i = b_i$, for $i = 1,\dots,n$) if the resulting graph is symmetric, then the solution can be considered as a solution for the undirected case (where $a_1, a_2, \dots, a_n$ is a graphical sequence). Such a case can be viewed in Fig. 7b. But Algorithm 4 does not always give a symmetric graph as solution. As an example consider the case of the degree sequence $(2,2,2,2,2)$. See Fig. 8a for the resulting graph. In Fig. 9 on the left we can see the maximum flow obtained. Using the following alternating semipath (which is a closed one)

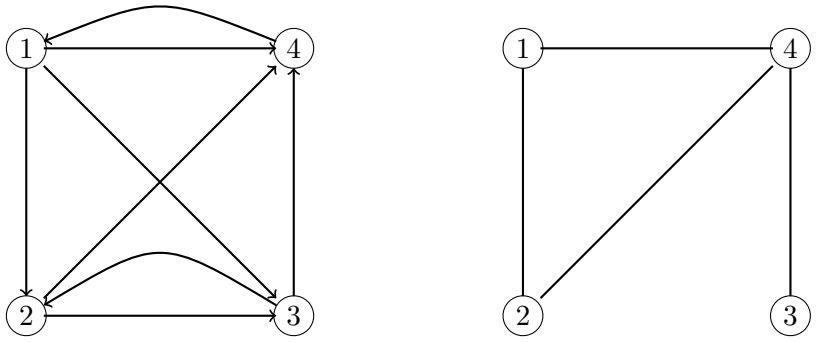
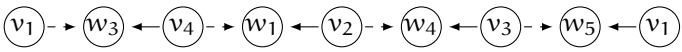


Figure 7: Solution for the bigraphical list: a) $(3,1)$, $(2,2)$, $(2,2)$, $(1,3)$, b) $(2,2)$, $(2,2)$, $(1,1)$, $(3,3)$ which corresponds to the graphical sequence $2,2,1,3$.



and changing in the maximum flow all dashed arcs from here with a thick one next to it, a new maximum flow will arise (on the right in Fig. 9). This solution yields a symmetric graph (Fig. 8.b), and the corresponding undirected graph is in Fig. 8.c. In the Appendix an algorithm for finding all closed alternating semipaths is given.

A different approach to get from a solution to another one, is based on the so called *square change* [5]. If we delete two arcs $(a,b),(c,d)$, then add the two arcs $(a,d),(c,b)$ the degree sequence does not change. In the case of Fig8*a)* the following square changes $(2,4),(3,5) \to (2,5),(3,4)$; $(1,3),(4,1) \to (1,1),(4,3)$; $(2,5),(1,1) \to (2,1),(1,5)$ will give the solution in Fig.8*b)*.

A modified Edmonds–Karp algorithm for undirected graphs. In the case of a bigraphical list with equal out- and in-degree sequences (i.e. $a_i = b_i$, for $i = 1,\ldots,n$) it is possible to modify the Edmonds–Karp algorithm in such a way that the resulting graph will be always symmetric without containing loops. In order to achieve this, we need to modify the breadth-first search by allowing only paths whose symmetric is also an augmenting path. Then each time we increase the flow along both paths.

As usual, given the edge capacities and flow values of the network, for every arc (x,y) in the original network by definition the residual network has an arc

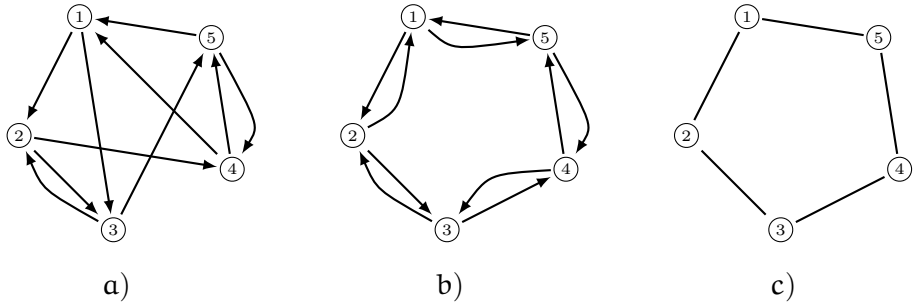


Figure 8: a) Solution for the bigraphical list $(2,2),(2,2),(2,2),(2,2),(2,2)$ given by the Algorithm 4. b) Solution after applying the alternating semipath method. c) In the solution of b) each pair of arcs (u,v) and (v,u) has been substituted by the edge $\{u,v\}$

.

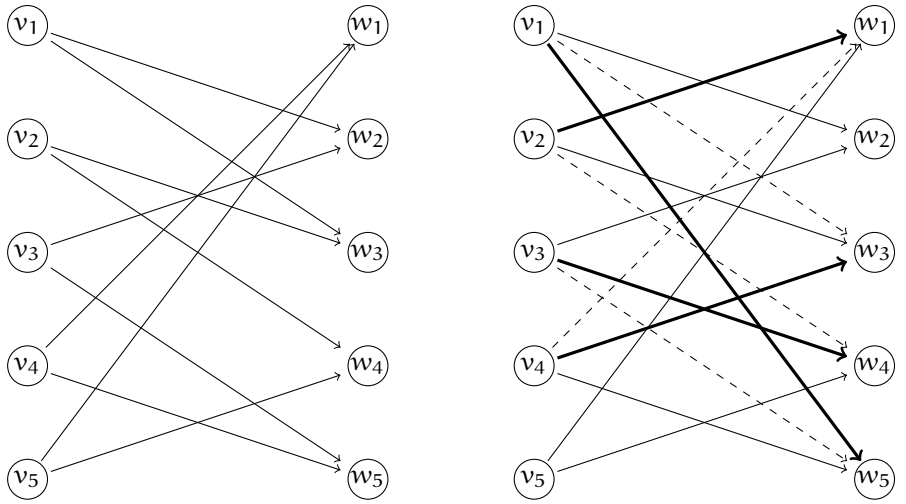


Figure 9: Using the alternating semipath method to obtain a new maximum flow which yields a symmetric graph as solution. Only the arcs which have flow equal to 1 are drawn. Eliminating the dashed arcs and introducing the thick ones, a new maximum flow is obtained.

(x,y) if $c(x,y) - f(x,y)$ is positive and an arc (y,x) if $f(x,y)$ is positive. In the residual network we set the capacities to the respective values.

Algorithm 5: Graph realization for the symmetric case

Input: Flow network (as described in (6) and (7))
Output: Adjacency matrix $A = (a_{ij})$ $(i, j = 1, 2, \ldots, n)$ of the solution graph
while BFS*(c, f, p)* **do**
  IncreaseFlow(c, f, p)
end
for $i := 1$ **to** n **do**
  for $j := 1$ **to** n **do**
    if $i = j$ **then** $a_{ii} = 0$
    else $a_{i,j} = f(v_i, w_j)$
  end
end

Algorithm 6: BFS(c, f)

Input: Flow network
Output: Parent sequence p describing one of the two paths
Push v to the end of q
while q *is not empty* **do**
  Pop the first element of q into x
  foreach y *such that* (x, y) *is an arc of the residual network* **do**
    if y *is not marked as visited* ***and*** Valid*(x, y, c, f, p, e)* **then**
      $p_y := x$
      if $x = v$ **then** $e_y := y$
      else $e_y := e_x$
      Mark y as visited and push it to the end of q
    end
  end
end
return True if w is marked as visited

In order to check whether the flow can be increased on the symmetric pair of a path, for each visited node x we store the value e_x which keeps the node that succeeds v on the path to x, thus the path will respect the following pattern: $v, e_x, \ldots, x$. The values of sequence e can be determined using simple recurrence relations, by applying dynamic programming.

Algorithm 7: VALID(x, y, c, f, p, e)

Input: Arc (x, y), flow network, sequences p and e
Output: TRUE if p_y should be set to x
if $y = w$ **then**
  Let $v_i = e_x$ and $w_j = x$
  if $v_j \neq e_x$ **then**
    return TRUE if arc (w_i, w) is in the residual network
  else
    return TRUE if arc (w_i, w) has capacity greater than 1 in the residual network
  end
else if $x = v_i$ ***and*** $y = w_j$ *for some* $i, j \in \{1, 2, \ldots, n\}$ **then**
  if (v_j, w_i) *is not in the residual network* **then return** FALSE
  Consider the path $B = v, v_{k_1} = e_x, w_{k_2}, v_{k_3}, w_{k_4}, \ldots, x$ ending in x according to p
  if $w_i \in B$ **then**
    return FALSE if v_j is the element before w_i in B
  end
else if $x = w_i$ ***and*** $y = v_j$ *for some* $i, j \in \{1, 2, \ldots, n\}$ **then**
  Check similarly to the previous case
end
return TRUE

We note that building path B introduces an additional linear factor to the time complexity to our algorithm compared to Algorithm 5, yielding to $O(n^6)$.

It's easy to see the correctness of the algorithm by the following argument. If there exists a solution, then at each step there must exist a pair of symmetric augmenting paths, and if such a pair of paths exist, the algorithm will always find one. After finding a pair of augmenting paths, the flow of the network will increase by 2, so by mathematical induction the algorithm will terminate correctly by finding a solution if one exists, when the flow of the network will be equal with twice the number of edges of the undirected graph we are looking for.

Algorithm 8: IncreaseFlow(c, f, p)

Input: Flow network, sequence p
Output: Flow network with increased flow
$y := w$
while $y \neq v$ **do**
  $x := p_y$
  Increase flow on arc (x, y)
  if $x = v_i$ *and* $y = w_j$ *for some* $i, j \in \{1, 2, \ldots, n\}$ **then**
    Increase flow on arc (v_j, w_i)
  else if $x = w_j$ *and* $y = v_i$ *for some* $i, j \in \{1, 2, \ldots, n\}$ **then**
    Increase flow on arc (w_i, v_j)
  else if $p_x = v$ **then**
    Let $v_i = x$
    Increase flow on arc (w_i, w)
  else if $y = w$ **then**
    Let $w_j = x$
    Increase flow on arc (v, v_j)
  end
  $y := x$
end

4 Conclusions

Despite the fact that the graph realization problem has been intensively studied, there are still many new ideas.

The necessary and sufficient conditions for the realization problem have been known for long, and from these it is easy to give algorithms, yet their exact description in pseudocode is not in vain, because it helps to investigate the complexity of these algorithms. We also examined the possibility of finding all solutions, excluding isomorphic graphs, and the possibility of a parallel approach for larger graph orders. The algorithm to solve the problem using network flows for directed graphs has been modified so that it can be applied to undirected graphs as well. In addition, we have presented an algorithm that solves the problem by integer linear programming.

In the Appendix a method to determine the closed alternating semipaths is presented, which can be used in the flow algorithm.

References

[1] T. M. Barnes, C. D. Savage, Efficient generation of graphical partitions, *Discrete Appl. Math.* **78,** 1-3 (1997) 17–26. ⇒269

[2] G. Chartrand, O. R. Oellermann: *Applied and Algorithmic Graph Theory*, McGraww-Hill, Inc.1993. ⇒269

[3] C. I. Del Genio, H. Kim, Z. Toroczkai, K.E. Bassler, Efficient and exact sampling of simple graphs with given arbitrary degree sequence. *PLoS ONE* 5,4 (2010) e10012. doi:10.1371/journal.pone.0010012 ⇒269

[4] J. Edmonds, R. M. Karp, Theoretical improvements in algorithmic efficiency for network flow problems, *Journal of the ACM* 19, 2 (1972) 248–264. ⇒282

[5] P. Erdős, T. Gallai, Gráfok előírt fokszámú pontokkal, *Matematikai Lapok* (in Hungarian), 11 (1960) 264–274. ⇒268, 269, 270, 285

[6] P. L. Erdős, I. Miklós: A simple Havel–Hakimi type algorithm to realize graphical degree sequences of directed graphs, *The Electronic Journal of Combinatorics* 17 (2010) #R66 ⇒269

[7] S. L. Hakimi, On realizability of a set of integers as degrees of the vertices of a linear graph I, *Journal of the Society for Industrial and Applied Mathematics*, 10 (1962) 496–506. ⇒268, 269

[8] V. Havel, A remark on the existence of finite graphs, *Časopis pro pěstování matematiky* (in Czech), 80 (1955) 477–480. ⇒268, 269

[9] A. Iványi, L. Lucz, T. F. Móri, P. Sótér: On Erdős–Gallai and Havel–Hakimi algorithms, *Acta Universitatis Sapientiae, Informatica*, **3,**, (2011) 230–268. ⇒268, 269, 271, 277, 278

[10] Z. Kása, On scattered subword complexity, *Acta Universitatis Sapientiae, Informatica*, **3,** 1 (2011) 127–136. ⇒291

[11] H. Kim, C. I. Del Genio, K. E. Bassler, Z. Toroczkai, Constructing and sampling directed graphs with given degree sequences, *New Journal of Physics* 14 (2012) 023012 (23pp) ⇒268

[12] H. Kim, Z. Toroczkai, P. L. Erdős, I. Miklós, L. A. Székely, Degree-based graph construction, *Journal of Physics A: Mathematical and Theoretical*, Volume 42, Number 39. ⇒268

[13] Kleitman, D. J.; Wang, D. L, Algorithms for constructing graphs and digraphs with given valences and factors, *Discrete Mathematics*, 6, 1 (1973) 79–88. doi:10.1016/0012-365x(73)90037-x ⇒268, 270

[14] A. Kohnert, Dominance order and graphical partitions, *Elec. J. Comb.* **11,** 1 (2004) No. 4. 17 pp. ⇒269

[15] V. M. Malhotra, M. P. Kumar, S. N. Maheshwari, An $O(|V|^3)$ algorithm for finding maximum flows in networks, *Information Processing Letters*, **7,** 6 (1978) 277–278. ⇒283
[16] M. Mihail, N. Vishnoi, On Generating Graphs with Prescribed Vertex Degrees for Complex Network Modeling, *Semantic Scolar*, Published 2003 Corpus ID: 52064211 ⇒280
[17] G. L. Nemhauser, L.A. Wolsey, *Integer and Combinatorial Optimization,* John Wiley & Sons, 1988. ⇒276
[18] F. Ruskey, R. Cohen, P. Eades, A. Scott, Alley CAT's in search of good homes, *Congr. Numer.* **102** (1994) 97–110. ⇒269
[19] A. Tripathi, H. Tyagi, A simple criterion on degree sequences of graphs, *Discrete Applied Mathematics* 156 (2008) 3513–3517. ⇒269
[20] A. Tripathi, S. Venugopalanb, D. B. West, A short constructive proof of the Erdös-Gallai characterization of graphic lists, *Discrete Math.* **310,** 4 (2010) 833–834. ⇒269
[21] L. A. Wolsey, *Integer Programming*, John Wiley & Sons, 1998. ⇒276

Appendix

Determining the closed alternating semipaths

Let $G = (V \cup W, E_1 \cup E_2)$ a bipartite digraph, where $V = \{v_1, v_2, \ldots, v_n\}$ and $W = \{w_1, w_2, \ldots, w_n\}$ are the set of vertices, E_1 the set of red arcs, E_2 the set of blue arcs, (an example is in Fig. 10). The arcs are (v_i, w_j) with $i \neq j$.

The problem is to find closed alternating semipaths in which the direction of the arcs and the colors also alternate. In Fig. 10 such a semipaths is:

Before presenting the algorithm, let us recall some notations ([10]) that will be used.

Let us consider a matrix $\mathcal{A}$ with the elements A_{ij} which are sets of strings. Initially elements of this matrix for $i, j = 1, 2, \ldots, n$ are defined as:

$$A_{ij} = \begin{cases} \{v_i w_j\}, & \text{if there exists an arc from } v_i \text{ to } w_j, \\ \emptyset, & \text{otherwise,} \end{cases} \tag{8}$$

If $\mathcal{A}$ and $\mathcal{B}$ are sets of strings, $\mathcal{AB}$ will be formed by the set of concatenation of each string from $\mathcal{A}$ with each string from $\mathcal{B}$, if they have no common letters:

$$\mathcal{AB} = \{ab \mid a \in \mathcal{A}, b \in \mathcal{B}, \quad \text{if } a \text{ and } b \text{ have no common letters}\}.$$

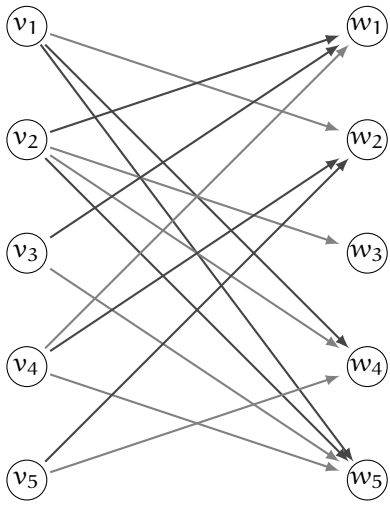


Figure 10:

If $s = s_1 s_2 \cdots s_p$ is a string, let us denote by $'s$ the string obtained from s by eliminating the first character: $'s = s_2 s_3 \cdots s_p$. Let us denote by $'A_{ij}$ the set A_{ij} in which we eliminate from each element the first character. In this case $'\mathcal{A}$ is a matrix with elements $'A_{ij}$.

Let us define for red and blue spanning subgraph of G respectively the matrices $\mathcal{R}$ and $\mathcal{B}$ as in the equation (8). $'\mathcal{R}$ and $'\mathcal{B}$ are defined as above. $\mathcal{B}^T$ represents the transposed matrix of $\mathcal{B}$ in which each element $v_i w_j$ is changed in $w_j v_i$.

The elements of the matrix

$$\mathcal{R}\Big('(\mathcal{B}^T)'\mathcal{R}\Big)^k, \quad \text{for } k = 1, 2, \ldots, n-1$$

are sets of strings of the form $s_1 s_2 \cdots s_{2k+1}$ (an alternating semipath). If there exists a blue arc (s_1, a_{2k+1}) then $s_1 s_2 \cdots s_{2k+1} s_1$ is a closed alternating semipaths. Algorithm 9 can be easily generalized to matrices of type $m \times n$.

Algorithm 9: Finding closed alternating semipaths

Input: Matrices $\mathcal{R}$ and $\mathcal{B}$ of type $n \times n$
Output: The closed alternating semipaths
$\mathcal{Y} := \mathcal{R}$
$\mathcal{X} := {}'(\mathcal{B}^{T})\,'\mathcal{R}$
for $k := 1$ **to** $n - 1$ **do**
  $\mathcal{Y} := \mathcal{Y}\mathcal{X}$
  for *each string* $s_1 s_2 \cdots s_{2k+1}$ *in each element of* $\mathcal{Y}$ **do**
    if *there exists a blue arc* (s_1, s_{2k+1}) **then**
      print $s_1 s_2 \cdots s_{2k+1} s_1$
    end
  end
end

For the example in Fig. 10 the initial matrices are

$$\mathcal{R} = \begin{pmatrix} \emptyset & \{v_1, w_2\} & \emptyset & \emptyset & \emptyset \\ \emptyset & \emptyset & \{v_2, w_3\} & \{v_2, w_4\} & \emptyset \\ \emptyset & \emptyset & \emptyset & \emptyset & \{v_3, w_5\} \\ \{v_4, w_1\} & \emptyset & \emptyset & \emptyset & \{v_4, w_5\} \\ \emptyset & \emptyset & \emptyset & \{v_5, w_4\} & \emptyset \end{pmatrix}$$

$$\mathcal{B} = \begin{pmatrix} \emptyset & \emptyset & \emptyset & \{v_1, w_4\} & \{v_1, w_5\} \\ \{v_2, w_1\} & \emptyset & \emptyset & \emptyset & \{v_2, w_5\} \\ \{v_3, w_1\} & \emptyset & \emptyset & \emptyset & \emptyset \\ \emptyset & \{v_4, w_2\} & \emptyset & \emptyset & \emptyset \\ \emptyset & \{v_5, w_2\} & \emptyset & \emptyset & \emptyset \end{pmatrix}$$

and the algorithm gives us the following closed alternating semipaths of length 4, 6 and 8 respectively (closed alternating semipath of length 10 can not exist):

- $v_1w_2v_5w_4v_1$, $v_5w_4v_1w_2v_5$,
- $v_1w_2v_4w_1v_2w_4v_1$, $v_1w_2v_4w_5v_2w_4v_1$, $v_2w_4v_1w_2v_4w_1v_2$, $v_2w_4v_1w_2v_4w_5v_2$, $v_4w_1v_2w_4v_1w_2v_4$, $v_4w_5v_2w_4v_1w_2v_4$,
- $v_1w_2v_4w_1v_3w_5v_2w_4v_1$, $v_2w_4v_1w_2v_4w_1v_3w_5v_2$, $v_3w_5v_2w_4v_1w_2v_4w_1v_3$, $v_4w_1v_3w_5v_2w_4v_1w_2v_4$.

From these only the following are different:
$v_1w_2v_5w_4v_1$, $v_1w_2v_4w_1v_2w_4v_1$, $v_1w_2v_4w_5v_2w_4v_1$, $v_1w_2v_4w_1v_3w_5v_2w_4v_1$.

Received: June 25, 2023 • Revised: September 25, 2023

Acta Univ. Sapientiae Informatica **15,** 2 (2023) 294–305

DOI: 10.2478/ausi-2023-0018

The eccentricity-based topological indices

Gul OZKAN KIZILIRMAK
Gazi University
Ankara, Turkey
email: gulozkan@gazi.edu.tr

Abstract. The aim of this paper is to obtain some relationships between eccentricity-based topological indices as the eccentric connectivity, connective eccentricity, total eccentricity, second Zagreb eccentricity, first Zagreb eccentricity connectivity, first eccentricity connectivity and first Zagreb eccentricity connectivity of a simple connected graph.

1 Introduction

Let $\mathcal{G}$ denote a graph with k vertices and s edges, which has the vertex and edge sets as $V(\mathcal{G})$ and $E(\mathcal{G})$, respectively. The number of edges connected to vertex i is denoted as the degree of i and shown as $d(i)$. The minimum and maximum vertex degrees are represented by δ and Δ, respectively. In this study, we are interested in simple undirected graph $\mathcal{G}$ which consists of no loops and multiple edges.

In the literature, there are many interesting studies in graph theory related to the distance of any two vertices. The eccentricity $\epsilon(t)$ of a vertex $t \in V(\mathcal{G})$ is defined as the maximum distance between t and any other vertex y in $\mathcal{G}$ and shown as $\epsilon(t) = max\{d(t,y) : y \in V(\mathcal{G})\}$. The maximum and minimum eccentricities over all vertices of $\mathcal{G}$ are called the diameter $d = diam(\mathcal{G})$ and the radius $r = rad(\mathcal{G})$ of $\mathcal{G}$, respectively [3, 7].

Key words and phrases: Eccentricity, topological indices

It is known that topological indices can be used to characterize of a graph. One of the most studied indices is the first Zagreb index $\mathcal{M}_1(\mathcal{G}) = \sum_{r\in V(\mathcal{G})} d(r)^2$.

There exist some studies on eccentricity based topological indices in the literature [1, 2, 13, 14]. One of them is the eccentric connectivity index and was introduced by Sharma et al. [12], which was defined as

$$\xi^c(\mathcal{G}) = \sum_{r\in V(\mathcal{G})} d(r)\epsilon(r).$$

Similarly, the connective eccentricity index of a graph $\mathcal{G}$ was defined in [6] and denoted as

$$\xi^{ce}(\mathcal{G}) = \sum_{r\in V(\mathcal{G})} \frac{d(r)}{\epsilon(r)}.$$

Also, the total eccentricity index was introduced by Farooq et al. [4] as:

$$\zeta(\mathcal{G}) = \sum_{r\in V(\mathcal{G})} \epsilon(r),$$

and moreover the first and second Zagreb eccentricity indices were defined in [5] as:

$$E_1(\mathcal{G}) = \sum_{r\in V(\mathcal{G})} \epsilon^2(r).$$

and

$$E_2(\mathcal{G}) = \sum_{rs\in E(\mathcal{G})} \epsilon(r)\epsilon(s).$$

Motivated by the eccentric-connectivity index, the first Zagreb eccentricity connectivity index $\mathcal{M}^1_{\mathcal{ECI}}$, the first eccentricity connectivity index $\mathcal{ECI}^1$ and the first Zagreb eccentricity connectivity index $\mathcal{M}^1_{\mathcal{ECI}^1}$ were introduced in [8] as:

$$\mathcal{M}^1_{\mathcal{ECI}}(\mathcal{G}) = \sum_{r\in V(\mathcal{G})} d^2(r)\epsilon(r).$$

$$\mathcal{ECI}^1(\mathcal{G}) = \sum_{r\in V(\mathcal{G})} d(r)\epsilon^2(r).$$

$$\mathcal{M}^1_{\mathcal{ECI}^1}(\mathcal{G}) = \sum_{r\in V(\mathcal{G})} d^2(r)\epsilon^2(r).$$

In this paper, some relationships between eccentricity-based topological indices are obtained in simple connected graphs.

Now, we give some lemmas. Both of these lemmas are crucial in proving the main results of this paper.

Lemma 1 [10] *If* t_j *and* y_j *(*$1 \leq j \leq k$*) are non-negative real numbers, then*

$$\sum_{j=1}^{k}(t_j)^2\sum_{j=1}^{k}(y_j)^2-\left(\sum_{j=1}^{k}t_jy_j\right)^2\leq\frac{k^2}{4}(M_1M_2-m_1m_2)^2,$$

where $M_1=\max_{1\leq j\leq k}\{t_j\}$, $M_2=\max_{1\leq j\leq k}\{y_j\}$; $m_1=\min_{1\leq j\leq k}\{t_j\}$, $m_2=\min_{1\leq j\leq k}\{y_j\}$.

Lemma 2 [11] *If* $c_j>0$, $d_j>0$, $p>0$, $j=1,2,...,k$, *then the following inequality holds:*

$$\sum_{j=1}^{k}\frac{c_j^{p+1}}{d_j^p}\geq\frac{\left(\sum\limits_{j=1}^{k}c_j\right)^{p+1}}{\left(\sum\limits_{j=1}^{k}d_j\right)^p}$$

with equality if and only if $\frac{c_1}{d_1}=\frac{c_2}{d_2}=...=\frac{c_k}{d_k}$.

Lemma 3 [9] *Let* $c_1,c_2,...,c_k$ *and* $d_1,d_2,...,d_k$ *be real numbers such that* $c\leq c_j\leq C$ *and* $d\leq d_j\leq D$ *for* $i=1,2,...,k$. *Then there holds*

$$\left|\frac{1}{k}\sum_{j=1}^{k}c_jd_j-\left(\frac{1}{k}\sum_{j=1}^{k}c_j\right)\left(\frac{1}{k}\sum_{j=1}^{k}d_j\right)\right|\leq\frac{1}{k}\left\lfloor\frac{k}{2}\right\rfloor\left(1-\frac{1}{k}\left\lfloor\frac{k}{2}\right\rfloor\right)(C-c)(D-d).$$

2 Main results

Theorem 4 *Let* $\mathcal{G}$ *be a simple connected graph with* k *vertices. Then we obtain*

$$\mathcal{M}_1(\mathcal{G})E_1(\mathcal{G})\leq(\xi^c(G))^2+\frac{k^2}{4}(\Delta d-\delta r)^2.$$

The equality holds for $\mathcal{G}\cong K_n$.

Proof. In Lemma 1, if we take $t_j=d(j)$ and $y_j=\epsilon(j)$, we get

$$\sum_{j=1}^{k}(d(j))^2\sum_{j=1}^{k}(\epsilon(j))^2-\left(\sum_{j=1}^{k}d(j)\epsilon(j)\right)^2 \leq \frac{k^2}{4}(\max(d(j))\max(\epsilon(j))-\min(d(j))\min(\epsilon(j)))^2.$$

By using the definitions of $\mathcal{M}_1(\mathcal{G}), E_1(\mathcal{G})$ and $\xi^{ce}(\mathcal{G})$, we have

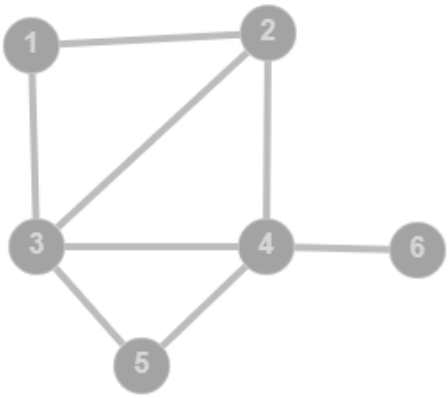
$$\mathcal{M}_1(\mathcal{G})E_1(\mathcal{G})-(\xi^c(\mathcal{G})^2 \leq \frac{k^2}{4}(\max(d(j))\max(\epsilon(j))-\min(d(j))\min(\epsilon(j)))^2.$$

Since $\max(d(j)) = \Delta$, $\max(\epsilon(j)) = D$, $\min(d(j)) = \delta$ and $\min(\epsilon(j)) = r$, we obtain

$$\mathcal{M}_1(\mathcal{G})E_1(\mathcal{G}) \leq (\xi^c(G))^2+\frac{k^2}{4}(\Delta D-\delta r)^2.$$

□

Example 5 *Let $\mathcal{G}$ be a simple connected graph with 6 vertices as follows.*



Then, we get $\mathcal{M}_1(\mathcal{G}) = 50, E_1(\mathcal{G}) = 34$ and $\xi^c(G) = 35$.

Since $\Delta = 4, D = 3, \delta = 1$ and $r = 2$, we obtain $\mathcal{M}_1(\mathcal{G})E_1(\mathcal{G}) = 1700$ and $(\xi^c(G))^2+\frac{k^2}{4}(\Delta D-\delta r)^2 = 1989$. Thus the inequality in Theorem 4 is satisfied.

Theorem 6 *If $\mathcal{G}$ is a simple connected graph with k vertices and s edges, then we get*

$$\xi^c(\mathcal{G})\xi^{ce}(\mathcal{G}) \leq 4s^2 + \frac{k^2}{4}\left(\Delta\sqrt{\frac{d}{r}} - \delta\sqrt{\frac{r}{d}}\right)^2.$$

The equality holds for $\mathcal{G} \cong K_n$.

Proof. In Lemma 1, we let $t_j = \sqrt{d(j)\epsilon(j)}$ and $y_j = \sqrt{\frac{d(j)}{\epsilon(j)}}$ to get

$$\sum_{j=1}^{k} d(j)\epsilon(j) \sum_{j=1}^{k} \frac{d(j)}{\epsilon(j)} - \left(\sum_{j=1}^{k} d(j)\right)^2 \leq \frac{k^2}{4}\left(\Delta\sqrt{\frac{d}{r}} - \delta\sqrt{\frac{r}{d}}\right)^2.$$

Since $\left(\sum_{j=1}^{k} d(j)\right)^2 = 4s^2$ and from the definitions of $\xi^c(\mathcal{G})$ and $\xi^{ce}(\mathcal{G})$, we get

$$\xi^c(\mathcal{G})\xi^{ce}(\mathcal{G}) \leq 4s^2 + \frac{k^2}{4}\left(\Delta\sqrt{\frac{d}{r}} - \delta\sqrt{\frac{r}{d}}\right)^2.$$

□

Theorem 7 *Let $\mathcal{G}$ be a simple connected graph with k vertices and s edges. Then we have*

$$\left|\frac{1}{k}\xi^c(\mathcal{G}) - \frac{2s}{k^2}\zeta(\mathcal{G})\right| \leq \frac{1}{4}\left(1 - \frac{1+(-1)^{k+1}}{2k^2}\right)(d-r)(\Delta-\delta).$$

The equality holds for $\mathcal{G} \cong K_n$.

Proof.

By using $r \leq \epsilon(j) \leq d$ and $\delta \leq d(j) \leq \Delta$ and choosing $c_j = \epsilon(j)$ and $d_j = d(j)$ in Lemma 3, we get

$$\left|\frac{1}{k}\sum_{j=1}^{k} \epsilon(j)d(j) - \left(\frac{1}{k}\sum_{j=1}^{k} \epsilon(j)\right)\left(\frac{1}{k}\sum_{j=1}^{k} d(j)\right)\right| \leq \frac{1}{k}\left\lfloor\frac{k}{2}\right\rfloor\left(1 - \frac{1}{k}\left\lfloor\frac{k}{2}\right\rfloor\right)(d-r)(\Delta-\delta).$$

Using the definitions of $\xi^c(\mathcal{G})$ and $\zeta(\mathcal{G})$, we get

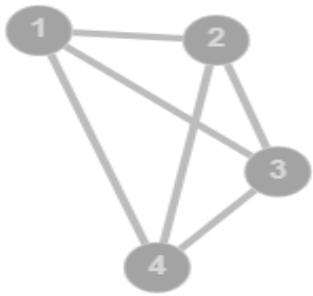
$$\left|\frac{1}{k}\xi^c(\mathcal{G}) - \frac{2s}{k^2}\zeta(\mathcal{G})\right| \leq \frac{1}{k}\left\lfloor\frac{k}{2}\right\rfloor\left(1 - \frac{1}{k}\left\lfloor\frac{k}{2}\right\rfloor\right)(d-r)(\Delta-\delta).$$

Since $\left\lfloor\frac{k}{2}\right\rfloor\left(1-\frac{1}{k}\left\lfloor\frac{k}{2}\right\rfloor\right) = \frac{k}{4}\left(1-\frac{1+(-1)^{k+1}}{2k^2}\right)$, we obtain

$$\left|\frac{1}{k}\xi^c(\mathcal{G}) - \frac{2s}{k^2}\zeta(\mathcal{G})\right| \leq \frac{1}{4}\left(1 - \frac{1+(-1)^{k+1}}{2k^2}\right)(d-r)(\Delta-\delta).$$

□

Example 8 *Let's consider* $\mathcal{G} = K_4$ *complete graph as follows.*



We can calculate as $\xi^c(\mathcal{G}) = 12$ *and* $\zeta(\mathcal{G}) = 4$. *Since* $\Delta = 3, d = 1, \delta = 3$ and $r = 1$, *we obtain*

$$\left|\frac{1}{k}\xi^c(\mathcal{G}) - \frac{2s}{k^2}\zeta(\mathcal{G})\right| = 0$$

and

$$\frac{1}{4}\left(1 - \frac{1+(-1)^{k+1}}{2k^2}\right)(d-r)(\Delta-\delta) = 0.$$

Hence, the equality holds.

Theorem 9 *If* $\mathcal{G}$ *is a simple connected graph, then we obtain*

$$\mathcal{M}^1_{\mathcal{ECI}}(\mathcal{G})\zeta(\mathcal{G}) \geq (\xi^c(\mathcal{G}))^2.$$

The equality holds for $\mathcal{G} \cong K_n$.

Proof. In Lemma 2, letting $c_j = \epsilon(j)d(j)$, $d_j = \epsilon(j)$ and $p = 1$ gives

$$\sum_{j=1}^{k} \frac{(\epsilon(j)d(j))^2}{\epsilon(j)} \geq \frac{\left(\sum_{j=1}^{k} \epsilon(j)d(j)\right)^2}{\sum_{j=1}^{k} \epsilon(j)}.$$

So, we have

$$\mathcal{M}^1_{\mathcal{ECI}}(\mathcal{G}) \geq \frac{(\xi^c(\mathcal{G}))^2}{\zeta(\mathcal{G})}.$$

Hence, it follows that

$$\mathcal{M}^1_{\mathcal{ECI}}(\mathcal{G})\zeta(\mathcal{G}) \geq (\xi^c(\mathcal{G}))^2.$$

□

Theorem 10 *Let $\mathcal{G}$ be a simple connected graph with k vertices. Then we have*

$$E_1(\mathcal{G}) \leq \frac{1}{k\delta^4}(\mathcal{M}^1_{\mathcal{ECI}}(\mathcal{G}))^2 + \frac{k}{4\delta^4}(d\Delta^2 - r\delta^2)^2.$$

The equality holds for $\mathcal{G} \cong K_n$.

Proof. In Lemma 1, we choose $t_j = \epsilon(j)$ and $y_j = (d(j))^2$ to get

$$\sum_{i=1}^{k} (\epsilon(j))^2 \sum_{j=1}^{k} ((d(j))^2)^2 - \left(\sum_{j=1}^{k} \epsilon(j)d(j))^2\right)^2 \leq \frac{k^2}{4}(\max(\epsilon(j))\max((d(j))^2) - \min(\epsilon(j))\min((d(j))^2)^2.$$

Since $\sum_{j=1}^{k} ((d(j))^2)^2 \geq k\delta^4$, we have

$$k\delta^4 E_1(\mathcal{G}) - (\mathcal{M}^1_{\mathcal{ECI}}(\mathcal{G}))^2 \leq \frac{k^2}{4}(d\Delta^2 - r\delta^2)^2.$$

Hence, we obtain

$$E_1(\mathcal{G}) \leq \frac{1}{k\delta^4}(\mathcal{M}^1_{\mathcal{ECI}}(\mathcal{G}))^2 + \frac{k}{4\delta^4}(d\Delta^2 - r\delta^2)^2.$$

□

Theorem 11 *If $\mathcal{G}$ is a simple connected graph with k vertices, then we get*

$$\left|\frac{1}{k}\mathcal{M}^1_{\mathcal{ECI}}(\mathcal{G}) - \frac{1}{k^2}\zeta(\mathcal{G})\mathcal{M}_1(\mathcal{G})\right| \leq \frac{1}{4}(1 - \frac{1+(-1)^{k+1}}{2k^2})(d-r)(\Delta^2-\delta^2).$$

The equality holds for $\mathcal{G} \cong K_n$.

Proof. It is known that $r \leq \epsilon(j) \leq d$ and $\delta^2 \leq (d(j))^2 \leq \Delta^2$. So, we let $c_j = \epsilon(j)$ and $d_j = (d(j))^2$ in Lemma 3, then

$$\left|\frac{1}{k}\sum_{j=1}^{k}\epsilon(j)(d(j))^2 - \left(\frac{1}{k}\sum_{j=1}^{k}\epsilon(j)\right)\left(\frac{1}{k}\sum_{j=1}^{k}(d(j))^2\right)\right| \leq \frac{1}{k}\left\lfloor\frac{k}{2}\right\rfloor\left(1-\frac{1}{k}\left\lfloor\frac{k}{2}\right\rfloor\right)(d-r)(\Delta^2-\delta^2).$$

Using the definitions of $\mathcal{M}^1_{\mathcal{ECI}}(\mathcal{G})$, $\zeta(\mathcal{G})$ and $\mathcal{M}_1(\mathcal{G})$, we get

$$\left|\frac{1}{k}\mathcal{M}^1_{\mathcal{ECI}}(\mathcal{G}) - \frac{1}{k^2}\zeta(\mathcal{G})\mathcal{M}_1(\mathcal{G})\right| \leq \frac{1}{k}\left(\frac{k}{4}\left(1-\frac{1+(-1)^{k+1}}{2k^2}\right)\right)(d-r)(\Delta^2-\delta^2).$$

Thus, we obtain

$$\left|\frac{1}{k}\mathcal{M}^1_{\mathcal{ECI}}(\mathcal{G}) - \frac{1}{k^2}\zeta(\mathcal{G})\mathcal{M}_1(\mathcal{G})\right| \leq \frac{1}{4}\left(1-\frac{1+(-1)^{k+1}}{2k^2}\right)(d-r)(\Delta^2-\delta^2).$$

□

Example 12 *Let's consider $\mathcal{G} = K_4$ complete graph in Example 8. Since $\Delta = 3, d = 1, \delta = 3$ and $r = 1$, the right side of the inequality in Theorem 11 is 0. Since $\mathcal{M}^1_{\mathcal{ECI}}(\mathcal{G}) = 36, \zeta(\mathcal{G}) = 4$ and $\mathcal{M}_1(\mathcal{G}) = 36$, we have*

$$\left|\frac{1}{k}\mathcal{M}^1_{\mathcal{ECI}}(\mathcal{G}) - \frac{1}{k^2}\zeta(\mathcal{G})\mathcal{M}_1(\mathcal{G})\right| = 0$$

Hence, the equality holds.

Theorem 13 *If $\mathcal{G}$ is a simple connected graph with k vertices, then we obtain*

$$\mathcal{M}_1(\mathcal{G}) \leq \frac{1}{kr^4}(\mathcal{ECI}^1(\mathcal{G}))^2 + \frac{k}{4r^4}(d^2\Delta - r^2\delta)^2.$$

The equality holds for $\mathcal{G} \cong K_n$.

Proof. In Lemma 1, we choose $t_j = (\epsilon(j))^2$ and $y_j = d(j)$ and get

$$\sum_{j=1}^{k}(\epsilon(j))^4\sum_{j=1}^{k}(d(j))^2-\left(\sum_{j=1}^{k}(\epsilon(j))^2d(j)\right)^2 \leq \frac{k^2}{4}(\max(\epsilon(j))^2\max(d(j))-\min(\epsilon(j))^2\min(d(j))^2.$$

Then,

$$kr^4\mathcal{M}_1(\mathcal{G})-(ECI^1(G))^2\leq\frac{k^2}{4}(d^2\Delta-r^2\delta)^2.$$

So, we obtain

$$\mathcal{M}_1(\mathcal{G})\leq\frac{1}{kr^4}(\mathcal{ECI}^1(\mathcal{G}))^2+\frac{k}{4r^4}(d^2\Delta-r^2\delta)^2.$$

□

Theorem 14 *Let $\mathcal{G}$ be a simple connected graph with k vertices and s edges. Then we have*

$$\left|\frac{1}{k}\mathcal{ECI}^1(\mathcal{G})-\frac{2s}{k^2}E_1(\mathcal{G})\right|\leq\frac{1}{4}\left(1-\frac{1+(-1)^{k+1}}{2k^2}\right)(d^2-r^2)(\Delta-\delta).$$

The equality holds for $\mathcal{G}\cong K_n$.

Proof. By using $r^2\leq(\epsilon(j))^2\leq d^2$ and $\delta\leq d(j)\leq\Delta$. We let $c_j=(\epsilon(j))^2$ and $d_j=d(j)$ in Lemma 3, then

$$\left|\frac{1}{k}\sum_{j=1}^{k}(\epsilon(j))^2d(j)-\left(\frac{1}{k}\sum_{j=1}^{k}(\epsilon(j))^2\right)\left(\frac{1}{k}\sum_{j=1}^{k}d(j)\right)\right| \leq\frac{1}{k}\left\lfloor\frac{k}{2}\right\rfloor\left(1-\frac{1}{k}\left\lfloor\frac{k}{2}\right\rfloor\right)(d^2-r^2)(\Delta-\delta).$$

By using the definitions of $\mathcal{ECI}^1(\mathcal{G})$ and $E_1(\mathcal{G})$, we obtain

$$\left|\frac{1}{k}\mathcal{ECI}^1(\mathcal{G})-\frac{1}{k}E_1(\mathcal{G})\frac{2s}{k}\right|=\left|\frac{1}{k}\mathcal{ECI}^1(\mathcal{G})-\frac{2s}{k^2}E_1(\mathcal{G})\right| \leq\frac{1}{4}\left(1-\frac{1+(-1)^{k+1}}{2k^2}\right)(d^2-r^2)(\Delta-\delta).$$

□

Theorem 15 *If $\mathcal{G}$ is a simple connected graph with k vertices, then we obtain*

$$\frac{k^2}{4}(3r^4\delta^4 - d^2\Delta^2(d^2\Delta^2 - 2r^2\delta^2)) \leq (\mathcal{M}^1_{\mathcal{ECI}^1}(\mathcal{G}))^2.$$

The equality holds for $\mathcal{G} \cong K_n$.

Proof. If we choose $t_j = (\epsilon(j))^2$ and $y_j = (d(j))^2$ in Lemma 1, we have

$$\sum_{i=j}^{k}(\epsilon(j))^4 \sum_{j=1}^{k}(d(j))^4 - \left(\sum_{j=1}^{k}(\epsilon(j))^2(d(j))^2\right)^2 \leq \frac{j^2}{4}(\max(\epsilon(j))^2 \max(d(j))^2 - \min(\epsilon(j))^2 \min(d(j))^2)^2.$$

So, we get

$$kr^4\delta^4 - (\mathcal{M}^1_{\mathcal{ECI}^1}(\mathcal{G}))^2 \leq \frac{k^2}{4}(d^2\Delta^2 - r^2\delta^2)^2.$$

After simplifying the above expression, we get the desired result as

$$\frac{k^2}{4}(3r^4\delta^4 - d^2\Delta^2(d^2\Delta^2 - 2r^2\delta^2)) \leq (\mathcal{M}^1_{\mathcal{ECI}^1}(\mathcal{G}))^2.$$

□

Theorem 16 *If $\mathcal{G}$ is a simple connected graph with k vertices, then we get*

$$\left|\frac{1}{k}\mathcal{M}^1_{\mathcal{ECI}^1}(\mathcal{G}) - \frac{1}{k^2}E_1(\mathcal{G})\mathcal{M}_1(\mathcal{G})\right| \leq \frac{1}{4}\left(1 - \frac{1 + (-1)^{k+1}}{2k^2}\right)(d^2 - r^2)(\Delta^2 - \delta^2).$$

The equality holds for $\mathcal{G} \cong K_n$.

Proof. We use the inequalities $r^2 \leq (\epsilon(j))^2 \leq d^2$ and $\delta^2 \leq (d(j))^2 \leq \Delta^2$. In Lemma 3, we choose $c_j = (\epsilon(j))^2$ and $d_j = (d(j))^2$, we get

$$\left|\frac{1}{k}\sum_{j=1}^{k}(\epsilon(j))^2(d(j))^2 - \left(\frac{1}{k}\sum_{j=1}^{k}(\epsilon(j))^2\right)\left(\frac{1}{k}\sum_{j=1}^{k}(d(j))^2\right)\right| \leq \frac{1}{k}\left\lfloor\frac{k}{2}\right\rfloor\left(1 - \frac{1}{k}\left\lfloor\frac{k}{2}\right\rfloor\right)(d^2 - r^2)(\Delta^2 - \delta^2).$$

Hence, we obtain

$$\left|\frac{1}{k}\mathcal{M}^1_{\mathcal{ECI}^1}(\mathcal{G}) - \frac{1}{k^2}E_1(\mathcal{G})\mathcal{M}_1(\mathcal{G})\right| \leq \frac{1}{4}\left(1 - \frac{1+(-1)^{k+1}}{2k^2}\right)(d^2-r^2)(\Delta^2-\delta^2).$$

□

Example 17 *Let's consider the graph in Example 5. Then, we have* $\mathcal{M}_1(\mathcal{G}) = 50, E_1(\mathcal{G}) = 34$ *and* $\mathcal{M}^1_{\mathcal{ECI}^1}(\mathcal{G}) = 225$. *Since* $\Delta = 4, d = 3, \delta = 1$ and $r = 2$, *we get*

$$\left|\frac{1}{k}\mathcal{M}^1_{\mathcal{ECI}^1}(\mathcal{G}) - \frac{1}{k^2}E_1(\mathcal{G})\mathcal{M}_1(\mathcal{G})\right| = 9.72$$

and

$$\frac{1}{4}\left(1 - \frac{1+(-1)^{k+1}}{2k^2}\right)(d^2-r^2)(\Delta^2-\delta^2) = 18.75$$

Thus, the inequality in Theorem 16 is satisfied.

References

[1] A. Alqesmah, A. Saleh, R. Rangarajan, A. Yurttaş, İ. N. Cangul, Distance eccentric connectivity index of graphs, *Kyungpook Math. J.* **61** (2021) 61–74. ⇒295

[2] K.C. Das, Comparison between zagreb eccentricity indices and the eccentric connectivity index, the second geometric-arithmetic index and the graovac-ghorbani index. *Croat. Chem. Acta* **89** (2016) 505–510. ⇒295

[3] R. C. Entringer, D. E. Jackson, D. A.Snyder, Distance in graphs, *Czech. Math. J.* **26** (1976) 283–296. ⇒294

[4] R. Farooq, M. A. Malik, On some eccentricity based topological indices of nanostar dendrimers, *Optoelectron. Adv. Mater. Rapid. Commun.* **9** (2015) 842–849. ⇒295

[5] D. Vuki evi, A. Graovac, Note on the Comparison of the First and Second Normalized Zagreb Eccentricity Indices, *Acta Chim. Slov* **57** (2010) 524–528. ⇒295

[6] S. Gupta, M. Singh, A.K. Madan, Connective eccentricity index: a novel topological descriptor for predicting biological activity, *J.Mol. Graph. Model.* **18** (2000) 18–25. ⇒295

[7] I. Gutman, N. Trinajstić, Graph theory and molecular orbitals. Total pi-electron energy of alternant hydrocarbons, *Chem. Phys. Lett.* **17** (1972) 535–538. ⇒294

[8] S. M. Hosamani , S. S. Shirakol‡, M. V. Kalyanshetti, İ. N. Cangul, New eccentricity based topological indices of total transformation Graphs †arXiv:2008.10194v1 [math.CO] 24 Aug 2020 ⇒295

[9] X. Li, R. N. Mohapatra, R. S. Rodriguez, Grüss-type inequalities, *J. Math. Anal. Appl.* **267** (2002) 434–443. ⇒296
[10] N. Ozeki, On the estimation of inequalities by maximum and minimum values, Journal of College Arts and Science, **5** (1968) 199–203. (in Japanese) ⇒296
[11] J. Radon, Uber die absolut additiven mengenfunktionen, ¨ Wiener-Sitzungsber, (1913) 1295–1438. ⇒296
[12] V. Sharma, R. Goswami, A.K. Madan, Eccentric connectivity index: a novel highly discriminating topological descriptor for structure–property and structure–activity studies, *J. Chem. Inf. Comput. Sci* **37** (1997) 273–282. ⇒295
[13] K. Xu, Y. Alizadeh, K.C. Das, On two eccentricity-based topological indices of graphs, *Discrete Appl. Math.* **233** (2017) 240–251. ⇒295
[14] K. Xu, K.C. Das, H. Liu, Some extremal results on the connective eccentricity index of graphs, *J. Math. Anal. Appl.* **433** (2016) 803–817. ⇒295

Received: August 11, 2023 • Revised: November 28, 2023

Acta Univ. Sapientiae Informatica **15,** 2 (2023) 306–329

DOI: 10.2478/ausi-2023-0019

AnnoCerv: A new dataset for feature-driven and image-based automated colposcopy analysis

Dorina Adelina MINCIUNĂ
University of Medicine and Pharmacy
Iași, Romania

Demetra Gabriela SOCOLOV
University of Medicine and Pharmacy
Iași, Romania

Attila SZŐCS ✉
Ascorb Research S.R.L.
Târgu Mureș, Romania
email: szocsatti@gmail.com

Doina IVANOV
University of Medicine and Pharmacy
Iași, Romania

Tudor GÎSCĂ
University of Medicine and Pharmacy
Iași, Romania

Valentin NECHIFOR
University of Medicine and Pharmacy
Iași, Romania

Sándor BUDAI
Cattus Distribution S.R.L.
Târgu Mureș, Romania

Attila GÁL
Cattus Distribution S.R.L.
Târgu Mureș, Romania

Ákos BÁLINT
Cattus Distribution S.R.L.
Târgu Mureș, Romania

Răzvan SOCOLOV
University of Medicine and Pharmacy
Iași, Romania

David ICLANZAN
Sapientia Hungarian University of Transylvania,
Târgu Mureș, Romania
ORCID: 0000-0003-2587-9106

Key words and phrases: colposcopy imaging, cervical cancer diagnosis, lesion segmentation, automated image analysis

Abstract. Colposcopy imaging is pivotal in cervical cancer diagnosis, a major health concern for women. The computational challenge lies in accurate lesion recognition. A significant hindrance for many existing machine learning solutions is the scarcity of comprehensive training datasets.

To reduce this gap, we present AnnoCerv: a comprehensive dataset tailored for feature-driven and image-based colposcopy analysis. Distinctively, AnnoCerv include detailed segmentations, expert-backed colposcopic annotations and Swede scores, and a wide image variety including acetic acid, iodine, and green-filtered captures. This rich dataset supports the training of models for classifying and segmenting low-grade squamous intraepithelial lesions, detecting high-grade lesions, aiding colposcopy-guided biopsies, and predicting Swede scores – a crucial metric for medical assessments and treatment strategies.

To further assist researchers, our release includes code that demonstrates data handling and processing and exemplifies a simple feature extraction and classification technique.

1 Introduction

Cervical cancer, characterized by a malignant tumor in the cervix, ranks as the fourth most prevalent cancer in women worldwide [19]. It accounts for approximately 6.6% of all female cancer cases due to its high incidence rate [19]. A critical concern is the absence of symptoms in the early stages, leading to a notably high mortality rate. According to the World Health Organization, there were an estimated 604000 new cases and 342000 deaths in 2020 [27]. Distressingly, around 90% of these instances were in low- and middle-income nations [27]. The key to combating this disease lies in the timely detection of precancerous lesions, early diagnosis, and prompt treatment. In this context, colposcopy emerges as a pivotal tool, significantly enhancing the cervical cancer detection rate and serving as an effective screening method for precancerous lesions [21, 24, 28].

Used primarily as a follow-up to abnormal Pap smear results, colposcopy provides a magnified view of the cervix, enabling healthcare providers to pinpoint potential areas of concern. This procedure aids in detecting and diagnosing various cervical issues, including cervical dysplasia, HPV infections, and inflammation [21, 24, 28]. By discerning the gravity and reach of these abnormalities, practitioners can make informed decisions. For instance, if anomalies are spotted during a colposcopy, a biopsy might be conducted. Furthermore, colposcopy is instrumental in monitoring treatment effectiveness for cervical

abnormalities. After certain treatments, like tissue removal, consistent colposcopy exams ensure the healing process is on track and no new abnormalities arise. In scenarios demanding a more intensive treatment approach, colposcopy can guide surgical interventions, such as the loop electrosurgical excision procedure (LEEP) or cold knife cone biopsy [21, 24, 28].

Recognizing the pivotal role of colposcopy images in the diagnosis of cervical cancer, it is imperative to emphasize the significance of image quality for accurate analysis, particularly precancerous cervical lesions [11]. The need for high-quality imagery is amplified in telemedicine discussions among multiple doctors. Given the potential impact of variables - such as camera angles, lighting and shaking - on image quality, defects such as low contrast and distortion can compromise diagnosis precision [11].

Extensive research confirms high-risk human papillomavirus infection as a primary cause of cervical cancer [12, 17, 8, 25]. Early screening, when paired with HPV testing and cytology, has the potential to identify 80.7–98.7% of cervical intraepithelial neoplasia [26, 3]. Colposcopy-guided biopsies are the gold standard for detecting cervical cancer and its precancerous lesions. However, the precision of diagnosis can be influenced by various factors, from the expertise of the gynecologist to the woman's menstrual status. In particular, even for experienced gynecologists, the sensitivity of colposcopy for identifying cancerous lesions ranges from 81.4% to 95.7%, with a specificity between 34.2% and 69% [7, 23, 22]. Consequently, improving colposcopy precision is becoming a priority in the management of intraepithelial cervical neoplasia.

In contemporary medicine, artificial intelligence (AI) and deep learning have carved a niche, enabling efficient analysis of vast clinical data. Recent findings highlight the utility of medical AI and computer-assisted diagnosis in identifying cancerous lesions, leveraging deep learning and medical image processing techniques. Studies spanning optical tomography [18], radiology [14], computerized tomography [9], colonoscopy [2], and morphopathology [10] suggest that with ample training data, machine learning can rival or even surpass clinicians in diagnostic accuracy.

Historically, Acosta et al. [1] employed the K-NN algorithm to discern normal from abnormal cervical tissues, achieving 71% sensitivity and 59% specificity. Asiedu et al. [4] reported 81.3% sensitivity and 78.6% accuracy in distinguishing between cervical neoplasia and normal tissues. Liming Hu et al.'s [15] seven-year cohort study trained a deep learning algorithm on colposcopy images, achieving higher accuracy than the Pap smear. Additionally, Bing Bai et al. [5] in 2018 used the K-means algorithm for automatic cervical region segmentation. Deep learning methods, with their capacity to autonomously

extract pertinent features from training data, underscore their value alongside conventional diagnostic techniques. However, a pertinent challenge remains: medical image datasets are often limited, constraining training capabilities.

Many studies keep their image datasets private. As a result, only a select few databases are available for developers [16, 30, 20, 13, 29]. Notably, the existing datasets primarily consist of acetic acid images. In light of this, there is an urgent need to develop or expand datasets, aiming to incorporate a diverse range of images, including acetic acid, iodine, and green-filter types.

In our study, we amassed a collection of colposcopy images. These images were meticulously segmented and annotated by specialists to distinctly visualize both healthy and pathological changes in cervical tissue. What sets this curated collection apart is its inclusion of acetic acid images, iodine images, and green-filtered images. This comprehensive dataset is now available for training machine learning models, aiding in the automatic classification and segmentation of low-grade squamous intraepithelial lesions (LSIL), detection of high-grade squamous intraepithelial lesions (HSIL), and assisting with colposcopy-guided biopsies. All curated data were cross-referenced with gynecological evaluations based on the patients' medical record.

2 Materials and methods

2.1 Comprehensive assessment in colposcopy: techniques and criteria

Colposcopy, a diagnostic procedure employed primarily in gynecological examinations, relies heavily on the discerning observation of cervical tissues to detect anomalies and potential malignancies. This procedure utilizes different techniques and criteria, each tailored to accentuate specific aspects of cervical tissue and enhance diagnostic precision. In the subsequent sub-sections, we will delve into the principal components and characteristics pivotal to colposcopy images, understand the significance of the Swede score evaluation as a diagnostic tool, and shed light on the transformation zone's classification methodology. Together, these criteria and methods offer a comprehensive overview, enabling a nuanced understanding of colposcopic examinations and the annotated images in the dataset.

2.1.1 Key elements and features of colposcopy images

The accurate diagnosis of cervical neoplasia using colposcopy is contingent on four primary features:

1. Intensity of Aceto-whitening: This refers to the color tone variations seen in the cervix upon application of acetic acid.
2. Demarcation and Surface Contour of Aceto-white Areas: This encompasses the clarity and texture of the white regions appearing after the acetic acid application.
3. Vascular Features: The visibility of blood vessels provides insights into the health of the cervical tissue.
4. Iodine-Induced Color Changes: Observing how the cervix responds to iodine application can give vital diagnostic clues.

Additional diagnostic considerations include:

- Anomalies in the transformation zone can be indicative of neoplasia.
- Expert gynecologists can differentiate between low-grade cervical intraepithelial neoplasia, immature squamous metaplasia, and inflammatory lesions.
- A biopsy, guided by colposcopy, becomes vital when the presence of neoplasia is uncertain.
- Recognizing dense and opaque aceto-white regions, particularly near the squamo-columnar junction, is essential for detecting intraepithelial neoplasia.

Characteristics of CIN (Cervical Intraepithelial Neoplasia):

- Low-grade CIN: Manifests as thin aceto-white lesions with irregular or feathered margins.
- High-grade CIN: These regions are more pronounced—thicker and more opaque with distinct boundaries. Their expansion might reach the endocervical canal, and they exhibit a rough, nodulated texture. Variability in color intensity can be noted within these lesions.

Vascular observations play a pivotal role:

- Both fine and pronounced vascular features, such as punctations and mosaics, are mostly confined to aceto-white areas.
- Low-grade malignancies often show fine punctations or mosaics.

- Conversely, coarse punctations or mosaics hint at high-grade lesions.
- Utilizing green filters can significantly enhance vascular visibility.

For more precise and standardized assessments, incorporating scoring systems like the Swede score [6] can offer valuable guidance in colposcopic evaluations and determinations.

2.1.2 Swede score evaluation

The Swede score [6] is an established metric used in colposcopic evaluations. It provides a systematic approach to assess cervical lesions based on specific characteristics. Each characteristic is scored according to the criteria presented in Table 1, and the cumulative score predicts the severity of the lesion according to the brackets presented in Table 2.

Characteristics	**0**	**1**	**2**
Uptake of acetic acid	Zero or transparent	Shady, milky (not transparent, not opaque)	Distinct, opaque white
Margins and surface	Diffuse	Sharp but irregular, jagged, "geographical". Satellites	Sharp and even; difference in surface level, including "cuffing"
Vessels	Fine, regular	Absent	Coarse or atypical
Lesion size	< 5 mm	$5 - 15$ mm or spanning 2 quadrants	> 15 mm or spanning $3 - 4$ quadrants, or endocervically undefined
Iodine staining	Brown	Faint or patchy yellow	Distinct yellow

Table 1: Swede score assessment

Score	Colposcopic Prediction
0–4	Low-grade/CIN 1
5–6	High-grade/non-invasive cancer/CIN2+
7–10	High-grade/suspected invasive cancer/CIN2+

Table 2: Interpretation of Swede score

2.1.3 Classification of the transformation zone (TZ) in Colposcopy

In colposcopic evaluations, the visibility and positioning of the squamocolumnar junction play a crucial role in categorizing the transformation zone. On the basis of this, the transformation zone can be systematically classified as:

Type 1: The transformation zone, which encompasses the entire squamocolumnar junction, is located in the ectocervix. In simpler terms, the entirety of the upper limit of the TZ is ectocervical.

Type 2: The upper boundary of the TZ is partially or entirely observed within the canal, ensuring visibility throughout a 360-degree angle.

Type 3: The upper boundary of the TZ remains elusive, implying that the upper limit is not visible during examination.

2.1.4 Categorization of aceto-white changes in abnormal colposcopic findings

Post the application of acetic acid during colposcopy, typical aceto-white changes manifest, helping identify potential abnormalities. These can be grouped based on severity as:

Minor (Grade 1): This category predominantly presents with:

- A slender aceto-white epithelium complemented by an irregular, 'geographical' boundary.
- Presence of delicate structures like fine mosaic and fine punctation patterns, indicating lesser severity.

Major (Grade 2): More severe changes in this category are characterized by:

- A pronounced, dense aceto-white epithelial layer that showcases aceto-whitening rapidly upon acid application.

- Noticeable cuffed crypt or gland openings, indicative of potential concerns.
- The epithelium may exhibit coarse mosaic and punctation patterns. In addition, distinct features like a sharp border, the inner border sign, and ridge sign further solidify its classification as major changes.

2.2 Dataset and automatic processing

During our research phase, we sourced 527 colposcopy images from 100 medical records. Expert specialists segmented and annotated each image to differentiate between healthy and pathological cervical tissues.

The segmented and annotated image set, Swede scores and the accompanying code are available at the following address: `https://github.com/iclx/AnnoCerv`. This work is licensed under a Creative Commons Attribution 4.0 International License [1].

2.2.1 Dataset structure and format description

The organization of the dataset is hierarchical, ensuring ease of navigation and clarity. Here is a detailed breakdown of its structure:

Directory Structure: Each individual case is encapsulated within its own unique folder, named "Case ID".

Image Files: Within each case folder, there are one or more cervix images saved in the JPG format. The images within a single case can encompass various types, namely acetic acid images, iodine images, and green-filtered images.

Filename convention: The naming convention for the images is standardized for clarity. It consists of a case identifier, followed by the image type, and an index enclosed within parentheses to distinguish multiple images of the same type.

Annotation files: Each acetic acid image (denoted by 'Aceto' in the filename) has a corresponding PNG annotation file. This file carries the same primary filename but with a .png extension. For instance, an image named C1Aceto (1).jpg has its annotations in C1Aceto (1).png.

Annotation encoding: The PNG annotation files utilize a specific color encoding to represent various observed features:

- **blue** for the squamous-cylindrical junction,

[1] `http://creativecommons.org/licenses/by/4.0/`

- **purple** for aceto-white areas,
- **red** for atypical vessels and punctations,
- **brown** for mosaics,
- **yellow** for Naboth cysts, and
- **black** for cuffed gland openings.

The background of these PNG files is transparent. In scenarios where the medical professional did not detect any notable features, the PNG remains entirely transparent without any colored pixels.

Swede scores are cataloged in the CSV file "swede_scores.csv", where each row corresponds to the score of its respective case.

2.2.2 Automated processing

Given the clearly delineated Dataset Structure and Format Description, the systematic processing of the image set becomes inherently straightforward from a computational perspective. The procedure entails the following methodical steps:

Directory iteration: We commence by traversing each folder, wherein every individual folder signifies a distinct case, warranting content exploration and analysis.

Image type verification: In each case folder, we check for the presence of image types that are of interest for our analysis.

Annotation examination: For every Aceto image, we open its associated PNG file. This step helps us identify different pixel colors, which correspond to specific medical notes.

Statistical aggregation: After collecting all the required data, we can proceed to calculate statistics of interest.

In code listing 1, we exemplify this process to determine several pertinent statistics:

1. The number of cases containing iodine images.
2. The number of cases containing green-filtered images.
3. The number of cases where the squamous-cylindrical junction is not visible (evidenced by the absence of blue pixels).

4. The total count of cases exhibiting atypical regions (atypical vessels, Naboth cysts or cuffed gland openings).

Source code 1: Case processing and statistics extraction – Python code snippet

```
# Iterate through each folder (case)
for folder in os.listdir(base_path):
    folder_path = os.path.join(base_path, folder)

    print(f'Processing folder {folder}')

    if os.path.isdir(folder_path):
        iodine_present = False
        green_present = False
        blue_absent = True
        atypical_regions = 0

        # Check for image types and corresponding annotations
        for file in os.listdir(folder_path):
            print(f'\tProcessing {file}')
            if "Iod" in file and file.endswith(".jpg"):
                iodine_present = True
            elif "Green" in file and file.endswith(".jpg"):
                green_present = True
            elif "Aceto" in file and file.endswith(".jpg"):
                annotation_file = os.path.join(folder_path,
                ↪  file.replace(".jpg", ".png"))

                if os.path.exists(annotation_file):
                    img = Image.open(annotation_file)
                    pixels = list(img.getdata())

                    for pixel in pixels:
                        # Check for transparency (Alpha channel)
                        if len(pixel) == 4 and pixel[3] > 0:
                            if pixel[:3] == colors['blue']:
                                blue_absent = False
                            elif pixel[:3] != colors['purple']:
                                atypical_regions += 1
```

In our effort to promote accessibility, the code is readily available in the GitHub repository as a Google Colab Notebook named "data_summary.ipynb". The notebook can be easily extended or modified to compute different statistics of interest.

2.2.3 Feature extraction and classification

The GitHub repository additionally contains a Google Colab Notebook named "data_modelling.ipynb" that exemplifies foundational operations, serving as a primer for individuals unfamiliar with image processing and machine learning tasks in the domain of medical imaging.

Source code 2: Feature computation – Python code snippet

```
from skimage import feature, color

def extract_features(img_path):
    img = cv2.imread(img_path)
    intensity = cv2.cvtColor(img, cv2.COLOR_BGR2GRAY)

    # Texture feature using Local Binary Pattern
    texture_r5 = feature.local_binary_pattern(intensity, P=8*5, R=5,
    ↪  method='uniform')

    # Gradient features using Sobel operator
    grad_x = cv2.Sobel(intensity, cv2.CV_64F, 1, 0, ksize=3)
    grad_y = cv2.Sobel(intensity, cv2.CV_64F, 0, 1, ksize=3)

    # Spatial features: simply the x and y coordinates
    x = np.arange(img.shape[1])
    y = np.arange(img.shape[0])
    x, y = np.meshgrid(x, y)

    # Color-based features
    hsv_img = color.rgb2hsv(img)
    hue = hsv_img[:, :, 0]
    saturation = hsv_img[:, :, 1]
    value = hsv_img[:, :, 2]

    # Stack all features together
    features = np.dstack((intensity, texture_r5, grad_x, grad_y, x, y, hue,
    ↪  saturation, value))

    return features
```

The operations demonstrated include:

Dataset download: Copy of the images to the local machine.

Image resizing: A programmatic approach to altering image and annotation dimensions.

Feature extraction : Focused on the classification of pixels representing the squamous-cylindrical junction (depicted as blue pixels in the annotated PNG files). The features extracted encompass: i) Intensity Features: the direct utilization of pixel intensity; ii) Texture Features: just one Local Binary Patterns (LBP) of radius 5 is used; iii) Gradient Features: the magnitude and direction of image gradients are computed; iv) Spatial Features: the pixel's x and y coordinates is stored; v) Color-based Features: the extraction of Hue, Saturation, and Value (HSV) from pixels. The operations are also depicted in the Code Listings 2 and 3.

Feature scaling: a common operation before fitting the models to the data. By normalizing the features to a consistent scale, we ensure that each one contributes appropriately to the model's outcomes, facilitating faster convergence for gradient-based methods and potentially boosting the model's overall performance.

Data preparation: Given the imbalanced nature of the classification task, the notebook exemplifies a basic balancing method via undersampling. Also, the dataset is divided into training and testing subsets.

Feature correlation analysis and pair plot technique: These methods help assesing the relationships between different features and the target classification variable. Feature correlation analysis quantifies the interdependencies, offering insights into the intrinsic structure of the data. The pair plot technique visualizes pairwise relationships in a dataset. Together, these tools facilitate a comprehensive understanding of the data, and can guide the selection and prioritization of the most pertinent features for model training.

Model training: A Random Forest classifier, utilizing its default parameters, is trained on the extracted, balanced small dataset.

Evaluation: The computation of relevant metrics such as the F1 score and the ROC curve is exemplified.

It is important to emphasize that the methods and techniques highlighted in the notebook are foundational, designed primarily to serve as a rapid, cloud-based experimentation tool for newcomers and enthusiasts. While they offer a convenient starting point for those new to the field, they do not embody the cutting-edge of current research or advanced methodologies. The primary

aim is to demonstrate a workflow that is accessible and can be tried out and executed in the cloud in a matter of minutes.

Source code 3: Feature extraction for each case – Python code snippet

```
# Collect data and labels
X_data = []
y_labels = []

for img_file in os.listdir(output_folder):
    if img_file.endswith('.jpg'):
        print(f'\tComputing features for {img_file}')
        features = extract_features(os.path.join(output_folder, img_file))
        X_data.append(features)

        annotation_path = os.path.join(output_folder, img_file.replace('.jpg',
        ↪  '.png'))
        img = cv2.imread(annotation_path)
        annotation = np.array(img[:, :, 0] == 255) & np.array(img[:, :, 2] ==
        ↪  0)
        # Convert blue pixels to label 1, others to 0
        is_junction = annotation.astype(np.int)
        y_labels.append(is_junction)

X_data = np.array(X_data).reshape(-1, 9)
y_labels = np.array(y_labels).reshape(-1)
```

3 Results and discussion

3.1 Segmented and annotated images

To provide insight into our database, we display representative examples of segmented images in Figures 1, 2, 3, and 4, with the annotations superimposed on the cervix images. These images underscore the variety and depth of the content within the dataset.

Figure 1 showcases squamous-cylindrical junctions, aceto white areas, Naboth cyst, punctuation, mosaic, and fine vessels.

In Figure 2, the emphasis is on highlighting the squamous-cylindrical junctions, aceto white areas, polyps, Naboth cysts, and glandular openings.

The rationale for the iodine test is rooted in cellular chemistry: mature squamous epithelium, both original and newly formed, contains glycogen. In contrast, neoplastic and invasive cancer cells typically have minimal or no glycogen. As a result, they do not absorb iodine, appearing as distinct mustard yellow or saffron-colored regions. Following this principle, neoplastic aceto

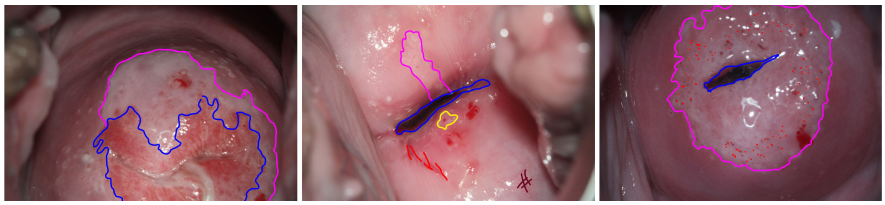


Figure 1: Annotations: blue – squamous-cylindrical junction, purple – aceto white area, red – atypical vessels, punctations, brown – mosaic, yellow – Naboth cyst

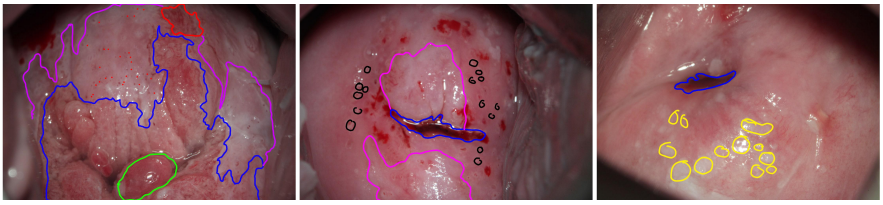


Figure 2: Annotations: blue – squamous-cylindrical junction, purple – aceto white area, yellow – Naboth cysts, black – cuffed gland opening, green – polyp.

white areas remain unaffected by iodine. This characteristic can be observed in Figure 3, where iodine images act as confirmatory markers for suspected lesions.

Colposcopy with a green filter allows visualization of vascular changes. Figure 4 offers insight into this, depicting key vascular alterations like punctuation, mosaic, atypical fine vessels, and larger vessels.

3.2 Exploratory data analysis

Derived from the previously mentioned Google Colab Notebook for data processing, this section delves into patterns and insights within the dataset related to cervical health diagnostics.

The case based image type distribution is presented in Figure 5. A predominant 94% of the cases contain iodine images, highlighting their important role in confirmation and diagnostics. Conversely, green-filtered images, which primarily aid in the evaluation of vascular changes, are present in only 11% of cases. This differential suggests that such vascular evaluations might be less frequently necessitated in the overall diagnostic spectrum. In 13 cases, the squamous-cylindrical junction is not visible, marked by an absence of blue

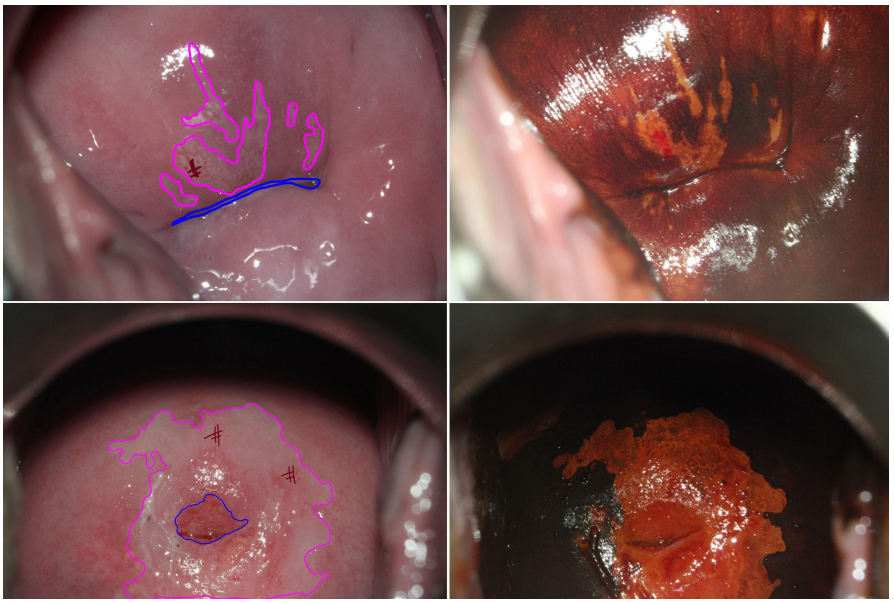


Figure 3: Aceto white neoplastic areas (purple) confirmed with Iodine solution.

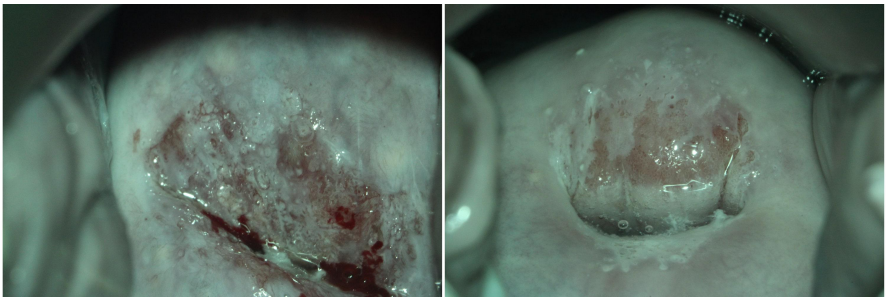


Figure 4: Colposcopy images taken with green filter to highlight vascular changes.

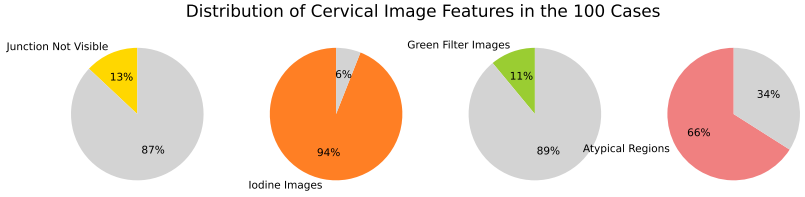


Figure 5: Extracted properties from the 100 colposcopy imaging cases.

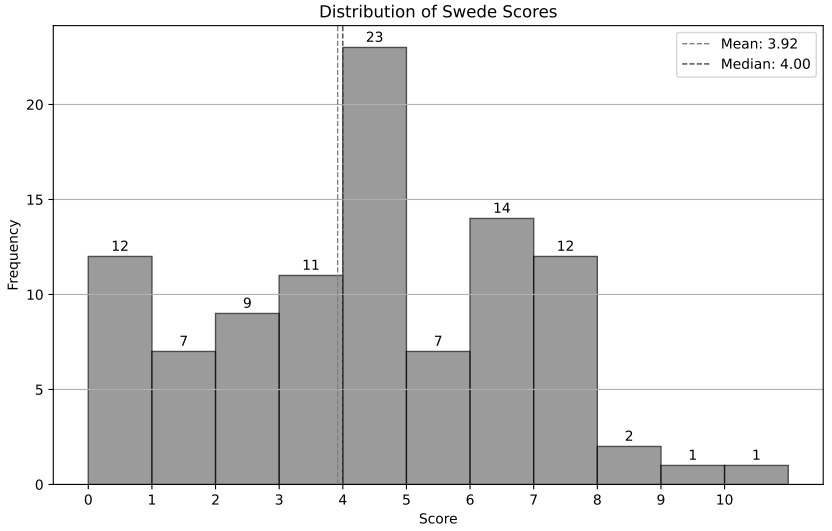


Figure 6: Swede scores distribution and central tendency.

pixels in the annotations. A significant 66% of cases manifest atypical regions, encompassing atypical vessels, Naboth cysts, or cuffed gland openings, underscoring the critical nature of in-depth cervical health assessments.

Figure 6 provides a visual representation of the distribution of Swede scores. Scores, which range from 0 to 10, reveal a spectrum of health conditions. Although 12 cases boast an optimal score of 0, a considerable portion, specifically 23 cases, cluster around a score of 4. However, a handful of cases with high scores of 9 and 10 highlight the existence of severe abnormalities. Statistically, with a mean of 3.92, a median at 4.00, and a standard deviation of 2.40, it is evident that the majority of the cases hover around a moderate risk range.

The dataset offers a detailed snapshot of cervical health through its various image types and score distributions. With atypical regions evident in 66% of cases, the need for meticulous diagnostics becomes even more evident. Although a considerable portion of cases fall within the low-to-moderate risk categories, the presence of high-risk outliers emphasizes the dataset's potential as a valuable resource for training advanced machine learning models. The mix of iodine and green-filtered images within the dataset lays a foundation for exploring a variety of diagnostic methodologies.

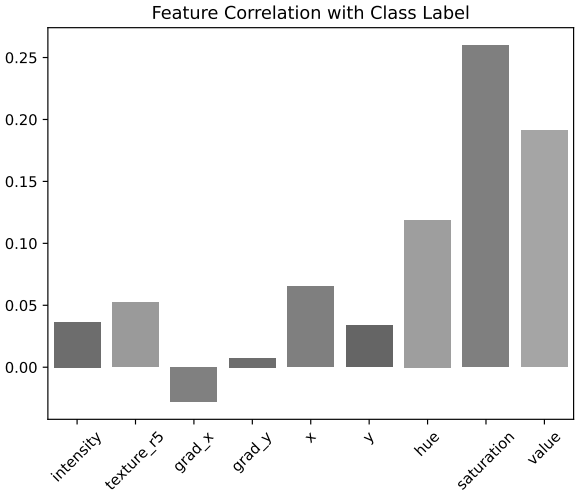


Figure 7: Correlations between the features and the class label.

3.3 Feature correlation

Understanding the relationships between features and the target classification variable is pivotal for effective model building. The two key techniques exemplified in this endeavor are Feature Correlation Analysis and the Pair Plot Technique. The former provides a quantitative measure of interdependencies between features, shedding light on their internal structure and importance. Meanwhile, the Pair Plot Technique offers a visual representation of pairwise relationships across the dataset, enabling a holistic grasp of data intricacies.

A visual representation of the correlations of features with the class label is illustrated in Figure 7. Positional features (mainly 'x') and color-based features stand out with relatively higher positive correlations to the target, suggesting they might play an essential role in classification. In contrast, the feature 'grad_y' shows no correlation, and 'grad_x' seems to be inversely correlated.

Moving onto the Pair Plot in Figure 8, certain observations emerge:

Intensity vs. value: A direct correlation is evident between Intensity (as derived from the gray-scale image) and Value (the brightness of the color). This relationship, expected given that changing the Value in HSV, we are generally increasing or decreasing the brightness of the RGB channels, which in turn will affect the grayscale intensity.

y vs. saturation: A distinctive pattern emerges. Most of the 'Junction' class instances cluster towards the right, indicating a potential joint link between these features and the target variable.

x & y distribution: The scatter plot between x and y showcases a stark disparity in the distribution of our two classes. A potential interpretation is the prevalence of the 'Junction' predominantly towards the center of the image.

Intensity vs. value distribution: Their distribution peaks also differ noticeably, reinforcing the idea of some inherent structural differences within the dataset.

3.4 Classification performance

The Random Forest classifier, using default parameters, served as a baseline to distinguish between 'Non-Junction' and 'Junction' classes in a balanced dataset. Detailed performance metrics are presented in Table 3.

The model achieved an accuracy of around 80%, illustrating a consistent prediction rate for both classes. Precision, which represents the fraction of correct positive predictions, was similar for both classes. The slightly higher recall for the 'Junction' class suggests the model's marginally better ability to detect these instances. With F1-Scores of 0.79 and 0.80 for 'Non-Junction' and 'Junction' respectively, the model demonstrated a balanced performance for both classes, harmonizing precision and recall.

While the current results provide valuable insights, it's worth noting that the model's performance might vary with different configurations or when applied to other datasets. Exploring alternative machine learning algorithms and fine-tuning parameters can potentially unearth more robust classification strategies.

In tandem with the table, Figure 9 visualizes the Receiver Operating Characteristic (ROC) Curve, offering an in-depth view of the performance of the classifier. The curve's area of 0.73 indicates its acceptable discriminative capability, with ample room for improvement. A prominent inflection point at a True Positive Rate (TPR) of 0.8 and a False Positive Rate (FPR) of about 0.37 suggests an optimal threshold. While the TPR is commendable, an FPR of 0.37 highlights the misclassification of a substantial number of negative instances.

The Random Forest classifier, even in its default configuration, yields satisfactory results. Understanding the feature importance provided by the Ran-

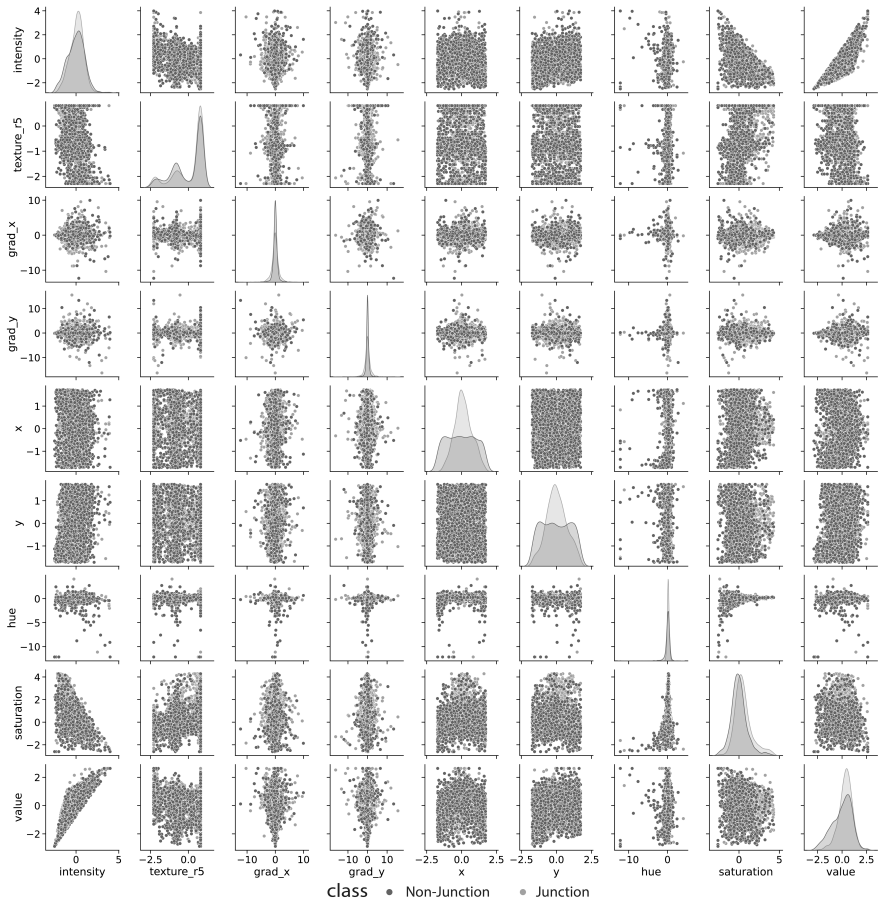


Figure 8: Pairwise relationship of the features.

dom Forest classifier can also offer insights into which variables have a greater influence on the classification decision. A thorough examination of these importance metrics could guide feature engineering efforts, possibly leading to enhanced performance by emphasizing on the most influential features. This introspective approach not only strengthens the model's predictive power but also adds an interpretative dimension to the model, bridging the gap between machine learning predictions and domain-specific knowledge. This study balanced the dataset through subsampling to simplify the classification task for demonstration purposes. For the genuine, heavily imbalanced dataset, har-

	Precision	Recall	F1-Score	Support
Non-Junction	0.80	0.78	0.79	400
Junction	0.79	0.81	0.80	400
accuracy			0.80	800
macro avg	0.80	0.80	0.79	800
weighted avg	0.80	0.80	0.79	800

Table 3: Performance metrics

nessing advanced techniques such as Convolutional Neural Networks (CNNs) and transfer learning could yield superior outcomes.

4 Conclusions

Cervical cancer remains a pressing health concern for women worldwide. While computational methods offer promising avenues for improved diagnosis, their effectiveness is intrinsically linked to the quality and comprehensiveness of available training datasets. Recognizing a discernible gap in this area, we introduce AnnoCerv, a dataset that provides a detailed perspective on cervical colposcopy images. These 527 samples, derived from 100 medical records, present an array of expert-annotated, feature-rich images that aim to support a range of analysis, from basic lesion recognition to Swede score predictions.

AnnoCerv represents our effort to enhance the resources available to researchers and practitioners in the field. While the accompanying code provides an introduction to image processing and machine learning tasks, it's primarily designed for those less familiar with the domain. We acknowledge its foundational nature, emphasizing that there remains a significant opportunity and need for the development of more sophisticated and nuanced methods.

Choosing to present examples via Google Colab Notebooks was a deliberate strategy to enhance accessibility. This approach streamlines the initial setup, allowing users to rapidly interact with the dataset.

We hope that the AnnoCerv image set and code can serve as valuable resources for further research, innovation, and developments in the field of cervical health and diagnostics.

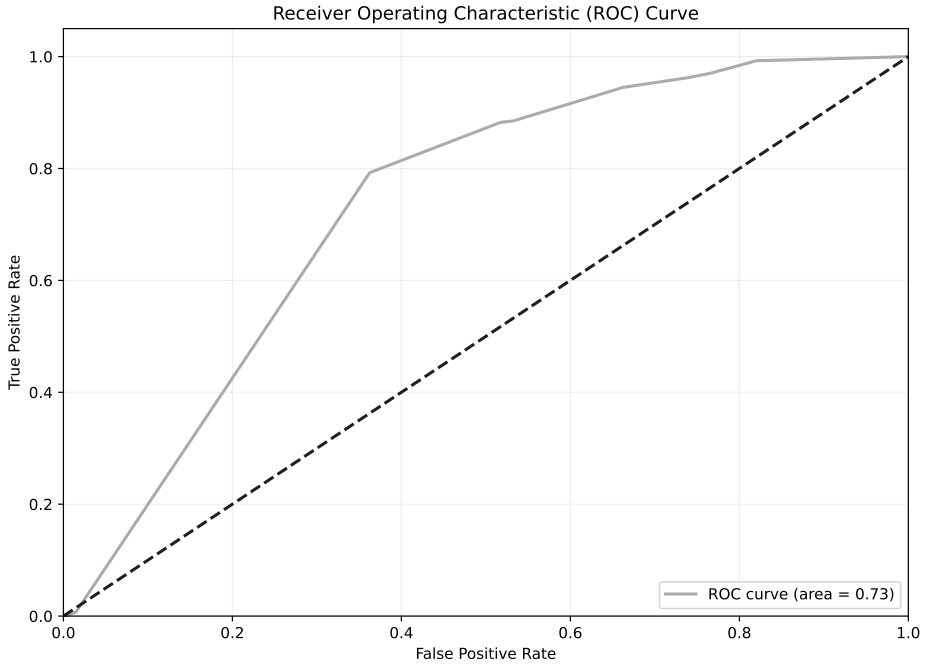


Figure 9: Operating characteristic curve.

Acknowledgements

The work was supported by Financing contract no. 75/221_ap2/21.08.2020, in the Competitiveness Operational Program 2014–2020, Priority Axis 2 - "Information and communication technology (TIC) for a competitive digital economy", SMIS Code 130106.

References

[1] H.-G. Acosta-Mesa, N. Cruz-Ramírez, R. Hernández-Jiménez, Aceto-white temporal pattern classification using k-nn to identify precancerous cervical lesion in colposcopic images. *Computers in biology and medicine* **39,** 9 (2009) 778–784. ⇒308

[2] O. F. Ahmad, A. S. Soares, E. Mazomenos, P. Brandao, R. Vega, E. Seward, D. Stoyanov, M. Chand, M., L. B. Lovat, Artificial intelligence and computer-aided diagnosis in colonoscopy: current evidence and future directions. *The lancet Gastroenterology & hepatology* **4,** 1 (2019) 71–80. ⇒308

[3] M. Arbyn, F. Verdoodt, P. J. Snijders, V. M. Verhoef, E. Suonio, L. Dillner, S. Minozzi, C. Bellisario, R. Banzi, F.-H. Zhao, et al. Accuracy of human papillomavirus testing on self-collected versus clinician-collected samples: a meta-analysis. *The lancet oncology* **15,** 2 (2014) 172–183. ⇒308

[4] M. N. Asiedu, A. Simhal, U. Chaudhary, J. L. Mueller, C. T. Lam, J. W. Schmitt, G. Venegas, G. Sapiro, G., N. Ramanujam, Development of algorithms for automated detection of cervical pre-cancers with a low-cost, point-of-care, pocket colposcope. *IEEE Transactions on Biomedical Engineering* **66,** 8 (2018) 2306–2318. ⇒308

[5] B. Bai, P.-Z. Liu, Y.-Z. Du, Y.-M. Luo, Automatic segmentation of cervical region in colposcopic images using k-means. *Australasian physical & engineering sciences in medicine*, **41** (2018) 1077–1085. ⇒308

[6] J. Bowring, B. Strander, M. Young, H. Evans,P. Walker, The swede score: evaluation of a scoring system designed to improve the predictive value of colposcopy. *Journal of lower genital tract disease* **14,** 4 (2010) 301–305. ⇒311

[7] B. H. Brown, J. A. Tidy, The diagnostic accuracy of colposcopy–a review of research methodology and impact on the outcomes of quality assurance. *European Journal of Obstetrics & Gynecology and Reproductive Biology* **240** (2019) 182–186. ⇒308

[8] X. Castellsagué, Natural history and epidemiology of hpv infection and cervical cancer. *Gynecologic oncology* **110,** 3 (2008) S4–S7. ⇒308

[9] S. Chilamkurthy, R. Ghosh, S. Tanamala, M. Biviji, N. G. Campeau, V. K. Venugopal, V. Mahajan, P. Rao, P., P. Warier, Deep learning algorithms for detection of critical findings in head ct scans: a retrospective study. *The Lancet* **392,** 10162 (2018) 2388–2396. ⇒308

[10] N. Coudray, P. S. Ocampo, T. Sakellaropoulos, N. Narula, M. Snuderl, D. Fenyö, A. L. Moreira, N. Razavian, A. Tsirigos, Classification and mutation prediction from non–small cell lung cancer histopathology images using deep learning. *Nature medicine* **24,** 10 (2018) 1559–1567. ⇒308

[11] J. Fan, J. Liu, S. Xie, C. Zhou, Y. Wu, Cervical lesion image enhancement based on conditional entropy generative adversarial network framework. *Methods* **203** (2022) 523–532. ⇒308

[12] A. Goodman Hpv testing as a screen for cervical cancer. *BMJ* **350** (2015). ⇒308

[13] P. Guo, Z. Xue, Z. Mtema, K. Yeates, O. Ginsburg, M. Demarco, L. R. Long, M. Schiffman, M., S. Antani, Ensemble deep learning for cervix image selection toward improving reliability in automated cervical precancer screening. *Diagnostics* **10,** 7 (2020) 451. ⇒309

[14] A. Hosny, C. Parmar, J. Quackenbush, L. H. Schwartz, H. J. Aerts, Artificial intelligence in radiology. *Nature Reviews Cancer* **18,** 8 (2018) 500–510. ⇒308

[15] L. Hu, D. Bell, S. Antani, Z. Xue, K. Yu, M. P. Horning, N. Gachuhi, B. Wilson, M. S. Jaiswal, B. Befano, et al. An observational study of deep learning and automated evaluation of cervical images for cancer screening. *JNCI: Journal of the National Cancer Institute* **111,** 9 (2019) 923–932. ⇒308

[16] International Agency for Research on Cancer (IARC). Cervical image bank, 2021. Accessed on 19th October 2023. ⇒309

[17] J. Jin, J. Hpv infection and cancer. *Jama* **319,** 10 (2018) 1058–1058. ⇒308

[18] D. S. Kermany, M. Goldbaum, W. Cai, C. C. Valentim, H. Liang, S. L. Baxter, A. McKeown, G. Yang, X. Wu, F. Yan, et al. Identifying medical diagnoses and treatable diseases by image-based deep learning. *Cell* **172,** 5 (2018) 1122–1131. ⇒308

[19] S. Pimple, G. Mishra, G. Cancer cervix: Epidemiology and disease burden. *Cytojournal* **19** (2022). ⇒307

[20] M. E. Plissiti, P. Dimitrakopoulos, G. Sfikas, C. Nikou, O. Krikoni, A. V. Charchanti, Sipakmed: A new dataset for feature and image based classification of normal and pathological cervical cells in pap smear images. *2018 25th IEEE International Conference on Image Processing (ICIP)* (2018) 3144–3148. ⇒309

[21] W. Prendiville, R. Sankaranarayanan, *Colposcopy and treatment of cervical precancer*. International Agency for Research on Cancer, World Health Organization, 2017. ⇒307, 308

[22] M. Sideri, P. Garutti, S. Costa, P. Cristiani, P. Schincaglia, P. Sassoli de Bianchi, C, Naldoni, L. Bucchi, et al. Accuracy of colposcopically directed biopsy: results from an online quality assurance programme for colposcopy in a population-based cervical screening setting in italy. *BioMed Research International 2015* (2015). ⇒308

[23] M. Underwood, M. Arbyn, W. Parry-Smith, S. De Bellis-Ayres, R. Todd, C. Redman, E. Moss, E. Accuracy of colposcopy-directed punch biopsies: a systematic review and meta-analysis. *BJOG: An International Journal of Obstetrics & Gynaecology* **119,** 11 (2012) 1293–1301. ⇒308

[24] J. Valls, A. Baena, G. Venegas, M. Celis, M. González, C. Sosa, J. L. Santin, M. Ortega, A. Soilán, E. Turcios, et al. Performance of standardised colposcopy to detect cervical precancer and cancer for triage of women testing positive for human papillomavirus: results from the estampa multicentric screening study. *The Lancet Global Health* **11,** 3 (2023) e350–e360. ⇒307, 308

[25] J. M. Walboomers, M. V. Jacobs, M. M. Manos, F. X. Bosch, J. A. Kummer, K. V. Shah, P. J. Snijders, J. Peto, C. J. Meijer, N. Muñoz, Human papillomavirus is a necessary cause of invasive cervical cancer worldwide. *The Journal of pathology* **189,** 1 (1999) 12–19. ⇒308

[26] J. Wang, Analysis of the application values of different combination schemes of liquid-based cytology and high-risk human papilloma virus test in the screening of high-grade cervical lesions. *Brazilian Journal of Medical and Biological Research* **52** (2018). ⇒308

[27] World Health Organization. Cervical cancer – fact sheet, Year. Accessed on 19th October 2023. ⇒307

[28] P. Xue, M. T. A. Ng, Y. Qiao, The challenges of colposcopy for cervical cancer screening in lmics and solutions by artificial intelligence. *BMC medicine* **18** (2020) 1–7. ⇒307, 308

[29] X. Yang, Z. Zeng, S. G. Teo, L. Wang, V. Chandrasekhar, S. Hoi, Deep learning for practical image recognition: Case study on kaggle competitions. In *Proceedings of the 24th ACM SIGKDD international conference on knowledge discovery & data mining* (2018), pp. 923–931. ⇒309

[30] Y. Yu, J. Ma, W. Zhao, Z. Li, S. Ding, S. MSCI: A multistate dataset for colposcopy image classification of cervical cancer screening. *International journal of medical informatics* **146** (2021) 104352. ⇒309

Received: August 18, 2023 • *Revised: October 21, 2023*

Acta Univ. Sapientiae Informatica **15,** 2 (2023) 330–344

DOI: 10.2478/ausi-2023-0020

Enhancing healthcare services recommendation through sentiment analysis

Muhammad Rizwan Rashid RANA
University Institute of Information Technology, Pir Mehr Ali Shah Arid Agriculture University Rawalpindi, Pakistan
email: rizwanrana315@gmail.com

Asif NAWAZ
University Institute of Information Technology, Pir Mehr Ali Shah Arid Agriculture University Rawalpindi, Pakistan
email: asif.nawaz@uaar.edu.pk

Tariq ALI
University Institute of Information Technology, Pir Mehr Ali Shah Arid Agriculture University Rawalpindi, Pakistan
email: tariq.ali@uaar.edu.pk

Ghulam MUSTAFA
University Institute of Information Technology, Pir Mehr Ali Shah Arid Agriculture University Rawalpindi, Pakistan
email: gmustafa@uaar.edu.pk

Abstract. As technology advances, most people use social media sites like Twitter, Facebook, and Flickr to share information and communicate with others. The volume of free-text data is growing daily due to the widespread use of these social media platforms. These platforms contain a substantial amount of unstructured information. Patient opinions expressed on social media platforms play a significant role in healthcare improvement and impact health-related policymaking. In this research, we introduce a machine learning approach for the optimal identification of healthcare-related features. This approach is based on a novel synthetic

Key words and phrases: sentiment Analysis; opinion mining; entropy; feature extraction; quality of services;

method. Additionally, we employ an entropy-based technique to classify free-text comments from hospital data into positive, negative or neutral. The experimental results and evaluations show 85%, 82.3%, 78.2% and 87% accuracy between ratings of health care. We observed that there is a minor association between our technique, expert opinion and patient interviews. Through the use of machine learning techniques, we achieve an accuracy level that suggests we are capable of providing an accurate and reasonable assessment of the ideal healthcare center for a patient. Our proposed novel framework predicts the healthcare experience at hospitals based on patient reviews posted on social media. This innovative approach outperforms traditional methods, such as surveys and expert opinions.

1 Introduction

Patient feedback helps in improving overall QoS in healthcare systems [9, 11]. Conventional procedures of patient feedback through surveys and reports reveal sustainable development in health services. These techniques are expensive and require some basic questions that are conducted frequently. In the present era, patients around the globe share their healthcare experience on the internet, social media websites or demonstrate in health reports in the form of blogs on different online healthcare communities [10, 6]. However, such type of information is much unstructured and difficult to understand.

In the case of unstructured data, it is very difficult to understand about patient's experience. Studies demonstrate that 85% of grown-ups utilize the web, 26% of individuals perused another person's encounters about health and 12% of individuals utilize online reviews of hospitals or some therapeutic minding sites [5]. In recent years, sentiment analysis has turned out to be progressively prevalent for preparing internet-based life information on online networks, wikis, micro-blogging stages, and other online cooperative media [23]. Intended to characterize text use sentiment analysis which is a part of full effective computing research. But it can also classify sound and video into positive, negative or neutral [19]. A large portion of the writing is in the English dialect; however, the number of publications suffers from multi-lingual issues [12].

It is very crucial to understand the main attributes and behaviors of approaches such as opinion mining, sentiment analysis and natural language processing. For example, in elections forecast sentiment analysis classifies natural language data into different positive or negative emotions [2]. On the other hand, if we apply the same approach to health services, for the translation of

literary data about the patient's experience on a marvelous scale use different analytical techniques [14]. As a result of its composition nature, it maintained a strategic distance from the investigative spotlight of ordinary quantitative analysis. Alemi et al., proposed a technique related to the use of sentiment analysis of patient reviews in the form of real-time patient surveys [1]. They show that the comments of a specific patient are either positive or negative, as they set different classes for those sentiments. Moreover, they suggest that we need to compare sentiment analysis with old methods to know about patients' experiences.

SODA and RedMat allow different patients to define their experience with their health and different services at all hospitals in a particular country. People add around a million reviews per day about different services, particularly health care, QoS and management services. It contains an average of about 700 reviews about various hospitals [13, 7]. These comments contain both rating and free-text descriptions. They also calculate the experience of patients with the help of a survey about hospitals.

The rest of the paper is organized as follows: section 2 intro machine learning for patient comments. Section 3 demonstrates the conceptual model which is purposed by us. Section 4 represents the experimental results. Discussion is delivered in Section 5. The conclusion and future work is represented in Section 6.

2 Machine learning for patient comments

We conducted a method test using patient feedback obtained from RedMet and the SODA Choices website. Our primary objective was to estimate patient responses based on their comments. To achieve this, we utilized a machine learning algorithm to categorize the comments into distinct groups. We then compared our results to manually assessed comments by domain experts and conducted interviews to verify accuracy.In our effort to recommend improvements in Quality of Service (QoS), we thoroughly investigated free-text patient comments across various categories, including General Medical Services (GMS), Health Services (HS), Social Services (SS), and Management Services (MS). We developed a predictive model to assess a patient's likelihood of choosing a particular hospital based on the feedback we received. Additionally, our model assessed the hospital's hygiene and patient treatment.

General practitioners and physicians provide essential services through GMS, which form the foundation of healthcare services. The hospital offers HS, en-

compassing emergency services and ensuring patient comfort during painful situations. MS refers to services integrated by the hospital administration to enhance the well-being of patients and improve their service ratings. SS encompasses services provided by hospital staff, including nursing and housekeeping, aimed at creating a healthy and conducive environment for a swift recovery. All of these services collectively contribute to the overall enhancement of hospital QoS.

We used a set of data to test the predicted accuracy of the process. After the very first step i.e., pre-processing, we collect 71% valid data from total comments. We also compare our predicted results with the patient's rating which is based on interview and expert ratings.

3 Proposed model

Keeping in mind the end goal to get more fruitful results, we ensured that the execution of our recommended technique would be comparable to or better than the currently accepted solutions to healthcare difficulties in the field of sentiment analysis, the feature identification module demonstrated that the proposed approach outperforms existing techniques on supplied data. The proposed model is shown in Figure 1.

3.1 Datasets

Most datasets were obtained from Medicare under terms and conditions of use and security caveats. We choose one of them from the Socrata [4], Open Data API (SODA) [22]. Data facilitated in Socrata destinations are accessed using SODA software. We can look for information in various categories, including healing facilities, nursing homes, hospices, long-term care, and supplier directories. The official Centers for Medicare and Medicaid Services information is available on this site as a resource (CMS). The SODA API supports the JSON format, the most commonly used configuration for API answers. Socrata recommends this format since it is the simplest and most productive. Clients can submit and follow up on reviews of medical and dental experts, psychologists, urgent care centers, group practitioners, and hospitals on RateMDs.com. Both patients and doctors will benefit from this site. Table 3.1 shows a description of the dataset.

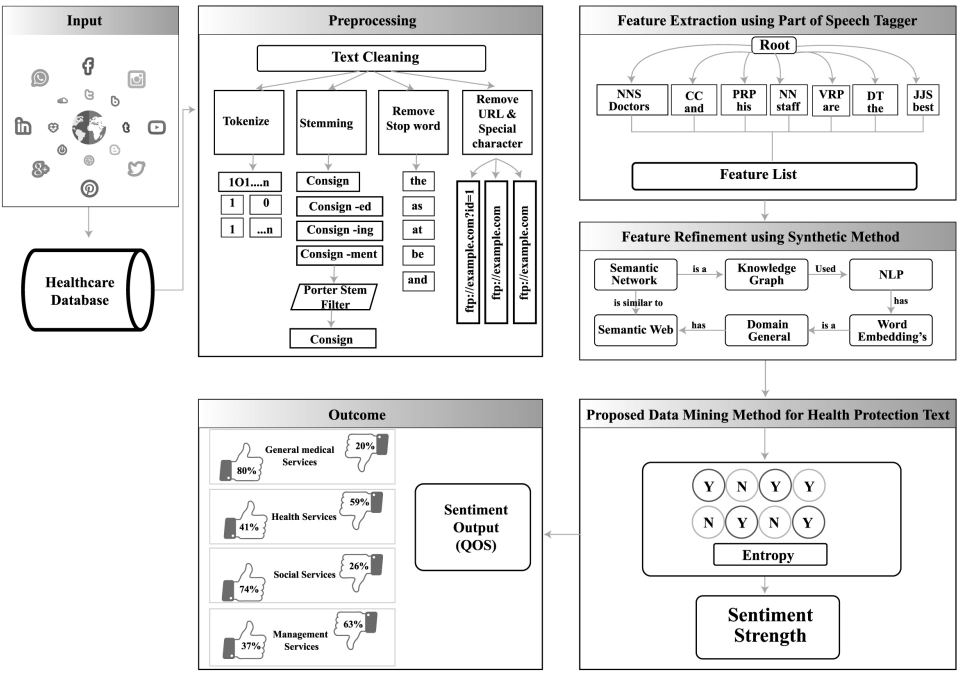


Figure 1: Proposed model

3.2 Pre-processing

The input text is split into sentences in this segment. The POS labeling and stemming are achieved with the help of Stanford CoreNLP [17, 3]. Words with positive polarity are used in sentences like "this is anything but a decent doctor's facility," however the refuting term NOT changes the polarity of a sentence. Furthermore, unigram characteristics do not show relationships between words in the material. Before extracting the unigram features, the negative word's impact should be reflected. Figure 2 depicts the data preprocessing.

3.3 Health protection feature extraction

Any item's reviews or tweets may contain unique characteristics that can be noted separately. Each sentence of a specific review associated with health care can be considered a sack of words at this level. The Modified POS tagger recognizes all highlights as well as opinions from the pack of words. A POS

#	Type	Dataset	Reviews/Tweets
1	RateMDs	Reviews of Patient	689
2	HOADF	Reviews of Physician	122,716
3	Medicare	Hospital	61358
4	Medicare	Nursing Home	40976
5	Medicare	Hospice	45787
6	Medicare	Long Term Care	35000
7	Medicare	Supplier Directory	20000

Table 1: Dataset breakdown [16]

tagging tool looks for grammatical form as well as linguistic linkages in other sentences in a sentence. Each word in a sentence is labeled as an action verb (VB), a noun (NN) and noun phrase (NNP), a proper noun (NNP), adjectives (JJ), and so on.

3.4 Health protection feature refinement using net (synthetic method)

Researchers have successfully applied machine learning algorithms to split sentiments in a document[15, 20]. However, as the feature set of data grows larger, the temporal complexity of these strategies grows. Furthermore, insignificant and repeated features play a role in determining the sentiment of a given document, causing the algorithm's accuracy to vary[8]. The primary goal of this step is to reduce the dimensionality of the feature space to obtain the most perfect feature and reduce computing costs

ConceptNet is a large-scale system that was launched in 1999 to understand the semantics of words [18]. Below are a few stages associated with the arrangement of our conceptual mode.

- ConceptNet, in all of its versions, includes social learning of the English dialect, and its sibling effort provides information on other well-known dialects.
- ConceptNet makes use of DBpedia's subset. It uses Wiktionary, a multilingual vocabulary, to extract knowledge from Wikipedia articles. This dictionary contains information on various topics, including health care.

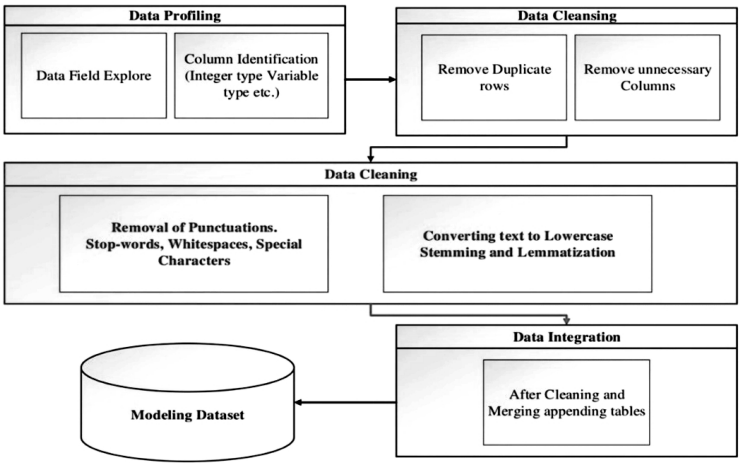


Figure 2: Proposed model

- For expanded information, the WordNet multi-dialect dictionary is also used.
- "Games with Purpose" provided some insight into people's feelings. The Japanese made this game for the GWAP challenge.

Figure 3 depicts a Concept Net improved model that was used in our suggested methodology as a tree with various healthcare-related properties.

3.5 Proposed data mining method for health protection text

The maximum entropy model is a modestly developed statistic model that was originally used to handle enormous amounts of authentic text. Gradually, it became clear that the maximum entropy model is also applicable to natural language processing [21]. The central concept is to provide a known event set, identify potential requirements based on it, and then choose a model that meets the imperative condition. Meanwhile, the probability distribution of the unknowns, distributed equally as may be expected given the conditions, is not completely understood. The probability is calculated using the maximum entropy approach [24].

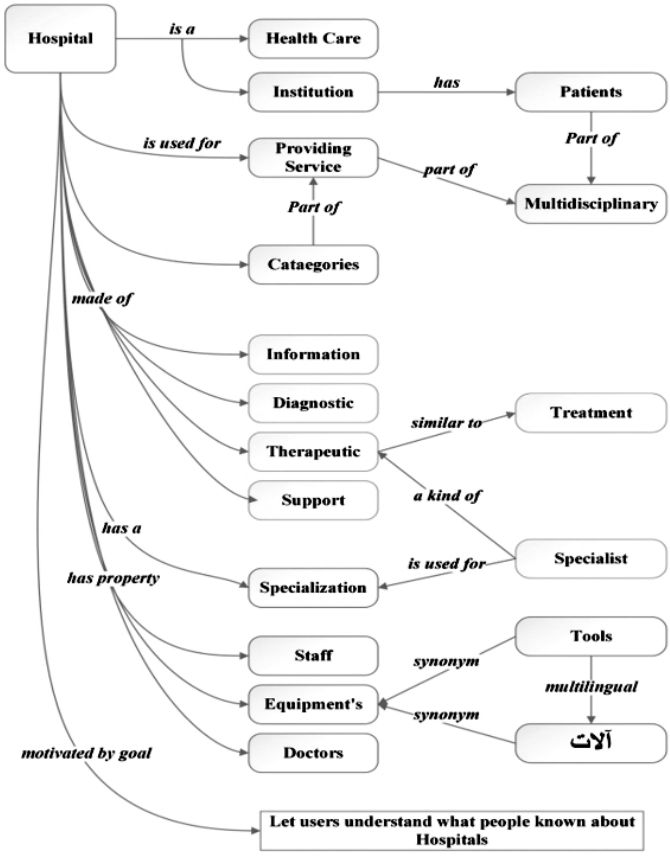


Figure 3: ConceptNet model for healthcare feature extraction

3.6 The text classification process

The preparation and testing of a classifier is required for text classification. The training data must preprocess and depict the text categorization process using the maximum entropy model. Different text features are created by handling training text following word segmentation, removing stop words, feature extraction, and word recurrence measurements. Portraying distinctive content as per the element of the greatest entropy work technique to ascertain the different parameters requires a maximum entropy classifier.

3.7 Strengths and limitations

Sentiment analysis using the technique is as good as the learning data set that we give as input. We can utilize numerous numbers of appraisals in the learning set than in different investigations. Also, with the use of sentiment analysis in medical services information, analysts need to prepare the framework themselves by evaluating reviews and attributing qualities to enable the technique to learn. We utilized a huge dataset that allowed us to specifically look at free text reviews and ratings posted by similar patients which help us to remove the potential base of reviewers during the assignment of a review.

Online comments left on a website without being solicited are likely to tend to gravitate toward models of both negative and excellent comments. Additionally, these online reviews are for the most part contributed by youthful young affluent people. Also, there are parts of patients' reviews that are difficult to process. There are a few words, for example, 'Irony', 'humor' and 'sarcasm' are regularly used in the English dialect and these words are hard for an algorithm to preprocess. The usage of earlier polarity enhanced the outcomes and gave some great results, yet, there were difficulties in understanding the context. Content that is trimmed again and again, for instance, "stank of pee" or "like a holy messenger", is effortlessly categorized as negative or positive. The sense of other ordinarily utilized expressions was hard to develop without knowing about their context. It was extremely hard to predict expressions without knowing the main context.

Currently, our approach cannot use such types of sentences that are looked clear on case to case basis. They are not using the most cutting-edge machine learning algorithms or approaches to classification selection in this early exploratory work, as observed in other industries [2, 24]. However, we believe that further work may have the capacity to embrace this.

4 Results

We had our proposed system assessed by a panel of experts from an NGO called 'X' that provides healthcare solutions. Two Medical Specialists, one Analyst, and two Quality Affirmation Specialists make up their experienced team. We've presented our proposed structure, as well as the knowledge base and user interface we've created. To their respectful expert group, our demonstration covers the strategy of research, organization and administration. We investigated whether such a medicinal services system on a specific topic has been built using our intended healthcare structure and whether it will aid in

improving QoS in hospitals by removing ambivalence and collisions between patients and doctors. For the progress of QoS, we presented parameters based on GMS, MS, HS, and SS. They've started a rating system that goes from 0 to 10. Not Agreed is 0-3 points, moderately Agreed is 4-6 points, and Agreed is 7-10 points. We liquidated the outcomes based on agreement, partially agreed, and not agreed by applying the mean to all of the parameter's resultant predictions. The principle of expert opinions is based on three variables: agreed, disagreed, and somewhat agreed. Following the advice of the experts, it was widely agreed that the proposed technique is ideal for dealing with healthcare facilities in similar hospitals. The proposed approach to collecting patient data is simple to implement and aids in the reduction of comments and reviews.

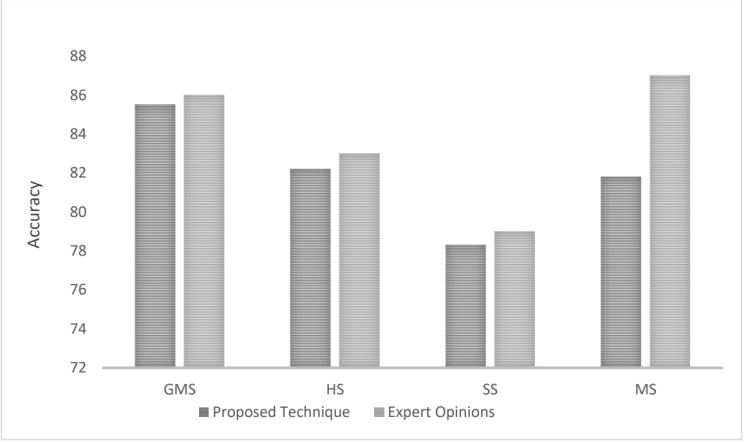


Figure 4: Comparison of proposed technique with expert opinion

Following the evaluation of our predicted method, we compare the achieved results to the interviews conducted with various patients who came to the hospitals at random. This is a poorly controlled process that is used on haphazardly selected patients who are admitted to hospitals and are suffering from various diseases. This interview policy applied to nearly 30 hospitals and included both general and specific questions. In these meetings, we simply select specific information, such as our anticipated topic. Every question has a set of character choices that range from "excellent" to "poor." The suggested analysis ranking differs from the patient ranking. In the first experiment, the obtained results are distinguished by the inclusion of expert suggestions. It demonstrates that the accuracy depicted as a completion measure of the sug-

gested approach is optimistic. Each piece of hospital advice is a perfect match for expert opinions. Figure 4 depicts the graphical representation. The comparison of proposed technique with the interview-based results is shown in Figure 5. Figure 6 represents an overall comparison of the suggested technique with expert opinions and interviews, demonstrating that it performed well in terms of performance measurement. This comparison clearly shows that, given the complexity of the free text and the difficulties of obtaining expert opinions and conducting interviews, the proposed technique performs admirably. Without the challenges of conducting interviews and appointing an expert panel, the accuracy gained is nearly the same.

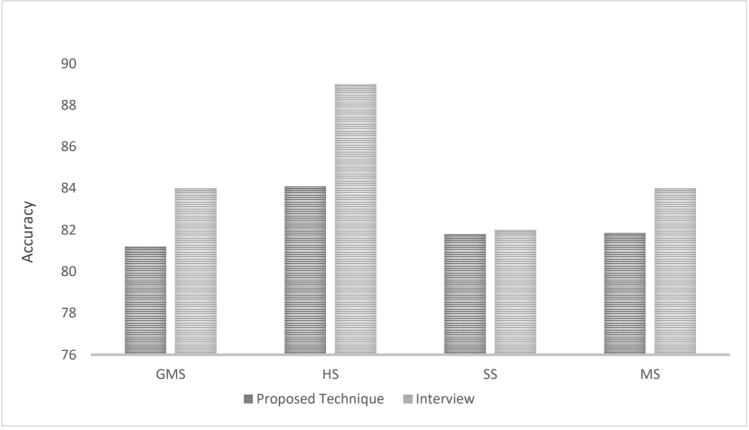


Figure 5: Comparison of proposed technique with interviews conducted

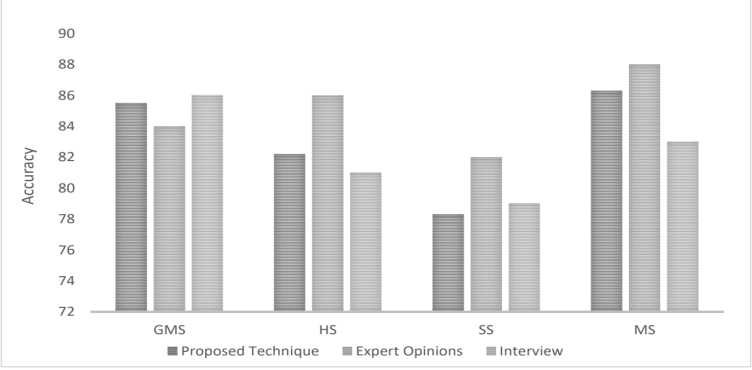


Figure 6: Comparison of proposed technique with the expert opinion and interview

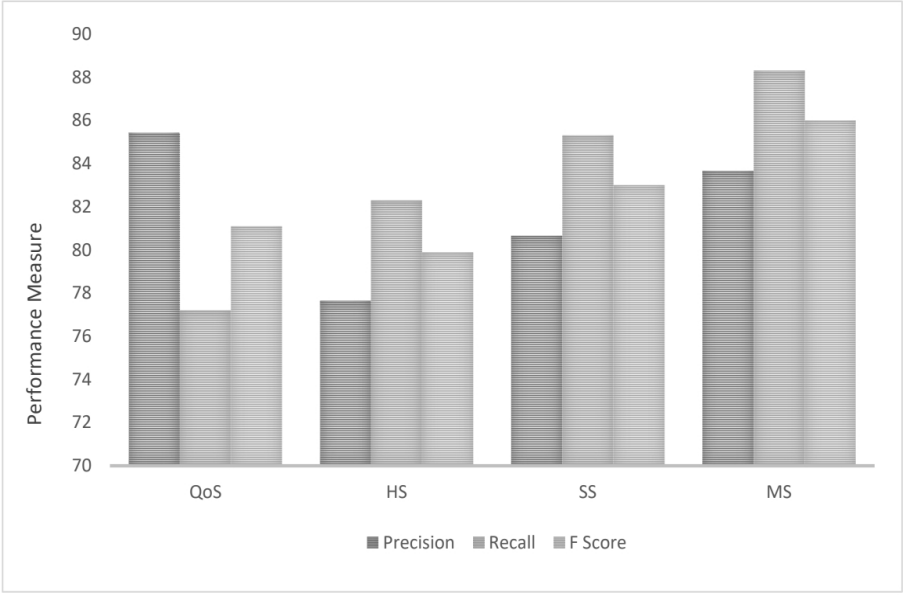


Figure 7: Experimental results in terms of precision, recall and f-score

5 Discussion

There was an agreement between our forecast and patient ratings regarding whether or not they would suggest a facility based on QoS. Our sentiment analysis prediction is accurate 78.5 percent to 87 percent of the time, depending on the classification method we utilized. GMS sentiment analysis is between 84 percent and 86 percent, HS prediction is between 82 percent and 84 percent, SS prediction is between 78 percent and 80 percent, and MS prediction is between 86 percent and 88 percent using these. Using ConceptNet, our suggested strategy yields promising findings and outperforms existing state-of-the-art methods such as surveys and interviews. Precision, recall, and F-score values are shown in Figure 7 for GMS, HS, SS, and MS.

6 Conclusion

This work predicts the sentiment analysis of patients' reviews related to their experience of medicinal services in some hospitals. This novel methodology is

related to patient experience estimated by old methodologies i.e. surveys. This work contributes to understanding the patient reviews related to hospital QoS by analyzing their reviews posted on social media websites. Although there are some future possibilities by processing these reviews in real-time to get useful results. We improve it by refining the data polarity. At the same time, it is helpful to think about the general basic systems that are used in this process with different tools and platforms utilized for opinion mining and sentiment analysis i.e., SentiWordNet and WordNet Affect.

Future efforts to improve the characteristic of this algorithm include the improvement of its preprocessing ability to precisely predict the nature of comments given by the patients. Moreover, there is need to enhance the execution of sentiment analysis techniques, enhancement in the procedure to extract different types of free-content data on the Internet and analyze the connections between comments and clinical QoS. For example, there are a different features that are also added to improve the process by including a higher number of n-grams.

References

[1] F. Alemi, M. Torii, L. Clementz, D. C. Aron, Feasibility of real-time satisfaction surveys through automated analysis of patients' unstructured comments and sentiments, *Quality Management in Healthcare* **21** (2012) 9–19. ⇒332

[2] M. R. Chauhan, A. Sharma, G. Sikka, The emergence of social media data and sentiment analysis in election prediction, *Journal of Ambient Intelligence and Humanized Computing* **12** (2021) 2601–2627. ⇒331, 338

[3] A. Chiche, B. Yitagesu, Part of speech tagging: a systematic review of deep learning and machine learning approaches, *Journal of Big Data* **9** (2022) 1–25. ⇒334

[4] J. S. Erickson, A. Viswanathan, J. Shinavier, Y. Shi, J. A. Hendler, Open government data: A data analytics approach, *IEEE Intelligent Systems* **28** (2013) 19–23. ⇒333

[5] G. G. Gao, J. S. McCullough, R. Agarwal, A, K. Jha, A changing landscape of physician quality reporting: analysis of patientsâ€™ online ratings of their physicians over a 5-year period, *Journal of medical Internet research* **14** (2012) 38. ⇒331

[6] M. Godovykh, A. Pizam, Measuring Patient Experience in Healthcare, *International Journal of Hospitality Management* **112** (2023) 103405. ⇒331

[7] F. Greaves, C. Millett, Consistently increasing numbers of online ratings of healthcare in England, *Journal of Medical Internet Research* **14** (2013) 94. ⇒332

[8] L. He, T. Yin, K. Zheng, They May Not Work! An evaluation of eleven sentiment analysis tools on seven social media datasets, *Journal of Biomedical Informatics* **1**32 (2022) 104142. ⇒335

[9] K. Herng Leong, D. Putri Dahnil, Classification of Healthcare Service Reviews with Sentiment Analysis to Refine User Satisfaction, *International Journal of Electrical and Computer Engineering Systems* **13** (2022) 323–330. ⇒331

[10] JA. M. Hopper, M. Uriyo, Using Sentiment Analysis to Review Patient Satisfaction Data Located on the nternet, *Journal of health organization and management* **29** (2015) 221–233. ⇒331

[11] J. W. Huppertz, P. Otto, Predicting HCAHPS Scores from Hospitals' Social Media Pages: A Sentiment Analysis, *Health Care Management Review* **43** (2018) 359–367. ⇒331

[12] M. R. Kanfoud, A. Bouramoul, SentiCode: A New Paradigm for One-time Training and Global Prediction in Multilingual Sentiment Analysis, *Journal of Intelligent Information Systems* **59** (2022) 501–522. ⇒331

[13] T. Lagu, S. L. Goff , N. S. Hannon, A mixed-methods analysis of patient reviews of hospital care in England: implications for public reporting of health care quality data in the United States, *The Joint Commission Journal on Quality and Patient Safety* **39** (2013) 1–7. ⇒332

[14] S. T. Lai, R. Mafas, Tentiment Analysis in Healthcare: Motives, Challenges and Opportunities pertaining to Machine Learning, *In 2022 IEEE International Conference on Distributed Computing and Electrical Circuits and Electronics (ICDCECE)*, 2022 1–4. ⇒332

[15] B. Liang, H. Su, L. Gui, E. Cambria, R. Xu, Aspect-based sentiment analysis via affective knowledge enhanced graph convolutional networks, *Knowledge-Based Systems* **235** (2022) 107643. ⇒335

[16] A. Nawaz, *Polarity Estimation and Aggregation in Aspect BasedSentiment Analysis,* PhD Thesis, International Islamic University, 2019 ⇒335

[17] A. Nawaz, T. Ali, Y. Hafeez, S. U. Rehman, M. R. Rashid, Mining public opinion: a sentiment based forecasting for democratic elections of Pakistan, *Spatial Information Research* **30** (2022) 169–181. ⇒334

[18] S. P. V. Rajeswaran, Bayesian analysis of ConceptNet relations on PubMed dataset, *Journal of Algebraic Statistics* **1**3 (2022) 2016–2025. ⇒335

[19] M. R. R. Rana, A. Nawaz, J. Iqbal, A Survey on Sentiment Classification Algorithms, Challenges and Applications, *Acta Universitatis Sapientiae, Informatica* **10** (2018) 58–72. ⇒331

[20] M. R. R. Rana, S. U. Rehman, A. Nawaz, T. Ali, A. Imran, A. Alzahrani, A. Almuhaimeed, Aspect-based Sentiment Analysis for Social Multimedia: A Hybrid Computational Framework, *Computer Systems Science and Engineering* **46** (2023) 2415–2428. ⇒335

[21] A. Rusiecki, Trimmed categorical cross-entropy for deep learning with label noise, *Electronics Letters* **5**5 (2019) 319–320. ⇒336

[22] H. Won, J. Han, M. S. Gil, Y. S. Moon, SODAS: Smart Open Data as a Service for Improving Interconnectivity and Data Usability, *Electronics* **12** (2023) 1237–2023. ⇒333

[23] H. Wu, C. Huang, S. Deng, Improving aspect-based sentiment analysis with Knowledge-aware Dependency Graph Network, *Information Fusion* **92** (2023) 289–299. ⇒331

[24] X. Xie, S. Ge, F. Hu, M. Xie, N. Jiang, An improved algorithm for sentiment analysis based on maximum entropy, *Soft Computing* **2**3 (2019) 599–611. ⇒336, 338

Received: July 31, 2023 • Revised: November 4, 2023

Acta Univ. Sapientiae Informatica **15,** 2 (2023) 345–358

DOI: 10.2478/ausi-2023-0021

Applications of edge analytics: a systematic review

Darko ANDROČEC
Faculty of Organization and Informatics,
University of Zagreb
Pavlinska 2, 42000 Varaždin, Croatia
email: dandrocec@foi.unizg.hr

Abstract. With the development and expansion of the Internet of Things, computing at the edge is becoming increasingly important, especially for applications where real-time response is important. In this paper, we made a systematic review of the literature on analytics at the edge. We extracted data from 40 selected primary relevant studies from the complete set of 419 papers retrieved from scientific databases. In our analysis of the full text of every primary study we investigated: temporal distribution of primary studies, publication types, domain and application areas of the primary papers, used machine learning and deep learning methods. We also elaborated on the main themes of the primary studies and recommended some possible interesting future research possibilities.

1 Introduction

With the expansion of the Internet of Things, the number of smart devices equipped with various sensors that generate a large amount of data is also increasing. The most common use case is that all this data is then sent to a centralized server or cloud. Due to the large amount of data, there can be problems with applications that need to work in real time. In addition, the costs of sending and storing data on the cloud are not negligible. This

Key words and phrases: machine learning, toxic comment, deep learning, systematic review

is why computing at the edge is becoming more and more popular, which processes all or part of the IoT data processing closer to the smart devices themselves, without sending all the data to the cloud. Edge computing is more efficient because latency is greatly reduced. At its most basic, edge computing brings data processing and storage closer to the devices that collect the data, rather than relying on a central location that may be physically thousands of kilometers away. Also a lot of AI/Machine Learning/Deep Learning algorithms can be implemented more efficiently closer to the source of the data.

There are some existing reviews on the edge analytics (e.g. most comprehensive are [32] and [34]), but non have focus on applications and domains of edge analytics and do not include the newest papers from 2022 and 2023. Application and domains are very important information in edge analytics, from which practitioners and researchers can conclude which applications are the most prevalent and suitable for the use of edge analytics, and which still need further research. That part is elaborated in detail in our paper and is its main contribution. In addition, we have listed at the end possible future research questions that may be useful to readers and researchers from this and related scientific fields. Analytics at the edge is increasingly important today, as it enables real-time processing and avoids the latency that is inevitable if data is immediately sent to the cloud, and can include diagnostics, descriptive or predictive analytics.

This work is organized as follows: Section 2 describes in detail the steps of the performed systematic literature review procedure. The section 3 explains the results of the systematic review on applications of edge analytics. Last section in this paper lists the conclusions and possible future research possibilities.

2 Research methodology

We have performed our review by using the systematic literature review (SLR) methodology described by Kitchenham and Charters [23] that is developed to be suitable for software engineering research reviews. According to the mentioned SLR methodology there are three main review implementation phases: planning, conducting, and reporting. These main steps of the SLR protocol are listed and elaborated in the next subsections.

2.1 Planning

Planning step of the chosen SLR methodology deals with the explanation of the necessity of conducting a certain systematic literature review with a concrete

theme. In the Introduction section of this work, we have defined the specific need to do a systematic review on applications and domains of edge analytics.

Next, the following main research questions were defined:
RQ1: How has research on edge analytics evolved over time?
RQ2: How is edge analytics research reported and what is the maturity level of this research field?
RQ3: What are main applications and domains of the edge analytics?
RQ4: What types of data are processed mostly in the edge analytics?
RQ5: Which machine learning and deep learning methods are mostly used in edge analytics?

When defining our review protocol, we have decided to include the following scientific databases: IEEE Xplore, Science Direct, and Web of Science Core Collection. The search strings/keywords were simply defined as "edge analytics". We have defined the following inclusion criteria (IC): the main objective of the paper must discuss or investigate an issue related to the applications of edge analytics; the paper must be a research (scientific) work written in English; it should be published as a conference or a journal paper.

We excluded the following papers(EC): studies that are not related to the our defined research questions; papers reported only by abstracts or slides; duplicate studies; papers that do not show any applications/proof-of-concept /domains where edge analytics is used.

2.2 Conducting

The second step of the chosen SLR procedure is conducting. We have performed the search operation on the mentioned three electronic sources using search string *"edge analytics"* on 7th July 2023. We have used a reference management software *Zotero* to easy our SLR. We have organized the retrieved papers in Zotero collection together with their bibliographic information and full texts. Our first search resulted in 202 extracted papers. First search resulted in 419 paper: IEEE Xplore - 128 papers, Science Direct - 114 papers, and Web of Science Core Collection (177). First, we have removed the duplicate papers, because some papers were in more than one scientific databases. After additionally performing all the inclusion/exclusion criteria defined in the previous subsection on titles and abstracts, 129 papers remained. After excluding the unrelated works that are not focused or do not have any applications or domains for edge analytics defined, 78 papers remained. The final selection

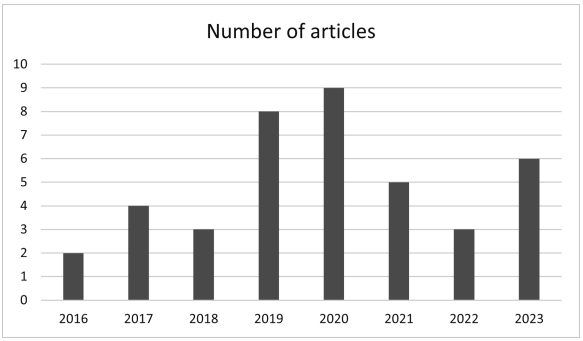


Figure 1: Distribution of the primary studies per year

was done by reading the whole text of the papers, and after this phase, we have selected 40 primary studies for our SLR (Table 1 and Table 2).

2.3 Reporting

Reporting phase includes specifying dissemination mechanisms and formatting the main report. We have extracted data from the 40 selected primary studies and did a synthesis taking into consideration the defined research questions. The results of our systematic literature review on applications of edge analytics are shown in the next sections, and this paper is main dissemination mechanisms of our performed SLR.

3 Results and discussion

3.1 Temporal overview of studies

The earliest found works on applications of edge analytics were published in 2016 (see Fig.1). The peak of number of papers was is 2019 and 2020. But this year (2023) is also very promising, and we expect that this research theme will be also very popular in the near future. We must take into consideration the date of our SLR search (7th July 2023), so our data for 2023 is actually only for the first halve of the year.

3.2 Types of publications

The publication types of primary studies in SLR is demonstrated in Fig. 2. Of the primary studies, 22 were conference papers and 18 were journal papers.

ID	Title of the paper
P1	A Cloud Analytics-Based Electrical Energy Management Architecture Empowered by Edge Analytics AIduino with Push Notifications for Demand-Side Management [26]
P2	Adaptive Edge Analytics for creating memorable customer experience and venue brand engagement, a scented case for Smart Cities [16]
P3	Adaptive Edge Analytics for Distributed Networked Control of Water Systems [19]
P4	Adaptive Recovery of Incomplete Datasets for Edge Analytics [28]
P5	An Efficient Edge Analytical Model on Docker Containers for Automated Monitoring of Public Restrooms in India [13]
P6	Asset Monitoring using Smart Sensing and Advanced Analytics for the Distribution Network [24]
P7	Camera-Based Edge Analytics for Drilling Optimization [12]
P8	Cyber-Physical Analytics: Environmental Sound Classification at the Edge [8]
P9	Deep Learning for Reliable Mobile Edge Analytics in Intelligent Transportation Systems: An Overview [10]
P10	Edge Based Decision Making In Disaster Response Systems [43]
P11	Edge Computing for Having an Edge on Cancer Treatment: A Mobile App for Breast Image Analysis [5]
P12	Enabling Far-Edge Analytics: Performance Profiling of Frequent Pattern Mining Algorithms [1]
P13	GLEAN: Generalized-Deduplication-Enabled Approximate Edge Analytics [15]
P14	Heuristic Algorithms for Co-scheduling of Edge Analytics and Routes for UAV Fleet Missions [21]
P15	Implementation of Intrusion Detection Methods for Distributed Photovoltaic Inverters at the Grid-Edge [17]
P16	Improved Algorithms for Co-Scheduling of Edge Analytics and Routes for UAV Fleet Missions [22]
P17	Intelligent edge analytics for load identification in smart meters [40]
P18	IoT based Ocean Acidification monitoring system with ML based Edge Analytics [18]
P19	Leveraging edge analysis for Internet of Things based healthcare solutions [29]
P20	LiHEA: Migrating EEG Analytics to Ultra-Edge IoT Devices With Logic-in-Headbands [41]

Table 1: Selected primary studies of SLR on edge analytics (part1)

ID	Title of the paper
P21	ML-assisted IC Test Binning with Real-Time Prediction at the Edge [14]
P22	On Delay-Sensitive Healthcare Data Analytics at the Network Edge Based on Deep Learning [9]
P23	Real-Time Monitoring of Agricultural Land with Crop Prediction and Animal Intrusion Prevention using Internet of Things and Machine Learning at Edge [35]
P24	Realising Edge Analytics for Early Prediction of Readmission: A Case Study [33]
P25	Towards Resource-Efficient Wireless Edge Analytics for Mobile Augmented Reality Applications [6]
P26	Using Edge Analytics to Improve Data Collection in Precision Dairy Farming [4]
P27	Using Siamese Networks to Detect Shading on the Edge of Solar Farms [38]
P28	An energy efficient IoT data compression approach for edge machine learning [2]
P29	An OCF-IoTivity enabled smart-home optimal indoor environment control system for energy and comfort optimization [20]
P30	Application of MES/MOM for Industry 4.0 supply chains: A cross-case analysis [31]
P31	Edge computing for Internet of Things: A survey, e-healthcare case study and future direction [37]
P32	Federated Learning for improved prediction of failures in Autonomous Guided Vehicles [39]
P33	Genetically optimized Fuzzy C-means data clustering of IoMT-based biomarkers for fast affective state recognition in intelligent edge analytics [25]
P34	A Deep Learning Approach for Voice Disorder Detection for Smart Connected Living Environments [42]
P35	A Novel Edge Analytics Assisted Motor Movement Recognition Framework Using Multi-Stage Convo-GRU Model [30]
P36	Design and Implementation of Cloud Analytics-Assisted Smart Power Meters Considering Advanced Artificial Intelligence as Edge Analytics in Demand-Side Management for Smart Homes [7]
P37	Edge analytics for anomaly detection in water networks by an Arduino101-LoRa based WSN [3]
P38	Liver Disease Detection: Evaluation of Machine Learning Algorithms Performances With Optimal Thresholds [36]
P39	Novel smart home system architecture facilitated with distributed and embedded flexible edge analytics in demand-side management [27]
P40	Smart surveillance system for real-time multi-person multi-camera tracking at the edge [11]

Table 2: Selected primary studies of SLR on edge analytics (part2)

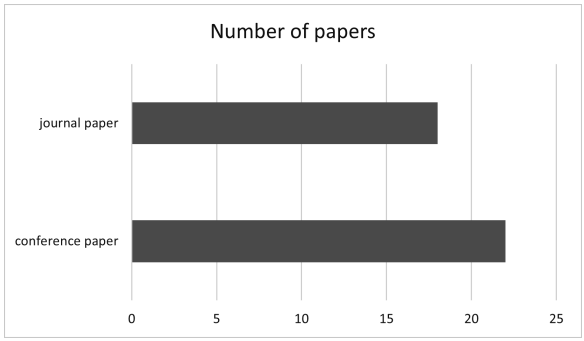


Figure 2: Publication types of primary studies in SLR

We did not find any relevant scientific book chapter. The number of papers in scientific journals and conferences is actually quite similar, which suggests that this research topic is already quite mature.

3.3 Data types used in edge analytics

Most of the primary studies have used the following data types in edge analytics: numeric sensor data (32 papers), video data (6 papers), and voice/sound data (2 papers). This distribution is shown in Fig. 3. Another view of the data type sphere can be according to the domain from which the data comes. In this sense, we can single out the following types of data that are most common in the set of primary studies obtained by performed SLR methodology: health data (10 papers), environment data (temperature, humidity etc. - 9 papers), industrial sensor data (5 papers), and security and disaster management data (4 papers). Graphical depiction of the mentioned distribution can be found at Fig. 4.

3.4 Applications and domains of edge analytics

Table 3. lists the most important applications and domains of edge analytics in primary studies of our SLR. Some of the papers can have more than one application/domain, and for some papers it is impossible to determine the used domain. The most often domain is healthcare. More concretely in our set of SLR's primary studies we have papers using edge analytics to help cancer treatment (P11 [5]), patient monitoring (P19 [29], P24 [33], P35 [30]), EEG analytics (P20 [41]), e-healthcare (P31 [37]), voice disorder (P35 [30]), liver disease detection (P38 [36]). The labels in parentheses are the codes of the

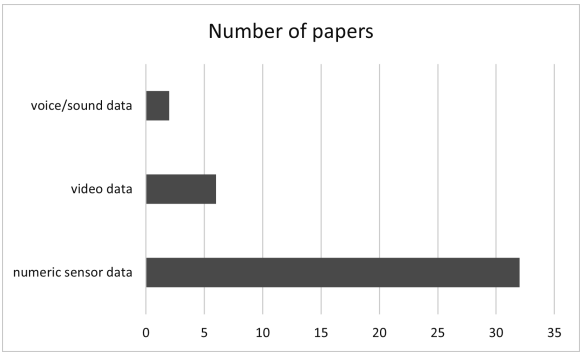


Figure 3: Data types used in edge analytics

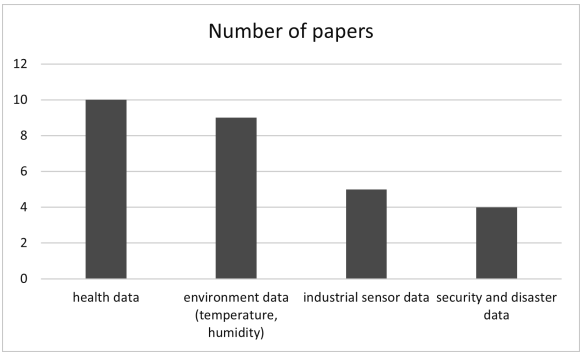


Figure 4: Types of data by domain

primary studies listed in Table 1 and Table 2 of this work (selected primary studies list of SLR on edge analytics part 1 and part 2).

The next domains with more papers are smart home, energy management, and transportation. In the smart home domains, most investigated themes are: smart power meters (P1 [26], P3 [19], P36 [7]), smart buildings (P4 [28]), smart sanitization system (P5 [13]). Papers focused on energy management deal with the following issues: smart energy management (P1 [26]), water leak management (P3 [19]), electric distribution network monitoring (P6 [24]), smart electric network (P17 [40]), and energy savings (P29 [20]). Regarding transportation domain in edge analytics, we have found the following themes in primary studies: intelligent transportation system (P9 [10]), Unmanned Aerial Vehicles or drones (P14 [21], P16 [22]), and driving behavior monitoring (P28 [2]).

Agriculture domains includes the paper on a smart irrigation (P23 [35]) and one work on a precision dairy farming (P26 [4]). Using edge analytics for disaster management is described in the paper P10 [43] and for security (more specifically intrusion detection) in the paper P16 [22]. Marketing domain has two papers, one on electronic scent diffusers (P2 [16]), and another one on market basket analysis (P12 [1]). Regarding industry domain, we have found one paper on drilling automation in oil and gas industry (P7 [12]) and one paper on IC test in semiconductor manufacturing (P21 [14]).

Domain of the application	**Number of papers**
Healthcare	9
Agriculture	2
Smart home	5
Energy management	5
Transportation	5
Disaster management	1
Security	1
Marketing	2
Industry	2

Table 3: Applications and domains of edge analytics

3.5 Used machine and deep learning methods

Table 4 lists the most used machine learning and deep learning methods in edge analytics according to primary studies set of our performed SLR. The most used methods are convolutional neural networks (7 papers), other types of artificial networks (4 papers), Kalman filter, k-nearest neighbours, recurrent neural network, and YOLO (all 5 mentioned in 3 papers). It turns out that the most commonly used methods are different forms of neural networks that are implemented on the edge, regardless of the limitations of edge devices in terms of processing capabilities as well as data storage capacity. The type of artificial neural network often depends on the type and characteristics of data processed at the edge. For anomaly detection, Kalman filter method is often used.

Machine and deep learning methods	Number of papers
Convolutional Neural Network	7
Kalman filter	3
Random Forest	2
K-Nearest Neighbours	3
Support Vector Machine	2
Recurrent Neural Network (RNN)	3
Polynomial regression	1
Support Vector Clustering (SVC)	2
Other types of Artificial Neural Network	4
XGBoost	1
Genetic algorithms	1
Naïve Bayes	1
Logistic Regression	2
YOLO	3
Siamese neural network	2

Table 4: Used machine and deep learning methods

4 Conclusions

Recently, the number of different smart devices that facilitate various activities in industry, smart homes, healthcare and other domains has been increasing. Many applications that use smart devices and related solutions need to work in real time. Because of this, it is often impractical to send the large amounts of data generated by today's IoT devices to a remote cloud. In many cases, edge computing is being used, which enables data processing and storage to be done closer to the devices themselves, thus reducing latency. In this paper, we made a systematic review of the literature on the applications of analytics at the edge.

In this conclusion, we look back at the research questions we defined at the beginning of the SLR procedure and look at the results after a detailed reading of all 40 primary studies.

RQ1: How has research on edge analytics evolved over time?

Research on edge analytics started in 2016, and the most papers were published in 2019 and 2020. In the first half of the year 2023 we can see a repeated growth in the popularity of this research topic.

RQ2: How is edge analytics research reported and what is the maturity level of this research field?

The maturity of this research topic is quite high, because in the case of primary

studies, we have almost the same number of works performed at scientific conferences and in scientific journals.
RQ3: What are main applications and domains of the edge analytics?
The main application and domains of the edge analytics are healthcare, smart home, energy management, and transportation. More detailed analysis can be found in Section 3.4.
RQ4: What types of data are processed mostly in the edge analytics?
Numeric sensor data, video data, and voice/sound data are most common data types used in edge analytics.
RQ5: Which machine learning and deep learning methods are mostly used in edge analytics? Generally, the most used methods are different types of artificial neural networks. More detailed analysis on this topic can be found in Section 3.5 of this work.

Future research can include techniques, methods, and methodologies to enable and efficiently execute deep learning (various types of artificial neural networks) on the edge devices that are limited by processing power as well as the amount of data they can store. Interoperability of edge analytics data, different IoT devices and services, and edge devices and frameworks is also challenging future research theme. Edge analytics as a service can be another interesting possibility for future research.

References

[1] K. A. Alam, R. Ahmad, et al., Enabling far-edge analytics: performance profiling of frequent pattern mining algorithms, *IEEE Access* **5** (2017) 8236–8249. ⇒349, 353

[2] J. Azar, et al., An energy efficient IoT data compression approach for edge machine learning, *Future Generation Computer Systems* **96** (2019) 168–75. ⇒ 350, 352

[3] M. Babazadeh, Edge analytics for anomaly detection in water networks by an Arduino101-LoRa based WSN, *ISA Transactions*, **92** (2019) 273–85. ⇒350

[4] K. Bhargava, et al., Using edge analytics to improve data collection in precision dairy farming, *2016 IEEE 41st Conference on Local Computer Networks Workshops (LCN Workshops)*, Dubai, United Arab Emirates, 2016, pp. 137–44. ⇒ 350, 353

[5] E. Charteros, I. Koutsopoulos, Edge computing for having an edge on cancer treatment: a mobile app for breast image analysis, *2020 IEEE International Conference on Communications Workshops (ICC Workshops)*, virtual, 2020, pp. 1–6. ⇒349, 351

[6] L. E. Chatzieleftheriou, et al., Towards resource-efficient wireless edge analytics for mobile augmented reality applications, *2018 15th International Symposium on Wireless Communication Systems (ISWCS)*, Lisbon, Portugal, 2018, pp. 1–5. ⇒350

[7] Y.Y. Chen, et al., Design and implementation of cloud analytics-assisted smart power meters considering advanced artificial intelligence as edge analytics in demand-side management for smart homes, *Sensors*, **19**, 9 (2019), 2047. ⇒350, 352

[8] D. Elliott, et al., Cyber-physical analytics: environmental sound classification at the edge, *2020 IEEE 6th World Forum on Internet of Things (WF-IoT)*, New Orleans, LA, USA, 2020, pp. 1–6. ⇒349

[9] Z. M. Fadlullah, et al., On delay-sensitive healthcare data analytics at the network edge based on deep learning, *2018 14th International Wireless Communications Mobile Computing Conference (IWCMC)*, Limassol, Cyprus, 2018, pp. 388–93. ⇒350

[10] A. Ferdowsi, et al., Deep learning for reliable mobile edge analytics in intelligent transportation systems: an overview, *IEEE Vehicular Technology Magazine*, **14**, 1 (2019) 62–70. ⇒349, 352

[11] B. Gaikwad, A. Karmakar, Smart surveillance system for real-time multi-person multi-camera tracking at the edge, *Journal of real-time image processing*, **18**, 6 (2021) 1993–2007. ⇒350

[12] C. P. Gooneratne, Camera-based edge analytics for drilling optimization, *2020 IEEE International Conference on Edge Computing (EDGE)*, virtual, 2020, pp. 111–15. ⇒349, 353

[13] R. Gore, et al., An efficient edge analytical model on Docker containers for automated monitoring of public restrooms in India, *2020 IEEE International Conference on Advanced Networks and Telecommunications Systems (ANTS)*, New Delhi, India, 2020, pp. 1–6. ⇒349, 352

[14] T. Honda, et al., ML-assisted IC test binning with real-time prediction at the edge, *2023 7th IEEE Electron Devices Technology & Manufacturing Conference (EDTM)*, Seoul, Korea, 2023, pp. 1–4. ⇒350, 353

[15] A. Hurst, et al., GLEAN: Generalized-deduplication-enabled approximate edge analytics, *IEEE Internet of Things Journal*, **10**, 5 (2023) 4006–4020. ⇒349

[16] A. Ilapakurti, et al., Adaptive edge analytics for creating memorable customer experience and venue brand engagement, a Scented Case for Smart Cities, *2017 IEEE SmartWorld, Ubiquitous Intelligence & Computing, Advanced & Trusted Computed, Scalable Computing & Communications, Cloud & Big Data Computing, Internet of People and Smart City Innovation (SmartWorld/SCALCOM/UIC/ATC/CBDCom/IOP/SCI)*, San Francisco, CA, USA, 2017, pp. 1–8. ⇒349, 353

[17] C. B. Jones, et al., Implementation of intrusion detection methods for distributed photovoltaic inverters at the grid-edge, *2020 IEEE Power & Energy Society Innovative Smart Grid Technologies Conference (ISGT)*, Washington, DC, USA, 2020, pp. 1–5. ⇒349

[18] K. V. Gopika, et al., IoT based ocean acidification monitoring system with ML based edge analytics, *2022 4th International Conference on Inventive Research in Computing Applications (ICIRCA)*, Coimbatore, India, 2022, pp. 345–53. ⇒ 349

[19] S. Kartakis, et al., Adaptive edge analytics for distributed networked control of water systems, *2016 IEEE First International Conference on Internet-of-Things Design and Implementation (IoTDI)*, Berlin, Germany, 2016, pp. 72–82. ⇒349, 352

[20] A. N. Khan, et al., An OCF-IoTivity enabled smart-home optimal indoor environment control system for energy and comfort optimization, *Internet of Things*, **22** (2023) 100712. ⇒350, 352

[21] A. Khochare, Y. Simmhan, et al., Heuristic algorithms for co-scheduling of edge analytics and routes for UAV fleet missions, *IEEE INFOCOM 2021 - IEEE Conference on Computer Communications*, Vancouver, British Columbia, Canada, 2021, pp. 1–10. ⇒349, 352

[22] A. Khochare, F. B. Sorbelli, et al., Improved algorithms for co-scheduling of edge analytics and routes for UAV fleet missions, *IEEE/ACM Transactions on Networking*, **1** (2023) 1–17. ⇒349, 352, 353

[23] B. Kitchenham, S. Charters, Guidelines for performing systematic literature reviews in software engineering(2007). ⇒346

[24] S. Kulkarn, et al., Asset monitoring using smart sensing and advanced analytics for the distribution network, *2019 North American Power Symposium (NAPS)*, Wichita, USA, 2019, pp. 1–6. ⇒349, 352

[25] A. Kumar, et al., Genetically optimized fuzzy C-Means data clustering of IoMT-based biomarkers for fast affective state recognition in intelligent edge analytics, *Applied Soft Computing*, **109** (2021) 107525. ⇒350

[26] Y.H. Lin, A cloud analytics-based electrical energy management architecture empowered by edge analytics AIduino with push notifications for demand-side management, *2019 IEEE 2nd International Conference on Power and Energy Applications (ICPEA)*, Signapore, 2019, pp. 1–6. ⇒349, 352

[27] Y.H. Lin, Novel smart home system architecture facilitated with distributed and embedded flexible edge analytics in demand-side management, *International Transactions on Electrical Energy Systems*, vol. 29, no. 6, June 2019, p. 1214. ⇒350

[28] I. Lujic, et al., Adaptive recovery of incomplete datasets for edge analytics, *2018 IEEE 2nd International Conference on Fog and Edge Computing (ICFEC)*, Washington DC, DC, USA, 2018, pp. 1–10. ⇒349, 352

[29] K. J. Madukwe, et al., Leveraging edge analysis for Internet of things based healthcare solutions, *2017 IEEE 3rd International Conference on Electro-Technology for National Development (NIGERCON)*, Owerri, Nigeria, 2017, pp. 720–25. ⇒349, 351

[30] A. Manocha, R. Singh, A novel edge analytics assisted motor movement recognition framework using multi-stage Convo-GRU model, *Mobile Networks & Applications*, **27**, 2 (2022) 657–676. ⇒350, 351

[31] S. Mantravadi, et al., Application of MES/MOM for Industry 4.0 supply chains: a cross-case analysis, *Computers in Industry*, **148** (2023) 103907. ⇒350

[32] M.G.S. Murshed, et al., Machine learning at the network edge: a survey, *ACM Computing Surveys*, **54**, 8 (2022) 1–37. ⇒346

[33] Y. Nan, et al., Realising edge analytics for early prediction of readmission: a case study, *2020 IEEE International Conference on Cloud Engineering (IC2E)*,Sydney, Australia, 2020, pp. 95–104. ⇒350, 351

[34] S. Nayak, et al., A review on edge analytics: issues, challenges, opportunities, promises, future directions, and applications, *Digital Communications and Networks*, (2022). ⇒346

[35] R. Nikhil, et al., Real-time monitoring of agricultural land with crop prediction and animal intrusion prevention using internet of things and machine learning at edge, *2020 IEEE International Conference on Electronics, Computing and Communication Technologies (CONECCT)*, Bangalore, India, 2020, pp. 1–6. ⇒350, 353

[36] A. Pan, et al., Liver disease detection: evaluation of machine learning algorithms performances with optimal thresholds, *International journal of healthcare information systems and informatics*, **17**, 2 (2022). ⇒350, 351

[37] P. P. Ray, et al., Edge computing for Internet of things: a Survey, e-healthcare case study and future Direction, *Journal of Network and Computer Applications*, **140** (2019) 1–22. ⇒350, 351

[38] S. Shapsough, et al., Using Siamese networks to detect shading on the edge of solar farms, *2020 7th International Conference on Internet of Things: Systems, Management and Security (IOTSMS)*, Paris, France, 2020, pp. 1–8. ⇒350

[39] B. Shubyn, et al., Federated learning for improved prediction of failures in autonomous guided vehicles, *Journal of Computational Science*, **68** (2023) 101956. ⇒350

[40] T. Sirojan, et al., Intelligent edge analytics for load identification in smart meters, *2017 IEEE Innovative Smart Grid Technologies - Asia (ISGT-Asia)*, Auckland, New Zealand, 2017, pp. 1–5. ⇒349, 352

[41] T. Tazrin, et al., LiHEA: migrating EEG analytics to ultra-edge IoT devices with logic-in-headbands, *IEEE Access*, **9** (2021) 138834–138848. ⇒349, 351

[42] L. Verde, et al., A deep learning approach for voice disorder detection for smart connected living environments, *ACM Transactions on Internet techbology*, **22**, 1 (2022) 8. ⇒350

[43] J. Wagner, M. Roopaei, Edge based decision making in disaster response systems, *2020 10th Annual Computing and Communication Workshop and Conference (CCWC)*, Las Vegas, NV, USA, 2020, pp. 10469–73. ⇒349, 353

Received: September 22, 2023 • Revised: November 14, 2023

Acta Univ. Sapientiae Informatica **15,** 2 (2023) 359–403

DOI: 10.2478/ausi-2023-0022

Precognition of mental health and neurogenerative disorders using AI-parsed text and sentiment analysis

Attila BIRÓ
Department of Physiotherapy, University of Malaga, 29071 Malaga, Spain
Department of Electrical Engineering and Information Technology, George Emil Palade University of Medicine, Pharmacy, Science, and Technology of Targu Mures, 540088, Romania
Biomedical Research Institute of Malaga (IBIMA), 29590 Malaga, Spain
email: abiro@uma.es
ORCID: 0000-0002-0430-9932

Antonio Ignacio CUESTA-VARGAS
Department of Physiotherapy, University of Malaga, 29071 Malaga, Spain
Biomedical Research Institute of Malaga (IBIMA), 29590 Malaga, Spain
Faculty of Health Science, School of Clinical Science, Queensland University Technology, Brisbane 4000, Australia
email: acuesta@uma.es
ORCID: 0000-0002-8880-4315

László SZILÁGYI
Physiological Controls Research Center, Óbuda University, 1034 Budapest, Hungary
Computational Intelligence Research Group, Sapientia Hungarian University of Transylvania, 540485 Targu Mures, Romania
email: lalo@ms.sapientia.ro
ORCID: 0000-0001-6722-2642

Key words and phrases: mental health disorder, neurogenerative disorder, anomaly detection, sports safety, performance sports, BERT, GPT-3, sentiment analysis, multimodal learning

Abstract. The paper examines the potential of artificial intelligence (AI) in parsing text and conducting sentiment analysis to identify early markers of mental health and neurodegenerative disorders. Through the analysis of textual data, we investigate whether AI can provide a non-invasive, continuous, and objective complement to traditional diagnostic practices. *Background*: the early detection of mental health (such as depression, anxiety, psychotic disorders, Alzheimer's disease and dementia) and neurodegenerative disorders (like Parkinson's disease) remains a critical challenge in clinical practice. Traditional diagnostic methods rely on clinical evaluations that may be subjective and episodic. Recent advancements in AI and natural language processing (NLP) have opened new avenues for precognitive health assessments, suggesting that variations in language and expressed sentiments in written text can serve as potential biomarkers for these conditions. *Materials and Methods*: the research used a dataset comprising various forms of textual data, including anonymized social media interactions, transcripts from patient interviews, and electronic health records. NLP algorithms were deployed to parse the text, and machine learning models were trained to identify language patterns and sentiment changes. The study also incorporated a sentiment analysis to gauge emotional expression, a key component of mental health diagnostics. *Results*: the AI models were able to identify language use patterns and sentiment shifts that correlated with clinically validated instances of mental health symptoms and neurodegenerative conditions. Notably, the models detected an increased use of negative affect words, a higher frequency of first-person singular pronouns, and a decrease in future tense in individuals with depression. For neurodegenerative conditions, there was a notable decline in language complexity and semantic coherence over time. *Conclusions*: the implemented pipeline of AI-parsed text and sentiment analysis appears to be a promising tool for the early detection and ongoing monitoring of mental health and neurodegenerative disorders. However, these methods are supplementary and cannot replace the nuanced clinical evaluation process. Future research must refine the AI algorithms to account for linguistic diversity and context, while also addressing ethical considerations regarding data use and privacy. The integration of AI tools in clinical settings necessitates a multidisciplinary approach, ensuring that technological advancements align with patient-centered care and ethical standards.

1 Introduction

Research using social media posts and other digital footprints as potential sources of health-related information have started to become hot topic nowadays, taking into consideration that any tool developed for disease identifica-

tion from text [21] would need to be highly accurate and validated extensively to be used in a clinical setting. Identification of diseases through text messages [76] can be challenging yet feasible within certain contexts. The approach relies on analyzing patterns in language that may be indicative of cognitive, psychological, or even some physical health conditions. Here's how it might work for various conditions, like: (1) *Mental health disorders* [28, 29]: (a) changes in the frequency of communication, use of negative words, and shifts in the complexity of language may suggest *depression or anxiety* [7]; (b) the disorganized thought patterns that emerge in how a person composes messages could be a warning sign of *psychotic disorders* [30]; or (c) the repeated questions, simpler sentence structures, or a notable decline in the complexity of language could indicate cognitive decline, like *Alzheimer's disease* and *dementia* [4]; (2) *Neurodegenerative diseases* [55, 49]: while not directly identifiable through text, subtle changes in typing speed and fine motor skills required for typing might provide early clues of *Parkinson's disease* [39, 54, 1, 20]; (3) *Sleep disorders* [73, 66]: late-night time stamps and content that indicates restlessness or consistent complaints about lack of sleep may be suggestive of *insomnia*; (4) *Infectious diseases* [71, 12, 72, 16]: it is less likely to identify infectious diseases from text messages unless the content explicitly describes symptoms or experiences related to the *infectious disease* [71]; (5) *Chronic diseases* [41, 43, 45]: if someone frequently discusses feelings of tiredness, changes in weight, or other symptomatic experiences, this could indirectly hint at chronic conditions like *diabetes or thyroid issues* [31]. While text message analysis may provide signals indicative of a health issue [58, 75], this method is far from diagnostic so this can be just a preventive supportive tool for specialists (see Figure 1).

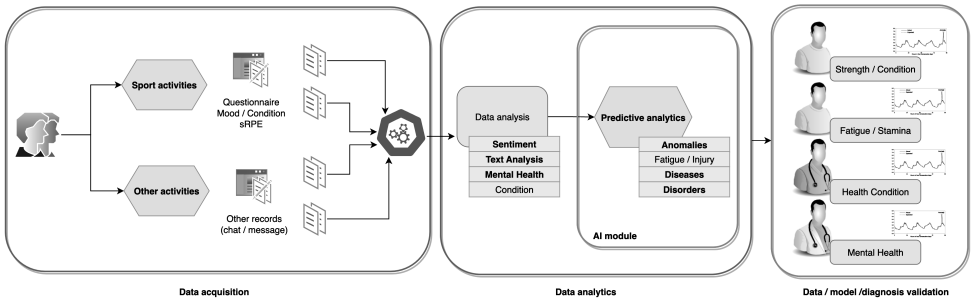


Figure 1: Text to diagnosis with AI: Conceptual approach

The identification of diseases requires thorough clinical evaluation and should not be done solely on the basis of text message analysis. Moreover, ethical con-

siderations around privacy and consent are paramount when analyzing personal communications for health-related insights. From a data science and AI perspective, the process would involve: (1) data collection or/and preparation; (2) Natural Language Processing (NLP); (3) Machine Learning (ML) and (4) Validation. What is of paramount importance is to obtain a large, reproducible, valid dataset of text messages with appropriate permissions. This process is followed by employing techniques to analyze semantic content, sentiment [40], and changes in language patterns over time. The next step is to develop predictive models that could correlate certain text features with disease markers, which is followed by a rigorous testing and validation process with clinical data to ensure that predictions have real-world applicability and do not result in false positives or negatives.

In the quest to comprehend and enhance mental health and neurodegenerative disorder diagnosis, it is imperative to *examine the current diagnostic methodologies* (Section 1.1) meticulously, acknowledge their *inherent limitations* (Section 1.2), and underscore the *significance of ongoing research in sports and sports safety* (Section 1.3), which offers a unique perspective on cognitive and psychological well-being.

1.1 Existing methods of mental health and neurogenerative disorders diagnosis

The diagnosis of mental health and neurodegenerative disorders is an intricate and multifaceted process, which traditionally involves a combination of clinical evaluation, neuropsychological testing, biomarker analysis, and neuroimaging techniques. Clinicians rely on structured interviews and standardized questionnaires [23] to ascertain the presence of symptomatology consistent with diagnostic criteria, such as those outlined in the Diagnostic and Statistical Manual of Mental Disorders (DSM) [17] or the International Classification of Diseases (ICD). Neuropsychological assessments [78] provide a quantitative measure of cognitive functions, including memory, attention, language, and executive function, which can be indicative of specific neurocognitive disorders.

In the realm of neurodegenerative diseases, the utilization of biomarkers obtained from cerebrospinal fluid (CSF) and blood tests [3, 68] can offer biochemical evidence of underlying pathophysiology, such as the presence of amyloid-beta or tau proteins in Alzheimer's disease [68]. Neuroimaging techniques, including magnetic resonance imaging (MRI), positron emission tomography (PET), and computed tomography (CT), serve to visualize structural and

functional brain changes [69, 61, 27]. These traditional methods, while robust, are complemented by emerging technologies such as digital phenotyping and machine learning algorithms that analyze patterns in speech, typing, and daily activity data to enhance early detection and personalized care approaches. However, the integration of these novel methods into clinical practice remains a subject of ongoing research and ethical scrutiny.

1.2 Limitations of existing methods of mental health and neurogenerative disorders diagnosis

The diagnostic paradigms for mental health and neurodegenerative disorders [25] are constrained by several limitations that can impede their efficacy. Clinically, reliance on self-reported symptoms and observable behavior during patient interviews is subject to biases [51] and the variable ability of patients to accurately convey their experiences. The heterogeneity of symptom presentation and the overlap between different disorders can lead to diagnostic ambiguity. Neuropsychological tests, while invaluable, are time-consuming, require specialized personnel to administer, and may be influenced by an individual's educational background, cultural factors, and test-retest variability [35].

In the context of neurodegenerative diseases [44], the definitive diagnosis [77] is often only possible post-mortem through histopathological examination, with current in vivo techniques providing primarily supportive evidence. Biomarkers, although a promising avenue for early detection, are not universally available and can lack specificity, as many are not exclusive to a single disorder. Neuroimaging [64], while offering detailed insights into brain structure and function, is expensive, not always accessible, and can yield false negatives, especially in the early stages of neurodegenerative processes. These limitations underscore the necessity for continuous refinement of diagnostic tools and the development of more accessible, objective, and precise methods of assessment.

1.3 The importance of research in sports and sports safety

The precognition of mental health and neurodegenerative disorders within the sporting domain [74] harnesses the potential of AI-parsed text and sentiment analysis as an innovative means to safeguard athlete well-being and longevity in sports. Athletic performance is intricately linked to mental health, and the early detection of disorders using AI can significantly mitigate risks, enhance performance longevity, and promote a healthier sporting environment.

AI-parsed text analysis and sentiment analysis [42] afford a unique opportunity for the monitoring of athletes' mental health by analyzing communication patterns in interviews, social media [79] posts, and other written or spoken narratives. Athletes, who often face immense pressure to perform, may exhibit early signs of stress, anxiety, or depression in their language use, which AI can detect more consistently and objectively than traditional self-report measures. Moreover, neurodegenerative diseases, such as chronic traumatic encephalopathy (CTE) [57], which are a concern in contact sports, may manifest early cognitive and behavioral changes that subtly surface in linguistic expression—changes to which AI algorithms can be finely attuned.

The implementation of such technology in the sports sector offers a proactive strategy for identifying athletes at risk, enabling timely interventions. It also aligns with the broader movement towards personalized medicine in sports, where individual mental health trajectories inform tailored support programs. Furthermore, it emphasizes the duty of care that sporting organizations have towards their athletes, extending beyond physical health to encompass cognitive and emotional well-being.

AI's capability to analyze sentiment and detect mood fluctuations can serve as a barometer for athlete burnout [70] or diminished motivation, both of which are pivotal for sports safety and performance optimization. In providing a continuous, data-driven monitoring system, AI tools can alert coaches and medical teams to potential mental health issues before they escalate, potentially averting the long-term consequences associated with prolonged stress or undiagnosed conditions.

The methods of this paper in sports and sports safety are expected to represent a significant advancement towards fostering safer sporting environments and ensuring the holistic health of athletes. By employing psycholinguistic analysis, it is possible to examine an athlete's sentiments by studying their distinctive linguistic patterns. This approach involves monitoring the trends and evolution of their communication style, including rhythm, speed, and complexity. Through this analysis, it becomes feasible to identify trends that may indicate a deteriorating mental condition, such as semantic anomalies or pragmatic failures. The utilization of artificial intelligence (AI) in the field of sentiment analysis holds the potential to provide a prognostication of health warning indicators for sports safety. It necessitates a collaborative effort involving data scientists, clinicians, and sports professionals to refine these tools for the nuances of athletic communication and to integrate them ethically and effectively into sports health practices.

2 Objectives

This study examines the efficacy of an AI-driven pipeline for accurately predicting distinct mental health and neurodegenerative problems [15] based on textual data. The objective of this study is to provide a systematic methodology for evaluating the mental health of athletes, with the intention of minimizing the influence of self-reporting and third-party bias [51]. An additional secondary objective of this study is to integrate the findings as a valuable resource within the athletic training and health management ecosystem to enhance athlete safety and performance. This study aims to contribute to the expanding field of sports science by showcasing the potential effectiveness of utilizing sophisticated methodologies, such as NLP and ML, for the purpose of managing mental health in high-performance sports.

3 Novelties of the approach

Connecting text analysis with sentiment analysis [8] can enhance the ability to identify potential health issues. Sentiment analysis is a form of NLP that assesses the affective state of the text, which could relate to the emotional state of the individual writing the message. In research settings, sentiment analysis is increasingly being used to study large datasets from social media to identify public health trends and even to monitor the well-being of specific individuals (with their consent). Sports psychology shows that athletes' mental health profiles differ from those of non-athletes. Many athletes, especially those in competitive and high-stakes contexts, learn resilience, stress management, discipline, and focus, which might affect their mental health differently than non-athletes. Further, the sort of sport done can affect mental health; team sports encourage community and teamwork, whereas solo sports emphasize self-reliance and personal goal setting. Due to the cognitive demands of extreme sports, practitioners frequently have a higher risk tolerance and better fear and anxiety management. These differences demonstrate the complex link between athletics and psychology. However, for sentiment analysis to be used effectively in a healthcare context, it should be part of a broader diagnostic and treatment framework overseen by healthcare professionals. Novel fields of interest as follows:

1. *Mental health monitoring* [13]: changes in sentiment could correlate with mental health issues such as depression, anxiety, or mood swings. For example, a person who typically expresses positive sentiments but shows a

sudden shift to predominantly negative sentiments might be experiencing emotional distress.

2. *Emotional well-being monitoring* [14]: A consistent decline in positive sentiments or an increase in language that indicates stress or anger could be indicative of psychological distress or even social isolation, which is an important factor in overall health.
3. *Trend analysis* [36]: By analyzing sentiment over time, it might be possible to identify trends that are indicative of a deteriorating or improving condition. For instance, the progression of a neurological condition like Alzheimer's disease might be subtly reflected in increasingly negative or confused sentiments in text messages.
4. *Treatment monitoring* [32]: For individuals undergoing treatment for conditions like depression, changes in sentiment over time could potentially indicate how well the treatment is working.
5. *Predictive analysis* [34]: Sentiment analysis could contribute to predictive models that attempt to forecast health events, such as depressive episodes, by identifying warning signs in text communication.

In the realm of psycholinguistics and clinical diagnostics, linguistic and paralingistic elements serve as critical indicators that may reveal underlying cognitive and emotional processes. Linguistically, alterations in syntax, such as simplified sentence structures or reduced complexity in clause embedding, may signal cognitive impairment. Lexical access difficulties are often manifested in increased word-finding pauses, a reduced vocabulary range, and a reliance on nonspecific words like "thing" or "stuff," which can be indicative of neurodegenerative decline. Semantic anomalies, including tangentiality or the use of inappropriate words, and pragmatic failures, like the inability to adhere to conversational norms, also form part of the linguistic tapestry that may suggest pathology.

Paralinguistically, changes in speech prosody, such as a monotonous tone, reduced pitch variation, and altered speech rate, can indicate emotional distress or neurological disorders [60]. Further, the detection of micro-expressions, or subtle facial movements, alongside analysis of gesture frequency and congruence with verbal output, can provide additional context to the emotional state and cognitive functioning of an individual. These deviations from normative patterns, when systematically analyzed, can yield significant insights into the presence and progression of mental health and neurodegenerative disorders, offering a rich substrate for AI-enhanced assessment tools.

Sentiment analysis, through its nuanced parsing of affective language, offers a granular perspective on emotional state, providing a continuous, unobtrusive proxy for mood and affect, which are core components of many psychiatric evaluations [5]. In clinical applications it extends beyond basic positive or negative classifications to encompass the intricate spectrum of human emotions relevant to psychiatric evaluations. It leverages computational linguistics to dissect the subtleties of affective language, offering insights into the intensity and fluctuations of emotional states. For instance, the assessment of written or spoken discourse through sentiment analysis can identify linguistic markers of depression, such as an increased frequency of words associated with negative affect, a heightened use of first-person singular pronouns, or a diminished use of future tense, which may suggest hopelessness or a lack of forward-looking perspective.

Furthermore, the technology can detect patterns of speech indicative of anxiety, characterized by language expressing excessive worry, hyperarousal, and uncertainty. In the context of neurodegenerative disorders, sentiment analysis might reveal a decline in the complexity of emotional expression or a growing incongruence between the expressed sentiment and the discussed topic, often seen in the early stages of such conditions. By quantifying these linguistic features, sentiment analysis offers a granular, continuous, and unobtrusive means of monitoring mood and affect, which are pivotal in the diagnosis and management of mental health disorders. It provides a supplementary dimension to traditional psychiatric assessments, which typically rely on intermittent and subjective self-reported measures, enhancing the longitudinal tracking of mental health states. Let us have some concrete linguistic patterns illustrations.

In *individuals experiencing depression*, there may be a notable increase in the use of words that convey sadness, loneliness, and other negative emotions, like:

1. "I feel **hopeless** and **overwhelmed** by everything."
2. "It's like there's a constant feeling of **gloom** hovering over me."
3. "I'm just so **tired** of feeling **worthless** all the time."

Research suggests that a *high frequency of first-person singular pronouns* can be a linguistic marker of *self-focused attention*, which can be related to depression or anxiety, like:

1. "**I** can't seem to do anything right."
2. "**I** am always the one who messes things up."
3. "**I** feel like **I**'m a burden to everyone."

A *lack of forward-looking statements* can indicate a *pessimistic outlook or a sense of hopelessness*, which is often found in depressive speech patterns, for instance:

1. "There's no point in trying to plan for anything."
2. "Why bother with what's going to happen tomorrow?"
3. "It's not like things will get better for me."

Each of these linguistic cues, when observed in natural language communication, can provide mental health professionals with additional context to understand a patient's emotional state.

Despite its potential, there are several important considerations and challenges: (1) Sentiment analysis algorithms must be sophisticated enough to understand context, sarcasm, and nuanced language use, which can vary widely between individuals and cultures; (2) analyzing personal text messages raises significant privacy concerns. It is crucial to have explicit consent from individuals to the analysis of their data with robust data protection measures; (3) the accuracy of sentiment analysis can vary, and false positives or negatives could have serious implications. Validation with clinical data is necessary; (4) there is an ethical dimension to consider regarding the surveillance of individuals' communications, which needs careful ethical oversight and regulation.

4 Materials and methods

The research used datasets comprising various forms of textual data [80, 82], including anonymized social media interactions, transcripts from patient interviews, and electronic health records [10]. NLP algorithms were deployed to parse the text, and machine learning models were trained to identify language patterns and sentiment changes. The study also incorporated a sentiment analysis to gauge emotional expression, a key component of mental health diagnostics.

5 Methods

5.1 Data processing

For data annotation we used human experts, as well as already validated datasets [80, 82]. A collaborative, AI-supported solution for wider dataset annotation is planned to be used in the next phases.

5.2 Environment

The experiments were conducted on Google Colab Pro Environment, by using python-based notebooks with PyTorch, Pandas, NumPy, MatPlotlib, seaborn, Transformers, SciPy, Scikit Learn, Natural Language Toolkit (NLTK), TextBlob, LightGBM.

5.3 Mental health disorder identification

Translating an algorithm for detecting mental health disorders from text into mathematical language involves defining a series of functions and representations for the processes of feature extraction, transformation, and classification. The mathematical formulation of the problem is as follows. We first introduce the variables and functions: (1) let D be a set of documents $\{\mathsf{d}_1, \mathsf{d}_2, ..., \mathsf{d}_n\}$, where each document d_i represents a piece of text; (2) let L be a set of labels $\{\mathsf{l}_1, \mathsf{l}_2, ..., \mathsf{l}_m\}$, where each label corresponds to a mental health disorder (e.g., depression, psychosis, Alzheimer's); (3) Preprocess(d) is a function that takes document d and returns a preprocessed document; (4) Tokenize(d) is a function that takes a preprocessed document d and returns a set of tokens T; (5) Vectorize(T) is a function that converts tokens T to a feature vector $\mathbf{x}$; (6) Classify($\mathbf{x}$) is a function that assigns a label l to a feature vector $\mathbf{x}$. For term frequency-inverse document frequency (TF-IDF), we define tfidf($\mathsf{t}, \mathsf{d}, \mathsf{D}$) which computes the TF-IDF score for a term t in document d within the set of documents D.

$$\mathrm{tf}(\mathsf{t}, \mathsf{d}) = \frac{\mathsf{f}_{\mathsf{t},\mathsf{d}}}{\sum_{\mathsf{t}' \in \mathsf{d}} \mathsf{f}_{\mathsf{t}',\mathsf{d}}} \tag{1}$$

$$\mathrm{idf}(\mathsf{t}, \mathsf{D}) = \log \frac{|\mathsf{D}|}{|\{\mathsf{d} \in \mathsf{D} | \mathsf{t} \in \mathsf{d}\}|} \tag{2}$$

$$\mathrm{tfidf}(\mathsf{t}, \mathsf{d}, \mathsf{D}) = \mathrm{tf}(\mathsf{t}, \mathsf{d}) \times \mathrm{idf}(\mathsf{t}, \mathsf{D}) \tag{3}$$

Sentiment analysis might use a predefined sentiment lexicon S where each word w has an associated sentiment score $\mathsf{s}(w)$ [67]. The sentiment score of a document d is a sum of sentiment scores of its words:

$$\mathrm{SentimentScore}(\mathsf{d}) = \sum_{w \in \mathsf{d}} \mathsf{s}(w) \tag{4}$$

Complexity measures involves computing the diversity of words Diversity(d) or the readability Readability(d) of the document. A classifier can be repre-

sented by a function C that maps feature vectors to labels. If using a support vector machine (SVM), for instance, the decision function for a binary classification looks like:

$$C(\mathbf{x}) = \text{sgn}(\mathbf{w}^T\mathbf{x} + b) \tag{5}$$

where $\mathbf{w}$ is the weight vector, b is the bias, and sgn is the sign function.

For a probabilistic output, such as from a logistic regression or neural network with a softmax layer, the classifier could output a probability vector over labels:

$$C(\mathbf{x}) = \text{softmax}(\mathbf{W}\mathbf{x} + \mathbf{b}) \tag{6}$$

where $\mathbf{W}$ is a weight matrix and $\mathbf{b}$ is a bias vector. The mathematical representation of the algorithm is as follows:

1. Preprocessing:
$$\forall d_i \in D, \tilde{d_i} = \text{Preprocess}(d_i) \tag{7}$$
2. Tokenization and Vectorization:
$$\forall \tilde{d_i}, T_i = \text{Tokenize}(\tilde{d_i}) \tag{8}$$
$$\forall T_i, \mathbf{x_i} = \text{Vectorize}(T_i) \tag{9}$$
Now, let us include the sentiment and complexity features into $\mathbf{x_i}$.
3. Classification:
$$\forall \mathbf{x_i}, l_i = C(\mathbf{x_i}) \tag{10}$$
4. Model Training (if not using a predefined model): find $\mathbf{W}$ and $\mathbf{b}$ that minimize a loss function
$$\mathcal{L}(C(\mathbf{x_i}), y_i) \tag{11}$$
over the training data, where y_i is the true label.
5. Inference: for a new document d, we will calculate
$$l = C(\text{Vectorize}(\text{Tokenize}(\text{Preprocess}(d)))) \tag{12}$$

5.3.1 Examples of mental health disorder identification

Each mental health disorder has unique linguistic patterns that the algorithm has to learn from training data. The specific model and its parameters must be tailored based on empirical evidence from this data. This abstract representation simplifies many of the complexities involved in NLP and ML for better understandig of the approach. Implementing this algorithm requires careful consideration of the representativeness and bias in the training data, the interpretability of the model, and ethical considerations, especially given the sensitive nature of mental health. Identifying mental health disorders from text and its sentiment analysis involves the computational interpretation of language use, which may suggest underlying psychological conditions. Here are some concrete examples illustrating how sentiment analysis and text examination can point toward mental health concerns:

1. Example 1: Potential Depression Detection [22]
   (a) *Text*: "I just don't want to get out of bed anymore. Nothing really makes me happy, and I can't see the point in trying. It's all just too much."
   (b) *Analysis*: High frequency of negative affect words ("don't want," "nothing," "can't," "pointless," "too much"), low sentiment score, and minimal positive language use.
   (c) *Indicators*: Anhedonia (lack of pleasure), feelings of helplessness, and low mood, which are symptomatic of depression.
2. Example 2: Potential Anxiety Detection [6]
   (a) *Text*: "Every time I have to leave the house, I get this overwhelming dread. What if something terrible happens? I'm always so worried."
   (b) *Analysis*: The Anxiety-related words ("overwhelming," "dread," "terrible," "worried"), high use of words that express uncertainty and fear.
   (c) *Indicators*: Excessive worry about future events, physical sensation of dread, consistent with anxiety disorders.
3. Example 3: Potential Bipolar Disorder Detection [47] (During Depressive Phase)
   (a) *Text*: "I've lost interest in seeing my friends or doing any of my hobbies. I feel empty and sad most days now. My life seems like a series of disappointments."

(b) *Analysis*: Negative sentiment prevalence, decreased mention of engaging in activities, expression of emptiness and sadness.

(c) *Indicators*: Social withdrawal and persistent sadness could indicate a depressive episode in the context of bipolar disorder.

4. Example 4: Potential Bipolar Disorder Detection [47] (During Manic Phase)

   (a) *Text*: "I've started a million projects this week, and I feel on top of the world! Sleep is for the weak, I can get by on an hour a night, no problem!"

   (b) *Analysis*: Extremely positive sentiment, grandiosity, possible engagement in high-risk activities, reduced need for sleep.

   (c) *Indicators*: The manic phase may be characterized by an inflated self-esteem, decreased need for sleep, and racing thoughts.

5. Example 5: Potential Schizophrenia Detection [37]

   (a) *Text*: "Voices are telling me not to trust anyone. I know they are plotting against me because I can hear them whispering when I'm alone."

   (b) *Analysis*: Mention of hallucinations ("voices"), paranoia ("plotting against me"), and delusional thinking.

   (c) *Indicators*: Auditory hallucinations and paranoid delusions are common symptoms of schizophrenia.

6. Example 6: Potential Post-Traumatic Stress Disorder (PTSD) [53] Detection

   (a) *Text*: "I can't stop thinking about the accident. It replays in my head every time I close my eyes. I'm always on edge."

   (b) *Analysis*: Persistent recounting of a traumatic event, high incidence of stress-related words, ongoing sense of tension.

   (c) *Indicators*: Intrusive memories of the event, hypervigilance, and strong stress response suggest PTSD.

In practice, sentiment analysis can be conducted using ML models trained on annotated datasets, where texts are labeled with associated mental health conditions. The models can learn to detect patterns that frequently correspond to specific disorders. However, it is important to validate AI findings with clinical assessments, as sentiment analysis tools can complement but *NOT REPLACE* professional diagnosis.

5.4 Neurodegenerative diseases detection

Detecting neurodegenerative diseases like Alzheimer's [46] from text could involve similar strategies to those used for mental health disorders, but the focus might shift towards detecting cognitive impairment, which can manifest in language in different ways. The mathematical framework should be adapted to capture these nuances. First we define the variables and functions for neurodegenrative diseases: (1) let C denote cognitive features extracted from text, such as coherence, vocabulary richness, or syntactic complexity; (2) define CoherenceScore(d) as a function that measures the logical flow and clarity of ideas within the text; (3) define SyntacticComplexity(d) as a function that measures the complexity of sentence structures; (4) define VocabularyRichness(d) as a function that measures the diversity of vocabulary used in the document. In feature engineering the coherence could be measured through the consistency of topics or entities mentioned in a text. Let $\mathcal{T}$ be a topic model, then coherence can be quantified as:

$$\text{CoherenceScore}(d) = \sum_{i=1}^{n-1} \text{similarity}(\mathcal{T}(s_i), \mathcal{T}(s_{i+1})) \tag{13}$$

where $\mathcal{T}(s)$ is the topic distribution of sentence s and similarity is a function measuring the similarity between topic distributions (e.g., cosine similarity). *Syntactic complexity* might involve parsing trees and measuring their depth or branch factor:

$$\text{SyntacticComplexity}(d) = \frac{1}{|S|} \sum_{s \in S} \text{TreeDepth}(\text{ParseTree}(s)) \tag{14}$$

where S is the set of sentences in d, and ParseTree(s) is the syntactic parse tree of sentence s. The *Vocabulary richness* might use metrics like the type-token ratio (TTR) or unique words:

$$\text{VocabularyRichness}(d) = \frac{|\text{unique words in } d|}{|\text{tokens in } d|} \tag{15}$$

The Classification Model, which will be a ML model in this case, potentially a NN with architecture suited to capture temporal and cognitive features, could be represented as:

$$C(\mathbf{x}) = \text{softmax}(\mathbf{W}_2 \cdot \text{ReLU}(\mathbf{W}_1 \mathbf{x} + \mathbf{b}_1) + \mathbf{b}_2) \tag{16}$$

where $\mathbf{x}$ includes traditional NLP features along with the neurodegenerative-specific cognitive features, $\mathbf{W}_1, \mathbf{W}_2$ are weight matrices, $\mathbf{b}_1, \mathbf{b}_2$ are bias vectors, and ReLU is the Rectified Linear Unit activation function.

The mathematical representation will be as follows:

1. Preprocessing and Feature Engineering:

$$\forall d_i \in D, \tilde{d}_i = \text{Preprocess}(d_i) \tag{17}$$

$$\forall \tilde{d}_i, T_i = \text{Tokenize}(\tilde{d}_i), C_i = \text{CognitiveFeatures}(\tilde{d}_i) \tag{18}$$

$$\forall T_i, \mathbf{x}_i^{\text{text}} = \text{Vectorize}(T_i) \tag{19}$$

$$\mathbf{x}_i = [\mathbf{x}_i^{\text{text}}, C_i] \tag{20}$$

   where C_i includes coherence, complexity, and vocabulary richness.

2. Model Training: Now can optimize $\mathbf{W}_1, \mathbf{W}_2, \mathbf{b}_1, \mathbf{b}_2$ by minimizing the loss function $\mathcal{L}(C(\mathbf{x}_i), y_i)$.

3. Inference: for a new document d,

$$l = C(\text{Vectorize}(\text{Tokenize}(\text{Preprocess}(d))), \text{CognitiveFeatures}(d)) \tag{21}$$

Detecting neurodegenerative diseases from text requires rigorous validation and should be supplemented with clinical assessments. The model presented in this paper contains a concept with a minimal dataset but needs to be expanded with a large, validated dataset. The cognitive features are required to be specifically tailored to the types of language deficits or changes associated with the particular neurodegenerative disease in question.

5.4.1 Examples of neurodegenerative diseases or disorder identification

Identifying neurodegenerative diseases from text and sentiment analysis involves detecting changes in language that may be symptomatic of cognitive decline. Here are concrete examples of how text analysis might reveal signs of neurodegenerative conditions:

1. Example 1: Potential Alzheimer's disease
   (a) *Text*: "I went to the... umm... the place where you buy food. I forgot what it's called. And I couldn't remember why I was there."
   (b) *Analysis*: Hesitations and word-finding difficulties, use of nonspecific language ("place where you buy food"), and memory lapses.
   (c) *Indicators*: These linguistic patterns can indicate episodic memory impairment and semantic memory issues, common in early-stage Alzheimer's disease.
2. Example 2: Potential Parkinson's disease [59]
   (a) *Text*: "My hands have been shaking a lot lately, making it hard to type or write. I feel stiff and slow."
   (b) *Analysis*: Mention of physical symptoms affecting fine motor skills, change in activity due to physical limitations.
   (c) *Indicators*: Motor symptoms affecting writing could indirectly be observed through changes in typing patterns, such as increased time to type messages or more typographical errors.
3. Example 3: Potential frontotemporal dementia [63]
   (a) *Text*: "My family says I've been acting inappropriately, but I think they're overreacting. I don't see anything wrong with what I'm doing."
   (b) *Analysis*: Possible lack of insight into socially inappropriate behaviors, which family members notice.
   (c) *Indicators*: Changes in social conduct, personality, and a decline in judgment, which are often seen in frontotemporal dementia.
4. Example 4: Potential vascular dementia [65]
   (a) *Text*: "I've been feeling confused lately, especially when trying to handle my bills or planning things. It wasn't like this before."
   (b) *Analysis*: Expression of confusion in complex tasks, recognition of change from previous abilities.
   (c) *Indicators*: Impairments in executive function may suggest vascular dementia, particularly if there is a history of strokes or cardiovascular disease.

5. Example 5: Potential Lewy body dementia [26]
   (a) *Text*: "I've been seeing things that aren't there, particularly at night. It's very unsettling."
   (b) *Analysis*: Reference to visual hallucinations, a sense of distress related to these experiences.
   (c) *Indicators*: Visual hallucinations are a hallmark symptom of Lewy Body Dementia, especially when coupled with sleep disturbances.
6. Example 6: Potential amyotrophic lateral sclerosis (ALS) [38]
   (a) *Text*: "Speaking has become exhausting. People can't understand me well anymore."
   (b) *Analysis*: Indication of speech difficulties, increased effort required for communication.
   (c) *Indicators*: ALS can affect speech muscles, leading to dysarthria, which could be reflected in brief and effortful communication.

In sentiment analysis, while emotional content might not directly indicate a neurodegenerative disease, drastic changes in sentiment over time could reflect the emotional impact of living with such diseases, like frustration or sadness due to loss of autonomy. Moreover, sentiment analysis might detect less obvious changes in emotional expression or response that could be associated with cognitive changes. It is vital to emphasize that these text-based observations are not conclusive for diagnosis but may prompt further clinical evaluation. Neurodegenerative diseases are complex and require comprehensive medical assessment, including neurological examination, cognitive testing, and imaging, for accurate diagnosis. Text and sentiment analysis can serve as supplementary tools that might flag potential issues for further investigation.

5.5 Detecting chronic diseases

Detecting chronic diseases such as diabetes or thyroid disorders through text analysis can be quite challenging because the symptoms and signs of these diseases are typically not as directly reflected in language as thos of some mental or neurodegenerative disorders. However, if we consider that individuals might express concerns, experiences, or symptoms related to their physical health in text messages, sentiment analysis and certain linguistic cues might provide indirect indicators. For this, the mathematical framework must incorporate

sentiment analysis and pay attention to specific lexicon related to physical symptoms, healthcare management, and possibly, lifestyle aspects that can be linked to chronic conditions. Not the disease itself would be detected but rather cues that might warrant further medical investigation. Let us introduce the variables and functions first: (1) define SymptomLexicon($\mathtt{d}$) as a function that identifies the presence of words or phrases associated with symptoms or management of chronic diseases; (2) let H represent healthcare management features, such as mentions of medication, doctor visits, or treatment-related activities; (3) let L represent lifestyle features that may include dietary habits, physical activity levels, or other behaviors relevant to chronic diseases; (4) SentimentAnalysis($\mathtt{d}$) remains a function that assigns a sentiment score to document $\mathtt{d}$, but may also flag emotionally charged language that could be indicative of stress or frustration related to disease management. On the *feature engineering* side for chronic diseases the *symptom mention* can be a binary or frequency-based feature indicating the presence of terms from a curated symptom lexicon:

$$\text{SymptomFeature}(\mathtt{d}) = \sum_{\mathtt{t} \in \text{SymptomLexicon}} \text{Ind}(\mathtt{t} \in \mathtt{d}) \tag{22}$$

where Ind is an indicator function returning 1 if the term $\mathtt{t}$ is present in document $\mathtt{d}$ and 0 otherwise. The *healthcare management features* can be quantified similarly by counting mentions of healthcare-related activities:

$$\mathsf{H}(\mathtt{d}) = \sum_{\mathtt{t} \in \text{HealthcareLexicon}} \text{Ind}(\mathtt{t} \in \mathtt{d}) \tag{23}$$

The *lifestyle features* can also be derived from mentions of activities or habits:

$$\mathsf{L}(\mathtt{d}) = \sum_{\mathtt{t} \in \text{LifestyleLexicon}} \text{Ind}(\mathtt{t} \in \mathtt{d}) \tag{24}$$

The *sentiment analysis* can be adapted to give higher weight to sentiments expressed in the context of health:

$$\text{HealthSentimentScore}(\mathtt{d}) = \sum_{w \in \mathtt{d}} s(w) \cdot \text{Ind}(w \in \text{HealthContext}) \tag{25}$$

where HealthContext is a set of contexts where the sentiment might be particularly relevant to chronic disease. As classifier we used the well-known models,

such as random forest or gradient boosting machine, which can handle sparse features effectively:

$$C(\mathbf{x}) = \text{softmax}(\mathbf{W}\mathbf{x} + \mathbf{b}) \tag{26}$$

where $\mathbf{x}$ includes traditional NLP features, sentiment scores, symptom features, healthcare management features, and lifestyle features. The complete mathematical model for chronic diseases can be formulated as follows: (1) *Feature Engineering*: for each document $d_i \in D$: (1) preprocess and extract text features to form $\mathbf{x}_i^{\text{text}}$, (2) extract symptom, healthcare management, and lifestyle features to form $\mathbf{x}_i^{\text{chronic}}$(3) perform sentiment analysis tailored to the health context and then (4) Combine all features:

$$\mathbf{x}_i = [\mathbf{x}_i^{\text{text}}, \mathbf{x}_i^{\text{chronic}}, \text{HealthSentimentScore}(d_i)] \tag{27}$$

(2) *Model Training*: train classifier C with features $\mathbf{x}_i$ to predict labels y_i related to the likelihood of chronic disease-related discourse; (3) *Inference*: for a new document d, calculate the likelihood of chronic disease-related discourse: $l = C(\mathbf{x})$.

5.5.1 Examples of chronic diseases identification

Identifying chronic diseases from text and sentiment analysis typically involves looking for patterns that suggest a person is experiencing symptoms or challenges associated with their condition. It is important to note that while sentiment analysis can reveal emotions and concerns related to health, it is not a diagnostic tool for chronic physical diseases. Similarly to the other categories, it can signal when further medical evaluation may be warranted. Here are some examples:

1. Example 1: Potential diabetes management [18]
   (a) *Text*: "I'm feeling really drained lately, no matter how much I rest. My feet have been tingling, too. I'm worried because my mom has diabetes, and these were her first signs."
   (b) *Analysis*: Negative sentiment expressed through words like "drained" and "worried," along with the mention of symptoms associated with diabetes (fatigue, neuropathy).
   (c) *Indicators*: Textual cues suggest the individual may be experiencing symptoms commonly associated with diabetes, warranting further medical investigation.

2. Example 2: Potential thyroid disorders [62]
   (a) *Text*: "I just can't seem to lose weight, and I'm always cold. My hair is falling out, and I'm feeling down most days. It's so frustrating!"
   (b) *Analysis*: Expressions of frustration and physical symptoms that may be consistent with hypothyroidism, such as unexplained weight gain, cold intolerance, and hair loss.
   (c) *Indicators*: The combination of sentiment and symptom-related language can point toward possible thyroid dysfunction.
3. Example 3: Potential chronic obstructive pulmonary disease (COPD) [52]
   (a) *Text*: "I get out of breath just walking to the mailbox. It's scary and makes me anxious about leaving the house."
   (b) *Analysis*: The expression of anxiety linked to difficulty breathing, a key symptom of COPD.
   (c) *Indicators*: Descriptions of breathlessness during low-exertion activities could suggest a respiratory issue like COPD.
4. Example 4: Potential rheumatoid arthritis [50]
   (a) *Text*: "My joints have been so stiff and sore lately, especially in the morning. It's been making me quite miserable."
   (b) *Analysis*: Words indicating physical discomfort and a negative emotional state, coupled with specific timing (morning stiffness) which is characteristic of rheumatoid arthritis.
   (c) *Indicators*: Joint symptoms and their impact on mood may be indicative of a chronic inflammatory condition.
5. Example 5: Potential chronic pain conditions [33]
   (a) *Text*: "I'm in constant pain, and nothing seems to help. It's hard to concentrate on work or even enjoy time with my family."
   (b) *Analysis*: Persistent negative sentiment and references to ongoing pain impacting daily life.
   (c) *Indicators*: Chronic pain conditions can lead to the observed textual expressions of suffering and its effects on quality of life.
6. Example 6: Potential heart disease [9]

(a) *Text*: "I've had more chest pain and discomfort this week. Feeling a bit nervous about it."

(b) *Analysis*: Concern and physical symptoms suggestive of cardiac issues, with an emotional response indicating awareness of potential severity.

(c) *Indicators*: Symptoms like chest pain, when mentioned in text, can be a red flag for cardiovascular conditions and should be followed up clinically.

In these examples, sentiment analysis might help to quantify the emotional burden of the symptoms or the disease management process itself. The negative sentiments expressed in conjunction with mentions of specific symptoms can lead to a holistic understanding of the patient's experience. While AI cannot diagnose chronic diseases from text alone, it can provide valuable insights into a person's subjective health experience, which is useful for healthcare providers to know when to probe further or monitor symptoms more closely.

No.	**Disease**	**Text length**
1	Alzheimer's Disease and Dementia	39.435497
2	Depression and Anxiety	91.884543
3	None (No symptoms)	85.082874
4	Psychosis	41.777778
5	Suicide and Depression	1068.948119

Table 1: Text length distribution by label groups

No.	**Disease**	**Text labels**
1	None (No symptoms)	7976
2	Suicide and Depression	2525
3	Alzheimer's Disease and Dementia	2496
4	Psychosis	2448
5	Depression and Anxiety	1585

Table 2: Text label distribution

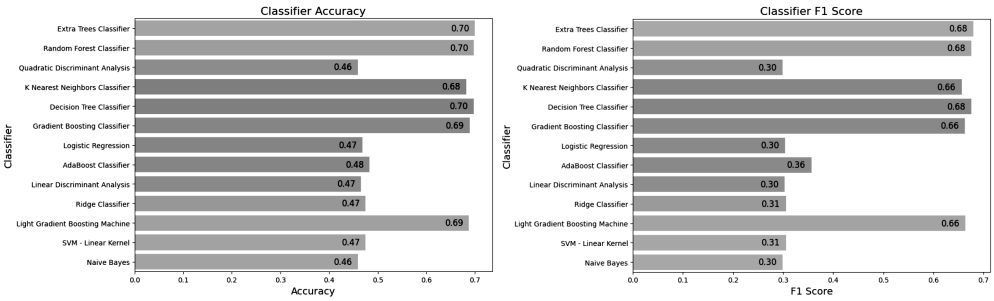


Figure 2: Classifier Accuracy (left), Classifier F1 Score(right)

6 Experiments

We used the most well-known classifiers in our experiments, as shown in Table 6, resulting the following overall initial results. The most promizing were the: (1) Random Forest, (2) Extra Trees, (3) Decision Tree, (4) Light Gradient Boosting Machine [11].

Classifier	**Accuracy**	**Precision**	**Recall**	**F1 Score**
Extra Trees Classifier	**0.699941**	**0.697118**	**0.699941**	**0.679721**
Random Forest Classifier	**0.697592**	**0.693373**	**0.697592**	**0.676143**
Quadratic Discriminant Analysis	0.459190	0.221843	0.459190	0.299158
K Nearest Neighbors Classifier	0.682032	0.667897	0.682032	0.656546
Decision Tree Classifier	**0.697005**	**0.694699**	**0.697005**	**0.676126**
Gradient Boosting Classifier	0.688784	0.681584	0.688784	0.663057
Logistic Regression	0.468878	0.261091	0.468878	0.304289
AdaBoost Classifier	0.483265	0.294791	0.483265	0.356398
Linear Discriminant Analysis	0.465942	0.260414	0.465942	0.303054
Ridge Classifier	0.474750	0.225388	0.474750	0.305663
Light Gradient Boosting Machine	0.687317	0.677241	0.687317	0.663735
SVM - Linear Kernel	0.474750	0.225388	0.474750	0.305663
Naive Bayes	0.459190	0.221843	0.459190	0.299158

Table 3: Experiments results with classifier comparisons

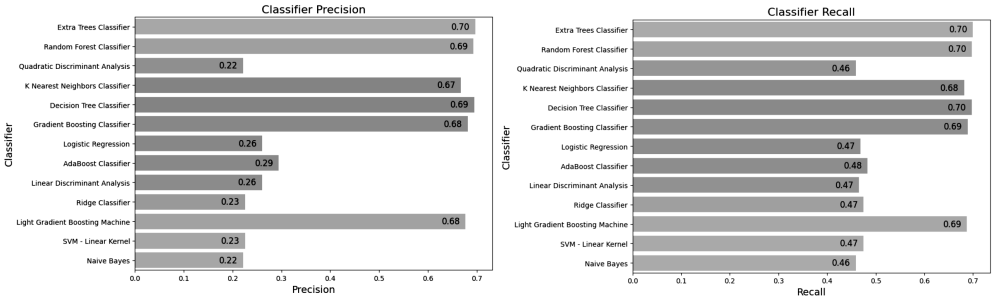


Figure 3: Classifier Precision (left), Classifier Recall (right)

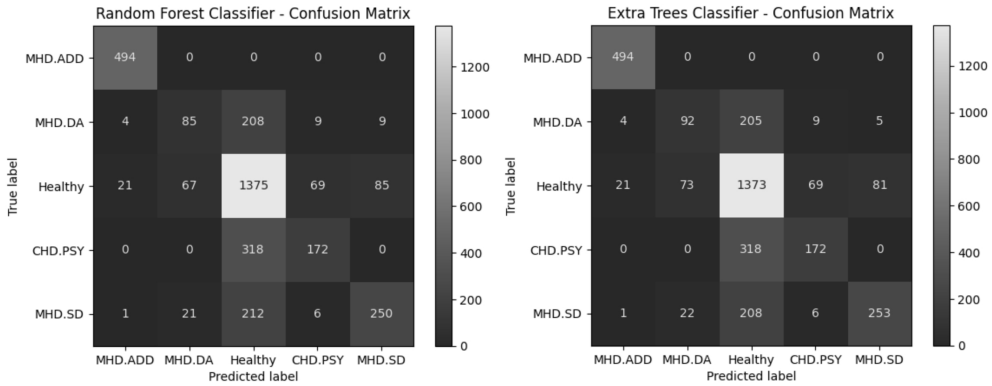


Figure 4: Confusion Matrix - Random Forest (left), Extra Trees (right)

7 Results

The utilization of AI for parsing text (using label distribution as is presented in Table 5.5.1) and sentiment analysis has yielded promising results in the realm of precognition of mental health and neurodegenerative disorders. Studies leveraging these technologies have demonstrated AI's capability to identify linguistic patterns (see Table 5.5.1 for text length distribution) that correlate with symptomatic manifestations of various conditions. The research has shown that individuals with depression tend to exhibit a higher frequency of negative affect words and self-referential pronouns in their communication, which AI algorithms can detect with notable accuracy (see Figure 7 and Figure 8).

In the domain of neurodegenerative diseases, preliminary findings suggest that changes in language complexity, such as simplified syntax and dimin-

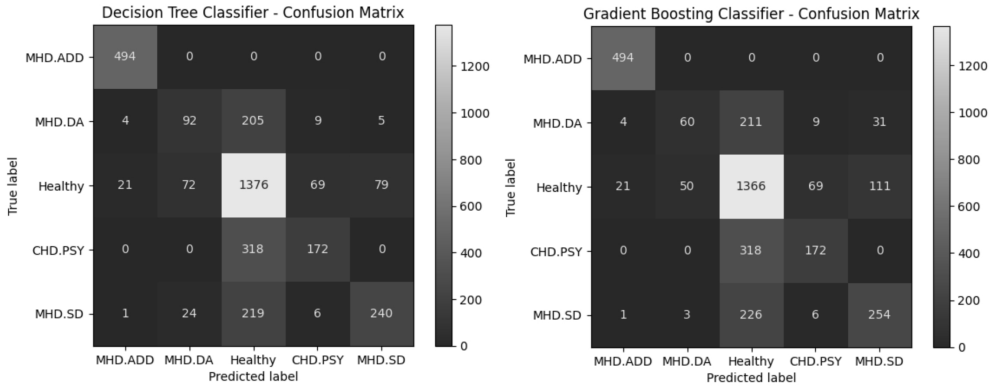


Figure 5: Confusion Matrix - Decision Tree (left), Gradient Boosting (right)

ished vocabulary diversity (Table 7 and (see Table 7), may be quantifiable through AI analysis before these symptoms are clinically evident. This can be particularly observed in conditions like Alzheimer's disease, where progressive cognitive decline has a direct impact on language function. Furthermore, sentiment analysis monitors fluctuations in mood and affect, which are integral to mental health assessment, providing a supplemental, non-invasive tool for tracking patient well-being.

The paper facilitates the monitoring of individuals' communication over time, enabling the identification of trends that may indicate the onset or progression of a disorder. This approach offers a continuous, objective assessment that can complement intermittent clinical evaluations. Moreover, it holds the potential for remote monitoring, which is invaluable for patient populations that may have limited access to regular healthcare services (see Figure 9).

However, these advancements come with the caveat that such technologies are adjuncts and not replacements for traditional diagnostic methods. AI-based predictions require validation through clinical expertise and should be viewed within the broader context of comprehensive medical assessment. The ethical implications of using personal communication data for health monitoring are also under scrutiny, necessitating transparent data handling and stringent privacy safeguards.

The presented methods are emerging as significant contributors to the early detection and monitoring of mental health and neurodegenerative disorders, offering a novel lens through which to understand and manage these complex conditions. The continued refinement of these tools, alongside advances in

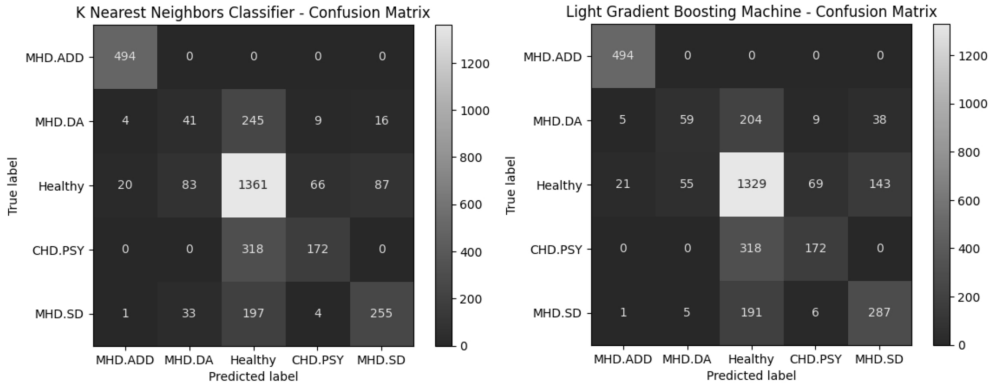


Figure 6: Confusion Matrix - K Nearest Neighbors (left), Light Gradient Boosting Machine (right)

No.	ID	Feature	Importance
1	3254	plotting	**0.051509**
2	5000	sentiment	0.045933
3	4747	voices	0.044404
4	3164	people	0.037201
5	1994	hear	0.034005
6	2610	lost	0.032294
7	3520	real	0.028020
8	3231	places	0.027559
9	3606	remember	0.025091
10	1039	dates	0.024884

Table 4: Top 10 features with Importance levels

machine learning and NLP (see Table 7), promises to enhance the capabilities of medical professionals and improve outcomes for patients.

8 Discussion

The incorporation of AI-based text and sentiment analysis into the diagnostic milieu of mental health and neurodegenerative disorders heralds a novel frontier in early detection and monitoring. This computational approach can transcend conventional constraints by analyzing linguistic and paralinguistic elements in patients' speech or written text, potentially unveiling subtle devi-

Rank	Alzheimer's & Dementia		Depression & Anxiety		None		Psychosis		Suicide & Depression	
No.	**ID**	**IMP**	**ID**	**IMP**	**ID**	**IMP**	**ID**	**IMP**	**ID**	**IMP**
1	familiar	**5.775062**	anxious	**6.343930**	morning	**1.450505**	hear	**8.617279**	suicide	**5.172760**
2	places	5.702947	restless	5.998787	coffee	1.445807	plotting	8.378585	help	4.807741
3	remember	5.150686	worried	5.552964	buy	1.287901	arena	5.983609	kill	4.515317
4	lost	5.135937	nervous	4.982023	cool	1.249352	people	4.887970	life	4.319223
5	hard	4.962242	worry	4.118800	holiday	1.237574	voices	4.455395	just	4.166109
6	dates	4.882358	sleep	2.832804	tweet	1.230690	things	3.417450	wish	4.115592

ID = feature name, IMP - feature importance

Table 5: Top six features with Importance levels per labels

Disease	**Precision**	**Recall**	**F1-Score**	**Support**
Alzheimer's Disease and Dementia	**0.95**	**1.00**	**0.97**	494
Depression and Anxiety	0.49	0.29	0.37	315
None	0.65	0.85	0.74	1617
Psychosis	0.67	0.35	0.46	490
Suicide and Depression	0.75	0.52	0.61	490
accuracy			0.70	3406
macro avg	0.70	0.60	0.63	3406
weighted avg	0.70	0.70	0.68	3406

Table 6: Extra Trees Classifier Classification Report

Disease	**Precision**	**Recall**	**F1-Score**	**Support**
Alzheimer's Disease and Dementia	**0.95**	**1.00**	**0.97**	494
Depression and Anxiety	0.49	0.27	0.35	315
None	0.65	0.85	0.74	1617
Psychosis	0.67	0.35	0.46	490
Suicide and Depression	0.73	0.51	0.60	490
accuracy			0.70	3406
macro avg	0.70	0.60	0.62	3406
weighted avg	0.69	0.70	0.68	3406

Table 7: Random Forest Classifier Classification Report

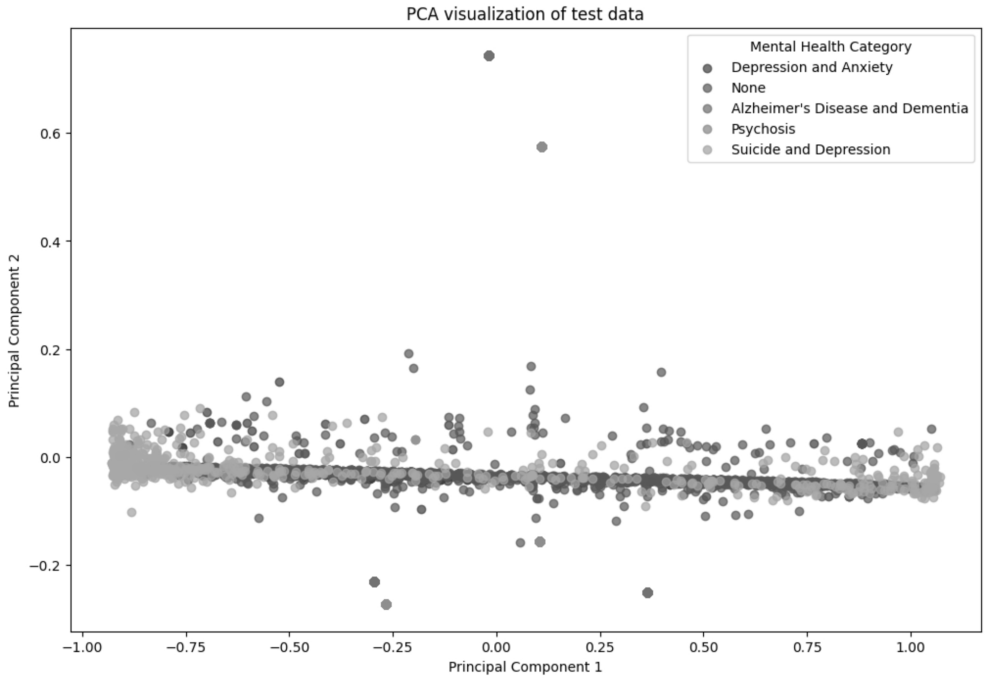


Figure 7: PCA visualization of test data

ations from normative communication patterns indicative of cognitive decline or emotional distress.

The ***Extra Trees Classifier*** (see Table 7) reveals insights into the model's performance across various classes, which in this case are different mental health conditions. The report includes precision, recall, f1-score, and support for each class, as well as overall accuracy and averages. (1) *Alzheimer's Disease and Dementia*: the model demonstrates high precision (0.95) and perfect recall (1.00) in identifying Alzheimer's Disease and Dementia, leading to an excellent f1-score of 0.97. This suggests that the classifier is highly effective in identifying this condition, with a low rate of false negatives and false positives; (2) *Depression and Anxiety*: the performance significantly drops in this category, with a precision of 0.49 and a recall of 0.29, resulting in a f1-score of 0.37. This indicates challenges in accurately classifying cases of Depression and Anxiety, with a higher likelihood of both false positives and false negatives; (3) *None (Healthy)*: the model performs reasonably well in identifying healthy subjects, with a precision of 0.65 and a higher recall of 0.85, culminating in

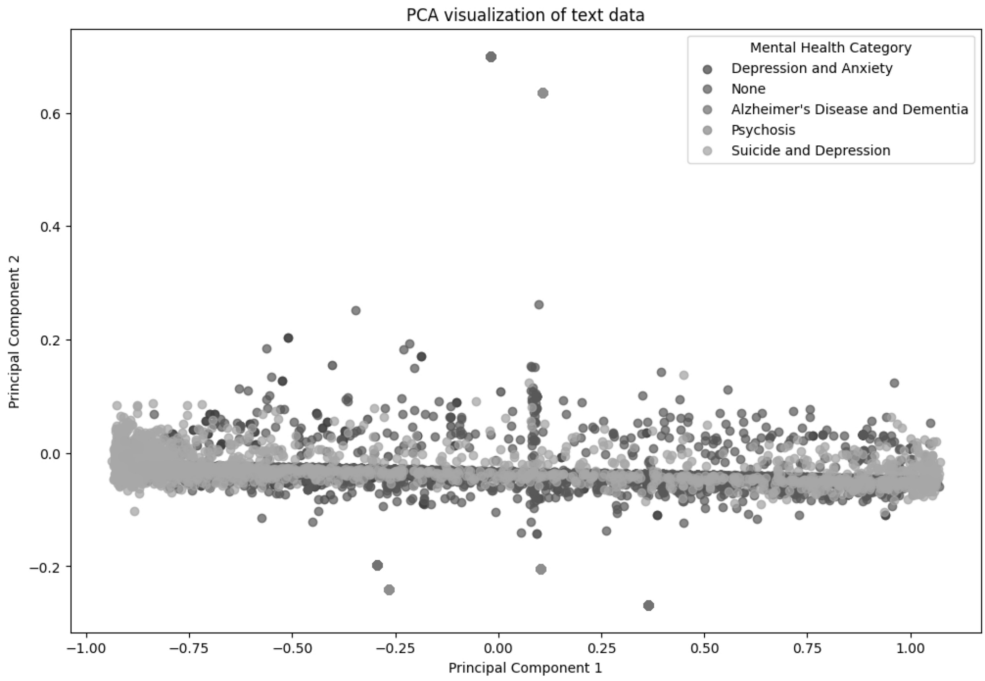


Figure 8: PCA visualization of text data

a f1-score of 0.74. This suggests that while the model can reliably identify healthy individuals, there is still room for improvement in reducing false positives; (4) *Psychosis*: for Psychosis, the model shows a moderate precision of 0.67 but a lower recall of 0.35, leading to a f1-score of 0.46. This implies that while the classifier is relatively reliable when it identifies a case as Psychosis (fewer false positives), it misses a significant number of true Psychosis cases (higher false negatives); (5) *Suicide and Depression*: the model shows fairly good precision (0.75) but moderate recall (0.52) in this category, with a resulting f1-score of 0.61. This suggests a better balance between false positives and false negatives, although there is still a notable number of missed cases; (6) *Overall Performance*: the overall accuracy of the model is 0.70, which is satisfactory but indicates potential for improvement. The macro average f1-score (0.63) and weighted average f1-score (0.68) reflect moderate performance across classes, with better accuracy in some (like Alzheimer's Disease and Dementia) and challenges in others (like Depression and Anxiety); In conclusion, the *Extra Trees Classifier* shows varying levels of effectiveness in identifying

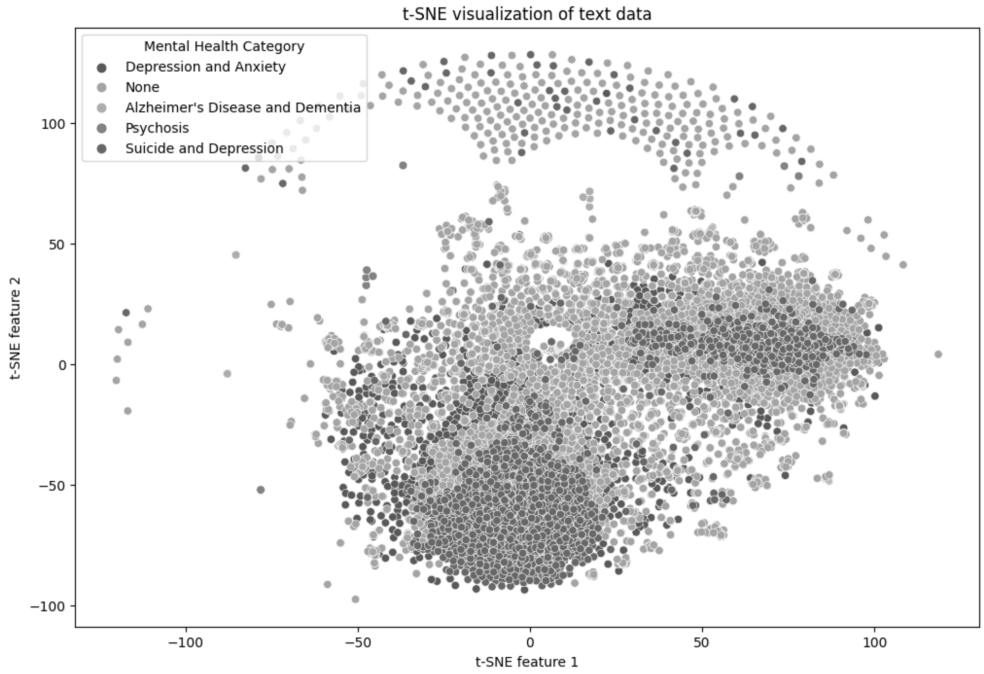


Figure 9: t-SNE visualization of text data

different mental health conditions. Its strength lies in identifying *Alzheimer's Disease* and *Dementia* and the general healthy population, while it faces challenges in accurately classifying Depression and Anxiety, Psychosis, and Suicide and Depression. This variance suggests the need for further model tuning or exploration of alternative features that could improve classification accuracy, especially for the underperforming categories. Additionally, the imbalanced performance across classes indicates the potential benefit of employing techniques to handle class imbalances effectively.

The ***Random Forest Classifier*** (see Table 7) provides a detailed view of the model's performance across different categories related to mental health disorders. This analysis focuses on precision, recall, f1-score for each category, and overall accuracy. (1) *Alzheimer's Disease and Dementia*: the Random Forest Classifier demonstrates excellent performance in identifying Alzheimer's Disease and Dementia, evidenced by high precision (0.95) and perfect recall (1.00), culminating in an f1-score of 0.97. This indicates a strong ability of the model to correctly classify cases of Alzheimer's Disease and Dementia with minimal false positives and negatives; (2) *Depression and Anxiety*: the

Disease	**Precision**	**Recall**	**F1-Score**	**Support**
Alzheimer's Disease and Dementia	**0.95**	**1.00**	**0.97**	494
Depression and Anxiety	0.49	0.29	0.37	315
None	0.65	0.85	0.74	1617
Psychosis	0.67	0.35	0.46	490
Suicide and Depression	0.74	0.49	0.59	490
accuracy			0.70	3406
macro avg	0.70	0.60	0.63	3406
weighted avg	0.69	0.70	0.68	3406

Table 8: Decision Tree Classifier Classification Report

model shows reduced effectiveness in this category with a precision of 0.49 and a lower recall of 0.27, leading to an f1-score of 0.35. This suggests a significant challenge in correctly identifying cases of Depression and Anxiety, as indicated by a considerable number of false negatives and a moderate rate of false positives; (3) *None (Healthy)*: in classifying healthy individuals, the model exhibits decent performance with a precision of 0.65 and a higher recall of 0.85, resulting in a f1-score of 0.74. This indicates a relatively reliable identification of healthy cases, albeit with some room for reducing false positive rates; (4) *Psychosis*: for the category of Psychosis, the model displays a moderate precision of 0.67 but a lower recall of 0.35, yielding an f1-score of 0.46. This points to a reasonable accuracy when the model predicts Psychosis (lower false positives) but a substantial number of missed true cases (higher false negatives); (5) *Suicide and Depression*: the model achieves a fairly good precision of 0.73 and a moderate recall of 0.51, resulting in an f1-score of 0.60. This suggests a somewhat balanced performance in terms of false positives and false negatives, though with room for improvement in recall; (6) *Overall Performance*: the overall accuracy of the model stands at 0.70, reflecting a competent level of performance across all categories. However, the macro average f1-score (0.62) and weighted average f1-score (0.68) indicate a disparity in performance across different categories, with strengths in certain areas like Alzheimer's Disease and Dementia and weaknesses in others such as Depression and Anxiety. In summary, the Random Forest Classifier demonstrates a robust capability in identifying Alzheimer's Disease and Dementia and reasonably good performance in distinguishing healthy individuals. However, it faces challenges in accurately classifying conditions like Depression and Anxiety, Psychosis, and Suicide and Depression. These variations in performance

highlight the need for further model refinement or exploration of additional or alternative features, particularly for the categories where performance is lacking. The disparity in classification effectiveness across different mental health conditions also suggests the potential utility of more tailored approaches or models for specific conditions, as well as the importance of considering class imbalance in the training data.

The ***Decision Tree Classifier*** (see Table 7) elucidates its performance in diagnosing various mental health conditions. The report details the model's precision, recall, f1-score for each category, and overall accuracy, offering a comprehensive assessment of its predictive capabilities. (1) *Alzheimer's Disease and Dementia*: in this category, the Decision Tree Classifier exhibits exemplary performance, as evidenced by its high precision (0.95) and perfect recall (1.00), leading to an impressive f1-score of 0.97. This indicates the model's robust ability to accurately identify cases of Alzheimer's Disease and Dementia with minimal error; (2) *Depression and Anxiety*: the model shows limited effectiveness in classifying Depression and Anxiety, with a precision of 0.49 and a recall of 0.29, resulting in an f1-score of 0.37. This suggests considerable challenges in accurately detecting cases of Depression and Anxiety, as indicated by a high rate of false negatives and a significant number of false positives; (3) *None (Healthy)*: the classifier demonstrates reasonable performance in identifying healthy individuals, with a precision of 0.65 and a higher recall of 0.85, yielding an f1-score of 0.74. This suggests a reliable identification of healthy cases, though there is scope for improvement in precision; (4) *Psychosis*: in the Psychosis category, the model achieves moderate precision (0.67) but a lower recall (0.35), resulting in an f1-score of 0.46. This points to a moderate level of accuracy in predicting Psychosis (fewer false positives), but with a notable number of missed true cases (higher false negatives); (5) *Suicide and Depression*: the classifier shows fairly good precision (0.74) but moderate recall (0.49), culminating in an f1-score of 0.59. This balance suggests a better equilibrium between false positives and false negatives, although the number of missed cases is still significant; (6) *Overall Performance*: the Decision Tree Classifier achieves an overall accuracy of 0.70, indicating a satisfactory level of performance across all classes. However, the macro average f1-score (0.63) and weighted average f1-score (0.68) highlight disparities in performance across different categories, with significant effectiveness in some areas (such as Alzheimer's Disease and Dementia) and limitations in others (notably Depression and Anxiety); In conclusion, the Decision Tree Classifier shows a strong capacity to accurately identify Alzheimer's Disease and Dementia and reasonably good ability to distinguish healthy individuals. However,

it faces considerable challenges in effectively classifying conditions like Depression and Anxiety, Psychosis, and Suicide and Depression. These variances underscore the need for further refinement of the model or exploration of more sophisticated or specialized features to enhance its predictive accuracy, particularly in the underperforming categories. The observed disparities also suggest the necessity of adopting strategies to address potential class imbalances in the training process to improve the model's diagnostic capabilities across a broader spectrum of mental health conditions.

Moreover, AI-driven methodologies can harness large datasets to identify linguistic biomarkers that may be imperceptible to human clinicians, thereby augmenting the sensitivity of early screening efforts. This innovation also promises to democratize mental health diagnostics by enabling remote and scalable tools that can reach underserved populations, circumventing barriers such as stigma and geographical isolation. In the domain of neurodegenerative disorders, text analysis might track longitudinal changes in language usage over time, facilitating a more dynamic understanding of disease progression. Collectively, these advances stand to refine diagnostic accuracy, enhance patient engagement in their mental wellness, and tailor interventions to the linguistic and emotional profiles discerned through AI analysis.

9 Limitations of AI-parsed text analysis on prediagnosis

The precognition of mental health and neurodegenerative disorders through AI-parsed text and sentiment analysis, while innovative, encounters specific constraints that limit its current clinical utility. One significant limitation is the potential for algorithmic bias, where AI models may exhibit skewed performance due to imbalances or lack of diversity in training datasets. Such biases can result in differential accuracy across populations, leading to misclassification or underrepresentation of certain demographic groups.

Moreover, the complexity of human language, replete with sarcasm, metaphor, and cultural idioms, presents a formidable challenge for AI interpretation [2]. Sentiment analysis algorithms may misclassify the emotional valence of such nuanced communication, potentially yielding false indicators of a disorder. The inherent ambiguity in language, especially when considering text out of context or in short snippets typical of digital communication, further exacerbates the risk of erroneous conclusions.

Data privacy is a critical issue, as the use of personal text data for AI analysis [10] necessitates rigorous consent protocols and data protection measures to safeguard against breaches of confidentiality. Ethical considerations also extend to the implications of false positives or negatives, which can have profound effects on individuals' lives, including unwarranted distress, stigmatization, or inappropriate medical intervention.

The variability in individual communication styles and changes over time adds to the complexity of establishing consistent and reliable diagnostic criteria through text analysis. For neurodegenerative diseases, which progress over time, establishing a baseline for comparison can be challenging, and deviations from the baseline may be subtle and gradual, making them difficult to detect reliably.

Lastly, the current diagnostic standards for mental health and neurodegenerative disorders involve a multifaceted clinical approach, including direct patient interviews, cognitive assessments, and medical examinations. AI-parsed text and sentiment analysis, while providing valuable supplementary information, cannot yet replicate the depth and breadth of these traditional methods. Clinicians must interpret AI-generated data with caution, integrating it with a comprehensive clinical picture to make informed decisions regarding diagnosis and treatment.

10 Future development

Our objective is to conduct a comprehensive investigation using a validated dataset, while also enhancing the model by incorporating speech analysis within a high-performance computing (HPC) environment [81]. The present stage of the investigation encounters constraints within the Google Colab Pro platform.

The future development lines of AI will offer more sensitive, specific, and timely identification of mental health and neurodegenerative disorders, ultimately leading to better patient outcomes through early and personalized care. The trajectory of AI in the precognition of mental health and neurodegenerative disorders is trending towards several specific lines of development:

1. **Voice analysis expansion** [19]: The tonal quality, pitch fluctuations, speech rate, and pause patterns in voice data can be quantitatively analyzed to detect early subtle signs of cognitive decline or emotional distress. For example, monotone speech may be an early indicator of Parkinson's disease, while a decrease in speaking rate and increased pause time

may suggest Alzheimer's disease. Future AI models could be trained to detect these vocal biomarkers with greater precision, utilizing deep learning techniques to learn from a vast array of voice samples;

2. **Contextual and idiomatic language understanding** [24]: Developing AI systems with an enhanced understanding of context will involve creating more sophisticated NLP algorithms capable of detecting sarcasm, irony, and humor. This requires training on diverse datasets that include various dialects and colloquialisms to reflect the true range of human language use;

3. **Neuroimaging integration** [48]: Combining AI-parsed text and sentiment analysis with data from neuroimaging techniques like fMRI or PET scans could lead to more accurate identifiers of disease. AI could help correlate changes in speech and writing with specific neural patterns associated with neurodegeneration;

4. **Genomic correlations** [22]: AI could be used to find associations between linguistic changes and genetic markers. By analyzing the genetic profiles of individuals alongside language symptoms, researchers can identify hereditary patterns in neurodegenerative disease manifestation;

5. **Real-time wearable monitoring** [10]: The future may see the proliferation of wearable devices that not only track physical health metrics but also capture speech and writing in real-time. AI systems could analyze this data continuously to identify trends predictive of mental health and cognitive conditions;

6. **Digital phenotyping** [56]: This involves the collection and analysis of data on behavior and lifestyle as manifested in the digital realm, from typing speeds on smartphones to interaction patterns on social media. Such phenotyping could provide additional clues to cognitive and mental health status;

7. **Ethical data governance**: As AI systems gain access to more sensitive personal data, the development of rigorous ethical frameworks to govern data use will be crucial. This includes transparent AI operations, user consent protocols, and privacy-preserving analytics techniques such as federated learning;

8. **AI education**: Efforts will be made to increase the explainability of AI models in healthcare, enabling clinicians to understand and trust AI-

driven assessments. Explainable AI will be essential for integrating AI insights into clinical decision-making.

Each of these development lines aims to *enhance* the *precision*, *reliability*, and *ethical integrity of AI applications* in mental health and neurodegenerative disease care, promising a future of proactive and personalized healthcare solutions for performance sports platforms and sports safety solutions [83].

11 Conclusions

The integration of AI in parsing textual data and performing sentiment analysis has been progressively recognized as a substantial adjunct in the field of mental health (such as depression, anxiety, psychotic disorders, Alzheimer's disease and dementia) and neurodegenerative disorders (like Parkinson's disease). Empirical research has underscored AI's potential in flagging early symptomatology and in monitoring the progression of such conditions through the analysis of linguistic cues. Individuals exhibiting mental health disorders, for instance, have been found to demonstrate distinctive patterns in language and sentiment that AI algorithms can discern, often with considerable accuracy. These patterns include a prevalence of negatively connotated language and a proclivity for certain grammatical structures, indicative of affective disturbances or cognitive decline.

For neurodegenerative disorders, shifts in linguistic ability, such as a waning in vocabulary richness and sentence complexity, may serve as early indicators detectable by AI before traditional diagnostic methods yield definitive results. Sentiment analysis has augmented this detection capacity by providing a continuous measure of emotional states, thus facilitating a dynamic assessment framework that aligns closely with the episodic and fluctuating nature of these disorders.

However, the translation of these analytical advancements into clinical practice entails navigating the challenges associated with data diversity, representativeness, and privacy. The applications of AI in this context also raise ethical questions pertaining to the handling of sensitive personal data and the implications of predictive analytics on patient autonomy and stigma. Furthermore, the current outcomes from AI analytics in this sphere are predominantly correlative rather than causative, necessitating cautious interpretation and integration with clinical expertise.

In conclusion, while AI-parsed text and sentiment analysis represent burgeoning fields with transformative potential for precognitive assessments, they

are complemented by an array of limitations that require careful management. Future development in this domain is contingent upon methodological enhancements, multidisciplinary collaborations, and the establishment of rigorous ethical and operational protocols to safeguard against the misapplication of AI and to secure the confidentiality and integrity of patient data.

Overall, while AI-parsed text and sentiment analysis offer promising adjunctive tools for the precognition of certain disorders, they cannot yet replace the nuanced judgment of healthcare professionals. Their role is currently best suited to being one of several streams of data informing a holistic clinical picture. Clinicians must interpret AI-generated data with caution, integrating it with a comprehensive clinical picture to make informed decisions regarding diagnosis and treatment.

The expected outcomes of AI-parsed text and sentiment analysis *in the sports domain* is becoming increasingly vital, particularly for the early recognition of mental health concerns and the precursors to neurodegenerative disorders among athletes. The intense physical demands, psychological stress of competition, and high-impact nature of many sports can precipitate or aggravate conditions such as depression, anxiety, and CTE (Chronic Traumatic Encephalopathy).

Fields of interests are as follows: (1) **Mental health**: AI's ability to analyze linguistic patterns can reveal signs of mental distress that might be overlooked in traditional assessments. For example, a football player's social media posts could be analyzed for changes in emotional tone, indicating stress or depression, which could be related to on-field performance pressure or injury recovery processes. An AI system that notes an increase in language expressing anxiety or negative sentiments could trigger early psychological support interventions; (2) **Neurodegenerative disorder predictions**: In contact sports, repeated head injuries are a known risk factor for neurodegenerative diseases. AI can monitor athletes' speech and writing for coherence, word-finding difficulties, and other linguistic impairments over time. This longitudinal analysis could indicate early cognitive changes suggestive of conditions like CTE well before clinical symptoms manifest, enabling preemptive health measures and informing decisions on career longevity; (3) **Concussion management**: AI can play a pivotal role in post-concussion care. Athletes' communication before and after head injuries can be scrutinized for changes in language processing, which can be subtle and not immediately apparent. Consistent monitoring of an athlete's linguistic expression post-injury can aid in tailoring individualized recovery programs and determining the safest time for return to play; (4) **Performance and well-being correlations**: The sentiment analysis of

athlete interviews and press conferences can offer insights into the relationship between an athlete's psychological state and performance. By quantifying sentiment, AI could help teams and coaches understand how emotional factors influence game-day performance, contributing to strategies that optimize athlete well-being and effectiveness; (5) **Cognitive baselines long-term monitoring**: Establishing cognitive and linguistic baselines for athletes and tracking any deviations from these over time can allow for the early detection of potential health issues. AI systems can be employed to perform this tracking unobtrusively, analyzing routine communications without requiring formal clinical testing.

In summary, the integration of the methods presented in this paper into sports safety initiatives offers a more nuanced and proactive approach to monitoring the mental and neurological health of athletes. It provides an additional layer of protection by identifying potential issues early, thereby facilitating timely interventions and contributing to the overall well-being and longevity of sports professionals.

Acknowledgements

1. **Funding**: The work of L. Szilágyi was supported by the Consolidator Researcher Program of Óbuda University, Budapest, Hungary. The work of A. Biró was supported by University of Málaga and ITware, Hungary.

2. Special thanks to Dr. Katalin Tünde Jánosi-Rancz, from Department of Mathematics-Informatics, Sapientia University, Tg. Mures, Romania for the inspiration and motivation on text and sentiment analysis, to Prof. Dr. Sándor Miklós Szilágyi, from the Department of Electrical Engineering and Information Technology, George Emil Palade University of Medicine, Pharmacy, Science, and Technology of Targu Mures, and to Prof. Dr. Jaime Martin-Martin, from University of Málaga for mentoring and support.

3. **Data Availability Statement**: Data are available upon request. Figures and the results charts of this study can be found at: `https://doi.org/10.6084/m9.figshare.24584871` (accessed on 24 November 2023).

Abbreviations

The following abbreviations are used in this manuscript:

AI	artificial intelligence
NLP	natural language processing
ML	machine learning
DSM	Diagnostic and Statistical Manual of Mental Disorders
ICD	international Classification of Diseases
CSF	cerebrospinal fluid
MRI	magnetic resonance imaging
PET	positron emission tomography
CT	computed tomography
TF-IDF	term frequency-inverse document frequency
SVM	support vector machine
HPC	High Performance Computing

References

[1] J. Abhishek, R. Raja, AI for the detection of neurological condition: Parkinson's disease & emotions, *i-manager's Journal on Artificial Intelligence & Machine Learning (JAIM)* **1**, 1 (2023) 34–40. ⇒361

[2] G. Ahmad, J. Singla, A. Anis, A. Reshi, A. Salameh, Machine learning techniques for sentiment analysis of code-mixed and switched indian social media text corpus - a comprehensive review, *International Journal of Advanced Computer Science and Applications* **13**, 2 (2022) 455–467. ⇒391

[3] Z. Alirezaei, M. Pourhanifeh, S. Borran, M. Nejati, H. Mirzaei, M. Hamblin, Neurofilament light chain as a biomarker, and correlation with magnetic resonance imaging in diagnosis of cns-related disorders, *Molecular Neurobiology* **57**, (2019) 469–491. ⇒362

[4] I. Akushevich, J. Kravchenko, A. Yashkin, P. Doraiswamy, C. Hill, Expanding the scope of health disparities research in alzheimer's disease and related dementias, *Alzheimer's & Dementia: Diagnosis, Assessment & Disease Monitoring* **15**, 1 (2023) e12415. ⇒361

[5] F. Amato, L. Borzì, G. Olmo, J. Orozco-Arroyave, An algorithm for parkinson's disease speech classification based on isolated words analysis, *Health Information Science and Systems* **9**, (2021) 32. ⇒367

[6] K. Atchison, S. Shafiq, D. Ewert, A. Leung, Z. Goodarzi, Detecting anxiety in long-term care residents: a systematic review, *Canadian Journal on Aging / La Revue Canadienne Du Vieillissement* **42**, 1 (2022) 92-101. ⇒371

[7] I. Baek, E. Lee, J. Kim, Differences in anxiety sensitivity factors between anxiety and depressive disorders, *Depression and Anxiety* **36**, 10 (2019) 968–974. ⇒361

[8] Q. Baker, F. Shatnawi, S. Rawashdeh, M. Al-Smadi, Y. Jararweh, Detecting epidemic diseases using sentiment analysis of arabic tweets, *Journal of Universal Computer Science* **26**, 1 (2020) 50–70. ⇒365

[9] F. Bessière, B. Mondésert, M. Chaix, P. Khairy, Arrhythmias in adults with congenital heart disease and heart failure., *Heart Rhythm* O^2 **2**, 6 (2020) 744–753. ⇒379

[10] A. Bíró, K.T. Jánosi-Rancz, L. Szilágyi, A.I. Cuesta-Vargas, J. Martín-Martín, S.M. Szilágyi, Visual Object Detection with DETR to Support Video-Diagnosis Using Conference Tools, *Applied Sciences* **12**, 12 (2022) 5977. ⇒368, 392, 393

[11] G. Bologna, A rule extraction technique applied to ensembles of neural networks, random forests, and gradient-boosted trees. Algorithms, 14(12), (2021) 339. ⇒381

[12] A. Bothra, Y. Cao, J. Černý, G. Arora, The epidemiology of infectious diseases meets AI: a match made in heaven, *Pathogens* **12**, 2 (2023) 317. ⇒361

[13] J. Breslau, E. Leckman-Westin, H. Yu, B. Han, R. Pritam, D. Guarasi, M. Horvitz-Lennon, D.M. Scharf, H.A. Pincus, M.T. Finnerty Impact of a mental health based primary care program on quality of physical health care, *Administration and Policy in Mental Health and Mental Health Services Research* **45**, 2 (2017) 276–285. ⇒365

[14] E. Brindal, N. Kakoschke, S. Golley, M. Rebuli, D. Baird, Effectiveness and feasibility of a self-guided mobile app targeting emotional well-being in healthy adults: 4-week randomized controlled trial, *Jmir Mental Health* **10**, (2023) e44925. ⇒366

[15] R. Calleja, J. Mas, S. Abraha, J. Nolan, O. Harrison, G. Tadros, A. Matic, Machine learning model to predict mental health crises from electronic health records, *Nature Medicine* **28**, 6 (2022) 1240–1248. ⇒365

[16] S. Chakraborty, H. Paul, S. Ghatak, S. Pandey, A. Kumar, K. Singh, M. Shah, An AI-based medical chatbot model for infectious disease prediction, *IEEE Access* **10**, (2022) 128469–128483. ⇒361

[17] R. Cooper, Diagnostic and statistical manual of mental disorders (dsm), *Knowledge Organization* **44**, 8 (2017) 668–676. ⇒362

[18] J. Davis, A. Fischl, J. Beck, L. Browning, A. Carter, J. Condon, *et al.*, 2022 national standards for diabetes self-management education and support, *Diabetes Spectrum* **35**, 2 (2022) 137–149. ⇒378

[19] D. Dixit, V. Mittal, Y. Sharma, Voice parameter analysis for the disease detection, *IOSR Journal of Electronics and Communication Engineering* **9**, 3 (2014) 48–55. ⇒392

[20] S. Dixit, K. Bohre, Y. Singh, Y. Himeur, W. Mansoor, S. Atalla, S., K. Srinivasan, A comprehensive review on ai-enabled models for Parkinson's disease diagnosis, *Electronics* **12**, 4 (2023) 783. ⇒361

[21] H. Dong, V. Suárez-Paniagua, H. Zhang, M. Wang, A. Casey, E. Davidson, J. Chen, B. Alex, W. Whiteley, H. Wu, Ontology-driven and weakly supervised rare disease identification from clinical notes, *BMC Med Inform Decis Mak* **23** 1, (2023) 86. ⇒361

[22] J. Elwood, E. Murray, A. Bell, M. Sinclair, G. Kernohan, J. Stockdale, A systematic review investigating if genetic or epigenetic markers are associated with postnatal depression, *Journal of Affective Disorders* **253**, (2019) 51–62. ⇒371, 393

[23] L. Erkoreka, N. Ozamiz-Etxebarria, O. Ruiz, J. Ballesteros, Assessment of psychiatric symptomatology in bilingual psychotic patients: a systematic review and meta-analysis, *International Journal of Environmental Research and Public Health* **17**, 11 (2020) 4137. ⇒362

[24] S. Fakharian, P. Cook, Contextualized embeddings encode monolingual and cross-lingual knowledge of idiomaticity, *17th Workshop on Multiword Expressions* **17**, (2021) 23–32. ⇒393

[25] S. Franklyn, J. Stewart, C. Beaurepaire, E. Thaw, R. McQuaid, Developing symptom clusters: linking inflammatory biomarkers to depressive symptom profiles, *Translational Psychiatry* **12**, (2022) 133. ⇒363

[26] J.E. Galvin, S. Chrisphonte, I. Cohen, K.K. Greenfield, M.J. Kleiman, C. Moore, M.L. Riccio, A. Rosenfeld, N. Shkolnik, M. Walker, L.C. Chang, M.I. Tolea, Characterization of dementia with lewy bodies (dlb) and mild cognitive impairment using the lewy body dementia module (lbd-mod), *Alzheimer's & Dementia* **17**, 10 (2021) 1675–1686. ⇒376

[27] L. Gambogi, L. Souza, P. Caramelli, How to differentiate behavioral variant frontotemporal dementia from primary psychiatric disorders: practical aspects for the clinician, *Arquivos De Neuro-Psiquiatria* **80**, 5s1 (2022) 7–14. ⇒363

[28] V. Gouttebarge, A. Bindra, C. Blauwet, N. Campriani, A. Currie, L. Engebretsen, B. Hainline, E. Kroshus, D. McDuff, M. Mountjoy, R. Purcell, M. Putukian, C.L. Reardon, S.M. Rice, R. Budgett, International olympic committee (ioc) sport mental health assessment tool 1 (smhat-1) and sport mental health recognition tool 1 (smhrt-1): towards better support of athletes' mental health, *British Journal of sports Medicine* **55**, 1 (2020) 30–37. ⇒361

[29] V. Gouttebarge, J.M. Castaldelli-Maia, P. Gorczynski, B. Hainline, M.E. Hitchcock, G.M. Kerkhoffs, S.M. Rice, C.L. Reardon, Occurrence of mental health symptoms and disorders in current and former elite athletes: a systematic review and meta-analysis, *British Journal of sports Medicine* **53**, 11 (2019) 700–706. ⇒361

[30] H. Griffiths, The acceptability and feasibility of using text messaging to support the delivery of physical health care in those suffering from a psychotic disorder: a review of the literature, *Psychiatric Quarterly* **91**, 4 (2020) 1305–1316. ⇒361

[31] D. Gruson, P. Dabla, S. Stankovic, E. Homsak, B. Gouget, S. Bernardini, B. Macq, Artificial intelligence and thyroid disease management, *Biochemia Medica* **32**, 2 (2022) 182–188. ⇒361

[32] K. Hallgren, Remotely assessing mechanisms of behavioral change in community substance use disorder treatment to facilitate measurement-informed care: pilot longitudinal questionnaire study, *JMIR Formative Research* **6**, 11 (2022) e42376. ⇒366

[33] S. Holmes, J. Upadhyay, D. Borsook, Delineating conditions and subtypes in chronic pain using neuroimaging, *Pain Reports* **4**, 4 (2019) e768. ⇒379
[34] H. Isah, P. Trundle, D. Neagu, Social media analysis for product safety using text mining and sentiment analysis, *14th UK Workshop on Computational Intelligence (UKCI)* **14**, 6930158 (2014) 1–7. ⇒366
[35] K. Jaeschke, F. Hanna, S. Ali, N. Chowdhary, T. Dua, F. Charlson, Global estimates of service coverage for severe mental disorders: findings from the who mental health atlas 2017, *Global Mental Health* **8**, (2021) e27. ⇒363
[36] S. Kim, Analysis of sentiment analysis research trends using text mining, *techrxiv 21903441* 23 January 2023. ⇒366
[37] D. Koshiyama, K. Kirihara, M. Tada, T. Nagai, M. Fujioka, K. Usui, T. Araki, K. Kasai, Reduced auditory mismatch negativity reflects impaired deviance detection in schizophrenia, *Schizophrenia Bulletin* **46**, 4 (2020) 937–946. ⇒372
[38] K.A. Kvam, M. Benatar, A. Brownlee, T. Caller, R.R. Das, P. Green, S. Kolodziejczak, J. Russo, D. Sanders, N. Sethi, K. Stavros, J. Stierwalt, N.G. Walters, A. Bennett, S.R. Wessels, B.R. Brooks, Amyotrophic lateral sclerosis quality measurement set 2022 update, *Neurology* **101**, 5 (2023) 223–232. ⇒376
[39] M. Landers, S. Saria, A. Espay, Will artificial intelligence replace the movement disorders specialist for diagnosing and managing parkinson's disease?, *Journal of Parkinson's Disease* **11**, s1 (2021) S117–S122. ⇒361
[40] S. Lee, S. Ma, J. Meng, J. Zhuang, T. Peng, Detecting sentiment toward emerging infectious diseases on social media: a validity evaluation of dictionary-based sentiment analysis, *International Journal of Environmental Research and Public Health* **19**, 11 (2022) 6759. ⇒362
[41] C. Lee, B. Jo, H. Woo, Y. Im, R. Park, C. Park, Chronic disease prediction using the common data model: development study, *JMIR AI* **1**, 1 (2022) e41030. ⇒361
[42] J. Liu, J. Kong, X. Zhang, Study on differences between patients with physiological and psychological diseases in online health communities: topic analysis and sentiment analysis, *International Journal of Environmental Research and Public Health* **17**, 5 (2020) 1508. ⇒364
[43] S. Mahadevan, A. Wojtusciszyn, L. Favre, S. Boughorbel, J. Shan, K. Letaief, N. Pitteloud, L. Chouchane, Precision medicine in the era of artificial intelligence: implications in chronic disease management, *Journal of Translational Medicine* **18**, (2020) 472. ⇒361
[44] N. Mahesh, R. Donati, Neurodegenerative diseases and potential early detection methods, *Journal of Student Research* **11**, 4 (2022) 3441. ⇒363
[45] S. Malakar, S. Roy, S. Das, S. Swaraj, J. Velásquez, R. Sarkar, Computer based diagnosis of some chronic diseases: a medical journey of the last two decades, *Archives of Computational Methods in Engineering* **29**, 7 (2022) 5525–5567. ⇒361
[46] B. Muqaku, P. Oeckl, Peptidomic approaches and observations in neurodegenerative diseases, *International Journal of Molecular Sciences* **23**, 13 (2022) 7332. ⇒373

[47] C. Musket, N. Hansen, K. Welker, K. Gilbert, J. Gruber, A pilot investigation of emotional regulation difficulties and mindfulness-based strategies in manic and remitted bipolar I disorder and major depressive disorder, *International Journal of bipolar Disorders* **9**, 1 (2021) 2. ⇒371, 372

[48] M.A. Myszczynska, P.N. Ojamies, A.M.B. Lacoste, D. Neil, A. Saffari, R. Mead, G.M. Hautbergue, J.D. Holbrook, L. Ferraiuolo, Applications of machine learning to diagnosis and treatment of neurodegenerative diseases, *Nature Reviews Neurology* **16**, 8 (2020) 440–456. ⇒393

[49] B. Nichol, A. Hurlbert, J. Read, Predicting attitudes towards screening for neurodegenerative diseases using oct and artificial intelligence: findings from a literature review, *Journal of Public Health Research* **11**, 4 (2022) 227990362211276. ⇒361

[50] R. Nithyashree, R. Deveswaran, A comprehensive review on rheumatoid arthritis, *Journal of Pharmaceutical Research International* **32**, 12 (2020) 18–32. ⇒379

[51] N. Norori, Q. Hu, F. Aellen, F. Faraci, A. Tzovara, Addressing bias in big data and ai for health care: a call for open science, *Patterns* **2**, 10 (2021) 100347. ⇒363, 365

[52] J.A. Ohar, G.T. Ferguson, D.A. Mahler, M.B. Drummond, R. Dhand, R.A. Pleasants, A. Anzueto, D.M.G. Halpin, D.B. Price, G.S. Drescher, H.M. Hoy, J. Haughney, M.W. Hess, O.S. Usmani, Measuring peak inspiratory flow in patients with chronic obstructive pulmonary disease, *International Journal of Chronic Obstructive Pulmonary Disease* **19**, (2022) 79–92. ⇒379

[53] A. Palmisano, S. Meshberg-Cohen, I. Petrakis, M. Sofuoglu, A systematic review evaluating PTSD treatment effects on intermediate phenotypes of PTSD, *Psychological Trauma Theory Research Practice and Policy* **15**, (2023) in press. ⇒372

[54] H. Pandey, A. Shivnani, A. Chauhan, A. Singh, P. Khadakban, Application of AI for analysis of Parkinson's disease, *International Journal of Soft Computing and Engineering* **11**, 1 (2021) 33–39. ⇒361

[55] A. Patil, V. Biousse, N. Newman, Artificial intelligence in ophthalmology: an insight into neurodegenerative disease, *Current Opinion in Ophthalmology* **33**, 5 (2022) 432–439. ⇒361

[56] G. Pavarini, A. Yosifova, K. Wang, B. Wilcox, N. Tomat, J. Lorimer, L. Kariyawasam, L. George, S. Ali, I. Singh, Data sharing in the age of predictive psychiatry: an adolescent perspective, *BMJ Mental Health* **25**, 2 (2022) 69–76. ⇒393

[57] K. Pierre, V. Molina, S. Shukla, A. Avila, N. Fong, J. Nguyen, B. Lucke-Wold, Chronic traumatic encephalopathy: diagnostic updates and advances, *AIMS Neuroscience* **9**, 4 (2022) 519–535. ⇒364

[58] T. Quinton, B. Morris, M. Barwood, M. Conner, Promoting physical activity through text messages: the impact of attitude and goal priority messages, *Health Psychology and Behavioral Medicine* **9**, 1 (2021) 165–181. ⇒361

[59] V.R. Raju, Computational analysis of MER with STN DBS in parkinson's disease using machine learning techniques, *IP Indian Journal of Neurosciences* **6**, 4 (2020) 281–295. ⇒375

[60] V. Ramos, A. Lowit, L. Steen, H. Hernandez-Diaz, M. Huici, M. Bodt, G. Nuffelen, Acoustic identification of sentence accent in speakers with dysarthria: cross-population validation and severity related patterns, *Brain Sciences* **11**, 1ö (2021) 1344. ⇒366

[61] J.M. Ranson, M. Bucholc, D. Lyall, D. Newby, L. Winchester, N.P. Oxtoby, M. Veldsman, T. Rittman, S. Marzi, N. Skene, A. Al Khleifat, I.F. Foote, V. Orgeta, A. Kormilitzin, I. Lourida, D.J. Llewellyn Harnessing the potential of machine learning and artificial intelligence for dementia research, *Brain Informatics* **10**, (2023) 6. ⇒363

[62] S. Schneider, L. Tschaidse, N. Reisch, Thyroid disorders and movement disorders —a systematic review, *Movement Disorders Clinical Practice* **10**, 3 (2023) 360–368. ⇒379

[63] H. Sivasathiaseelan, C.R. Marshall, J.L. Agustus, E. Benhamou, R.L. Bond, J.E.P. van Leeuwen, C.J.D, Hardy, J.D. Rohrer, J.D. Warren, Frontotemporal dementia: a clinical review, *Seminars in Neurology* **39**, 2 (2019) 251–263. ⇒375

[64] M. Sobański, A. Zacharzewska-Gondek, M. Waliszewska-Prosół, M. Sssiadek, A. Zimny, J. Bladowska, A review of neuroimaging in rare neurodegenerative diseases, *Dementia and Geriatric Cognitive Disorders* **49**, 6 (2020) 544–556. ⇒363

[65] T. Strandberg, P. Tienari, M. Kivimäki, Vascular and Alzheimer disease in dementia, *Annals of Neurology* **87**, 5 (2020) 788–788. ⇒375

[66] Y. Sugawara, Y. Tomata, T. Sekiguchi, Y. Yabe, Y. Hagiwara, I. Tsuji, Social trust predicts sleep disorder at 6 years after the great east japan earthquake: data from a prospective cohort study, *BMC Psychology* **8**, 1 (2020) 69. ⇒361

[67] K. Szabó Nagy, J. Kapusta, TwIdw—A Novel Method for Feature Extraction from Unstructured Texts, *Applied Sciences* **13**, (2023) 6438. ⇒369

[68] J. Szarpak, D. Weronika, I. Gabka, D. Madycka, O. Wysokińska, The meaning of blood and cerebrospinal fluid biomarkers in early diagnosis of Alzheimer's disease, *Journal of Education Health and Sport* **10**, 9 (2020) 308–318. ⇒362

[69] Z.Q. Tan, H.Y. Wei, X.B. Song, W.X. Mai, J.J. Yan, W.J. Ye, X.Y. Ling, L. Hou, S.J. Zhang, S. Yan, H. Xu, L. Wang. Positron emission tomography in the neuroimaging of autism spectrum disorder: a review, *Frontiers in Neuroscience* **16**, (2022) 806876.

[70] Y. Tang, Y. Liu, L. Jing, H. Wang, J. Yang, Mindfulness and regulatory emotional self-efficacy of injured athletes returning to sports: the mediating role of competitive state anxiety and athlete burnout, *International Journal of Environmental Research and Public Health* **19**, 18 (2022) 11702. ⇒363 ⇒364

[71] N. Tran, C. Kretsch, C. LaValley, H. Rashidi, Machine learning and artificial intelligence for the diagnosis of infectious diseases in immunocompromised patients, *Current Opinion in Infectious Diseases* **36**, 4 (2023) 235–242. ⇒361

[72] N. Tran, S. Albahra, L. May, S. Waldman, S. Crabtree, S. Bainbridge, H. Rashidi, Evolving applications of artificial intelligence and machine learning in infectious diseases testing, *Clinical Chemistry* **68**, 1 (2021) 125–133. ⇒361

[73] E. Urtnasan, E. Joo, K. Lee, AI-enabled algorithm for automatic classification of sleep disorders based on single-lead electrocardiogram, *Diagnostics* **11**, 11 (2021) 2054 ⇒361

[74] S. Vella, M. Schweickle, J. Sutcliffe, C. Liddelow, C. Swann, A systems theory of mental health in recreational sport, *International Journal of Environmental Research and Public Health* **19**, 21 (2022) 14244. ⇒363

[75] Y. Wan, X. Wu, Y. Kou, The impact of text message on self-management for coronary heart disease: a meta-analysis of randomized controlled trials, *The Heart Surgery Forum* **23**, 1 (2020) E018-E024. ⇒361

[76] C.S. Wang, J.P. Troost, L.A. Greenbaum, T. Srivastava, K. Reidy, K. Gibson, H. Trachtman, J.D. Piette, C.B. Sethna, K. Meyers, K.M. Dell, C.L. Tran, S. Vento, K. Kallem, E. Herreshoff, S. Hingorani, K. Lemley, G. Oh, E. Brown, J.J. Lin, F. Kaskel, D.S. Gipson, Text messaging for disease monitoring in childhood nephrotic syndrome. *Kidney International Reports* **4**, 8 (2019) 1066–1074. ⇒361

[77] N. Younas, L. Flores, F. Hopfner, G. Höglinger, I. Zerr, A new paradigm for diagnosis of neurodegenerative diseases: peripheral exosomes of brain origin, *Translational Neurodegeneration* **11**, (2022) 28. ⇒363

[78] M. Zuylen, J. Kampman, O. Turgman, A. Gribnau, W. Hoope, B. Preckel, H.C. Willems, G.J. Geurtsen, J. Hermanides, Prospective comparison of three methods for detecting peri-operative neurocognitive disorders in older adults undergoing cardiac and non-cardiac surgery, *Anaesthesia* **78**, 5 (2023) 577–586. ⇒362

[79] D. Zhang, T. Guo, A. Han, S. Vahabli, M. Naseriparsa, F. Xia, Predicting mental health problems with personality, behavior, and social networks, *IEEE International Conference on Big Data* (2021) pp. 4537–4546. ⇒364

[80] ∗∗∗, Depression and Anxiety in Twitter (ID), Indonesian tweet entries potentially containing depression or anxiety behavior, last accessed on 15 November 2023. ⇒368

[81] ∗ ∗ ∗, Komondor, one of the greenest supercomputers in the world, HPC Competence Center, Last accessed on: 13 November 2023. ⇒392

[82] ∗ ∗ ∗, Suicide and Depression Detection, A dataset that can be used to detect suicide and depression in a text, last accessed on 15 November 2023. ⇒368

[83] ∗ ∗ ∗, Sunbears Cloud Campus, last accessed on 24 November 2023. ⇒394

Received: 10 November 2023 • Revised: November 30, 2023

Acta Univ. Sapientiae Informatica **15,** 2 (2023) 404–431

DOI: 10.2478/ausi-2023-0023

A generalized fuzzy-possibilistic c-means clustering algorithm

Mirtill-Boglárka NAGHI
Sapientia Hungarian University of Transylvania,
Cluj-Napoca, Romania
Óbuda University, Budapest, Hungary
Doctoral School of Applied Mathematics and
Applied Informatics
email: naghi.mirtill@ms.sapientia.ro
ORCID: 0009-0005-8936-7769

Levente KOVÁCS
Óbuda University, Budapest, Hungary
University Research, Innovation and Service Center
email: kovacs@uni-obuda.hu
ORCID:0000-0002-3188-0800

László SZILÁGYI
Computational Intelligence Research Group,
Sapientia Hungarian University of Transylvania,
Cluj-Napoca, Romania
Dept. of Electrical Engineering, Târgu Mureş
Óbuda University, Budapest, Hungary
University Research, Innovation and Service Center
email: lalo@ms.sapientia.ro
szilagyi.laszlo@uni-obuda.hu
ORCID: 0000-0001-6722-2642

Abstract. The so-called fuzzy-possibilistic c-means (FPCM) algorithm was introduced as an early mixed-partition method aiming to eliminate

Key words and phrases: fuzzy c-means algorithm, possibilistic c-means algorithm, mixed partition

some adverse effects present in the behavior of the fuzzy c-means (FCM) and the possibilistic c-means (PCM) algorithms. A great advantage of FPCM was the low number of its parameters, as it eliminated the possibilistic penalty terms used by PCM. Unfortunately, FPCM in its original formulation also has a weak point: the strength of the possibilistic term is in inverse proportion with the number of clustered data items, which makes FPCM act like FCM when clustering large sets of data. This paper proposes a modification of the FPCM algorithm by introducing an extra coefficient into the possibilistic term that allows us to control the strength of the possibilistic effect within the mixture model. The modified clustering model will be referred to as generalized FPCM, since a certain value of the extra parameter reduces it to the original FPCM, or in other words, FPCM is a special case of the proposed algorithm. The proposed method is evaluated using noise-free and noisy data as well.

1 Introduction

Data clustering represents one of the first applications of Zadeh's fuzzy logic [24]. The first fuzzy partitioning was defined by Ruspini [15] in 1969, while the first c-means clustering adopting fuzzy partitions is the ISODATA algorithm of Dunn introduced in 1974 [6], which was later generalized by Bezdek [3] and called fuzzy c-means (FCM) algorithm. FCM has been a very popular algorithm over the past decades in a wide range of sciences, in spite of its high sensitivity to noisy data, caused by the probabilistic constraint used by all c-means clustering models defined up to this point.

The necessity to relax the probabilistic constraint led to a series of c-means clustering approaches (e.g. [5, 9]), in which the fuzzy membership functions represented typicality values or the compatibility of data vectors with the clusters. These approaches were able to handle noisy data, to ignore them in establishing the clusters that represent the real, meaningful data vectors. However, they still had limitations: the first one was unable to handle clusters of different sizes (diameter), the second one frequently merged several clusters together.

To avoid this limitation, several mixed partition models of c-means clustering were proposed. In 1997, Pal et al. [11] introduced the fuzzy-possibilistic c-means (FPCM) algorithm, which proposed a mixture partition with a reduced number of parameters. This approach had the limitation of having a behavior strongly influenced by the size of the input dataset. The probabilistic and possibilistic components of the mixed partition were used as a linear combination. This scheme was then reused by the possibilistic-fuzzy c-means

(PFCM) algorithm proposed by Pal et al. [12], while Szilágyi [16]) later presented a different approach that combined the two partition components via multiplication. The most recent mixed partition models proved to be robust as they provide fine partitions both in case of absence and presence of outlier data.

This paper proposes to enhance the services provided by the FPCM algorithm by introducing a generalized formulation. We attempt to eliminate the limitations of PFCM by adding an extra coefficient to the possibilistic term. This parameter denoted by β defines the strength of the possibilistic term in the mixture partition. This modification represents a generalization of the FPCM algorithm because FPCM acts as a special case ($\beta = 1$) of the novel approach, while β can have any positive real value, each leading to a different algorithm.

The proposed clustering model is evaluated using standard datasets taken from the literature, in their original noise-free version, but with some added outliers as well. The evaluation process helps us in formulating recommendations regarding the parameters of the algorithm.

The rest of this paper is structured as follows: Section 2 presents the basic c-means clustering algorithms this work relies on. Section 3 exhibits the details of the proposed clustering model. Section 4 relates on the numerical evaluation of the proposed clustering model in comparison to other c-means clustering algorithms. Section 5 discusses the role of the main parameters and formulates recommendations regarding the use of the proposed method. Section 6 concludes the study.

2 Background works

All c-means clustering algorithms aim at grouping a set of object data $\mathbf{X} = \{\mathbf{x}_1, \mathbf{x}_2, \ldots \mathbf{x}_n\}$ into a fixed number of clusters. Clusters are denoted by Ω_i ($i = 1, \ldots, c$), where c is the number of clusters. As precondition, it is supposed that $c < n$. In real-life problems, usually the number of input data vectors exceeds the number of clusters by orders of magnitude. Each cluster Ω_i ($\forall i = 1, \ldots, c$) is represented by the cluster prototype $\mathbf{v}_i$, which is a vector of same type as the input data.

All c-means clustering models use a partition matrix. The partition matrix generally describes to what extent data vectors belong to each of the classes. In this study we only investigate clustering algorithms that use fuzzy partitions, meaning that all elements of the partition matrix represent fuzzy membership

functions. We use two different notations for the partition matrix: $\mathbf{U} = [u_{ik}] \in \mathcal{M}_{c\times n}$ and $\mathbf{T} = [t_{ik}] \in \mathcal{M}_{c\times n}$. The difference between these two matrices is that u_{ik} values satisfy the probabilistic constraint, meaning that all columns of matrix $\mathbf{U}$ sum up to 1, while the columns of $\mathbf{T}$ contain typicality values, t_{ik} ($i = 1, \ldots, c$; $k = 1, \ldots, n$) expressing how much vector $\mathbf{x}_k$ is compatible with cluster Ω_i.

2.1 The fuzzy c-means algorithm

The fuzzy c-means clustering algorithm minimizes the following objective function:

$$J_{FCM} = \sum_{i=1}^{c}\sum_{k=1}^{n} u_{ik}^{m}\|\mathbf{x}_k - \mathbf{v}_i\|_{\mathbf{A}}^2 = \sum_{i=1}^{c}\sum_{k=1}^{n} u_{ik}^{m} d_{ik}^2 \ , \tag{1}$$

being subject to the probabilistic constraint

$$\sum_{i=1}^{c} u_{ik} = 1 \qquad \forall k = 1, \ldots, n \ , \tag{2}$$

where $d_{ik} = \|\mathbf{x}_k - \mathbf{v}_i\|$ represents the distance between input vector $\mathbf{x}_k$ and cluster prototype $\mathbf{v}_i$, for any $i = 1, \ldots, c$, and $k = 1, \ldots, n$. Parameter $m > 1$ is the fuzzy exponent than controls the fuzziness of the algorithm. It is known, that the limit case $m \to 1$ reduces FCM to the k-means algorithm that uses binary logic to describe the partition. On the other side, if $m \to +\infty$, cluster prototypes merge together at the grand mean of the input data vectors. Raising the value of m makes the algorithm more fuzzy.

The optimization formulas of FCM are obtained from the zero gradient conditions of its objective function extended with special terms containing Lagrange multipliers that enforce the probabilistic constraint. The optimization formulas are obtained as:

$$u_{ik} = \frac{d_{ik}^{\frac{-2}{m-1}}}{\sum_{j=1}^{c} d_{jk}^{\frac{-2}{m-1}}} \qquad \begin{array}{l}\forall i = 1, \ldots, c \\ \forall k = 1, \ldots, n\end{array} \ , \tag{3}$$

$$\mathbf{v}_i = \frac{\sum_{k=1}^{n} u_{ik}^{m}\mathbf{x}_k}{\sum_{k=1}^{n} u_{ik}^{m}} \qquad \forall i = 1, \ldots, c \ . \tag{4}$$

The algorithm needs to be initialized with cluster prototypes differing from each other. The optimization is performed by alternately applying the formulas given in Eqs. (3) and (4) until convergence is reached. Convergence is reached when cluster prototypes stabilize. If we need to defuzzify the final partition, we may assign each data vector to the cluster whose prototype is closest, or the one with respect to which the fuzzy membership function has highest value. These two criteria are equivalent:

$$\mathbf{x}_k \to \Omega_i \quad \Leftrightarrow \quad i = \arg\min_j\{d_{jk}, j = 1, \ldots, c\} = \arg\max_j\{u_{jk}, j = 1, \ldots, c\} \quad (5)$$

Besides being a very popular algorithm in all sciences involving numerical data, a major disadvantage of FCM is its sensitivity to outlier data. A single distant outlier can attract cluster prototypes out of the range of the elements that it represents, or in extreme case the outlier may "steal" one of the cluster prototypes, causing poor partitioning of the real meaningful data.

2.2 The possibilistic c-means algorithm

The noise sensitivity of FCM was attributed to the probabilistic constraints of the partition, and thus several solutions emerged that relaxed this too strong limitation. The algorithm called FCM with extra noise class and also referred to as fuzzy $(c + 1)$-means defined an extra cluster Ω_0 which has no cluster prototype and is situated at an equal constant distance d_0 from all input vectors x_k $(k = 1, \ldots, n)$. The probabilistic constraint in this case looks like

$$\sum_{i=0}^{c} u_{ik} = 1 \qquad \forall k = 1, \ldots, n \ , \quad (6)$$

but now any noisy data vector x_k receives a high membership towards the noise class and this way it will hardly influence the cluster prototypes. The algorithm needs careful initialization, meaning that initial cluster prototypes should not be set to the noisy data vector. A limitation of this algorithm stands in the fact that similarly to FCM, it cannot handle clusters of different widths (radii).

Theoretically all c-means clustering models that use fuzzy partitions without constraining the fuzzy memberships with probabilistic condition could be called possibilistic c-means. However, the so-called possibilistic c-means algorithm is the one introduced by Krishnapuram and Keller [9]. PCM minimizes

the following objective function:

$$J_{PCM} = \sum_{i=1}^{c}\sum_{k=1}^{n} t_{ik}^{p} d_{ik}^{2} + (1 - t_{ik})^{p}\eta_i \ , \tag{7}$$

subject to the possibilistic constraint

$$0 < \sum_{i=1}^{c} t_{ik} < c \qquad \forall k = 1, \ldots, n \ , \tag{8}$$

which means that all data vectors must belong to at least one cluster to a nonzero extent, and none of the data vectors can be fully compatible with all clusters. Further on, η_i represents the possibilistic penalty term of cluster i ($i = 1, \ldots, c$) which is meant to control the width of the cluster, and $p > 1$ represents the so-called possibilistic exponent.

Similarly to the FCM algorithm, the optimization formulas are extracted from the zero gradient conditions of the objective function, but here there is no need to use Lagrange multipliers. The optimization formulas are obtained as:

$$t_{ik} = \left[1 + \left(\frac{d_{ik}^2}{\eta_i}\right)^{\frac{1}{p-1}}\right]^{-1} \qquad \begin{matrix} \forall i = 1, \ldots, c \\ \forall k = 1, \ldots, n \end{matrix} \ , \tag{9}$$

and

$$\mathbf{v}_i = \frac{\sum_{k=1}^{n} t_{ik}^{p} \mathbf{x}_k}{\sum_{k=1}^{n} t_{ik}^{p}} \qquad \forall i = 1, \ldots, c \ , \tag{10}$$

which are alternately applied until cluster prototypes stabilize. PCM can produce fine partitions even in the presence of outlier data, but unfortunately it frequently merges several or all clusters together. If we need to defuzzify the final partition, each input data vector is assigned to the cluster with which it shows the highest compatibility:

$$\mathbf{x}_k \to \Omega_i \quad \Leftrightarrow \quad i = \arg\max_{j}\{t_{jk}, j = 1, \ldots, c\} \tag{11}$$

2.3 Algorithms using mixed partitions

Since none of the two basic approaches of fuzzy logic based c-means clustering proved perfect, several attempts were made to merge the two partition matrices

into a mixed partition, and expected them to relax or attenuate each other's limitations. A linear combination of the classical FCM and PCM partitions was proposed by Pal et al. [12], referred to as possibilistic fuzzy c-means algorithm. The other approach called fuzzy possibilistic product partition c-means algorithm was proposed by Szilágyi [16] and later generalized for clusters with special shapes [17, 19]. However, this paper intends to generalize the method called fuzzy-possibilistic c-means (FPCM) algorithm introduced by Pal et al. [11], which uses an alternative definition for the possibilistic partition that is involved into a linear combination with the FCM partition matrix.

FPCM minimizes the following objective function:

$$J_{\mathrm{FPCM}} = \sum_{i=1}^{c}\sum_{k=1}^{n} \left(u_{ik}^{m} + t_{ik}^{p}\right) d_{ik}^{2} \ , \tag{12}$$

constrained by

$$\sum_{i=1}^{c} u_{ik} = 1 \qquad \forall k = 1, \ldots, n \ , \tag{13}$$

and

$$\sum_{k=1}^{n} t_{ik} = 1 \qquad \forall i = 1, \ldots, c \ , \tag{14}$$

where $m > 1$ and $p > 1$ represent the fuzzy and possibilistic exponents, respectively. Both constraints presented in Eqs. (13) and (14) may seem probabilistic at first sight. However, the elements of partition matrix $\mathbf{U}$ sum up to 1 in each column, while in $\mathbf{T}$ they sum up to 1 in each row.

The optimization formulas of FPCM are obtained from the zero gradient conditions of the objective functions, extended with terms that enforce the constraints by the use of Lagrange multipliers. The alternately applied optimization formulas are obtained as:

$$u_{ik} = \frac{d_{ik}^{\frac{-2}{m-1}}}{\sum_{j=1}^{c} d_{jk}^{\frac{-2}{m-1}}} \qquad \text{and} \qquad t_{ik} = \frac{d_{ik}^{\frac{-2}{p-1}}}{\sum_{l=1}^{n} d_{il}^{\frac{-2}{p-1}}} \qquad \begin{matrix} \forall i = 1, \ldots, c \\ \forall k = 1, \ldots, n \end{matrix} \ , \tag{15}$$

and

$$\mathbf{v}_i = \frac{\sum_{k=1}^{n} \left(u_{ik}^{m} + t_{ik}^{p}\right) \mathbf{x}_k}{\sum_{k=1}^{n} \left(u_{ik}^{m} + t_{ik}^{p}\right)} \qquad \forall i = 1, \ldots, c \ , \tag{16}$$

which are applied until cluster prototypes stabilize. The defuzzified partition is defined by the maximum value of the combined partition matrix, according to the formula:

$$\mathbf{x}_k \rightarrow \Omega_i \quad \Leftrightarrow \quad i = \arg\max_j \{u_{jk}^m + t_{jk}^p, j = 1, \ldots, c\} \ . \tag{17}$$

3 Methods

The problem formulation of the original FPCM does not offer equal chances to the probabilistic and possibilistic components to have their effect upon the final partition. This can be explained with the fact that the total sum of fuzzy membership functions in matrix $\mathbf{U}$ is n, which is the number of data vectors being fed to clustering, while in matrix $\mathbf{T}$ the total sum is c, the number of clusters. In the very frequent case, when $n >> c$, the possibilistic term hardly can influence the clustering process. This is why we need to introduce a compensating parameter denoted by β, which appears as a multiplying factor to the possibilistic term in the objective function.

The proposed clustering model, which in the following will be referred to as generalized fuzzy-possibilistic c-means algorithm (GFPCM), optimizes the following objective function:

$$J_{GFPCM} = \sum_{i=1}^{c}\sum_{k=1}^{n} \left(u_{ik}^m + \beta t_{ik}^p\right) \|\mathbf{x}_k - \mathbf{v}_i\|^2 = \sum_{i=1}^{c}\sum_{k=1}^{n} \left(u_{ik}^m + \beta t_{ik}^p\right) d_{ik}^2 \ , \tag{18}$$

subject to the same constraints as FPCM, presented in Eqs (13) and (14). All notations are the same as in PFCM, with the exception of β, which is a positive valued parameter. The proposed clustering model generalizes the original FPCM because FPCM is a special case of the proposed algorithm, namely the one that uses $\beta = 1$. For any other positive value of β we obtain a different algorithm. Another special case is the one defined by $\beta = 0$, setting that reduces GFPCM to FCM regardless to the value of p. At first sight it would seem logical to set $\beta = n/c$ so that the two components of the partition get the same strength. However, in this study we investigate the behavior of the algorithm in a wide range of β values, up to even the order of 10^6.

The optimization formulas of the GFPCM algorithm are obtained from the zero gradient conditions of the following functional:

$$\mathcal{L}_{GFPCM} = J_{GFPCM} + \sum_{k=1}^{n} \lambda_k \left(1 - \sum_{i=1}^{c} u_{ik}\right) + \sum_{i=1}^{c} \tau_i \left(1 - \sum_{k=1}^{n} t_{ik}\right) \ , \tag{19}$$

where λ_k $(k = 1, \ldots, n)$ and τ_i $(i = 1, \ldots, c)$ represent Lagrange multipliers needed to enforce the constraints during optimization. From the partial derivative with respect to u_{ik} $(\forall i = 1, \ldots, c, \forall k = 1, \ldots, n)$ we obtain:

$$\frac{\partial \mathcal{L}_{\text{GFPCM}}}{\partial u_{ik}} = 0 \Longrightarrow m u_{ik}^{m-1} d_{ik}^2 - \lambda_k = 0, \ , \tag{20}$$

which implies

$$u_{ik} = \left(\frac{\lambda_k d_{ik}^{-2}}{m} \right)^{\frac{1}{m-1}} = \left(\frac{\lambda_k}{m} \right)^{\frac{1}{m-1}} d_{ik}^{\frac{-2}{m-1}} \ . \tag{21}$$

We know from Eq. (13), that for any $k = 1, \ldots, n$

$$\sum_{j=1}^{c} u_{jk} = 1 \Longrightarrow 1 = \left(\frac{\lambda_k}{m} \right)^{\frac{1}{m-1}} \sum_{j=1}^{c} d_{jk}^{\frac{-2}{m-1}} \ . \tag{22}$$

Dividing Eqs. (21) and (22) term by term, we obtain

$$u_{ik} = \frac{u_{ik}}{1} = \frac{d_{ik}^{\frac{-2}{m-1}}}{\sum_{j=1}^{c} d_{jk}^{\frac{-2}{m-1}}} \ , \tag{23}$$

which is exactly the partition update formula known from FCM. Similarly, from the partial derivatives with respect to t_{ik} $(\forall i = 1, \ldots, c, \forall k = 1, \ldots, n)$, we obtain:

$$\frac{\partial \mathcal{L}_{\text{GFPCM}}}{\partial t_{ik}} = 0 \Longrightarrow \beta p t_{ik}^{p-1} d_{ik}^2 - \tau_i = 0 \ , \tag{24}$$

which implies

$$t_{ik} = \left(\frac{\tau_i d_{ik}^{-2}}{\beta p} \right)^{\frac{1}{p-1}} = \left(\frac{\tau_i}{\beta p} \right)^{\frac{1}{p-1}} d_{ik}^{\frac{-2}{p-1}} \ . \tag{25}$$

We know from Eq. (14), that for any $i = 1, \ldots, c$

$$\sum_{l=1}^{n} t_{il} = 1 \Longrightarrow 1 = \left(\frac{\tau_i}{\beta p} \right)^{\frac{1}{p-1}} \sum_{l=1}^{n} d_{il}^{\frac{-2}{p-1}} \ . \tag{26}$$

Dividing Eqs. (25) and (26), we obtain

$$t_{ik} = \frac{t_{ik}}{1} = \frac{d_{ik}^{\frac{-2}{p-1}}}{\sum_{l=1}^{n} d_{il}^{\frac{-2}{p-1}}} \ , \tag{27}$$

which is exactly the possibilistic component update formula of FPCM.

The partition update formula is obtained from the partial derivatives with respect to cluster prototype vectors $\mathbf{v}_i$ $(i = 1, \ldots, c)$:

$$\frac{\partial \mathcal{L}_{\mathrm{GFPCM}}}{\partial \mathbf{v}_i} = 0 \Longrightarrow \sum_{k=1}^{n} \left(u_{ik}^m + \beta t_{ik}^p\right)(-2)(\mathbf{x}_k - \mathbf{v}_i) = 0 \ , \tag{28}$$

which implies

$$\mathbf{v}_i \sum_{k=1}^{n} \left(u_{ik}^m + \beta t_{ik}^p\right) = \sum_{k=1}^{n} \left(u_{ik}^m + \beta t_{ik}^p\right) \mathbf{x}_k \ , \tag{29}$$

and consequently we obtain the cluster prototype updated as

$$\mathbf{v}_i = \frac{\sum\limits_{k=1}^{n} \left(u_{ik}^m + \beta t_{ik}^p\right) \mathbf{x}_k}{\sum\limits_{k=1}^{n} \left(u_{ik}^m + \beta t_{ik}^p\right)} \qquad \forall i = 1, \ldots, c \ . \tag{30}$$

Just like in case of any other c-means clustering algorithm, the cluster prototypes are obtained as the weighted average of input data vectors $\mathbf{x}_k$ $(k = 1, \ldots, n)$, where the weights are obtained in the final partition matrices. The defuzzification rule can be formulated as follows:

$$\mathbf{x}_k \to \Omega_i \quad \Leftrightarrow \quad i = \arg\max_j \{u_{jk}^m + \beta t_{jk}^p, j = 1, \ldots, c\} \ . \tag{31}$$

When initializing cluster prototypes, it is recommendable to choose random vectors that are distant from any of the input data vectors $\mathbf{x}_k$ $(k = 1, \ldots, n)$, just as recently suggested in [14]. Let us suppose the contrary, and initialize for example $\mathbf{v}_a = \mathbf{x}_b$ with some valid values of a and b. In this case in the first iteration $t_{ab} = 1$ and $t_{ak} = 0$ for any $k \neq b$. Especially if we use a high value of parameter β, the algorithm will hardly be able to move the cluster prototype $\mathbf{v}_a$ away from $\mathbf{x}_b$.

The GFPCM algorithm can be summarized as follows:

1. Set parameters m, p, and β.
2. Initialize cluster prototypes outside the range of input data vectors.
3. Update the probabilistic term of the partition using Eq. (23).
4. Update the possibilistic term of the partition using Eq. (27).

5. Update the cluster prototypes using Eq. (30).
6. Repeat steps 3-5 until cluster prototypes stabilize.
7. Defuzzify the obtained partition if necessary using Eq. (31).

4 Evaluation

The proposed generalized FPCM method underwent a thorough evaluation process, which aimed to establish the behavior of the algorithm in comparison with its predecessors, mainly the FCM and the original FPCM. We did not expect to find the best clustering model that uses mixed partition. This is why we did not compare the performance of GFPCM with more sophisticated clustering models like PFCM or FPPPCM. So the main goal was to establish under what circumstances GFPCM provides fine partitions and to what extent it can eliminate the sensitivity to outlier data. Details of the evaluation are presented in the following.

4.1 Datasets

Three public datasets are involved in the evaluation process: the IRIS [8], WINE [1], and BreastCancer (Wisconsin) data [21]. The goal was to evaluate the proposed method in clustering problems with more and less dimensions as well. Details of the datasets are given in Table 1. These datasets are involved in clustering in their original format with values normalized in each dimension, and separately with an added outlier. In all cases the added outlier vector is represented as $(\delta, \delta, \ldots, \delta)^{\mathsf{T}}$ in the normalized space, where $\delta > 1$ is a parameter that controls the position of the outlier. By varying the value of δ we can establish to what extent the clustering models can handle an outlier vector in the input data.

4.2 Evaluation criteria

We have chosen the following indicators used in the literature to evaluate the final partitions obtained by the algorithm: purity (abbreviated as PUR) [7], adjusted Rand index (abbreviated as ARI) [13], and normalized mutual information (abbreviated as NMI) [10].

In the context of cluster partition evaluation, purity can be defined as a measure of how well-defined and homogeneous the clusters are. It is a measure of the extent to which each cluster contains instances of only a single class.

Property	IRIS data	WINE data	BreastCancer data
Items (vectors)	150	178	569
Dimensions	4	13	30
Clusters	3	3	2
Cluster sizes	50, 50, 50	59, 71, 48	357, 212
Source	[2, 8]	[1]	[21]

Table 1: Datasets involved in the evaluation process, and their main properties.

The purity of a cluster partition is always in the unit range $[0, 1]$. A purity of 1 indicates that all clusters are well-defined and homogeneous, whereas a purity of 0 indicates that the clusters are completely mixed. Given some clusters M and some set of classes Y, purity is calculated using the following formula:

$$\mathrm{PUR} = \frac{1}{n} \sum_{m \in M} \max_{y \in Y} |m \cap y| \ . \tag{32}$$

In simpler terms, for each cluster, the majority class of the cluster must be found and the number of data points belonging to that majority class must be summed. Finally, the total sum must be divided by the total number of data points (n).

However, this criterion has certain limitations. It does not perform well if the dataset is not balanced, i.e. the number of points belonging to the classes are different. In this case, the purity criterion favors the larger clusters, and as such, some data points from the smaller classes will also be assigned to the larger clusters. Because of this, in unbalanced datasets, a higher purity does not necessarily indicate that the clustering was successful. Therefore, purity may not reflect the true structure of the data in all cases. To alleviate the side effects of solely calculating purity on the cluster partitions of a potentially imbalanced dataset, other criteria must be used, such as the adjusted Rand index (ARI).

The adjusted Rand index is another widely used clustering evaluation criterion. It assesses the similarity of clustering outcomes. ARI is a suitable evaluation criterion for datasets with imbalanced cluster sizes. It takes unexpected cluster assignments into consideration, producing a result that is robust when faced with clusters with significantly different sizes.

The ARI value of a cluster partition is always in the range $[-1, 1]$. However, ARI values are mostly expected to be in the $[0, 1]$ range. An ARI value of 1

indicates a perfect match between the two measured cluster partitions. Otherwise, an ARI value of 0 indicates the baseline with respect to randomness. Negative ARI values represent a result that is worse than random clustering. As such, ARI by itself can also be used to compare two distinctly parameterized clustering results, making sure that any improvements in the clustering similarity are due to the better selection of parameters, rather than random fluctuations.

ARI can be computed with the help of a contingency table that encodes the pairwise relationship between two partitions. Let M and Y denote the two partitions such as $M = \{M_1, M_2, \ldots, M_r\}$ and $Y = \{Y_1, Y_2, \ldots, Y_s\}$. The contingency table, more precisely, the $r \times s$ table counts the pairs that are assigned to the same or different clusters in M and Y, i.e. each cell (n_{ij}) represents the number of data points that belong to both clustering partitions (for the intersection of M_i and Y_j would yield $n_{ij} = |M_i \cap Y_j|$). Naturally, the elements on the diagonal represent the number of data points that are assigned to the same cluster in both partitions. The other elements represent the remaining ones that are assigned to different clusters. Then the table is extended by one row and column that sum all the values in their respective row or column. Precomputing these sums enables easier computation.

Taking everything into consideration, the contingency table has the following structure:

	Y_1	Y_2	$\cdots$	Y_s	$\sum$
M_1	n_{11}	n_{12}	$\cdots$	n_{1s}	a_1
M_2	n_{21}	n_{22}	$\cdots$	n_{2s}	a_2
$\vdots$	$\vdots$	$\vdots$	$\ddots$	$\vdots$	$\vdots$
M_r	n_{r1}	n_{r2}	$\cdots$	n_{rs}	a_r
$\sum$	b_1	b_2	$\cdots$	b_s	

Then ARI is calculated as follows:

$$\mathrm{ARI} = \frac{\sum_{ij} \binom{n_{ij}}{2} - \frac{\sum_i \binom{a_i}{2} \sum_j \binom{b_j}{2}}{\binom{n}{2}}}{\frac{1}{2}\left[\sum_i \binom{a_i}{2} + \sum_j \binom{b_j}{2}\right] - \frac{\sum_i \binom{a_i}{2} \sum_j \binom{b_j}{2}}{\binom{n}{2}}}, \tag{33}$$

where n_{ij}, a_i, b_j are values taken from the contingency table.

Normalized mutual information is an evaluation criterion deeply rooted in information theory. It assumes that the more information is mutual between the two clustering outcomes the more valid the overall result is. It is commonly

used because of its capability to assess partitions even in scenarios where the number of clusters varies. The values range from 0 to 1, where a higher NMI value indicates better agreement between the clustering assignments and the true class labels. A value of 0 indicates no mutual information, whereas a value of 1 indicates a perfect correlation. Altering the order or values of the class or cluster labels through permutation does not impact the NMI value. Let M and Y denote the clustering assignments, then NMI can be calculated as follows:

$$\mathrm{NMI}(Y, M) = \frac{2 \cdot \mathrm{I}(Y; M)}{\mathrm{H}(Y) + \mathrm{H}(M)} \ , \tag{34}$$

where $\mathrm{I}(Y; M)$ is the mutual information between Y and M, $\mathrm{H}(Y)$ is the entropy of Y and similarly, $\mathrm{H}(M)$ is the entropy of M.

There is another common formulation of the normalized mutual information which is more computational heavy than the aforementioned one:

$$\mathrm{NMI}(Y, M) = \frac{\mathrm{I}(Y; M)}{\sqrt{\mathrm{H}(Y) \cdot \mathrm{H}(M)}} \ . \tag{35}$$

Both formulations compute the same results and are valid representations of NMI. Furthermore, NMI is symmetric in the sense that Y and M is interchangeable, i.e. yielding the same result when switched.

These measures can assess the similarity between two clustering partitions, according an overall overview of the efficiency of clustering methods.

4.3 Tests using the IRIS dataset

Clustering algorithms are reported to work fine enough on IRIS data if the number of correct decisions reaches 133 out of 150, which corresponds to PUR=0.8867. When using the IRIS dataset without the addition of noise, we are interested in establishing those cases where GFPCM produces this outcome or better than that. It is also known about IRIS data, that the FCM algorithm produces clusters of better and better purity if the fuzzy exponent m rises, culminating at PUR=0.9333 (140 correct decisions out of 150), even though this pure partition is of low validity according to any cluster validity index (CVI) from the literature [23, 18].

Figure 1 exhibits the benchmarks of the GFPCM algorithm achieved on the IRIS dataset in case of no added outliers. The evolution of the benchmarks are all plotted against the fuzzy exponent m, and the behavior of the algorithm is investigated within a wide range of m. FPCM produces a high-purity partition on IRIS, which is influenced even by a very weak possibilistic term ($\beta = 1$). The

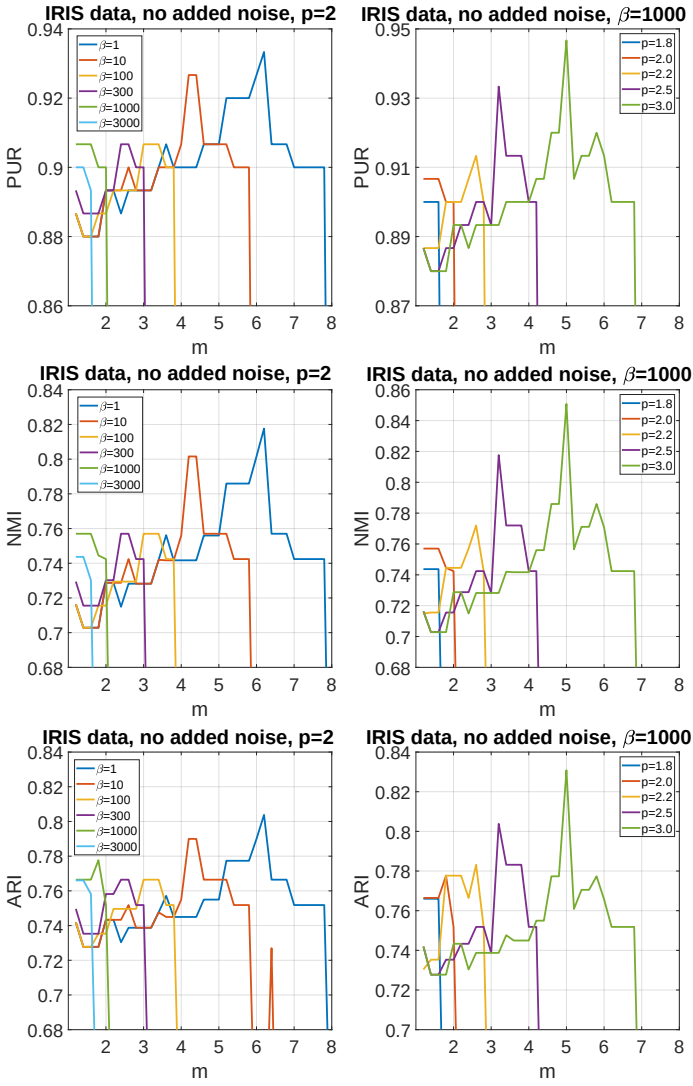


Figure 1: GFPCM benchmarks obtained on the IRIS dataset in case of no added noise. Graph representations in the left column show PUR, NMI and ARI values, respectively, all plotted against fuzzy exponent $\mathfrak{m}$, obtained with various values of trade-off parameter β, while possibilistic exponent was fixed at $\mathfrak{p} = 2$. Graphs in the right column represent PUR, NMI and ARI values plotted against $\mathfrak{m}$, obtained at fixed trade-off $\beta = 1000$, and selected values of exponent $\mathfrak{p}$.

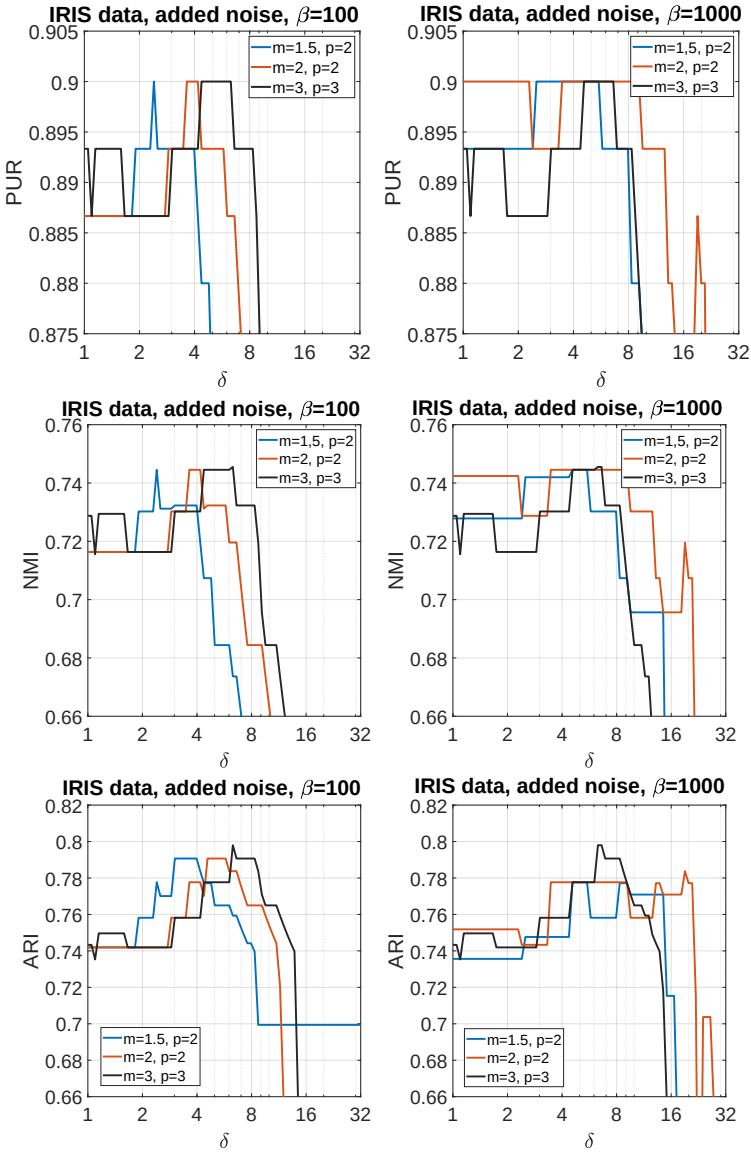


Figure 2: GFPCM benchmarks obtained on the IRIS dataset in case of an outlier added at $(\delta, \delta, \delta, \delta)^{\mathsf{T}}$. A comparison of the cases with trade-of parameter $\beta = 100$ and $\beta = 1000$ is presented, where the former works quite the same as FCM ($\beta = 0$) and FPCM ($\beta = 1$).

higher the value of β, the more restricted becomes the domain of acceptable partitions. However, let us clarify that this phenomenon is not a problem, because the recommended range of the exponent m hardly exceeds the value of 3 [20]. What we can see from the results is that it is not recommendable to use a very strong possibilistic component. This criterion restricts us to set the possibilistic parameters $p \geq 2$, and $\beta < 1000$. On the other hand, it is also visible that a high value of p (e.g. $p = 3$ is already high) weakens the effect of the possibilistic term within GFPCM.

Another thing that deserves to be remarked here: so far we did not see any reported case where any c-means algorithm provided 142 correct decision on the IRIS dataset. Figure 1 shows us such an example: the GFPCM algorithm used at $m = 5$, $p = 3$, and trade-off set to $\beta = 1000$ produced this outstanding PUR benchmark. This experience convinced us that a weak possibilistic term added to the objective function of the FCM algorithm can cause significant alterations in its behavior, even if it is not visible in every scenario.

Figure 2 presents the benchmarks of GFPCM, obtained on the IRIS dataset, with an added outlier whose position depends on parameter δ. The goal is to establish how far the outlier needs to stand to ruin the final partition. Conversely, we may ask what settings are needed for the GFPCM to assure a fine partition even in case of very distant outlier? Figure 2 relates on the examples of $\beta = 100$ (left column) and $\beta = 1000$ (right column). We experienced no intensive change in the behavior of GFPCM while varying the trade-off parameter between 0 and 100. However, as the possibilistic terms is becoming stronger while raising β further, the algorithm demonstrates an enhanced capability to accommodate outliers that are increasingly distant. The limit value of δ still tolerated by GFPCM in various circumstances is studied in Section 4.6.

4.4 Tests using the WINE dataset

WINE dataset contains vectors in a normalized 13-dimensional space, which are organized in unequal groups. The FCM algorithm can produce its partitioning with PUR ≈ 0.95 at reasonably low values of the exponent ($m \leq 2$), but this benchmark strongly drops if we increase the exponent. Figure 3 exhibits the behavior of the GFPCM algorithm when applied to the WINE dataset under various circumstances.

The original FPCM algorithm ($\beta = 1$) seems to perform the same as FCM ($\beta = 0$). However, if we increase further the value of β, GFPCM tends to extend the domain where it can provide acceptable partitions to higher value

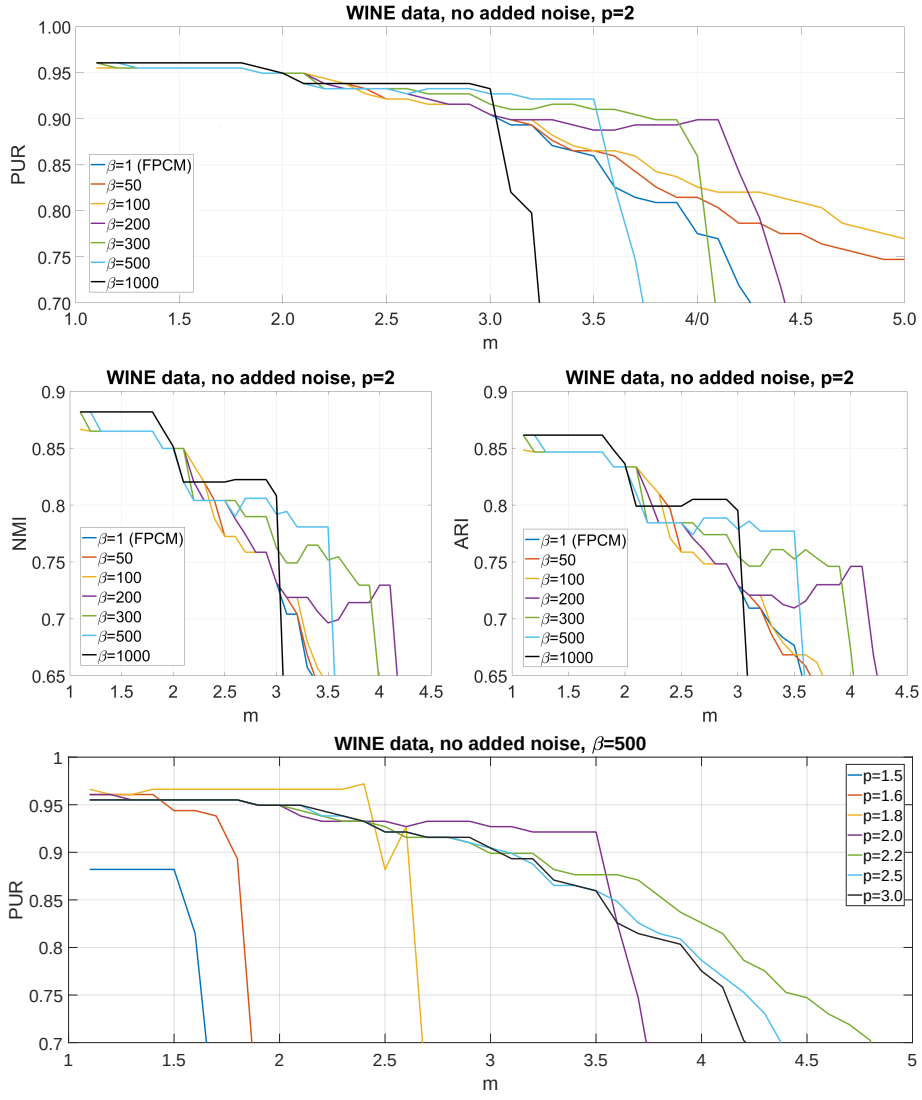


Figure 3: GFPCM benchmarks obtained on the WINE dataset in case of no added noise. Different curves relate on cases with various trade-off values of β at fixed $p = 2$, or at various values of p at fixed $\beta = 500$. Curves indicate up to which value of the fuzzy exponent m we have a stable solution in each scenario.

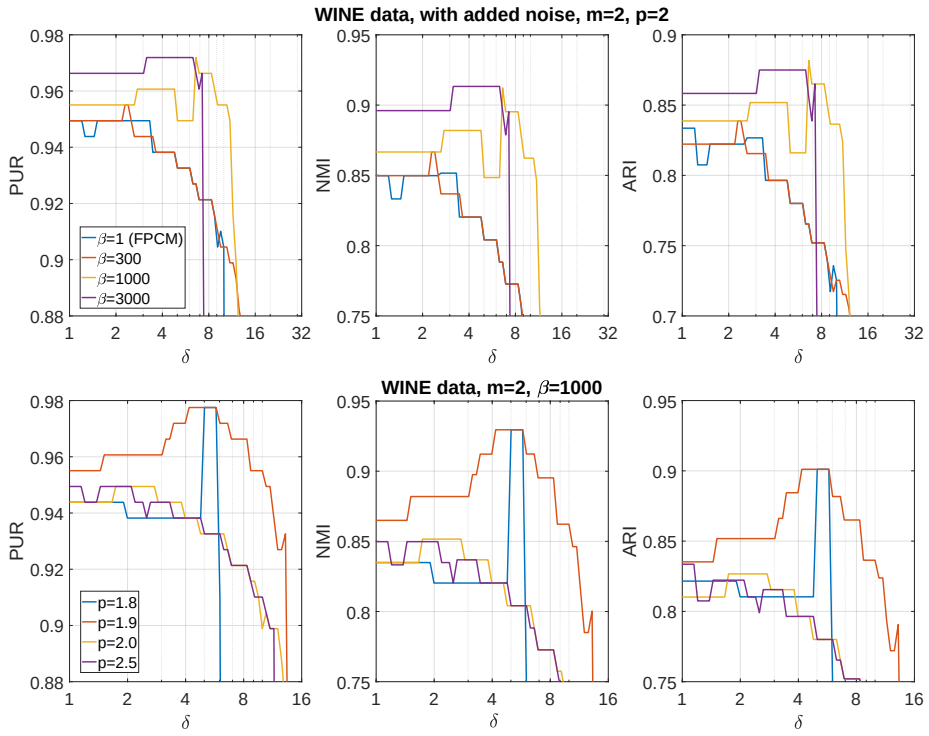


Figure 4: GFPCM benchmarks obtained on the WINE dataset, in case of an outlier added at $(\delta, \delta, \ldots, \delta)^{\mathsf{T}}$. The behavior of the GFPCM algorithm can be observed for scenarios of various trade-off values β at fixed $p = 2$ (upper row), and at various possibilistic exponent values p at fixed trade-off $\beta = 1000$. In all these tests, fuzzy exponent was set to $m = 2$.

of fuzzy exponent m. However, this effect saturates around $\beta = 100$. Above this value we can see that GFPCM provides finer partition than FCM or FPCM up to a certain limit of m, beyond which there is an abrupt drop in partition quality. The limit value of m seems to be in inverse proportion with trade-off value β. On the other hand, if we fix the trade-off parameter at a reasonably high value (e.g., in our case, $\beta = 500$), we can study the effect of various possibilistic exponents p upon the behavior of the algorithm. The possibilistic effect becomes stronger as p approaches 1. Experiments showed that reducing p far below 2 damages the clustering outcome. The best partitions in this case were obtained at $m \in [2.0, 2.5]$ and $p = 1.8$.

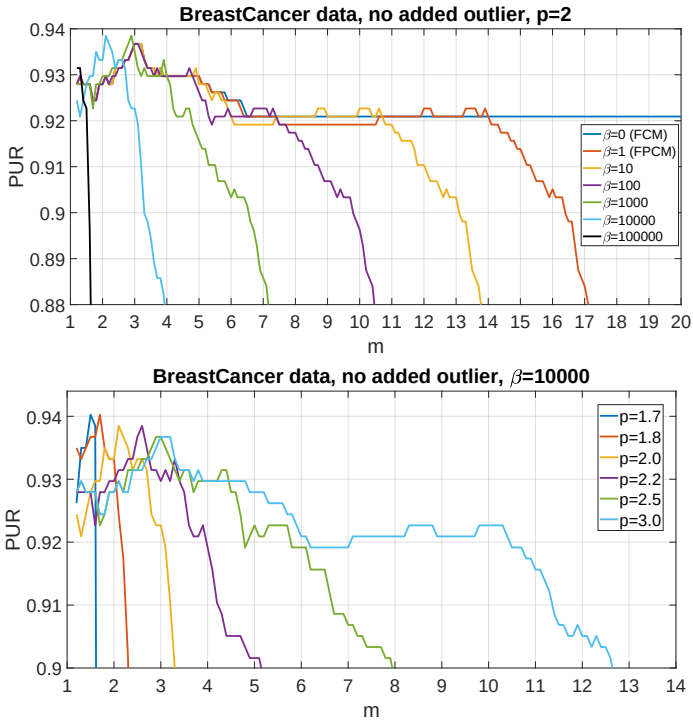


Figure 5: Purity benchmarks obtained on the BreastCancer dataset, with no added outliers. PUR is plotted against fuzzy exponent m, at fixed $p = 2$ and various trade-off values β (upper panel), and at fixed trade-off parameter $\beta = 10000$ and various possibilistic exponents p.

Figure 4 exhibits the behavior of the proposed clustering model in various scenarios, when applied to the WINE dataset with an added outlier whose position is controlled by the parameter δ. What we can observe is that there are settings which can extend the limit value of δ up to which we obtain a fine partitioning (e.g., $p = 1.9$ and $\beta = 1000$), while there are other settings which improve the purity of the obtained partition compared to FCM or FPCM if the outlier is very distant (e.g., PUR > 0.97 at $m = p = 2$ and $\beta = 3000$). Many of the phenomena are similar to the ones observed in IRIS data tests, but the best choice of β strongly depends on the data.

4.5 Tests using the BreastCancer dataset

The BreastCancer dataset presents vector data in a multi-dimensional setting. Each dimension was normalized before feeding the data to the clustering algorithms. Having only two clusters, we found it unnecessary to report NMI and ARI benchmarks, PUR contains all relevant information on the obtained partitions.

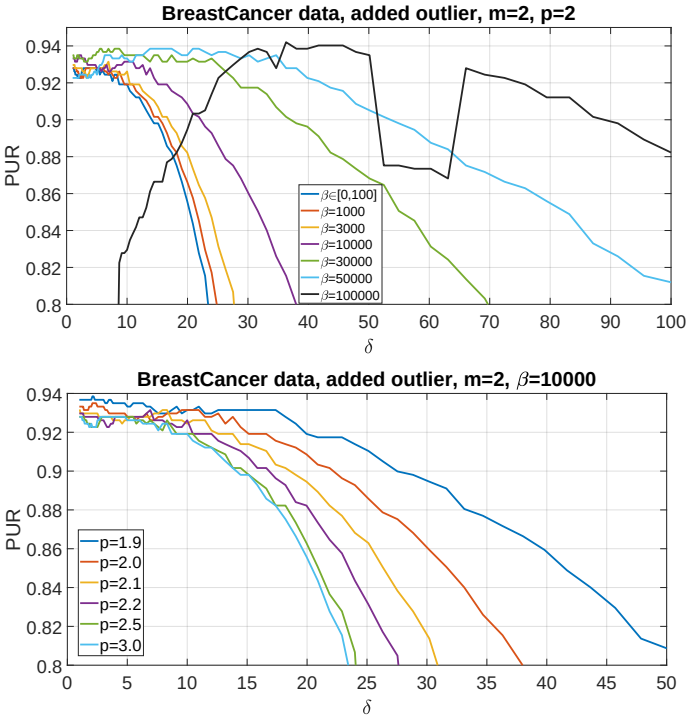


Figure 6: Purity benchmarks obtained on the BreastCancer dataset, with outlier added at $(\delta, \delta, \ldots, \delta)^T$ according to parameter $\delta > 1$. Graphs indicate how the position of the outlier influences the final partition produced by GFPCM, for various scenarios regarding parameters p and β. In all cases, fuzzy exponent was fixed at $m = 2$.

Figure 5 exhibits the clustering outcome of GFPCM at various settings, when applied to the BreastCancer dataset with no added outlier. The result of FCM obtained at $\beta = 0$ presents acceptable quality at any reasonable value of fuzzy exponent m, while the slightly modified version FPCM ($\beta = 1$) already

sets up a limit value for m above which we do not obtain fine partitioning. High PUR values are achieved at fuzzy exponents $m < 4$, especially when using trade-off parameter value $\beta \in [1000, 10000]$. If we investigate the effect of different possibilistic exponents upon the clustering outcome, the most convincing benchmarks are obtained at not much lower and not much higher than $p = 2$. Again, we need to mention that the reasonable and most frequented range of m is using values below 3.

Figure 6 presents how the parameter settings affect the clustering result in case of an added outlier, for various scenarios and outlier positions. When both exponents are fixed at $m = p = 2$, raising the trade-off parameter value extends the tolerance range of the outlier up to a certain extent. At $\beta < 100$ hardly any difference is visible between the behavior of GFPCM and FCM. Through changing the trade-off value up to $\beta = 50000$, GFPCM tends to tolerate the presence of an outlier at increasingly distant positions. However, at $\beta = 100000$ or higher, the algorithm no more produces fine partitions. If we fix $m = 2$ and $\beta = 10000$, and vary the possibilistic exponent value, we obtain similar phenomena to other datasets. GFPCM works best in the proximity of $p = 2$ or slightly below that, where it can provide partitions of better purity than FCM or FPCM. Larger values of p bring the performance of the algorithm close to FCM, which does not come as a surprise as with these settings we are weakening the possibilistic term in the objective function.

4.6 The limits of outlier tolerance

In case of all three datasets, we attempted to identify the maximum distance of the outlier defined by parameter δ, which is tolerated by the GFPCM algorithm without damaging the partition quality. Let us denote the limit value of δ by δ_{max}, and investigate how this value depends on the chosen dataset and the settings of the other three parameters m, p, and β. Further on, we denote by δ_{FCM} the maximum value tolerated by the FCM algorithm under the same circumstances (same m, but $\beta = 0$ and p irrelevant) where δ_{max} was established. The final partition was called acceptable if the PUR benchmark exceeded 0.88, 0.93, and 0.9 for the IRIS, WINE and BreastCancer datasets, respectively. These thresholds were established empirically.

A detailed summary of the obtained δ_{max} values is exhibited in Figure 7 and Table 2. Figure 7 shows us how the tolerated limit distance varies with trade-off parameter β when using various datasets and various settings for the fuzzy exponent, while the possibilistic exponent is fixed at $p = 2$. This is the main result of this study, as FPCM and GFPCM was meant to be an

extension of FCM to improve the way it handles outliers. A general thing that we can see in all these graphs is that the behavior of the GFPCM algorithm hardly changes below $\beta < 100$. Consequently, and not at all surprisingly, FCM ($\beta = 0$) and FPCM ($\beta = 1$) hardly manifest any visible difference.

However, if we raise the value of the trade-off parameter to a reasonable level, we may obtain a considerable extension of the tolerated noise range. When using the algorithm at low value of the fuzzy exponent, (e.g., $m = 1.5$), the ratio $\delta_{max}/\delta_{FCM}$ can be as high as 10. For higher values of the fuzzy exponent, the extension is approximately twofold. As an exception, in case of WINE dataset we do not achieve any improvement at $m > 2$.

Further on, we also need to remark that the best performance by GFPCM on various datasets is achieved at different values of the trade-off parameter β. This did not come as a surprise either, since β needs to compensate the disequilibrium caused by the difference between $\sum_i \sum_k u_{ik}^m$ and $\sum_i \sum_k t_{ik}^p$ within the objective function given in Eq. (18).

Table 2 presents a matrix of δ_{max} and their corresponding δ_{FCM} values, obtained at various settings of the two exponents m and p, and indicating the optimal β_{opt} trade-off value with which they were achieved. In this table, cases labeled as "not improving" mean that GFPCM does not bring any favorable change compared to FCM or FPCM, while "unstable" means that under those circumstances none of the FCM, FPCM or GFPCM can produce fine clustering outcome. This table suggests that using a possibilistic exponent in the proximity of $p = 2$ can significantly extend the tolerated distance of the outlier, with the condition that the necessary trade-off value is properly approximated.

5 Discussion

The main goal of this study was to eliminate some limitations of the FPCM algorithm, the behavior of which in its initial formulation strongly depended on the difference between the number input data vectors n and the number of clusters c. Whenever $n >> c$, the presence of the possibilistic part in the mixed partition is hardly observable. In our consideration, FPCM deserved an improvement because of the way it defined the possibilistic part of the mixture. It did not follow the conventional way indicated by the PCM algorithm [9] using the possibilistic penalty terms η_i ($i = 1, \ldots, c$). Instead of that, the typicality values represented by fuzzy membership functions t_{ik} ($i = 1, \ldots, c$;

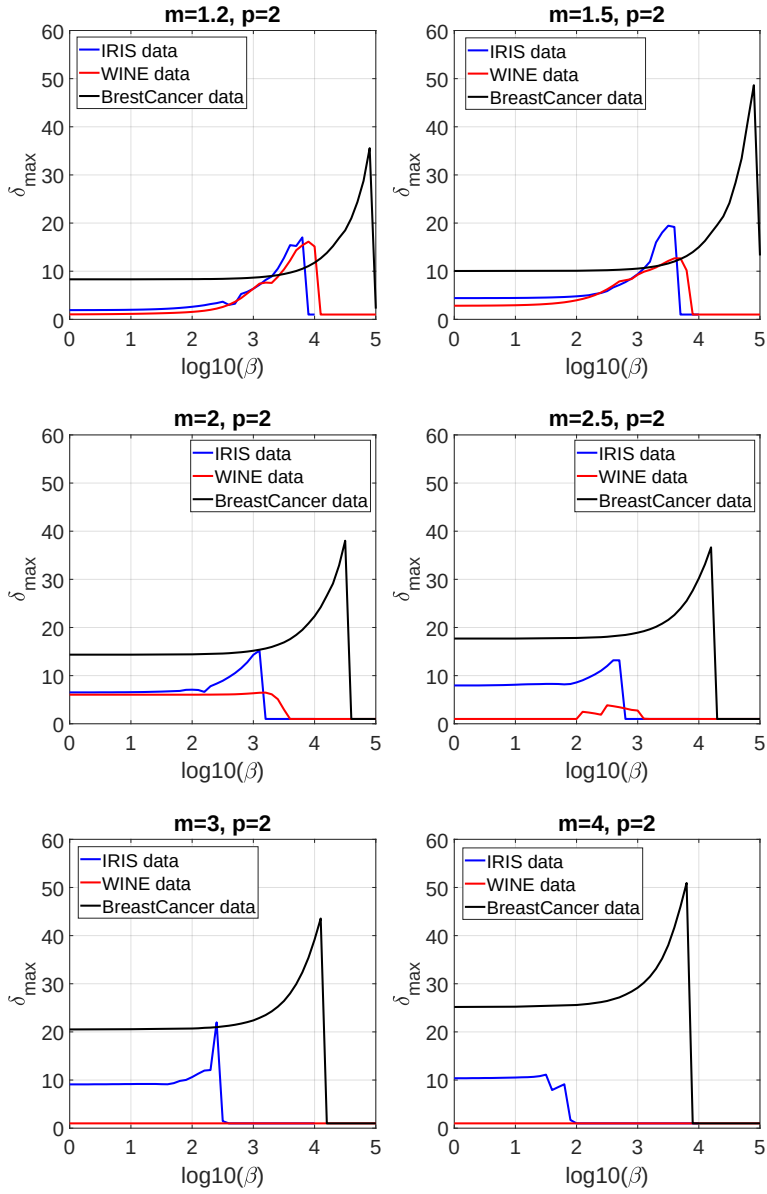


Figure 7: The evolution of the limit distance δ_{max} of the outlier plotted against the trade-off parameter β represented on a logarithmic scale, in case of various datasets and parameter settings.

Params		IRIS data			WINE data			BreastCancer data		
m	p	δ_{FCM}	δ_{max}	β_{opt}	δ_{FCM}	δ_{max}	β_{opt}	δ_{FCM}	δ_{max}	β_{opt}
1.2	1.8	1.91	16.14	1995	1.02	14.42	1995	8.32	24.10	15849
1.5	1.8	4.41	18.66	1000	2.82	12.85	1585	10.05	38.46	19953
2	1.8	6.50	22.91	501	6.04	9.02	501	14.35	37.76	10000
2.5	1.8	7.94	22.39	200	1.00	4.08	63	17.70	35.73	5012
3	1.8	9.10	16.14	100		unstable		20.51	38.37	3162
4	1.8	10.35	12.16	16		unstable		25.18	45.39	1585
1.2	2	1.91	17.02	6310	1.02	16.14	7943	8.32	35.56	79433
1.5	2	4.41	19.45	3162	2.82	12.68	3981	10.05	48.64	79433
2	2	6.50	15.21	1259	6.04	6.47	1585	14.35	38.02	31623
2.5	2	7.94	13.18	398	1.00	3.85	316	17.70	36.64	15849
3	2	9.10	21.98	251		unstable		20.51	43.55	12589
4	2	10.35	11.12	32		unstable		25.18	50.93	6310
1.2	2.2	1.91	14.03	10000	1.02	12.59	19953	8.32	18.84	100000
1.5	2.2	4.41	20.51	10000	2.82	10.16	12589	10.05	24.27	100000
2	2.2	6.50	17.26	1995	6.04	6.34	3981	14.35	37.07	100000
2.5	2.2	7.94	22.8	1585	1.00	3.48	1585	17.70	40.83	63096
3	2.2	9.10	22.28	794		unstable		20.51	42.46	39811
4	2.2	10.35	19.41	158		unstable		25.18	49.43	19953
1.2	2.5	1.91	7.13	10000	1.02	8.87	100000	8.32	10.00	100000
1.5	2.5	4.41	13.12	10000	2.82	8.83	79433	10.05	12.39	100000
2	2.5	6.50	16.14	10000	6.04	6.30	15849	14.35	18.03	100000
2.5	2.5	7.94	23.17	7943		unstable		17.70	23.17	100000
3	2.5	9.10	22.49	3981		unstable		20.51	28.97	100000
4	2.5	10.35	12.50	631		unstable		25.18	43.25	100000
1.2	3	1.91	2.40	10000	1.02	3.85	100000	8.32	not improving	
1.5	3	4.41	4.76	10000	2.82	6.98	100000	10.05	not improving	
2	3	6.50	7.06	10000	6.04	6.28	100000	14.35	not improving	
2.5	3	7.94	8.75	10000		unstable		17.70	not improving	
3	3	9.10	9.82	10000		unstable		20.51	not improving	
4	3	10.35	15.92	10000		unstable		25.18	not improving	

Table 2: The limit position of the outlier (δ_{max}) in case of various values of exponents m and p, and the value of trade-off parameter β_{opt} with which it is achieved.

$k = 1, \ldots, n$) were constrained probabilistically such a way, that they sum up to 1 with respect to each cluster.

The proposed modification in the objective function of FPCM, namely the introduction of trade-off parameter β enabled us to raise the strength of the possibilistic part of the mixed partition. The proposed clustering model (GFPCM) can be considered a generalization of FPCM, since FPCM is equivalent with the special case defined by $\beta = 1$, and FCM is obtained if $\beta = 0$ – the value of p is irrelevant in this case. Any other positive values of the trade-off parameter β lead to different partition mixtures, and consequently to different clustering algorithms.

The proposed clustering model uses three parameters, one more than FPCM. These are the fuzzy exponent m, the possibilistic exponent p, and the trade-off parameter β. To set the appropriate value of m, we may use the same criteria as we would use for FCM. For the general case, without knowing the properties of the input data, it is recommendable to keep m in the proximity of 1. There are several papers discussing the choice of this parameter, e.g., [4, 20, 22]. The experimental part of this study provided us enough evidence that the possibilistic exponent p should be chosen in the interval $p \in [1.8, 2.0]$. Lower values than that did not lead to convincing results in any of the circumstances. Higher values make the possibilistic part too weak, making the compensatory effect of GFPCM negligible.

For the trade-off parameter β, the ideal value seems to be proportional with $(n/c)^2$, but this remark needs further investigation. From the shape of the curves exhibited in Figure 7 we can easily realize that a careful prediction is needed for the choice of β, to place it below β_{opt}, but not very much below it. To provide a reliable approximation formula, a deeper investigation is needed, using several more datasets and experiments with multiple outliers as well. This is going to be the topic of a future study.

One of the relevant limitations of this study is the fact that we only tested the effect of a single outlier vector. Handling multiple outliers would have meant a lot more test cases, whose evaluation details hardly fit within the frame of such a study.

6 Conclusion

In this paper we proposed a generalization of the so-called fuzzy-possibilistic c-means algorithm, which in its original formulation had a strong limitation in the strength of the possibilistic part of the mixed partition. With the in-

troduction of a trade-off parameter we were able to amplify the phenomenon caused by the possibilistic extension of the fuzzy c-means objective function. The proposed clustering method was evaluated using three public datasets that contain real-life data. The proposed clustering model is capable to better handle datasets containing outlier data than its predecessors, namely the fuzzy c-means and the fuzzy-possibilistic c-means clustering algorithms.

Acknowledgements

This study was supported in part by the Collegium Talentum 2023 Programme of Hungary and the Consolidator Researcher Program of Óbuda University.

References

[1] S. Aeberhard, M. Floina, Wine, *UCI Machine Learning Repository* (1991) ⇒ 414, 415

[2] R. Andersen, Irises of the Gaspe Peninsula, *Bull. Amer, Iris Soc.* **59** (1935) 2–5. ⇒415

[3] J. C. Bezdek, *Pattern recognition with fuzzy objective function algorithms*, Plenum, New York (1981) ⇒405

[4] H. Choe, J.B. Jordan, On the optimal choice of parameters in a fuzzy c-means algorithm, *IEEE Int. Conf. on Fuzzy Systems (FUZZ-IEEE)*, 1992, pp. 349–354. ⇒429

[5] R.N. Davé, Characterization and detection of noise in clustering. *Pattern Recognition Letters*, **12**, 11 (1991) 657–664. ⇒405

[6] J.C. Dunn, A fuzzy relative of the ISODATA process and its use in detecting compact well separated clusters, *J. Cybern.* **3**, 3 (1974) 32–57. ⇒405

[7] B. Everitt, *Cluster analysis*, Chichester, West Sussex, U.K. (2011) ⇒414

[8] R.A. Fisher, Iris, *UCI Machine Learning Repository* (1988) ⇒414, 415

[9] R. Krishnapuram, J.M. Keller, A possibilistic approach to clustering, *IEEE Transactions on Fuzzy Systems* **1**, 2 (1993) 98–110. ⇒405, 408, 426

[10] T. O. Kvålseth, On normalized mutual information: Measure derivations and properties, *Entropy* **19**, 11 (2017) 613–114. ⇒414

[11] N.R. Pal, K. Pal, J.M. Keller, J.C. Bezdek, A mixed c-means clustering model, *IEEE Int. Conf. on Fuzzy Systems (FUZZ-IEEE)*, 1997, pp. 11–21. ⇒405, 410

[12] N.R. Pal, K. Pal, J.M. Keller, J.C. Bezdek, A possibilistic fuzzy c-means clustering algorithm, *IEEE Transactions on Fuzzy Systems* **13**, 4 (2005) 517–530. ⇒406, 410

[13] W. Rand, Objective criteria for the evaluation of clustering methods, *Journal of the American Statistical Association* **66**, 336 (1971) 846–850. ⇒414

[14] T.A. Runkler, A Convergence Study of the Possibilistic One Means Algorithm, *IEEE Int. Conf. on Fuzzy Systems (FUZZ-IEEE)*, 2023, art. no. 10309756, pp. 1–6. ⇒413

[15] E,H. Ruspini, A new approach to clustering *Information and Control* **15**, 1 (1969) 22–32. ⇒405

[16] L. Szilágyi, Fuzzy-Possibilistic Product Partition: a novel robust approach to c-means clustering, *Lect. Notes Comput Sci.* **6820** (2011) 150–161. ⇒406, 410

[17] L. Szilágyi, Robust spherical shell clustering using fuzzy-possibilistic product partition *Int. J. Intell. Syst.* **28**, 6 (2013) 524–539. ⇒410

[18] L. Szilágyi, S. M. Szilágyi, Generalization rules for the suppressed fuzzy c-means clustering algorithm, *Neurocomput.* **139** (2014) 298–309. ⇒417

[19] L. Szilágyi, Zs.R. Varga, S.M. Szilágyi, Application of the fuzzy-possibilistic product partition in elliptic shell clustering, *Lect. Notes Comput Sci.* **8825** (2014) 158–169. ⇒410

[20] V. Torra, On the selection of m for Fuzzy c-Means, *Conference of the International Fuzzy Systems Association*, 2015, pp. 1571–1577. ⇒420, 429

[21] W. Wolberg, O. Mangasarian, N. Street, W. Street, Breast Cancer Wisconsin (Diagnostic), *UCI Machine Learning Repository* (1995) ⇒414, 415

[22] K. L. Wu, Analysis of parameter selections for fuzzy c-means, *Pattern Recognition* **45**, 1 (2012) 407–415. ⇒429

[23] X. L. Xie, G. A. Beni, A validity measure for fuzzy clustering, *IEEE Trans. Pattern Anal. Mach. Intell.* **13**, 8 (1991) 841–847. ⇒417

[24] L. Zadeh, Fuzzy sets, *Information and Control* **8**, (1965) 338–353. ⇒405

Received: 15 November 2023 • Revised: December 6, 2023