UNIVERSIDADE FEDERAL DO RIO GRANDE DO SUL
INSTITUTO DE INFORMÁTICA
PROGRAMA DE PÓS-GRADUAÇÃO EM COMPUTAÇÃO

ALCIDES GONÇALVES LOPES JUNIOR

A Semantic Neighborhood Approach to Relatedness Evaluation on Well-Founded Domain Ontologies

Thesis presented in partial fulfillment
of the requirements for the degree of
Master of Computer Science

Advisor: Prof[a]. Dr[a]. Mara Abel

Porto Alegre
November 2019

CIP — CATALOGING-IN-PUBLICATION

Lopes Junior, Alcides Gonçalves

A Semantic Neighborhood Approach to Relatedness Evaluation on Well-Founded Domain Ontologies / Alcides Gonçalves Lopes Junior. – Porto Alegre: PPGC da UFRGS, 2019.

84 f.: il.

Thesis (Master) – Universidade Federal do Rio Grande do Sul. Programa de Pós-Graduação em Computação, Porto Alegre, BR–RS, 2019. Advisor: Mara Abel.

1. Knowledge-based measures. 2. Relatedness measures. 3. Semantic neighbors. 4. Ontological meta-properties. 5. Word sense disambiguation. I. Abel, Mara. II. Título.

ABSTRACT

In the context of natural language processing and information retrieval, ontologies can improve the results of the word sense disambiguation (WSD) techniques. By making explicit the semantics of the term, ontology-based semantic measures play a crucial role to determine how different ontology classes have a similar or related meaning. In this context, it is common to use semantic similarity as a basis for WSD. However, the measures generally consider only taxonomic relationships, which negatively affects the discrimination of two ontology classes that are related by the other relationship types. On the other hand, semantic relatedness measures consider diverse types of relationships to determine how much two classes on the ontology are related. However, these measures, especially the path-based approaches, have as the main drawback a high computational complexity to calculate the relatedness value. Also, for both types of semantic measures, it is unpractical to store all similarity or relatedness values between all ontology classes in memory, especially for ontologies with a large number of classes. In this work, we propose a novel approach based on semantic neighbors that aim to improve the performance of the knowledge-based measures in relatedness analysis. We also explain how to use this proposal into the path and feature-based measures. We evaluate our proposal on WSD using an existent domain ontology for well-core description. This ontology contains 929 classes related to rock facies. Also, we use a set of sentences from four different corpora on the Oil&Gas domain. In the experiments, we compare our proposal with state-of-the-art semantic relatedness measures, such as path-based, feature-based, information content, and hybrid methods regarding the F-score, evaluation time, and memory consumption. The experimental results show that the proposed method obtains F-score gains in WSD, as well as a low evaluation time and memory consumption concerning the traditional knowledge-based measures.

Keywords: Knowledge-based measures. Relatedness measures. Semantic neighbors. Ontological meta-properties. Word sense disambiguation.

Uma Abordagem Baseada em Vizinhos Semânticos para a Avaliação de Relacionamento em Ontologias Bem Fundamentadas

RESUMO

No contexto do processamento de linguagem natural e recuperação de informações, as ontologias podem melhorar os resultados das técnicas de desambiguação. Ao tornar explícita a semântica do termo, as medidas semânticas baseadas em ontologia desempenham um papel crucial para determinar como diferentes classes de ontologia têm um significado semelhante ou relacionado. Nesse contexto, é comum usar similaridade semântica como base para a desembiguação. No entanto, as medidas geralmente consideram apenas relações taxonômicas, o que afeta negativamente a discriminação de duas classes de ontologia relacionadas por outros tipos de relações. Por outro lado, as medidas de relacionamento semântico consideram diversos tipos de relacionamentos ontológicos para determinar o quanto duas classes estão relacionadas. No entanto, essas medidas, especialmente as abordagens baseadas em caminhos, têm como principal desvantagem uma alta complexidade computacional para sua execução. Além disso, tende a ser impraticável armazenar na memória todos os valores de similaridade ou relacionamento entre todas as classes de uma ontologia, especialmente para ontologias com um grande número de classes. Neste trabalho, propomos uma nova abordagem baseada em vizinhos semânticos que visa melhorar o desempenho das medidas baseadas em conhecimento na análise de relacionamento. Também explicamos como usar esta proposta em medidas baseadas em caminhos e características. Avaliamos nossa proposta na desambiguação utilizando uma ontologia de domínio preexistente para descrição de testemunhos. Esta ontologia contém 929 classes relacionadas a fácies de rocha. Além disso, usamos um conjunto de sentenças de quatro corpora diferentes no domínio Petróleo e Gás. Em nossos experimentos, comparamos nossa proposta com medidas de relacionamento semântico do estado-da-arte, como métodos baseados em caminhos, características, conteúdo de informação, e métodos híbridos em relação ao F-score, tempo de avaliação e consumo de memória. Os resultados experimentais mostram que o método proposto obtém ganhos de F-score na desambiguação, além de um baixo tempo de avaliação e consumo de memória em relação às medidas tradicionais baseadas em conhecimento.

Palavras-chave: Medidas baseadas em conhecimento, Medidas de relacionamento, Vizinhos semânticos, Meta-propriedades ontológicas, Desambiguação.

LIST OF ABBREVIATIONS AND ACRONYMS

BFO	Basic Formal Ontology
IC	Information Content
UFO	Unified Foundational Ontology
WSD	Word Sense Disambiguation

LIST OF FIGURES

Figure 2.1 The BFO hierarchy. 14
Figure 2.2 An example of the ontology structure after the value partition process. 17
Figure 2.3 Examples of is-a overloading and their solutions. 18
Figure 2.4 Example of an ambiguous class term (in red color) and its context window terms. 19
Figure 2.5 Example of the comparison of each sense of the target class term and their context window. 20

Figure 4.1 An example of the ontology classes and the relationships between them. 38
Figure 4.2 The path patterns of the direct paths. 39
Figure 4.3 The examples of the feature set of the traditional approach (left side) and with our approach (right side). 44

Figure 5.1 The evaluation time of the knowledge-based measures. 64
Figure 5.2 The memory consumption of the knowledge-based measures. 64

LIST OF TABLES

Table 5.1 The characteristics of each evaluated dataset. 50
Table 5.2 The evaluated knowledge-based measures based on Information Content. 51
Table 5.3 The F-score results of D1 using information content measures on WSD. 52
Table 5.4 The F-score results of D2 using information content measures on WSD. 52
Table 5.5 The F-score results of D3 using information content measures on WSD. 53
Table 5.6 The F-score results of D4 using information content measures on WSD. 53
Table 5.7 The evaluated knowledge-based measures based on features 54
Table 5.8 The F-score results of D1 using feature-based measures on WSD. 54
Table 5.9 The F-score results of D2 using feature-based measures on WSD. 55
Table 5.10 The F-score results of D3 using feature-based measures on WSD. 55
Table 5.11 The F-score results of D4 using feature-based measures on WSD. 56
Table 5.12 The evaluated knowledge-based measures based on paths 56
Table 5.13 Different strategies to use the local distance of a path pattern 57
Table 5.14 The F-score results of D1 using different local distance strategies on WSD. 57
Table 5.15 The F-score results of D2 using different local distance strategies on WSD. 58
Table 5.16 The F-score results of D3 using different local distance strategies on WSD. 58
Table 5.17 The F-score results of D4 using different local distance strategies on WSD. 58
Table 5.18 The F-score results of D1 using path-based measures on WSD. 59
Table 5.19 The F-score results of D2 using path-based measures on WSD. 60
Table 5.20 The F-score results of D3 using path-based measures on WSD. 60
Table 5.21 The F-score results of D4 using path-based measures on WSD. 61
Table 5.22 The evaluated hybrid knowledge-based measures 61
Table 5.23 The F-score results of D1 using hybrid measures on WSD. 62
Table 5.24 The F-score results of D2 using hybrid measures on WSD. 62
Table 5.25 The F-score results of D3 using hybrid measures on WSD. 62
Table 5.26 The F-score results of D4 using hybrid measures on WSD. 63
Table 5.27 Different strategies to existential dependence weights 65
Table 5.28 The F-score results of D1 using different existential dependence weights on WSD. 66
Table 5.29 The F-score results of D2 using different existential dependence weights on WSD. 66
Table 5.30 The F-score results of D3 using different existential dependence weights on WSD. 67
Table 5.31 The F-score results of D4 using different existential dependence weights on WSD. 67

CONTENTS

1 INTRODUCTION **9**
2 THEORETICAL FOUNDATION **12**
2.1 The Definition of Ontology and its Classifications **12**
2.2 The Basic Formal Ontology and the Strataledge® Ontology **13**
2.3 The Ontology Web Language and the Use of Value Partitions **16**
2.4 The Problem of "is-a" Overloading **17**
2.5 Knowledge-Based Word Sense Disambiguation **18**
3 RELATED WORKS **21**
3.1 The Semantic Proxies **21**
3.2 The Semantic Evidence from Structured Proxies **22**
3.3 State-of-the-Art of Semantic Measures based on Structured Proxies **24**
3.3.1 The Path-based Approaches 25
3.3.2 The Feature-based Approaches 27
3.3.3 The Information Content Approaches 28
3.3.4 The Hybrid Approaches 31
3.4 The Limitations of the Knowledge-based Approaches on Relatedness Evaluation **31**
3.4.1 The Path-Based Approaches 32
3.4.2 The Feature-Based Approaches 33
3.4.3 The Information Content Approaches 33
4 THE PROPOSAL TO SEMANTIC RELATEDNESS EVALUATION **34**
4.1 Building the Semantic Graph from a Well-Founded Ontology **34**
4.2 The Semantic Neighbors of an Ontology Class **36**
4.3 Using the Semantic Neighbors in Feature-Based Semantic Measures **43**
4.4 Using the Semantic Neighbors in Path-Based Semantic Measures **43**
4.4.1 The Local Distance 44
4.4.2 The Existential Dependence as a Piece of Semantic Proxy 45
4.4.3 Semantic Distance 46
5 EXPERIMENTS AND RESULTS **49**
5.1 The Analyzed Datasets **49**
5.2 Hypothesis 1: Similarity Measures are Ineffective to Distinguish Two Ontology Classes in Well-Founded Ontologies **51**
5.3 Hypothesis 2: Combining Taxonomic and Non-Taxonomic Relationships Improve the Distinction Capability of Feature-Based Measures **53**
5.4 Hypothesis 3: The Traditional Path-Based Measures are Effective to Distinguish Two Ontology Classes **56**
5.5 Hypothesis 3: The Traditional Path-Based Measures are Inefficient on Relatedness Evaluation **63**
5.6 Hypothesis 4: The Ontological Meta-Properties Improve the Distinction Performance of the Path-Based Measures **65**
6 DISCUSSION **68**
7 CONCLUSION **70**
REFERENCES **72**
APPENDIX A - THE WSD RESULTS WITH ANOTHER PARAMETER VALUES **77**

1 INTRODUCTION

In the last decades, several tasks have applied semantic measures in the document analysis, such as information retrieval (SIMOES et al., 2017; MUNIR; ANJUM, 2018; ASIM et al., 2019), natural language processing (ZHU; IGLESIAS, 2018; LASTRA-DÍAZ et al., 2019), cognitive science (ZHU; IGLESIAS, 2017; ZHANG; SUN; ZHANG, 2018), and artificial intelligence. In information retrieval domain (SIMOES et al., 2017; MUNIR; ANJUM, 2018; ASIM et al., 2019), the use of ontologies and semantic measures based on these ontologies are a hot topic and they are useful, for example, to the semantic word sense disambiguation, semantic indexing, semantic annotation, semantic queries, and the improvement of the precision and recall of the retrieval process. For information retrieval, the semantic measures with low evaluation performance become impractical for on-demand tasks.

Over the years, it is possible to note that knowledge-based semantic measures tend to explore more and more the semantics of the analyzed entities. One sign of this is the increasing application of the structured proxy semantics on the semantic measures, such as the WordNet[1], the SNOMED-CT[2], and a large number of biological and biomedical ontologies. After the popularization of the Open Biological and Biomedical Ontology (OBO) Foundry[3], many proposed ontologies have used top-level ontologies (e.g., UFO (GUIZZARDI, 2005); BFO (ARP; SMITH; SPEAR, 2015)) to provide well-founded definitions of their modeled entities. Besides the top-level ontologies, several ontology design patterns have been proposed to support common problems of modeling, reasoning, and representation of ontologies in computer-readable format. Although state-of-the-art semantic measures have not completely followed these ontological advances. The existence of efforts in this direction demonstrates the interest that the theme has aroused in the community.

A knowledge-based semantic measure is any mathematical function, algorithm, or approach to automatically calculate the degree of similarity or relatedness between two semantic entities based on a semantic evidence (e.g., the co-occurrence of the entities in corpora, the distance between the entities in an ontology, among others) extracted from a semantic proxy (e.g., a textual corpora, an ontology, a taxonomy, among others) (HARISPE et al., 2015). In this context, the literature orthogonally classifies the semantic

[1]https://www.w3.org/2006/03/wn/wn20/
[2]https://bioportal.bioontology.org/ontologies/SNOMEDCT
[3]http://www.obofoundry.org/

measures into semantic similarity and semantic relatedness measures. These two types differ according to the type of semantic relationships considered during their evaluation. The semantic similarity measures use only the taxonomic information to distinguish two ontology classes, while semantic relatedness measures use others relationships types besides taxonomic ones.

The knowledge-based similarity approaches have as the main drawback the inability to discriminate two ontology classes in situations where most of their relationships are not of the taxonomic type. On the other hand, the semantic relatedness measures do not present the disadvantage of similarity measures, but these measures, especially the path-based approaches, have high computational complexity on evaluation time. In this context, we refer to evaluation time as the time to calculate the value of similarity and relatedness between two ontology classes. Also, for both types of semantic measures, it is unpractical to store all values of similarity or relatedness between all ontology classes in memory, especially for large ontologies (DIEFENBACH et al., 2016).

In this work, we propose a novel strategy to compute the relatedness value between two ontology classes based on the semantic neighbors. These semantic neighbors is the set of related classes of a given class through a set of direct path patterns. We use an adaptation of the path patterns proposed by Hirst, St-Onge et al. (1998) to obtain our direct path patterns. Thus, with the semantic neighbors our main objective is to improve the performance of knowledge-based relatedness measures based on paths and features. In this work, when we talk about performance, we refer to the distinction capability, the evaluation time, and the memory required to perform the semantic measure.

To evaluate our proposal, we compare the original knowledge-based measure, as proposed in the literature, and their adaptation with our proposal on the word sense disambiguation (WSD) task, using an algorithm based on structured knowledge (PATWARDHAN; BANERJEE; PEDERSEN, 2003). We choose WSD because it evaluates the distinction capability of the knowledge-based measures (MCINNES; PEDERSEN, 2013). From this, as the input of the WSD algorithm, we use the set of sentences extracted from four different corpora on Oil&Gas domain, and the domain ontology for well-core description defined by Lorenzatti *et al.* (LORENZATTI et al., 2009) to support the Strataledge®[4] system. In this evaluation, we compare the F-score result of the different semantic measures approaches with nine different values for the size of the context window. Also, we compare their evaluation time and memory consumption during the

[4]Strataledge is a trademark of Endeeper Co. www.strataledge.com

relatedness evaluation.

The remainder of this document is structured as follows: Chapter 2 presents a brief background about ontologies and its classification, the Basic Formal Ontology (BFO), and some characteristics of well-founded ontologies. Chapter 3 describes the current state-of-the-art semantic measures, a deep analysis of these measures on relatedness evaluation, and the word sense disambiguation approach based on knowledge. Chapter 4 describes our proposal to find the semantic neighbors of an ontology class and how to use them into knowledge-based measures based on features and paths. Chapter 5 presents the experiments and results of the evaluation of the knowledge-based measures on word sense disambiguation. Chapter 6 describes the analysis of the word sense disambiguation results and the performance of our approach in this task. Finally, Chapter 7 concludes all the work presented in this document and define future directions.

2 THEORETICAL FOUNDATION

This chapter provides an overview of ontologies required to understand our contribution. In Section 2.1, we present the definition of ontology in the computer science context. In Section 2.2, we describe the Basic Formal Ontology (BFO) and the definitions of continuant entities and their sub-types. In Section 2.3, we present the basic notions of the Ontology Web Language (OWL) and the ontology design pattern of value partitions. In Section 2.4, we present the problem of "is-a" overloading and its impact on the ontology. Lastly, in Section 2.5, we describe the knowledge-based word sense disambiguation that we use to evaluate the distinction performance between the proposal of this work and the state-of-the-art on knowledge-based semantic measures.

2.1 The Definition of Ontology and its Classifications

In order to provide a better understanding of the propositions discussed throughout this proposal, we will first define a small set of terms that support our explanation. We use the term *class* (or concept) to refer to the mental abstraction of a portion of the reality. The term *instance* refers to the individual that extends this mental abstraction in space/time. The term *entity* is used to refer either class or instance.

In philosophy, the term *ontology* means a particular theory about the nature of things that exist (GUIZZARDI, 2005). In the last decades, ontologies have achieved great interest in the computer science community, especially in the areas of artificial intelligence, computational linguistics, and database theory. This interest is because ontologies offer a structured and unambiguous representation of the knowledge.

In the computer science context, there are many definitions of what an ontology is. Gruber (1993) defines an ontology as an explicit specification of a conceptualization. Borst (1997) extends the Gruber (1993) definition and defines an ontology as a formal specification of a shared conceptualization. After that, Studer, Benjamins and Fensel (1998) merge Gruber (1993) and Borst (1997) definitions and define an ontology as a formal and explicit specification of a shared conceptualization. Already Guarino (1998) considers ontology as a logical theory accounting for the intending meaning of formal vocabulary.

The concept of ontology can be subdivided according to the degree of generality (GUARINO, 1997):

- **Top-level ontologies:** this ontology type describes very general concepts that are independent of a particular problem or domain. In general, the concepts of top-level ontologies as proposed according to a set of ontological meta-properties deeply discussed in philosophy. This type of ontology makes possible the communication between domain ontologies providing a common ontological architecture (ARP; SMITH; SPEAR, 2015);
- **Domain ontologies:** this ontology type describes the vocabulary related to a specific domain (e.g., medicine, geography, or geology) as a structured representation of the entities and the relations between them. This type of ontology aims to support knowledge sharing and reuse (ARP; SMITH; SPEAR, 2015);
- **Task ontologies:** this ontology type has almost the same characteristics of a domain ontology, but this type of ontology focuses on a particular task and not a particular domain;
- **Application ontologies:** this ontology type describes the concepts depending on a particular domain or task. These concepts often correspond to roles played by domain entities while performing a certain activity.

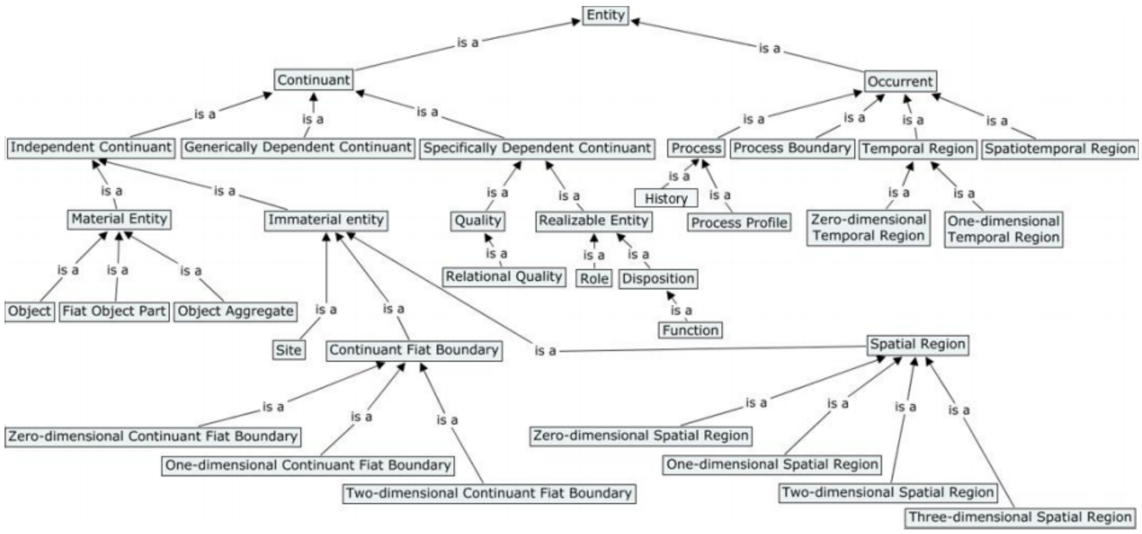
In this proposal, we are interested in the application of the knowledge-based semantic measures using *well-founded domain ontologies*, i.e., domain ontologies constructed based on definitions of top-level ontologies (e.g., the Basic Formal Ontology (ARP; SMITH; SPEAR, 2015), the Unified Foundational Ontology (GUIZZARDI, 2005), and others) or in the ontological meta-properties deeply discussed in philosophy (GUARINO; WELTY, 2009).

2.2 The Basic Formal Ontology and the Strataledge® Ontology

The Basic Formal Ontology (BFO) is a top-level ontology developed to support the integration of data obtained through scientific research and to support the interoperability of the multiple domain ontologies created in its terms (ARP; SMITH; SPEAR, 2015). Figure 2.1 shows the taxonomy structure of the BFO. The first subdivision of an entity in this top-level ontology regards the distinction between continuant and occurrent. In this work, we are interested only in continuant entities.

Continuants in BFO are entities that continue to exist over time while keeping their identities. Identity is an ontological meta-property and means how to recognize or

Figure 2.1: The BFO hierarchy.



Source: Arp, Smith and Spear (2015)

differentiate an entity over time (GUARINO; WELTY, 2009). These entities can gain and lose parts during their existence, but they have no temporal parts. Examples of continuants include a person, the qualities of the person (e.g., the person's weight) and the region of space occupied by a person at any given time. The subdivision of the BFO continuant is according to the type of existential dependence necessary for an entity to exist. An existentially dependent entity is one whose existence requires that a condition of a certain sort be met (CORREIA, 2008). In the BFO, there are three immediate subtypes of continuant: independent continuant, specifically dependent continuant, and generically dependent continuant.

The BFO defines as an independent continuant any continuant entity that is the bearer of qualities, i.e., this type of continuant have no existential dependence on other entities. The independent continuants maintain their identity and existence even by losing or gaining parts, dispositions, or roles, or even by changing their qualities. For example, consider the tomato is an instance of a Tomato class. The tomato instance can be left out in the sun and lose its moisture without ceasing to be the same instance of Tomato. The tomato instance may once have been green but is now red without ceasing to be the same instance of Tomato. The same occurs when a tomato instance is frozen, and thus loses its disposition to ripen, or if the chef selects a tomato instance, and thereby acquires the role of garnish to steak. Arp, Smith and Spear (2015) discuss more thoroughly all these examples.

The BFO defines as a specifically dependent continuant any dependent continuant that depends for its existence on some specific independent continuant that is its bearer.

Thus, a specifically dependent continuant is such that it cannot migrate from one bearer to another. Correia (2008) defines the specific dependence as the rigid necessitation that an entity requires the existence of a specific entity. For example, from an ontology built on OWL, this type of existential dependence holds between instances. An instance x is specifically dependent on instance y if whenever x exists, y must exist as well. Examples of specifically dependent continuants include the color of a tomato, the weight of a person, or the function of the heart to pump blood. The color of tomato could not exist without the tomato instance; the weight of a person could not exist without the person entity, and so on.

The non-migration axiom present in the definition of specifically dependent continuants is not true for all ontological entities. Some dependent entities are capable of such migration. For example, a PDF file is dependent, to be saved, on a storage device. However, the PDF file can be moved from one storage device to another, without ceases to exist. With this, the BFO incorporates the category of generically dependent continuant, defined as a continuant that is dependent on one or other independent continuants that can serve as its bearer. This type of dependence is a weaker type of existential dependence because, for example, from an ontology built on OWL, it holds between an instance x and a class C and indicates that instance x generally depends on any instance of class C.

In this work, we use a BFO version of the Strataledge® ontology (LORENZATTI et al., 2009). This ontology has 929 concepts that support the detailed and systematic description of sedimentary facies in drill cores. It includes all classes of lithologies, textures, structures, fractures, fossils, and other descriptive features about well-core description. Also, the Strataledge® ontology is originally proposed based on UFO (Unified Foundational Ontology) definitions. However, we use a BFO version of this ontology because of four factors: (a) BFO is extensively used in the biomedicine domain, where are proposed the most of research contributions using similarity or relatedness measures; (b) BFO presents an OWL version of its abstract concepts where it is possible to derive our domain classes; (c) BFO provides a centralized documentation of how to use its abstract concepts during the conceptual modeling of a particular domain; (d) in BFO, the distinction of each continuant type is made according to the type of existential dependence.

2.3 The Ontology Web Language and the Use of Value Partitions

The Ontology Web Language (OWL)[1] is a computational language to make the implicit knowledge explicit. In the OWL ontologies, we use the classes, individuals, and their respective properties to model the knowledge of a certain domain.

In the OWL ontologies, the property that describes the relationship between two entities is called *Object Property*. The OWL provides support to define the logical properties of the object properties (e.g., transitivity, asymmetry, reflexivity, and more). Also, the property that describes the relationship between an entity and its respective data values is called *Data Property*. Moreover, OWL provides support to define the restrictions of these two types of properties (domain, range, disjunctions, etc.).

A common requirement of creating ontologies in OWL is to represent descriptive features (e.g., qualities, attributes, or modifiers) as classes to add more semantics when dealing with this type of entity. With this, it is required to perform a value partition process[2] to achieve this requirement. This process is a design pattern to represent ontologies and restricts the range of possible values to an exhaustive list. In this work, the Strataledge® ontology uses the value partition process to represents the descriptive features about well-core description.

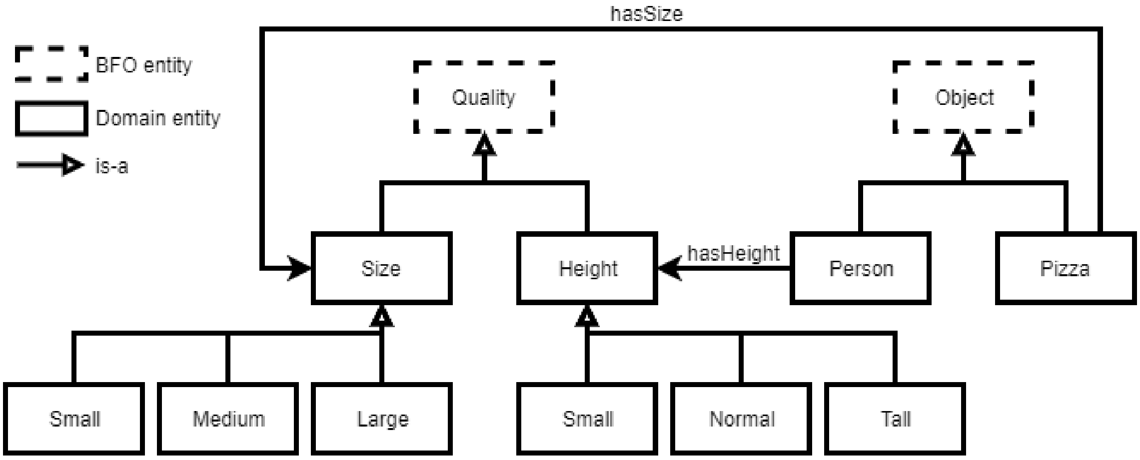
To exemplify the value partition process, consider the class *PizzaSizePartition* that restricts the range of possible values to *Small*, *Medium*, and *Large*. The value partition process, in the class level, consists on create a class *PizzaSizePartition* and, as its subclasses, the disjoint classes *Small*, *Medium*, and *Large*. After that, we create an object property called *hasSize*. The domain of this object property is the class *Pizza*, whose range is the value partition *PizzaSizePartition*. Finally, the classes *PizzaSizePartition*, *Small*, *Medium*, and *Large* are related through equivalent relationships.

The main impact of value partitions in semantic measures occurs when the classes that represent the descriptive features have the same taxonomic structure and are related to other ontology classes through non-taxonomic relationships. Consider the example presented in Figure 2.2, where we have two value partitions *Size* and *Height* in which the former is a *BFO:quality* of a *Pizza* and the later is a *BFO:quality* of a *Person*. From this, it is impossible to distinguish the two polysemic entities named Small, considering only the taxonomic structure of this ontology.

---

[1] https://www.w3.org/TR/2012/REC-owl2-overview-20121211/
[2] https://www.w3.org/TR/2005/NOTE-swbp-specified-values-20050517/

Figure 2.2: An example of the ontology structure after the value partition process.



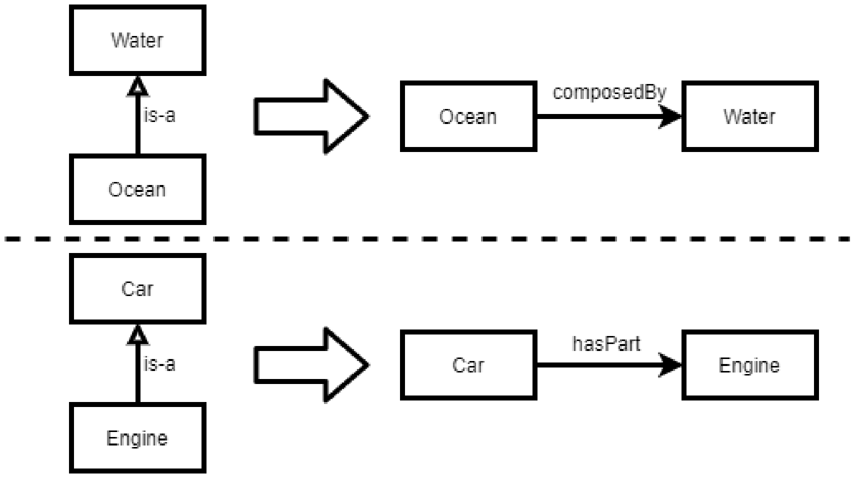
Source: The Authors.

2.4 The Problem of "is-a" Overloading

The use of domain ontologies is becoming increasingly common in many branches of science in the reflection of the increasing need to use computers for the handling of scientific data (ARP; SMITH; SPEAR, 2015). From this, many works in the literature are creating incompatible domain ontologies focused on their specific local needs (GUARINO, 1999). Many of these domain ontologies are created using the conceptual apparatus from some top-level ontology, but the philosophical discussions that define their general concepts, are ignored. From this, several semantic problems arise, among them the problem of the overload of "is-a" relations (or other types of hierarchical relations).

The basis of any ontology is its taxonomy (GUARINO, 1999), i.e., a set of classes related by hierarchical relations. One type of hierarchical relation is known as *is-a*. This relation means that the source class of the relationship is more specific than the target class. From this, the overload of *is-a* relationships usually occurs when it is considered the lexical relationship between the words that describe the ontology classes, rather than the ontological relationship between them. In the literature, many works exemplify and show how to solve the *is-a* overloading problem (GUARINO, 1999; GUARINO; WELTY, 2004; GUIZZARDI, 2005; ARP; SMITH; SPEAR, 2015). For example, Guarino and Welty (2004) use the ontological meta-properties of identity, essence, and unity, discussed in their work, to guide the construction of well-founded ontologies. Figure 2.3 presents examples of the "is-a" overloading on the left side and their solutions on the right side. The top-side example shows the violation of the unity principles and the misuse of the "is-a" relationship to represent a composition relation. Already the bottom-side example

Figure 2.3: Examples of is-a overloading and their solutions.



Source: Extracted from Guarino and Welty (2002).

shows the misuse of the "is-a" relationship to represent a part-whole relation. From this, it is possible to note that the entities no longer related through taxonomic relationships, but now they are related through non-taxonomic relationships. From this, if we consider only taxonomic relationships in an ontology containing these entities, the distinction becomes less precise because it is dependent on the taxonomic structure of the ontology.

2.5 Knowledge-Based Word Sense Disambiguation

Word sense disambiguation (WSD) is the task of automatically identifying the intended sense of an ambiguous term based on the context in which the term occurs (PATWARDHAN; BANERJEE; PEDERSEN, 2003; MCINNES; PEDERSEN, 2013). In the knowledge-based WSD, the semantic measures try to disambiguate two ontology entities named with the same term based on the ontology structure. Also, the WSD task is used to test the performance of semantic measures to distinguish two ontology entities (MCINNES; PEDERSEN, 2013).

In the knowledge-based WSD, Patwardhan, Banerjee and Pedersen (2003) propose to use a context window around the ambiguous class term. In this work, we use the class term to refer a term that names an ontology class, where an ambiguous class term represents the term that names two or more ontology classes.

The input of Patwardhan, Banerjee and Pedersen (2003) algorithm is a textual corpus, an ontology, and a knowledge-based semantic measure. From this corpora, the first step of this algorithm is to find all occurrences of the ambiguous class terms in this

Figure 2.4: Example of an ambiguous class term (in red color) and its context window terms.

rock	sandstone	**fold**	vein	fault

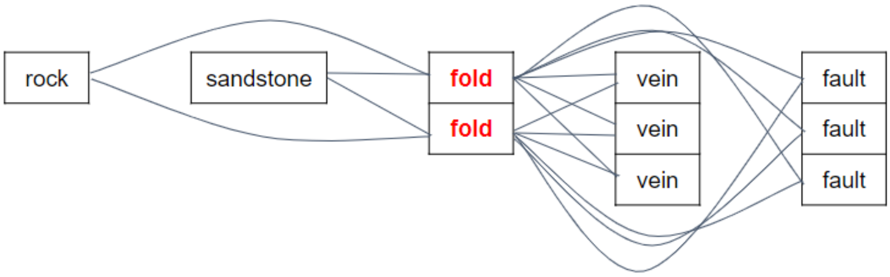
Source: The authors.

corpora. From each occurrence (target class term), the algorithm selects a context window. This context window represents a certain amount of class terms to the right and left of the target class term. In this step, to find the ambiguous class terms and their context window terms, are performed the stop-word removal and the stemming processes to remove the unnecessary words and to maintain only the stem of each word, respectively. For example, with a context window of size 2, we search for two class terms to the right of the target class term and two class terms to the left of the target class term. If the target term is at the beginning or end of a text, then are searched four class terms to the right of the target class term, or the left, respectively. For each sentence (the combination of the target class term and the context window), a domain expert evaluates the real sense of the target class term according to the context of the domain ontology used. Figure 2.4 shows the target class term *fold*, an example of ambiguous class term in the Strataledge® ontology. Also, with window size equals to 2, this figure shows the class terms that describe the context window of this ambiguous class term.

After finding all the class terms of a given context window size, the next step of the Patwardhan, Banerjee and Pedersen (2003) algorithm is to analyze the similarity or relatedness value between each sense of the target class term with each class term in the context window. In this process is used a semantic measure and, for each sense, are selected the sense with a higher sum of similarity or relatedness values between the sense and the context window terms. Figure 2.5 shows this analysis process. In this figure, the context window terms *vein* and *fault* have more than one sense. With this, is considered only the higher similarity or relatedness value between the senses of *vein* for each sense of *fold*.

After the completion of the analysis process in all sentences extracted from the corpora, it is possible to analyze the results of the Patwardhan, Banerjee and Pedersen (2003) algorithm, i.e., the output of the algorithm using some semantic measure. For each sentence, if the domain expert evaluates the target class term in some sense and the algorithm has as output the same sense, then it considered a true-positive result (TP). If the domain expert evaluates the target class term in some sense and the algorithm has as output a different sense, then it considered a false-positive result (FP). If the domain expert

Figure 2.5: Example of the comparison of each sense of the target class term and their context window.



Source: The authors.

does not evaluate the target class term in some sense of the ontology, and the algorithm finds a sense, then it considered a false-negative result (FN). If the domain expert does not evaluate the target class term in some sense of the ontology, and the algorithm does not find a sense, then it considered a true-negative result (TN). From this, it is possible to analyze the results in function of the precision, recall, and F1 (F-measure) scores. Equation 2.1 presents the formula to calculate the F-measure.

$$F1 = 2 * \frac{precision * recall}{precision + recall} \tag{2.1}$$

3 RELATED WORKS

In this chapter, we present the main approaches to evaluate the similarity or relatedness between two semantic entities. In this work, we focused on semantic measures based on structured knowledge resources (e.g., ontologies, taxonomies, and thesaurus).

We structure this chapter as follows: in Section 3.1, we review the sources of information used by the semantic measures, called semantic proxies. In Section 3.2, we describe how structured proxies are used to extract semantic evidence. This semantic evidence is used by semantic measures to evaluate the similarity or relatedness between the compared entities. In Section 3.3, we present the state-of-the-art in semantic measures based on structured proxies (or knowledge-based semantic measures), reviewing path-based, information content, feature, and hybrid approaches. Finally, in Section 3.4, we present a deep analysis of the current knowledge-based approaches on relatedness evaluation.

3.1 The Semantic Proxies

Semantic measures are widely used today to compare semantic entities such as units of language, instances, or concepts, according to information supporting their meaning (HARISPE et al., 2015). The main objective of these measures is to evaluate as closely as possible the human perception. With this, the semantic measures require a source of information. From this source of information, the semantic measures extract the semantic evidence to characterize the compared entities. This source of information is known as a semantic proxy. In the state-of-the-art are used two types of semantic proxies (HARISPE et al., 2015):

- **Corpora of texts:** corresponds to unstructured or semi-structured texts. These texts usually contain informal evidence of semantic relationships between units of language (terms). For example, consider corpora where the terms *car* and *engine* co-occur more than the terms *car* and *brain*. Intuitively, the term *car* is more related to the term *engine* than to the term *brain*. In the literature, mainly in the information content approaches based on corpora (see Section 3.3.3 for more details), the occurrences of the terms or the distribution of these terms in corpora are used as semantic evidence. Also, not only the units of language can obtain their relatedness values using this type of semantic proxy. For example, if the ontology entities have human-friendly names, they can also take advantage of the assumption that semantically related entities tend to co-occur in corpora;

- **Structured knowledge resources or structured proxies:** this proxy encompasses a broad range of knowledge models, from structured vocabularies to well-founded ontologies. These models explicitly present the knowledge about the entities that they define. From these models, semantic measures use their structures to extract semantic evidence of the compared entities. For example, in an ontology, intuitively, there is an explicit relationship between the entities *car* and *engine*, but not between the entities *car* and *brain*. With this, it is explicit that relatedness value between the entities *car* and *engine* is greater than between the entities *car* and *brain*.

3.2 The Semantic Evidence from Structured Proxies

From the availability of a semantic proxy, it is possible to extract semantic evidence used in the comparison between two semantic entities. This semantic evidence is expected to directly or indirectly characterize the meaning of compared entities (HARISPE et al., 2015). Harispe et al. (2015) define the semantic evidence as to any clue or indication based on semantic proxy analysis from which, often based on assumptions, a semantic measure is based. For example, in the semantic measures based on corpora, the semantic evidence is related to the degree of the co-occurrence, in this corpora, of the terms that describes the ontology classes. Also, the semantic evidence is dependent on the type of semantic proxy used. In this work, we are interested in semantic evidence from an ontology (a type of structured proxy) in the class level. With this, we do not review the semantic measures that use instances of an ontology class as semantic evidence. From this, the semantic measures presented in Section 3.3 use the following semantic evidence (or a combination of them) during the semantic similarity or relatedness evaluation:

- **The shortest path:** this is one of the most traditional semantic evidences to perform semantic similarity or relatedness. Rada et al. (1989) propose the use of the length of the shortest path between two ontology classes to evaluate the semantic distance between them. In this approach, the more similar two classes are, the smaller the semantic distance between them. Along with taxonomies, authors have proposed the use of the Lowest Common Ancestor (LCA) to improve the semantic distance calculation. The LCA is a function that returns the deepest common ancestor of the two analyzed classes. In this case, the semantic distance between the classes c_1 and c_2 equals to the sum of the differences (in modulus) of the depth between c_1 and $LCA(c_1, c_2)$, and c_2 and $LCA(c_1, c_2)$;

- **The most informative common ancestor (MICA):** this is another traditional semantic evidence to measure the similarity or relatedness. Resnik (1995) proposes that the MICA is the common ancestor of two ontology classes that have the maximum information content value (see Section 3.3.3 for more details). With this semantic evidence, the semantic similarity or relatedness increases according to the amount of information that these two classes have in common;
- **The common features:** this is one of the most classical semantic evidence proposals. Tversky (1977) proposes that the similarity or relatedness increases according to the number of features two ontology classes have in common. In the ontological point of view, it is considered as the set of features, the properties of an ontology class;
- **The depth of the ontology class:** corresponds to the distance from an ontology class to the root class of the ontology or the difference between their depths. This value informs the expressivity of an ontology class, i.e., the deeper a class in ontology, the more expressive it is;
- **The number of hyponyms:** this semantic evidence corresponds to the number of sub-classes or descendants of an ontology class. This value informs the expressivity of an ontology class, i.e., the greater the number of descendants an ontology class contains, the lower its expressiveness;
- **The number of hypernyms:** this semantic evidence corresponds to the number of super-classes, subsumers or ancestors of an ontology class. This value informs the expressivity of an ontology class, i.e., the greater the number of ancestors an ontology class contains, the greater its expressiveness;
- **The number of sibling classes:** this semantic evidence corresponds to the number of classes that have the same parent class within a given ontology class. This value informs the expressivity of an ontology class, i.e., the greater the number of sibling classes of an ontology class, the greater its expressiveness;
- **The number of leaf nodes:** this semantic evidence corresponds to the number of

descendant classes of an ontology class that does not have any descendants. This value informs the expressivity of an ontology class, i.e., the greater the number of leaf nodes an ontology class contains the lower its expressiveness. The total number of leaf nodes is also used to normalize the number of leaf nodes of an ontology class;

- **The diameter (or width) of the ontology:** this semantic evidence corresponds to the length of the longest shortest path between two ontology classes. This value informs the coverage of the ontology. In the semantic measures, the diameter of the ontology is used to normalize the shortest path between two ontology classes;
- **The depth of the ontology:** this semantic evidence corresponds to the maximal depth of a class in ontology. This value informs the degree of expressiveness/granularity of the ontology. In the semantic measures, the depth of the ontology is commonly used to normalize the depth of the class;
- **The number of ontology classes:** this semantic evidence corresponds to the total number of classes in ontology. This value informs the coverage of the ontology. In the semantic measure, the number of the ontology classes is commonly used to normalize the number of hyponyms, the number of hypernyms, the number of leaf nodes, or the number of sibling nodes.

3.3 State-of-the-Art of Semantic Measures based on Structured Proxies

In the literature, there are many tools, mathematical functions, algorithms, or approaches to automatically calculate the degree of similarity or relatedness according to structured proxy's semantics. The similarity measures use only the taxonomic information to distinguish two ontology classes, while relatedness measures use other relationship types besides taxonomic ones. In this context, we call knowledge-based semantic measure, any semantic measure based on structured proxies. From this, these measures usually use an ontology as a semantic graph in which is used to extract the semantic evidence.

Definition 1 (Semantic Graph). *Let $G = (V, E)$ be a directed graph that represents an ontology where V is a finite set of vertexes that represents the entities of this ontology, and E is a finite set of edges that represent the relationships between these entities. In an ontology that has only binary relationships, the tuple (c_i, r, c_j) describes an edge e, where $c_i \in V$ and $c_j \in V$ and c_i is the subject (or the source vertex), r is the* *predicate (or the relation) and c_j is the object (or the target vertex).*

In this section, we review the main proposals in the state-of-art of the knowledge-based semantic similarity and relatedness measures, more specifically, the semantic measures based on an ontology. In Section 3.3.1, we review the semantic measures based on path analysis. In Section 3.3.2, we describe the feature-based semantic measures. Finally, in Section 3.3.3, we present the semantic measures based on the information content of the evaluated entities. Finally, in Section 3.3.4, we present the union of the previous approaches, called hybrid approaches.

3.3.1 The Path-based Approaches

In this section, we present the main knowledge-based semantic measures that use the graph notion of the shortest path. These semantic measures estimate the similarity or relatedness value as a function of the length of the shortest path between two classes on the semantics graph that represents the ontology. In this approach, the similarity or relatedness value increases as the path length decrease. Like in graph theory, a sequence of edges (relationships) between two ontology classes constitute a path.

Definition 2 (Path). *Let $G = (V, E)$ be a directed graph. A path $P(c_i, c_j)$ between $c_i, c_j \in V$ is a sequence of edges $\{e_1, ..., c_k\} \in E$ with size n that relates the vertexes c_i and c_j such that there is no repetition of visited vertexes in the sequence.*

Since the path-based approaches use ontologies as a semantic graph, classical algorithms proposed in graph theory, such as the Dijkstra algorithm (DIJKSTRA, 1959), can be used to estimate the semantic distance between two ontology classes during the similarity or relatedness evaluation in path-based approaches. Overall, Equation 3.1 shows the function that calculates the shortest path between two entities in an ontology.

$$sp(c_i, c_j) = \min_{\forall P \in Paths(c_i, c_j)} \sum_{k=0}^{|P|} W(P_k) \tag{3.1}$$

Where $Paths$ is the set of all paths between c_i and c_j in the ontology, and $W(P_k)$ is the weight of k-th edge of path P.

Rada et al. (1989) define one of the first path-based measures in the literature. In their work, the length of the shortest path equals to the number of relationships between the evaluated classes. Rada et al. (1989) use a taxonomy to test their approach. However, their proposal also works as well considering other types of relationships besides taxonomic ones.

Take into account all ontology relationships in semantic distance calculation has as the main drawback the impact on the performance of the semantic measure. However, it is complex to decide which sequence of relationships is semantically incorrect to exclude them from the shortest path calculation. Hirst, St-Onge et al. (1998) classified the WordNet (MILLER, 1995) relationships into three categories: upward (UR), downward (DR), and horizontal (HR). The UR category represents the generalization relationship, DR represents the specialization relationship, and HR describes other relationship types. Also, in semantic distance evaluation, these authors consider the set of semantically correct path patterns through the relationship categories UR, UR-DR, UR-HR, DR-HR, UR-HR-DR, DR, HR-DR, and H. The notation "-" means the direction of the path, for example, in UR-DR paths a sequence of DR relationships follows a sequence of UR relationships.

The Rada et al. (1989) proposal fails to weigh all relationships of the ontology with the same weight. In the literature, some works use different weights regards the type of relationship between the evaluated classes or according to the semantic evidences of these classes. (ZHU; LI; SANCHO, 2017; CAI et al., 2018; QUINTERO et al., 2018). However, many of these weighting strategies are strictly dependent on a specific domain.

Besides the semantic evidence of the length of the shortest path, one the of most import semantic evidence is the definition of the lowest common ancestor (LCA) (WU; PALMER, 1994) (presented in Section 3.2). Wu and Palmer (1994) propose that the similarity or relatedness value between two classes as a ratio taking into account the depth of their LCA and the shortest path linking the evaluated classes.

From the addition of the depth in the path-based semantic measures, Leacock and Chodorow (1998) define that the similarity or relatedness value increases according to the negative logarithm of the shortest path, scaled by the double of the maximum depth of the ontology. Already Liu, Zhou and Zheng (2007) propose two versions of Wu and Palmer (1994) measure. The first version uses different weights for the shortest path and depth of ontology since the second version uses the first version in a non-linear function. In the path-based semantic measures that use non-linear functions, Li, Bandar and Mclean (2003) propose a semantic measure that combines the shortest path length and the depth of the LCA. Also, some measures use the depth to normalize the distance (LEACOCK; CHODOROW, 1998) or the distance to normalize the depth (WU; PALMER, 1994; Li; Bandar; Mclean, 2003; Liu; Zhou; Zheng, 2007). There are still methods that use both normalization methods (HAO et al., 2011).

In the literature, the use of the ontology depth or the depth of the LCA is a significant advance to distinguish two entities from an ontology, but some works go further. For example, Wei and Chang (2015) propose to use the ontology width in Wu and Palmer's measure to normalize its similarity or relatedness value. Since Jin et al. (2017) propose an extensive mathematical formulation in which combines the number of hyponyms, the maximum number of the ontology entities, the shortest path, the depth of the ontology and the LCA to calculate the semantic similarity between two ontology entities.

3.3.2 The Feature-based Approaches

In this section, we present the main proposals in the feature-based approaches. The feature-based approaches try to overcome the limitations of path-based approaches regarding the fact that the paths in a semantic graph do not necessarily represent uniform semantic distances (TVERSKY, 1977). In the feature-based approaches, the degree of overlap between the features of the two classes represents the similarity or relatedness value between them (TVERSKY, 1977; Rodriguez; Egenhofer, 2003; SÁNCHEZ et al., 2012). As mentioned in Section 3.2, the properties of a class in ontology represent the class features. In literature, there are considered the properties both the class relationships and the textual descriptions of the class.

Tversky (1977) proposes the first work in feature-based approaches. In this work, the common features tend to increase the similarity or relatedness value, and non-common ones tend to decrease it. In this semantic measure, a ratio model takes into account the weighted common and non-common features of compared classes. This weighting corresponds to the importance of common and non-common features in the similarity or relatedness evaluation. Equation 3.2 shows the Tversky (1977) semantic measure.

$$sim_{Tversky}(c_1, c_2) = \frac{f(\Psi(c_1) \cap \Psi(c_2))}{f(\Psi(c_1) \cap \Psi(c_2)) + \alpha * f(\Psi(c_1) - \Psi(c_2)) + \beta * f(\Psi(c_2) - \Psi(c_1))} \tag{3.2}$$

In Equation 3.2, the $\Psi(c_1)$ and $\Psi(c_2)$ are the set of features of c_1 and c_2, respectively; the signal $\cap$ means the intersection set between c_1 and c_2 features; the signal $-$ means the difference set between the two entities sets; α and β are two smoothing factors used to indicate the contribution of the difference set from c_1 and from c_2, respectively; f is a function that reflects the salience of a set of features (SÁNCHEZ et al., 2012).

The literature proposes some alternative methods to extend the initial definition of feature-based approaches. For example, Rodriguez and Egenhofer (2003) propose to use the classes feature with the synsets (or equivalent classes) and the features and synsets of the neighborhood of the evaluated classes. In this work, the neighborhood of a class corresponds to the set of ontology classes with the semantic distance less than a given radius, with the analyzed class. In other work, Petrakis et al. (2006) propose to use the synsets and the glosses (or textual descriptions) of the evaluated classes and their neighborhood (like in Rodriguez and Egenhofer (2003)). The authors consider that two classes are similar or related if their synsets and their glosses and the classes in their neighborhood are lexically similar.

The dependence of parameter tuning in the Tversky (1977) measure is one of its drawbacks. In this context, Rodriguez and Egenhofer (2003) proposed using the depth of the classes to avoid the dependence of the weights. Sánchez et al. (2012) presented a dissimilarity measure tanking into evidence the differences between the analyzed classes. In recent work, Likavec, Lombardi and Cena (2019) propose the use of a non-linear function in a variation of Tversky (1977) measure.

3.3.3 The Information Content Approaches

In this section, we present the main semantic measures based on information content. We split this section into two parts. Firstly, we present the information content (IC) models that have contributed to the advance of the state-of-the-art semantic evidence. Secondly, we present the knowledge-based semantic measures that use these IC models in the similarity or relatedness evaluation.

The works about semantic similarity evaluation based on the information content started from Resnik (1995) proposal. The information content value describes how specific and informative a class is (RESNIK, 1995; SECO; VEALE; HAYES, 2004). Resnik (1995) attempts to address this problem based on the estimation of the class probabilities through the frequency counting of term occurrences in a training corpus (LASTRA-DÍAZ; GARCÍA-SERRANO, 2015). This term is a textual representation of an ontology class in corpora.

$$freq(c) = \sum_{t \in Term(c)} Count(t) \tag{3.3}$$

where $Term(c)$ refers to the set of terms that describes the subsumed classes of the class c and $Count(t)$ to the frequency of the term t in the training corpus. Thus, Equation 3.4 describes the first IC model proposed by Resnik (1995).

$$IC_{Resnick}(c) = -log\left(\frac{freq(c)}{N}\right) \tag{3.4}$$

where $freq(c)$ returns the frequency of class c, and N is a constant value of the total number of observed nouns, except those which are not subsumed by any class in the structured knowledge resource, i.e., $freq(root)$.

The model proposed by Resnik (1995) presents the main disadvantage of the need of a training corpus. Thus, Seco, Veale and Hayes (2004) propose to use the intrinsic information of the classes in ontology to compute their IC value. In their work, Seco, Veale and Hayes (2004) present a comprehensive model based on the idea that the IC is inversely proportional to the number of hyponyms (or sub-classes) of a given class.

The main drawback of Seco, Veale and Hayes (2004) IC model is that the authors do not consider the depth of the analyzed class in ontology. With this, two classes with an equal number of hyponyms but very different depths, i.e., a different degree of expressiveness, can produce similar IC values (LASTRA-DÍAZ; GARCÍA-SERRANO, 2015). Thus, Zhou, Wang and Gu (2008) propose to use both the number of hyponyms (or sub-classes) and the depth of these classes in ontology. Already, Cai et al. (2018) propose to use the depth in a non-linear function.

Another drawback of Seco, Veale and Hayes (2004) IC model is that evaluating the set of hyponym is not appropriate to estimate the IC value of generic classes (classes with low depth in the ontology) because this set represents the classes that rarely occur in a corpus (SÁNCHEZ; BATET; ISERN, 2011). With this, Sánchez, Batet and Isern (2011) propose to use the number of leaves (classes that have no hyponyms) and the subsumers of a given class.

Like the Seco, Veale and Hayes (2004) IC model, the Sánchez, Batet and Isern (2011) IC model does not use the depth of the class as semantic evidence either. Thus, Meng, Gu and Zhou (2012) propose a new IC model to ensure that a class with more hyponyms have less IC value than the classes with fewer ones. Also, these authors propose that the deeper the class in ontology, the greater its IC value.

Sánchez and Batet (2012) note the incapability of the Seco, Veale and Hayes (2004) IC model to distinguish two leaf classes. Thus, the authors propose the concept of called *commonness*. In this concept, a ratio model takes into account the leaf nodes and the subsumers of the analyzed class and the same semantic evidence of the root class of ontology.

Adhikari et al. (2015) merge the Meng, Gu and Zhou (2012) and Sánchez and Batet (2012) IC models to solve their problems. The Meng, Gu and Zhou (2012) IC model considers that two classes that have a different number of subsumers but have the same hyponym structure and stay in the same depth have the same IC value. Since the Sánchez and Batet (2012) IC model does not distinguish two classes in different depths.

The IC models described above are only useful to evaluate the similarity between two ontology classes, i.e., they explore only taxonomic relationships of ontology. To solve this limitation, Pirró and Euzenat (2010) convert the Tversky (1977) feature-based model into an IC model in which allows evaluating other types of relationships besides taxonomical.

All the approaches described above are IC models. The IC and hybrid semantic measures use these IC models to evaluate the similarity between two ontology classes. In this context, Resnik (1995) proposes that the semantic similarity between two ontology classes depends on the amount of information two concepts have in common. From this, the most specific common ancestor (MICA) gives this shared information, i.e., the superclass that subsumes both evaluated classes with the highest IC value (RESNIK, 1995).

$$sim_{Resnik}(c_1, c_2) = IC(MICA(c_1, c_2)) \tag{3.5}$$

The similarity measure proposed by Resnik (1995) has some limitations, like two classes that have the same MICA possess the same similarity value. To solve this issue, Jiang and Conrath (1997), Sánchez and Batet (2011), and Lin (1998) proposed their similarity models. In these models, the similarity measure is an adaptation of path-based measures in terms of the IC value. For example, in Jiang and Conrath (1997) similarity measure, the distance between two ontology classes equals to the sum of their IC values decreased by twice the IC value of their MICA. Also, some authors attempt to explore this adaptation of path-based measures in terms of IC values through non-linear functions (CAI et al., 2018).

The IC-based measures also use the feature-based approaches in their definitions. For example, Pirró and Seco (2008) present a relatedness measure founded in the feature-based theory proposed by Tversky (1977). In this measure, the authors use the analogy that the Resnik (1995) similarity measure is approximately equal to the intersection of two feature sets. Already Pirró and Euzenat (2010) propose that the difference between two feature sets of two ontology classes is approximately equal to the difference between their IC values. Hence, unlike most information content measures, these measures based on the adaptation of feature-based theory explore other features through other relationships types besides the taxonomic ones (e.g., part-whole relationships).

3.3.4 The Hybrid Approaches

The hybrid approaches try to overcome the limitations of the knowledge-based semantic measure approaches by combining them. For example, Cai et al. (2018) and Zhu and Iglesias (2017) merge a path-based approach and an IC-based approach into a single semantic measure. In these works, each semantic measure approach performs its respective role, while the path-based measure evaluates how two ontology classes are closely related, the IC measure evaluates the specificity of the evaluated classes.

3.4 The Limitations of the Knowledge-based Approaches on Relatedness Evaluation

The main goal of the semantic measures is to estimate the strength of the semantic likeness between two analyzed entities to distinguish them. In the literature, many semantic measures use as a backbone the ontology because the ontologies represent a structured and unambiguous representation of the knowledge. However, by doing a careful study of the state-of-the-art semantic measures, we found that these measures are not practical, ideal, or flexible. These problems occur mainly in ontologies that avoid the "is-a" overloading or in ontologies that the non-taxonomic relationships provide stronger semantic evidence than taxonomic relationships. With this, when we apply the state-of-art-semantic measures in these situations, the relatedness evaluation is prejudiced because of three main reasons: the quality of the distinction between two ontology classes, the evaluation time on-demand, and the memory consumption to perform the relatedness evaluation. In this section, we present a detailed analysis of the disadvantages of each knowledge-based semantic measure approach presented in Section 3.3, and our hypotheses of each of these approaches in relatedness evaluation.

3.4.1 The Path-Based Approaches

We believe that the most coherent approaches to be used in relatedness evaluation are the path-based approaches. The main reason for this assertion is that path-based measures can be efficiently used to evaluate how closely two classes of the ontology are related (or dependent). However, the definitions of the path-based semantic measures presented in Section 3.3.1 have limitations when used in the relatedness evaluation, such as:

- **Weighting distinct relationships.** One of the main drawbacks of the state-of-the-art path-based approaches is to stipulate the weight value of a relationship. In some cases, this weight is according to the taxonomic features of the related classes. While in other cases, the weight value is dependent on the task performed by the ontology, i.e., the relationships have fixed weight values. The main limitation in these two cases is how to stipulate, independent of the domain, the weight value of a relationship based on some semantic evidence. For example, consider an ontology class *car* that has component a class *chassis* and the class *color* as one of its qualities. In the state-of-the-art, no approach can distinguish these two relationships with a well-founded semantic basis or with domain-independent semantics, such as a top-level ontology;
- **The possible paths.** In the path-based semantic measures, it is common to consider a structured proxy as a directed acyclic graph. Thus, it is common the use of the Dijkstra algorithm (DIJKSTRA, 1959) in the shortest path calculation. When using only ontology taxonomy, the path-based approaches have a low query cost because there are fewer possible paths between the two evaluated classes. However, when considering non-taxonomic relationships, the computational cost grows dramatically, making path-based approaches impractical in applications that evaluate the relatedness on-demand. One strategy to solve this problem is to pre-compute all relatedness values between all ontology classes and store them in memory at a quadratic memory cost. Another strategy is to limit the possible paths that aim to improve the Dijkstra algorithm by reducing the number of interactions (HIRST; ST-ONGE et al., 1998). Another problem in shortest path calculation is to consider the path through intransitive relationships. For example, consider an entity *heart* that is component of an entity *person*, and the entity *person* that is member of an entity *orchestra*, the path between *heart* and *orchestra* using *(heart, component of,*

person) and *(person, member of, orchestra)* relationships is incorrect because the violate the transitivity statements;

3.4.2 The Feature-Based Approaches

The feature-based approaches are proposed initially to evaluate the similarity between two ontology classes (TVERSKY, 1977). Thus the feature-based measures consider as the class features the set of classes related (or inferred) through taxonomic relationships. In this approach, the inference process plays a crucial role in finding an embracing set of features of an ontology class. In some works, authors perform the relatedness evaluation by using the overlapping of the glosses of the analyzed classes. However, this textual definition, besides being costly to provide, is often unavailable. In other works, authors consider as features of an ontology class, the classes related through non-taxonomic relationships. In the context of the features, the state-of-the-art feature-based approaches does not present a general rule to obtain class features. Thus, the critical issue of the feature-based approaches is how to get an embracing set of features to provide a precise distinction between two ontology classes.

3.4.3 The Information Content Approaches

As presented in Section 3.3.3, there exist two main methods in information content (IC) approaches: extrinsic and intrinsic. The former method requires the existence of corpora in which the information content value is extracted based on the frequency of a term that describes an ontology class in these corpora. However, this strategy is limited by the requirement of corpora, the problems of the text matching (e.g., the polysemy), and the low frequency of the generic classes of the ontology. The latter method solves these problems by extracting the semantic evidence from the taxonomic structure of the ontology. An exception is the Pirró and Euzenat (2010) proposal, where considers the part-whole relationships. Through our analysis, there is a trend in the exploration of new taxonomic features and composition of them. However, no matter which semantic evidence from the taxonomy a semantic measure uses, it is still insufficient in relatedness evaluation. This limitation is due to the inability of the approach to distinguish two classes that have the same taxonomic structure but have different non-taxonomic relationships.

4 THE PROPOSAL TO SEMANTIC RELATEDNESS EVALUATION

In this chapter, we present our proposal to improve the relatedness evaluation based on feature-based and path-based semantic measures. Our approach uses a set of relationship categories in order to promote the acquisition of the semantic neighbors of an ontology class. We use the semantic neighbors of an ontology class as its set of features, in feature-based semantic measures. Also, we use these semantic neighbors to compute the distance in semantic measures based on paths. Moreover, we describe how to use the ontological meta-property of existential dependence as semantic evidence. We use this semantic evidence to propose a novel domain-independent way to improve the distinction between ontology classes during relatedness evaluation.

This chapter is structured as follows: in Section 4.1, we describe how to transform an ontology represented in OWL (Ontology Web Language) into a semantic graph. In Section 4.2, we describe our proposal to obtain the semantic neighbors of an ontology class. In Section 4.3, we describe our proposal to use the semantic neighbors of an ontology class as the feature set in feature-based semantic measures. In Section 4.4, we describe our proposal to use the semantic neighbors in the semantic distance calculation, in path-based approaches.

4.1 Building the Semantic Graph from a Well-Founded Ontology

Before we can use an ontology in a knowledge-based semantic measure, the first step is to convert this ontology into a semantic graph. From this, in this work, we propose the following steps to perform the conversion of a well-founded ontology into a semantic graph.

- **Step 1 (Well-Founded Domain Ontology).** The main requirement to use our approach to relatedness evaluation is a well-founded domain ontology, i.e., a domain ontology constructed based on definitions of top-level ontologies or based on the ontological meta-properties of the modeled entities. As stated in Section 2.2, in this work, we recommend the BFO as the top-level ontology.

Assumption 1 (Inverse Relationship): since the edge e is oriented, we denote r^- the type of relation that has the inverse semantic of r and e^- the inverse semantic edge of the direct edge e. We consider that any relationship (c_i, r, c_j) implicitly implies

(c_j, r^-, c_i). For example, the hierarchical relationship *(sedimentary rock, sub-classof, rock)* implies the inverse hierarchical relationship *(rock, super-classof, sedimentary rock)*, considering *sub-classof*$^-$ $=$ *super-classof*. The same situation occurs when considering a sequence of edges. For example, the part-whole relationships *(cerebellum, part of, brain)* and *(brain, part of, person)* implies the inverse part-whole relationships *(brain, has part, cerebellum)* and *(person, has part, brain)*. The inverse relationship has not necessarily the same logical properties as the direct relation. The resulting graph G is strongly connected, i.e., any vertex c_i is reachable from any other vertex c_j, and vice versa.

- **Step 2 (Building the Class Hierarchy).** By default, OWL makes it possible to create class hierarchies through the *SubClassOf* axiom. The relationship *(c_2, $subClassOf$, c_1)* means that the class c_2 is more specific than the class c_1, and implies the inverse relationship *(c_1, superClassOf, c_2)* which means that the class c_1 is more generic than the class c_2 in the ontology class hierarchy. With this, the method creates two edges in the semantic graph for each relationship *(c_j, $subClassOf$, c_i)* founded in the ontology. Additionally, all edges created from *subClassOf* and *superClassOf* relationships preserve their logical properties of being irreflexive, asymmetric, and transitive;
- **Step 3 (Building the Initial Graph).** From the hierarchy tree, the OWL makes it possible to create other relationship types using the *OWLObjectProperty* with their respective logical properties. With this, for each *OWLObjectProperty* relation r between the classes c_i and c_j, the method creates two edges in the semantic graph, one from c_i to c_j through the relationship *(c_i, r, c_j)* and the another from c_j to c_i through the inverse relationship *(c_j, r^-, c_i)*. Additionally, all edges created from *OWLObjectProperty* relations, explicitly defined in the OWL ontology, preserve their respective logical properties;
- **Step 4 (Enriching the Initial Graph with the Ontological Meta-Properties).** The OWL provides support to use the logical properties of ontology relations (transitivity, reflexivity, symmetry, and more) but does not provide support to the ontological meta-properties of the ontology classes and relationships defined in this ontology. From this, we propose to use an OWL version of the BFO aiming to infer that, when a domain concept derives from some general concept of the BFO, this domain concept inherits all the ontological meta-properties of this general concept. For example, consider a domain ontology class c that derives the BFO *generic dependent continuant*, then c inherits all the ontological meta-properties of the *generic*

dependent continuant.

After performing the above steps, we have a semantic graph enriched with the ontological meta-properties of the classes of the well-founded ontology, where the vertexes represent the ontology classes and the edges represent the ontology relationships, as described in Definition 1.

4.2 The Semantic Neighbors of an Ontology Class

In this section, we propose several assumptions to support our strategy to find the semantic neighbors of an ontology class. These semantic neighbors are the set of direct related classes of a given ontology class. In this work, we store only the relationships between an ontology class and its semantic neighbors in memory. From this, we aim to propose an approach with low memory consumption and low evaluation time during the relatedness evaluation. Also, in this section, we describe our view about what makes two relationships distinct in well-founded domain ontologies, and the set of relationship categories presented in a domain ontology. Moreover, we explain how to use the relationship categories (or composition of them) to build the direct paths between an ontology class and its semantic neighbors.

Assumption 2 (Distinct Relationships): two ontological relationships are distinct if they relate two classes that have different types of dependence meta-property, or if they have different logical properties, or if the relations have different names. We are aware that certain relationships show different names, but they have the same semantics (synonymous relationships), but it is hard to distinguish these relationships in these cases. Besides, we classify the relationships of a domain ontology into four categories:

- **Equivalent category (EC):** includes any relationship that has logical properties of reflexivity, symmetry, and transitivity and conveys the idea that one class c_i is semantically equivalent to another class c_j. In path-based semantic measures, this type of relationship must have the greatest relatedness value as possible between two ontology classes;
- **Hierarchical category:** includes the relationships that convey the idea of hierarchy among the related classes (e.g., is-a, sub-classof, super-classof). We subdivide this category into **Upward (UC)** and **Downward (DC)** categories. The Upward category includes the relationships that start from a more specific vertex to a more

generic vertex (e.g., a relationship through *sub-classof* relation). The Downward category includes the relationships that start from a more generic vertex to a more specific vertex (e.g., a relationship through *super-classof* relation). Usually, the relationships in the hierarchical category have the logical properties the irreflexivity, asymmetry, and transitivity. This work considers only the classes of the same branch of the ontology taxonomy, during the analysis of which class is more generic or more specific than another;

- **Horizontal category (HC):** it includes any relationship that is not possible to classify in the categories described above, such as part-whole, characterization, constitution, among other relationships. We are aware that this category includes very different semantic relations. In future works, we will split this category in order to use Horizontal relationships in more specific ways.

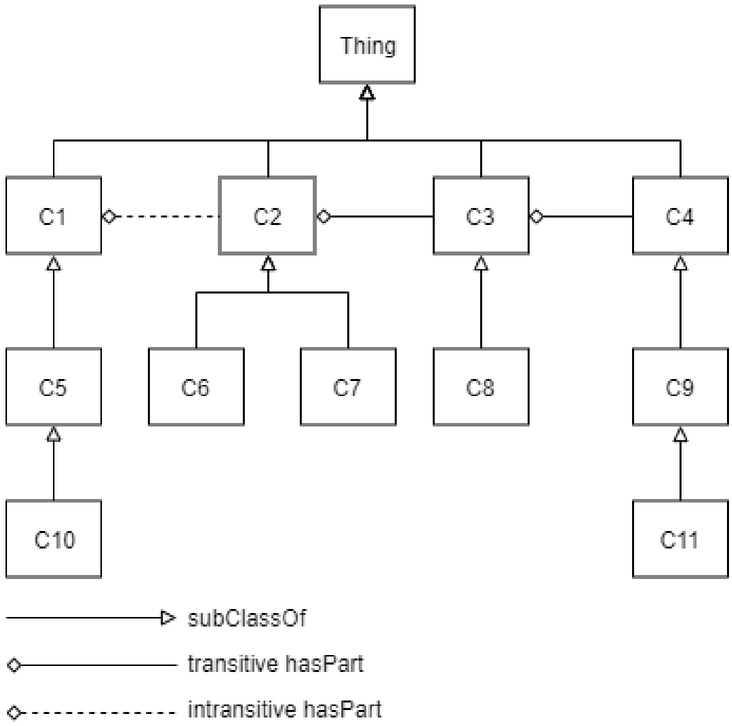
In this work, we use the relationship categories described in Assumption 2 to classify all the relationships of an ontology. From these categories, we propose to use a combination of these relationship categories in order to find a more embracing set of semantic neighbors of an ontology class. We call direct paths the path patterns resulting from these combinations.

Assumption 3 (Direct Path): we assume that there is a direct path between two ontology classes when a relationship of the categories described above (or composition of them) relate these two ontology classes. We consider that the set of the possible relationship compositions (or paths), in the semantic neighborhood discovery, are through the relationships of the following patterns: EC, UC, DC, HC, UC-HC, HC-DC, and UC-HC-DC. Also, the relationships in each relationship category need to be the same type, i.e., they not distinct (according to Assumption 2). The notation "-" means the direction of the path. For example, in UC-DC path, a sequence of DC relationships follows a sequence of UC relationships. In this path, the first relationship starts from the source class through a UC relationship, and the last relationship ends in the semantic neighbor through a DC relationship.

In Figure 4.1, we present an example of ontology classes and their relationships. In this example, the most abstract class is the class *Thing*, i.e., all the other ontology classes are sub-classes of the class *Thing*. Also, this example has three relationship types:

- The relationship *subClassOf* has as logical properties the irreflexivity, asymmetry, and transitivity, and has the relationship *superClassOf* as its inverse relationship,

Figure 4.1: An example of the ontology classes and the relationships between them.



Source: The authors.

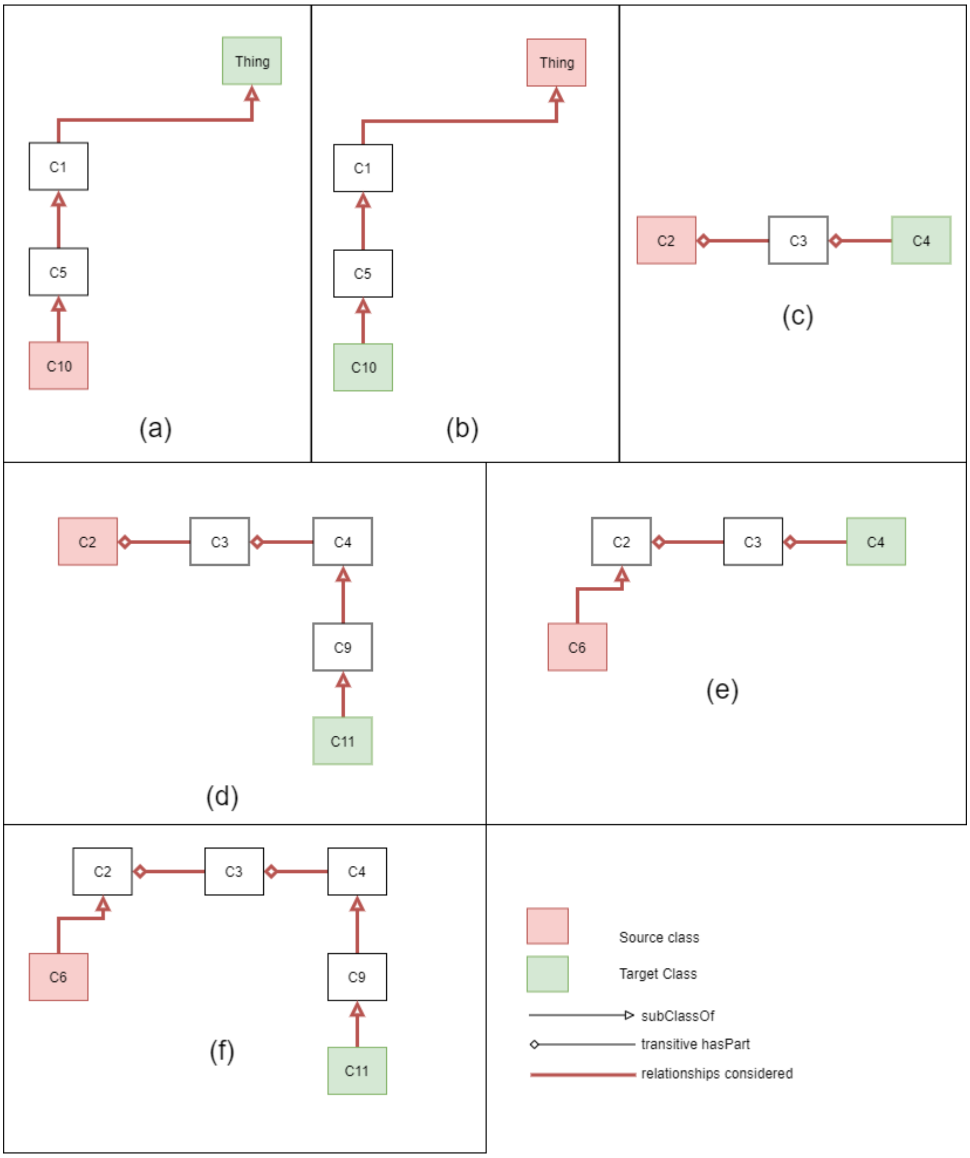
with the same logical properties. The category of the relationship *subClassOf* is UC, while the relationship *superClassOf* is DC;

- The relationship *hasPart*, with a solid line, has as logical properties the irreflexivity, asymmetry, and transitivity, and has the relationship *partOf* as its inverse relationship, with the same logical properties. The category of both relationships is HC;
- The relationship *hasPart*, with a dotted line, has the logical properties of irreflexivity, asymmetry, and intransitivity, and it has the relationship *partOf* as its inverse relationship, with the same logical properties. The category of both relationships is HC.

Using Figure 4.1 as an ontology example, we obtain the semantic neighbors through the direct paths following these approaches:

- **EC path:** it occurs when there are one or a sequence of EC relationships between the source class and the semantic neighbor (target class);
- **UC path:** the pattern (a) of Figure 4.2 exemplifies this path pattern, this direct path occurs when there is at least one or a sequence of UC relationships between the source class and the semantic neighbor (target class). Also, all equivalent classes (classes related through an EC relationship) of the target classes of this path are target classes of the source class. As described in Assumption 2, the relationships

Figure 4.2: The path patterns of the direct paths.



Source: The authors.

of this path carry the notion that the semantic neighbor is more generic than the source class. Since the UC relationships are usually transitive, the set of semantic neighbors comprehends all ontology classes that are more generic than the source class, including the class *Thing*. For example, the semantic neighbors of class *C9*, through this path, are the classes *C4* and *Thing*;

- **DC path:** the pattern (b) of Figure 4.2 exemplifies this path pattern, this direct path occurs when there is at least one or a sequence of DC relationships between the source class and the semantic neighbor (target class). Also, all equivalent classes (classes related through an EC relationship) of the target classes of this path are target classes of the source class. As described in Assumption 2, the relationships of this path carry the notion that the semantic neighbor is more specific than the source class. Since the DC relationships are usually transitive, the set of semantic neighbors comprehends all ontology classes that are more specific than the source class. For example, the semantic neighbors of class *C2*, through this path, are the classes *C6* and *C7*;

- **HC path:** the pattern (c) of Figure 4.2 exemplifies this path pattern, this direct path occurs when there is at least one or a sequence of HC relationships that are equal (based on Assumption 2) between the source class and the semantic neighbor. Also, all equivalent classes (classes related through an EC relationship) of the target classes of this path are target classes of the source class. We use the distinction provided by Assumption 2 because this path comprehends a large variety of semantic relations. For example, the ontology in Figure 4.1 has two relations called *hasPart*, but they have different logical properties (or they could have different meta-properties). Thus, if we consider that these relations are equal because they have the same name, then ontological inconsistencies arise. To exemplify this situation, consider that the *C1* represents the class *Orchestra*, the *C2* represents the class *Person*, the *C3* represents the class *Brain*, and the *C4* represents the class *Cerebellum*. If we consider the *Orchestra* as the source class and that the two *hasPart* relations are equal, then, through this path, the *Cerebellum* is a semantic neighbor of the *Orchestra*. However, this assertion is logically incorrect because the relationship between *Orchestra* and *Person* is intransitive;
- **HC-DC path:** the pattern (d) of Figure 4.2 exemplifies this path pattern. This direct path occurs when a DC path follows an HC path to relate the source class and the semantic neighbors. According to the pattern (d) of Figure 4.2, if a class, which is the target of some HC path has sub-classes, then all its sub-classes and all the equivalent classes of these sub-classes are semantic neighbors of the source class of the HC path. In this case, the relation between the source class and the semantic neighbor is the same as the HC path;
- **UC-HC path:** the pattern (e) of Figure 4.2 exemplifies this path pattern. This direct path occurs when an HC path follows a UC path to relate the source class and the semantic neighbor. According to the visual pattern (e) of Figure 4.2, if a class, which is the source of some HC path, have sub-classes, then all its sub-classes and all the equivalent classes of these sub-classes have the same semantic neighbors of the source class of the HC path. In this case, the relation between the source class and the semantic neighbor is the same as the HC path;

Input: class c_i
Output: The set of relationships SN from the class c_i to its semantic neighbors

```
SN ← ∅
for EC ∈ getEC(c_i) do
    add EC to SN
end
for DC ∈ getDC(c_i) do
    add DC to SN
end
for HC ∈ getHC(c_i) do
    add HC to SN
    c_j ← HC.target
    for DC ∈ getDC(c_j) do
        add HC to SN with DC.target
    end
end
for UC ∈ getUC(c_i) do
    add UC to SN
    c_j ← UC.target
    for HC ∈ getHC(c_j) do
        add HC to SN
        c_k ← HC.target
        for DC ∈ getDC(c_k) do
            add HC to SN with DC.target
        end
    end
end
return SN;
```

Algorithm 1: Algorithm applied to find the semantic neighbors of a class c_i through the direct paths.

- **UC-HC-DC path:** the pattern (f) of Figure 4.2 exemplifies this path pattern. This direct path occurs when a DC path follows a UC-HC path to relate the source class and the semantic neighbor. According to the pattern (f) of Figure 4.2, if two classes, which are source and target classes, respectively, of some HC path and they have sub-classes, then all the sub-classes of the target class and all the equivalent classes of these sub-classes are the semantic neighbors of all the sub-classes of the source class and all the equivalent classes of these sub-classes. In this case, the relation between the source class and the semantic neighbor is the same as the HC path;

Based on the descriptions of the assumptions 2 and 3, we propose Algorithm 1 to perform the semantic neighbor discovery. In this algorithm, we aim to find the set of relationships SN between the source class c_i and its semantic neighbors, where each of these relationships represents the characteristics of its respective direct path.

In Algorithm 1: at line 2, we discover the EC paths, and, in line 3, we add their relationships into SN; at line 5, we discover the DC paths, and, in line 6, we add their relationships into SN; at line 8, we discover the HC paths, and, in line 9, we add their relationships into SN; at line 11, we discover the HC-DC paths, and, in line 12, we add their relationships into SN; at line 15, we discover the UC paths, and, in line 16, we add their relationships into SN; at line 18, we discover the UC-HC paths, and, in line 19, we add their relationships into SN; at line 21, we discover the UC-HC-DC paths, and, in line 22, we add their relationships into SN. In this algorithm, the functions $getEC$, $getDC$, $getUC$, and $getHC$ discover the direct paths through the relationships of the same type, as proposed in Assumption 2.

Finally, applying Algorithm 1 in the ontology example, presented in Figure 4.1, the set of semantic neighbors of each ontology class are:

- **C1**: *Thing* through UC path; *C5*, and *C10* through DC paths; *C2* through HC path; *C6*, and *C7* through HC-DC paths;
- **C2:** *Thing* through UC path; *C6*, and *C7* through DC paths; *C1*, *C3*, and *C4* through HC paths; *C5*, *C10*, *C8*, *C9*, and *C11* through HC-DC paths;
- **C3:** *Thing* through UC path; *C8* through DC path; *C2*, and *C4* through HC paths; *C6*, *C7*, *C9*, and *C11* through HC-DC paths;
- **C4:** *Thing* through UC path; *C9*, and *C11* through DC paths; *C2*, and *C3* through HC paths; *C6*, *C7*, and *C8* through HC-DC paths;
- **C5:** *C1*, and *Thing* through UC paths; *C10* through DC path; *C2* through UC-HC path; *C6*, and *C7* through UC-HC-DC paths;
- **C6:** *C2*, and *Thing* through UC paths; *C1*, *C3*, and *C4* through UC-HC paths; *C5*, *C10*, *C8*, *C9*, and *C11* through UC-HC-DC paths;
- **C7:** *C2*, and *Thing* through UC paths; *C1*, *C3*, and *C4* through UC-HC paths; *C5*, *C10*, *C8*, *C9*, and *C11* through UC-HC-DC paths;
- **C8:** *C3*, and *Thing* through UC paths; *C2*, and *C4* through UC-HC paths; *C6*, *C7*, *C9*, and *C11* through UC-HC-DC paths;
- **C9:** *C4*, and *Thing* through UC paths; *C11* through DC path; *C2*, and *C3* through UC-HC paths; *C6*, *C7*, and *C8* through UC-HC-DC paths;
- **C10:** *C5*, *C1*, and *Thing* through UC paths; *C2* through UC-HC paths; *C6*, and *C7* through UC-HC-DC paths;

- **C11:** *C9*, *C4*, and *Thing* through UC paths; *C2*, and *C3* through UC-HC paths; *C6*, *C7*, and *C8* through UC-HC-DC paths.

4.3 Using the Semantic Neighbors in Feature-Based Semantic Measures

The traditional feature-based semantic measures consider the classes related directly or indirectly through the same relationship type as the feature set of an ontology class. From this, in our view, the set of semantic neighbors of an ontology class, presented in Section 4.2, provides a more embracing feature set than the traditional approach. We assume this because we obtain part of the semantic neighbors in the same way as the traditional approach, and we get the other part of the semantic neighbors by combining distinct relationships (taxonomic and non-taxonomic) through the relationship categories described in Section 4.2.

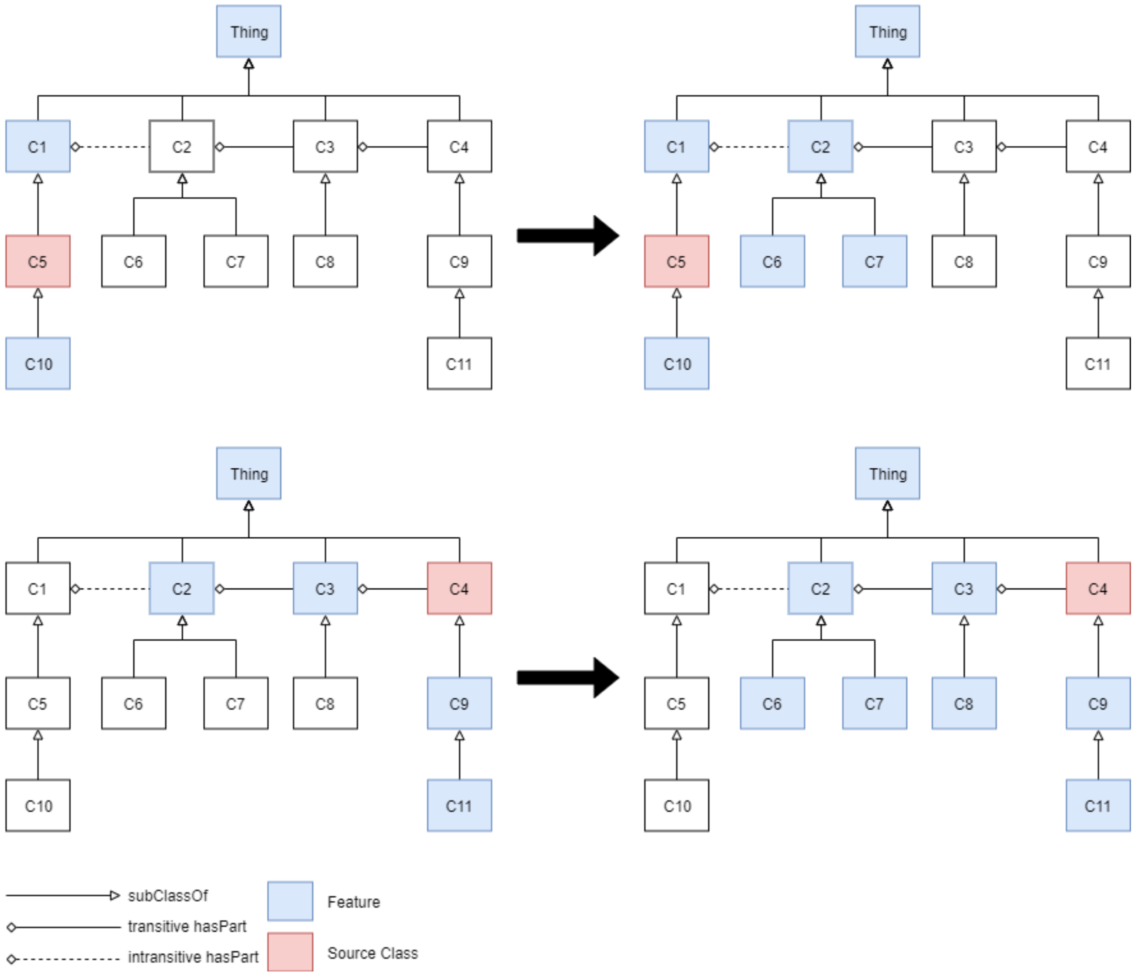
In the context of feature-based measures, we modify the traditional Ψ function of Tversky's measure (Equation 3.2). From this modification, the Ψ function returns the set of semantic neighbors of a given ontology class, according to Algorithm 1. In this case, Algorithm 1 returns the set of classes that are semantic neighbors of the source class, not the set of relationships between the source class and its semantic neighbors. Lastly, we do not modify the semantics of the intersection and difference functions of Tversky's measure, but we change the way to obtain the feature set.

Figure 4.3 presents two examples that describe the differences between the traditional approach to obtain the feature set (left side) and our approach (right side). In these examples, we use the classes *C4* and *C5* as the source classes. From this, using the traditional approach, the features of the class *C4* are the classes *Thing*, *C2*, *C3*, *C9*, and *C11*, and the features of the class *C5* are *Thing*, *C1*, and *C10*. Already, using our approach to obtain the feature set, the features of the class *C4* are the classes *Thing*, *C2*, *C3*, *C9*, *C11*, *C6*, *C7* and *C8*, and the features of the class *C5* are *Thing*, *C1*, *C10*, *C2*, *C6*, and *C7*. From this,

4.4 Using the Semantic Neighbors in Path-Based Semantic Measures

In this section, we describe some assumptions to use the semantic neighbors and the existential dependence meta-property in relatedness evaluation, using path-based mea-

Figure 4.3: The examples of the feature set of the traditional approach (left side) and with our approach (right side).



Source: The authors.

sures. Our main objective using the semantic neighbors presented in Section 4.2 in path-based measures is to limit the possible paths analyzed by these measures.

4.4.1 The Local Distance

The knowledge-based measures supported by paths use the semantic distance between the analyzed classes to perform the semantic relatedness evaluation. From this, these approaches use the length of the shortest path in the semantic graph, which represents the ontology to compute the semantic distance. In these approaches, the smaller the semantic distance, the greater the relatedness value between the two classes analyzed.

Assumption 4 (Local Distance): the local distance (LD) defines the semantic distance between a class and any of its semantic neighbors, discovered through a direct path. This semantic distance expresses how related these two classes are, i.e., the more related two classes are, the smaller the local distance value.

As described in Section 4.2, we find the semantic neighbors through the direct paths and store the relationships that represent these paths in memory. From this, these relationships also present information about the local distance between the related classes.

In this work, we use the local distance as a parameter of the final weight function of a relationship. From this parameter, we can stipulate the semantic closeness between two ontology classes, for example, regards their distance through the number of relationships in the direct path that link them. We do not define a general rule to this parameter because we can adjust it with different semantics. For example, consider the class *Person* is a subclass of the class *Mammal*, which is a subclass of the class *Animal*. If we consider that the classes *Person* and *Mammal* have the same relatedness value to the class *Animal*, then the local distance of the UC path between them is the same. However, if we consider that the class *Mammal* is more related to the class *Animal* than the class *Person*, then the local distance of the UC path between *Mammal* and *Animal* is lower than between *Person* and *Animal*.

4.4.2 The Existential Dependence as a Piece of Semantic Proxy

As described in Chapter 2, the ontological meta-properties are a set of properties deeply discussed in philosophy and adopted in top-level ontologies to explicit the meaning of their concepts, i.e., the ontological meta-properties help to distinguish and classify the generic concepts proposed in top-level ontologies, and the existential dependence is one of these ontological meta-properties.

In this work, we use the ontological meta-properties, more specifically the existential dependence, as a new semantic proxy where we can extract new semantic evidence. From this, we use the types of existential dependence to weight the local distance between an ontology class and its semantic neighbors. Thus, we aim to improve the distinction between the analyzed classes during the relatedness evaluation.

Assumption 5 (Existential Dependence Weight): we rely on the type of existential dependence as a way to weight a relationship that represents the direct path between the source class and any of its semantic neighbors. From this, we weight a relationship according to the type of existential dependence of its source class. Equation 4.1 shows the weighting process based on existential dependence meta-property.

$$EDW(r) = 1 - \begin{cases} S \\ G \\ I \end{cases} \tag{4.1}$$

where S is the weight of a relationship r with specific dependent source class; G is the weight of a relationship r with generic dependent source class; I is the weight of a relationship r with independent source class, with $0 < S, G, I < 1$.

Since the ontological meta-properties are a set of domain-independent properties, the process of weighting the relationships according to the existential dependence meta-property of their source class solves our problem in deciding weight factors that we can use in different domains. Also, this approach relies on the nature of the modeled classes in a domain ontology, i.e., we ignore the non-semantic aspects of the relationships (e.g., the name of the relationship, the number of related classes, among others).

From the local distance (presented in Section 4.4.1) and the weighting strategy based on the existential dependence type, described in this section, Equation 4.2 describes the final weight function of a relationship that we use in our semantic distance approach.

$$W(r) = EDW(r) * LD(r); \tag{4.2}$$

4.4.3 Semantic Distance

So far, we have discussed direct paths between a class and its respective semantic neighbors. However, in the semantic distance evaluation, many classes of ontology cannot be evaluated because there is no direct path between them, i.e., these classes are not neighbors. In order to solve this problem, we use the nearest common neighbor (NCN) as an intermediary class. From this, we evaluate the local distance of the indirect path between the two evaluated classes.

Assumption 6 (Common Neighbors): consider that the function SN of Algorithm 2 represents the output of Algorithm 1, sn_i the set of semantic neighbors of the class c_i, excluding the DC relationships, and sn_j the set of semantic neighbors of the class c_j, excluding the UC relationships, then the function $CN(sn_i, sn_j)$ (line 11 of Algorithm 2) returns the set of common neighbors between sn_i and sn_j. We exclude some types of relationships from sn_i and sn_j to maintain the restrictions of the direct path patterns described in Section 4.2.

Assumption 7 (Nearest Common Neighbor): we assume that the nearest common neighbor is the class whose value $distance$ is the smallest, within the set of common neighbors $ce \in CNs$ between sn_i and sn_j. The sets sn_i and sn_j are the neighbors of the classes c_i and c_2, respectively, and $distance = W(cc2c_i) + W(cc2c_j)$ (line 15 of Algorithm 2), where $cc2c_i$ is the relationship between cc and c_i and $cc2c_j$ is the relationship between cc and c_j (lines 13 and 14 of Algorithm 2, respectively).

Assumption 8 (Indirect Path): we assume that two classes c_i and c_j do not have a direct path between them but are related to the same common neighbor cc class and this common neighbor is the nearest common neighbor of both. In this case, the semantic distance between c_i and c_j equals to the sum of the weight value of the relationship between cc and c_i and the weight value of the relationship between cc and c_j.

Based on all the assumptions described in sections 4.2 and 4.4, we propose Algorithm 2 to compute the semantic distance between two ontology classes. In this algorithm, the semantic distance between two classes c_i and c_j assumes one value of the three possible situations:

- In the first situation, the two analyzed classes are equivalent (lines 2-4 of Algorithm 2). Thus, the semantic distance between them equals to 0;
- In the second situation, as discussed in Section 4.2, there is a direct path between the two analyzed classes (lines 5-9 of Algorithm 2). Thus, we retrieve from memory the semantic neighbors sn_i of the class c_i. If the class c_j is present in the relationships between c_i and its semantic neighbors sn_i, then the semantic distance equals to the weight value of the relationship between c_i and c_j;
- In the third situation, as discussed in this section, an indirect path exists between the two analyzed classes (lines 5 and 10-20 of the Algorithm 2). Thus, we retrieve from memory the semantic neighbors sn_j of the class c_j and we get the common neighbors between sn_i and sn_j. From the set of common neighbors, we search the nearest common neighbor cc between c_i and c_j. Finally, the semantic distance equals to the sum of the weight value of the relationship between cc and c_i and the weight value of the relationship between cc and c_j.

The main difference between our semantic distance approach and the traditional shortest path algorithm of Dijkstra (DIJKSTRA, 1959) is that we reduce the number of possible paths used in the shortest path calculation. Also, in our approach, two ontology classes are always related through a direct or indirect path.

Input: Source class c_i and Target class c_j
Output: Length value $minDistance$

```
minDistance ← MAXVALUE
if c_i ≡ c_j then
    minDistance = 0
else
    sn_i ← SN(c_i)
    if c_j ∈ sn_i then
        c_i2c_j ← c_i.getRelationshipTo(c_j)
        minDistance = W(c_i2c_j)
    else
        sn_j ← SN(c_j)
        CNs ← CN(sn_i, sn_j)
        for cc ∈ CNs do
            cc2c_i ← cc.getRelationshipTo(c_i)
            cc2c_j ← cc.getRelationshipTo(c_j)
            distance ← W(cc2c_i) + W(cc2c_j)
            if distance < minDistance then
                minDistance ← distance
            end
        end
    end
end
return minDistance;
```

Algorithm 2: The Semantic Distance Algorithm.

5 EXPERIMENTS AND RESULTS

In this chapter, we present the experiments and results performed using the adaptation of the knowledge-based measures based on paths and features with the semantic neighbors described in Chapter 4. In these experiments, we compare these adaptations with the original proposals presented in the literature. Also, we perform these experiments in the word sense disambiguation task (WSD) using the Patwardhan, Banerjee and Pedersen (2003) knowledge-based algorithm (described in Section 2.5). This task is extremely dependent on the distinction capability of the knowledge-based measures (MCINNES; PEDERSEN, 2013). As the input of WSD algorithm, we use the window size from 1 to 9, four different datasets on Oil&Gas domain to extract the sentences and the domain ontology of Strataledge® (LORENZATTI et al., 2009), described in Section 2.2, as the knowledge resource (semantic proxy).

This chapter is structured as follows: in Section 5.1, we describe the datasets from which we extracted the sentences where occur the ambiguous class terms of the Strateledge® ontology and our experimental hypotheses. In the remainder of this chapter, we present the experiments to prove our hypotheses.

5.1 The Analyzed Datasets

Since we will use the Patwardhan, Banerjee and Pedersen (2003) algorithm (described in Section 2.5) to disambiguate the polysemic class terms of the Strateledge® ontology, we extract the sentences from four different textual resources:

- **D1 (Polvo Project):** the Polvo Project is a Geology study developed by the Geoscience Institute of Federal University of Rio Grande do Sul in cooperation with Maersk Energia Company. The project comprises an integrated study of Petrology, Sedimentology, Seismic Sequence, Stratigraphy and Biostratigraphy, developed with data from the Polvo and Peregrino field area, Campos Basin, Brazil. In this corpora, we use only the final report document;
- **D2 (Scherer scientific articles):** this repository is a set of papers written by one of the stratigraphers that participated in the creation of the domain ontology of Strataledge®. The articles describe the analysis of facies architecture and the sequence stratigraphy of some fluvial and eolian reservoirs;

- **D3 (Sedimentary Geology journal):** this journal covers all aspects of sediments and sedimentary rocks at all spatial and temporal scales. The collection of articles must make a significant contribution to the field of study and must place the research in a broad context so that it is of interest to the diverse, international readership of the journal. This dataset includes four papers of the Sedimentary Geology journal (volume 379);
- **D4 (Sedimentology journal):** this journal publishes ground-breaking research across the spectrum of sedimentology, sedimentary geology, and sedimentary geochemistry. This dataset includes the papers of the Sedimentology journal (volume 66, issue 4).

During the sentence extraction, we do the string matching of the polysemic class terms of the Strataledge® ontology and the input documents of the corpora described above. In order to improve this matching, we perform the stop-word removal and stemming on the text of the input documents and the polysemic class terms. From this, we extract the sentences composed by the context window terms, where each term refers to a class in the ontology. After having all the extracted sentences, a geologist provided the groundtruth of the polysemic term of each sentence and classified the sentences if they are according to the same scale of analysis that the Strataledge® ontology covers. We do that because different scale of analysis share a lot of common geological terms. Also, some geological terms such as *massive*, *low*, *medium*, *high* are commonly used in different contexts. Thus, we aim to solve the polysemy problem only in Strataledge® context. Finally, as presented in Table 5.1, we extract a total of 1732 sentences. From these sentences, we consider 920 to perform the WSD because they are according to the domain and scale of geological analysis represented in the Strataledge® ontology.

Now that we have discussed the evaluated task, the domain ontology, and the details of the evaluated datasets, we present the list of hypotheses for which we designed the experiments. In these experiments, we evaluate the effectiveness of the knowledge-based measures regarding the distinction capability, and the efficiency regarding the memory consumption and the evaluation time.

Table 5.1: The characteristics of each evaluated dataset.

	D1	**D2**	**D3**	**D4**
Total no. of extracted sentences	325	433	288	686
No. of considered sentences	109	341	96	374

Source: The authors.

Hypothesis 1. The knowledge-based semantic similarity measures, i.e., semantic measures based on the taxonomic structure of the ontology are ineffective in distinguishing two classes in well-founded ontologies.

Hypothesis 2. The combination of taxonomic and non-taxonomic relationships improve the distinction performance in knowledge-based measures based on features.

Hypothesis 3. The knowledge-based relatedness approaches based on paths are effective in distinguishing two ontology classes, but they present a low performance on evaluation time.

Hypothesis 4. The ontological meta-properties improve the distinction performance of the path-based measures.

5.2 Hypothesis 1: Similarity Measures are Ineffective to Distinguish Two Ontology Classes in Well-Founded Ontologies

In this section, we present the experiments aiming to prove hypothesis 1. In these experiments, as presented in Table 5.2, we evaluate three different IC measures (Resnik (1995), Jiang and Conrath (1997), Lin (1998)) with four different IC models (Seco, Veale and Hayes (2004), Zhou, Wang and Gu (2008), Meng, Gu and Zhou (2012), Cai et al. (2018)) on WSD. Also, we keep the parameter values, presented in Table 5.2, in order to not generate a bias in one semantic evidence concerning the other. In this experiment, we evaluate only the IC-based measures because they use other semantic evidence besides the depth of the analyzed classes (commonly used in path-based measures), and the taxonomic distance proved ineffective to perform a good distinction based on Figure 2.2.

Table 5.3 presents the F-score results of WSD using the knowledge-based measures based on information content on D1 (see Section 5.1 for more details). In this ex-

Table 5.2: The evaluated knowledge-based measures based on Information Content.

ID	Semantic Measure	Parameter Values
IC1	Jiang and Conrath (1997) + IC Seco, Veale and Hayes (2004)	-
IC2	Jiang and Conrath (1997) + IC Zhou, Wang and Gu (2008)	$k = 0.5$
IC3	Jiang and Conrath (1997) + IC Meng, Gu and Zhou (2012)	-
IC4	Jiang and Conrath (1997) + IC Cai et al. (2018)	$\lambda = 0.5$
IC5	Lin (1998) + IC Seco, Veale and Hayes (2004)	-
IC6	Lin (1998) + IC Zhou, Wang and Gu (2008)	$k = 0.5$
IC7	Lin (1998) + IC Meng, Gu and Zhou (2012)	-
IC8	Lin (1998) + IC Cai et al. (2018)	$\lambda = 0.5$

Source: The authors.

periment, all the similarity measures achieved a poor F-score result on WSD, with F-score lower than 85%. The approach IC7 had the best F-score results of this experiment.

Table 5.3: The F-score results of D1 using information content measures on WSD.

	Window Size									
Semantic Measure	**1**	**2**	**3**	**4**	**5**	**6**	**7**	**8**	**9**	**AVG**
IC1	0.09	0.14	0.12	0.12	0.17	0.20	0.21	0.21	0.18	0.16
IC2	0.10	0.12	0.10	0.12	0.15	0.17	0.17	0.15	0.15	0.14
IC3	0.10	0.10	0.10	0.12	0.15	0.17	0.17	0.17	0.17	0.14
IC4	0.09	0.14	0.12	0.12	0.17	0.20	0.21	0.21	0.18	0.16
IC5	0.15	0.21	0.17	0.21	0.27	0.30	0.30	0.27	0.27	0.24
IC6	0.15	0.21	0.15	0.20	0.23	0.23	0.26	0.24	0.24	0.21
IC7	0.15	0.23	0.17	0.21	0.28	0.30	0.31	0.28	0.28	0.25
IC8	0.15	0.23	0.18	0.21	0.27	0.31	0.30	0.27	0.27	0.24

Source: The authors.

Table 5.4 presents the F-score results of WSD using the knowledge-based measures based on information content on D2 (see Section 5.1 for more details). In this experiment, all measures have poor F-measure results on WSD, with F-score lower than 85%. The approaches IC5, IC7, and IC8 had better F-score results in this experiment.

Table 5.4: The F-score results of D2 using information content measures on WSD.

	Window Size									
Semantic Measure	**1**	**2**	**3**	**4**	**5**	**6**	**7**	**8**	**9**	**AVG**
IC1	0.13	0.21	0.25	0.27	0.30	0.31	0.32	0.32	0.34	0.27
IC2	0.13	0.15	0.20	0.16	0.17	0.17	0.20	0.21	0.17	0.17
IC3	0.13	0.16	0.14	0.11	0.13	0.14	0.14	0.14	0.13	0.13
IC4	0.13	0.21	0.25	0.27	0.30	0.31	0.32	0.32	0.35	0.27
IC5	0.16	0.25	0.29	0.32	0.36	0.39	0.41	0.42	0.44	0.34
IC6	0.16	0.25	0.29	0.30	0.34	0.38	0.40	0.40	0.43	0.33
IC7	0.16	0.25	0.29	0.32	0.36	0.39	0.41	0.42	0.45	0.34
IC8	0.16	0.25	0.29	0.32	0.36	0.39	0.42	0.42	0.45	0.34

Source: The authors.

Table 5.5 presents the F-score results of WSD using the knowledge-based measures based on information content on D3 (see Section 5.1 for more details). In this experiment, all measures have poor F-measure results on WSD, with F-score lower than 85%. The approach IC7 had better F-score results in this experiment.

Table 5.6 presents the F-score results of WSD using the knowledge-based measures based on information content on D4 (see Section 5.1 for more details). In this experiment, the Lin (1998) IC measure obtained a better average F-score with different IC

Table 5.5: The F-score results of D3 using information content measures on WSD.

	Window Size									
Semantic Measure	**1**	**2**	**3**	**4**	**5**	**6**	**7**	**8**	**9**	**AVG**
IC1	0.10	0.17	0.17	0.21	0.25	0.29	0.29	0.29	0.32	0.23
IC2	0.10	0.14	0.12	0.17	0.21	0.24	0.24	0.24	0.29	0.19
IC3	0.10	0.14	0.12	0.17	0.21	0.22	0.22	0.24	0.29	0.19
IC4	0.10	0.17	0.17	0.21	0.25	0.29	0.29	0.29	0.32	0.23
IC5	0.22	0.27	0.27	0.32	0.37	0.41	0.40	0.40	0.41	0.34
IC6	0.22	0.27	0.25	0.29	0.39	0.40	0.41	0.43	0.41	0.34
IC7	0.22	0.27	0.25	0.30	0.39	0.43	0.43	0.43	0.45	0.35
IC8	0.22	0.27	0.27	0.32	0.37	0.41	0.40	0.40	0.41	0.34

Source: The authors.

models regards the other evaluated IC measures. However, these results are not expressive to the WSD task because the F-score is lower than 85%.

Table 5.6: The F-score results of D4 using information content measures on WSD.

	Window Size									
Semantic Measure	**1**	**2**	**3**	**4**	**5**	**6**	**7**	**8**	**9**	**AVG**
IC1	0.13	0.15	0.17	0.21	0.23	0.23	0.23	0.25	0.26	0.21
IC2	0.13	0.15	0.17	0.20	0.20	0.21	0.20	0.21	0.23	0.19
IC3	0.13	0.14	0.16	0.19	0.20	0.20	0.20	0.21	0.23	0.18
IC4	0.13	0.15	0.17	0.21	0.23	0.23	0.23	0.25	0.26	0.21
IC5	0.27	0.33	0.38	0.43	0.48	0.48	0.48	0.50	0.52	0.43
IC6	0.27	0.32	0.36	0.36	0.40	0.39	0.38	0.39	0.41	0.36
IC7	0.27	0.33	0.36	0.39	0.43	0.41	0.42	0.44	0.45	0.39
IC8	0.27	0.33	0.38	0.43	0.48	0.48	0.48	0.50	0.52	0.43

Source: The authors.

5.3 Hypothesis 2: Combining Taxonomic and Non-Taxonomic Relationships Improve the Distinction Capability of Feature-Based Measures

In this section, we present the experiments aiming to prove hypothesis 2. In these experiments, as presented in Table 5.7, we evaluate four different versions of Tversky (1977) measure and the feature-based approach proposed by Likavec, Lombardi and Cena (2019). Also, we adapt each of these measures with our approach of using the semantic neighbors in feature-based measures (described in Section 4.3), and we use the operator * to differentiate the original approaches from our adaptation. In the Tversky (1977) measure, we use different parameter values in order to benefit certain aspects of this measure

Table 5.7: The evaluated knowledge-based measures based on features

ID	Semantic Measure	Parameter Values
F1	Tversky (1977)	$\alpha = 0, \beta = 1$
F2	Tversky (1977)	$\alpha = 1, \beta = 0$
F3	Tversky (1977)	$\alpha = 1, \beta = 1$
F4	Tversky (1977)	$\alpha = 0.5, \beta = 0.5$
F5	Likavec, Lombardi and Cena (2019)	-

Source: The authors.

(as described in Equation 3.2) during the relatedness evaluation.

Table 5.8 presents the F-score results of WSD using the knowledge-based measures based on features on D1 (see Section 5.1 for more details). Overall, the results of this experiment are not expressive to the WSD task, with F-score lower than 85%. However, it is possible to note the improvement in the distinction of two ontology classes regards the traditional approaches to find the features of the classes. In this experiment, all adapted feature-based measures had a better F-score result in comparison to the original approaches. This improvement is true for all tested window size variations.

Table 5.8: The F-score results of D1 using feature-based measures on WSD.

	Window Size									
Semantic Measure	**1**	**2**	**3**	**4**	**5**	**6**	**7**	**8**	**9**	**AVG**
F1*	**0.58**	**0.69**	**0.70**	**0.73**	**0.73**	**0.72**	**0.75**	**0.73**	**0.75**	**0.71**
F1	0.09	0.18	0.18	0.18	0.24	0.27	0.27	0.30	0.31	0.23
F2*	**0.52**	**0.61**	**0.63**	**0.57**	**0.58**	**0.61**	**0.62**	**0.63**	**0.58**	**0.59**
F2	0.20	0.21	0.21	0.23	0.26	0.27	0.27	0.27	0.27	0.24
F3*	**0.58**	**0.69**	**0.69**	**0.71**	**0.70**	**0.70**	**0.72**	**0.70**	**0.72**	**0.69**
F3	0.20	0.23	0.20	0.23	0.27	0.30	0.30	0.30	0.30	0.26
F4*	**0.58**	**0.68**	**0.66**	**0.70**	**0.68**	**0.69**	**0.69**	**0.69**	**0.69**	**0.67**
F4	0.20	0.23	0.21	0.23	0.26	0.27	0.27	0.27	0.27	0.24
F5*	**0.54**	**0.60**	**0.63**	**0.65**	**0.65**	**0.68**	**0.69**	**0.67**	**0.68**	**0.64**
F5	0.20	0.23	0.23	0.24	0.27	0.30	0.30	0.30	0.30	0.26

Source: The authors.

Table 5.9 presents the F-score results of WSD using the knowledge-based measures based on features on D2 (see Section 5.1 for more details). In this experiment, all adapted feature-based measures had a better F-score result in comparison to the original approaches. This improvement is true for all tested window size variations. Also, the adapted Tversky (1977) measures had obtained expressive results, with F-score equals or greater than 85%, with a window size greater than 4, and the adapted Likavec, Lombardi and Cena (2019) measure with a window size of 9.

Table 5.9: The F-score results of D2 using feature-based measures on WSD.

	Window Size									
Semantic Measure	**1**	**2**	**3**	**4**	**5**	**6**	**7**	**8**	**9**	**AVG**
F1*	**0.56**	**0.75**	**0.79**	**0.82**	**0.85**	**0.88**	**0.89**	**0.90**	**0.91**	**0.82**
F1	0.12	0.21	0.26	0.30	0.32	0.34	0.38	0.38	0.41	0.30
F2*	**0.58**	**0.71**	**0.77**	**0.81**	**0.84**	**0.87**	**0.88**	**0.90**	**0.89**	**0.81**
F2	0.17	0.17	0.17	0.18	0.18	0.20	0.21	0.19	0.19	0.18
F3*	**0.61**	**0.78**	**0.82**	**0.84**	**0.87**	**0.89**	**0.90**	**0.91**	**0.92**	**0.84**
F3	0.20	0.27	0.31	0.31	0.32	0.35	0.38	0.36	0.36	0.32
F4*	**0.61**	**0.77**	**0.80**	**0.83**	**0.86**	**0.89**	**0.90**	**0.91**	**0.93**	**0.83**
F4	0.21	0.24	0.27	0.28	0.25	0.28	0.30	0.29	0.26	0.26
F5*	**0.57**	**0.71**	**0.76**	**0.80**	**0.81**	**0.83**	**0.84**	**0.84**	**0.85**	**0.78**
F5	0.19	0.26	0.31	0.33	0.34	0.34	0.36	0.36	0.38	0.32

Source: The authors.

Table 5.10: The F-score results of D3 using feature-based measures on WSD.

	Window Size									
Semantic Measure	**1**	**2**	**3**	**4**	**5**	**6**	**7**	**8**	**9**	**AVG**
F1*	**0.63**	**0.73**	**0.78**	**0.80**	**0.83**	**0.84**	**0.86**	**0.87**	**0.88**	**0.80**
F1	0.21	0.32	0.36	0.40	0.49	0.53	0.53	0.55	0.57	0.44
F2*	**0.55**	**0.62**	**0.64**	**0.68**	**0.72**	**0.75**	**0.72**	**0.73**	**0.72**	**0.68**
F2	0.04	0.08	0.08	0.12	0.19	0.21	0.19	0.21	0.25	0.15
F3*	**0.59**	**0.68**	**0.75**	**0.75**	**0.77**	**0.81**	**0.81**	**0.81**	**0.82**	**0.76**
F3	0.06	0.12	0.10	0.14	0.21	0.25	0.25	0.25	0.30	0.19
F4*	**0.56**	**0.68**	**0.72**	**0.75**	**0.76**	**0.79**	**0.79**	**0.81**	**0.80**	**0.74**
F4	0.06	0.12	0.08	0.14	0.19	0.22	0.21	0.22	0.27	0.17
F5*	**0.55**	**0.60**	**0.57**	**0.62**	**0.64**	**0.67**	**0.63**	**0.64**	**0.67**	**0.62**
F5	0.06	0.12	0.10	0.15	0.21	0.22	0.24	0.25	0.30	0.18

Source: The authors.

Table 5.10 presents the F-score results of WSD using the knowledge-based measures based on features on D3 (see Section 5.1 for more details). In this experiment, all adapted feature-based measures had a better F-score result in comparison to the original approaches. This improvement is true for all tested window size variations. Also, only the adapted Tversky (1977) measure F1* had expressive results on WSD, with F-score equals or greater than 85%, with a window size greater than 6.

Table 5.11 presents the F-score results of WSD using the knowledge-based measures based on features on D4 (see Section 5.1 for more details). In this experiment, all adapted feature-based measures had a better F-score result in comparison to the original approaches. This improvement is true for all tested window size variations. Also, the F1* measure had expressive results on WSD, with F-score equals or greater than 85%, with a window size greater than 5, while F3* with windows size greater than 6. The F2* and F4* measure have meaningful results with a window size greater than 6. The F5* measure have no significant results.

Table 5.11: The F-score results of D4 using feature-based measures on WSD.

	Window Size									
Semantic Measure	**1**	**2**	**3**	**4**	**5**	**6**	**7**	**8**	**9**	**AVG**
F1*	**0.60**	**0.73**	**0.78**	**0.82**	**0.84**	**0.85**	**0.86**	**0.87**	**0.88**	**0.80**
F1	0.21	0.27	0.32	0.38	0.41	0.44	0.45	0.48	0.51	0.39
F2*	**0.59**	**0.69**	**0.74**	**0.77**	**0.80**	**0.81**	**0.81**	**0.85**	**0.85**	**0.77**
F2	0.12	0.15	0.18	0.21	0.22	0.22	0.21	0.22	0.24	0.20
F3*	**0.61**	**0.73**	**0.77**	**0.81**	**0.82**	**0.83**	**0.85**	**0.87**	**0.88**	**0.80**
F3	0.13	0.16	0.19	0.22	0.24	0.23	0.25	0.26	0.26	0.21
F4*	**0.61**	**0.71**	**0.75**	**0.79**	**0.81**	**0.82**	**0.83**	**0.86**	**0.87**	**0.78**
F4	0.13	0.16	0.19	0.22	0.23	0.23	0.23	0.22	0.24	0.20
F5*	**0.57**	**0.64**	**0.64**	**0.66**	**0.66**	**0.66**	**0.67**	**0.69**	**0.69**	**0.66**
F5	0.10	0.12	0.13	0.17	0.19	0.19	0.19	0.20	0.22	0.17

Source: The authors.

5.4 Hypothesis 3: The Traditional Path-Based Measures are Effective to Distinguish Two Ontology Classes

In this section, we present the experiments aiming to prove the effectiveness of path-based measures to distinguish two ontology classes, as described in hypothesis 3. In these experiments, as presented in Table 5.12, we evaluate five different path-based measures in the WSD. Also, we subdivide this section into three different experiment categories in order to evaluate, besides the adaptation of the path-based measures, different strategies to use the local distance (described in Section 4.4.1), and the performance of our proposal in hybrid measures that use path-based approaches.

Table 5.12: The evaluated knowledge-based measures based on paths

ID	**Semantic Measure**	**Parameter Values**
P1	Rada et al. (1989)	-
P2	Li, Bandar and Mclean (2003)	$\alpha = 0.5, \beta = 0.5$
P3	Liu, Zhou and Zheng (2007) Strat 1	$\alpha = 0.5, \beta = 0.5$
P4	Liu, Zhou and Zheng (2007) Strat 2	$\alpha = 0.5, \beta = 0.5$
P5	Hao et al. (2011)	$\alpha = 0.5, \beta = 0.5$

Source: The authors.

In the first experiment category, we evaluate the Rada et al. (1989) measure with different strategies to use the local distance values for each direct path category, as presented in Table 5.13. In this experiment, we change the distance function of Rada et al. (1989) measure by our proposal (described in Section 4.4.3).

Table 5.13: Different strategies to use the local distance of a path pattern

Path pattern	**LD1**	**LD2**	**LD3**	**LD4**
UR	Num R	Num R	0	0
DR	Num R	Num R	Num R	Num R
HR	Num R	Num R	Num R	Num R
UR-DR	Num R	Num HR	Num HR	Num HR
HR-DR	Num R	Num HR	Num HR	Num HR-DR
UR-HR-DR	Num R	Num HR	Num HR	Num HR-DR

Source: The authors.

Table 5.14 presents the F-score results of WSD, on D1 (see Section 5.1 for more details), using different strategies to use the local distance in our proposal to semantic distance evaluation. In this experiment, it is possible to note the improvement of WSD using the LD1, LD2, and LD3 strategies in comparison to the original Rada et al. (1989) measure P1. Also, only the LD1, LD2, and LD3 had expressive results on WSD, with F-score equals or greater than 85%.

Table 5.14: The F-score results of D1 using different local distance strategies on WSD.

	Window Size									
Semantic Measure	**1**	**2**	**3**	**4**	**5**	**6**	**7**	**8**	**9**	**AVG**
P1	0.80	0.81	0.82	0.83	0.82	0.82	0.84	0.83	0.80	0.82
LD1	**0.86**	0.86	0.88	0.86	0.88	0.90	0.90	0.92	0.89	0.88
LD2	**0.86**	**0.89**	**0.92**	0.90	**0.94**	**0.95**	**0.96**	**0.95**	**0.95**	**0.92**
LD3	**0.86**	**0.89**	0.91	0.89	0.93	**0.95**	0.95	**0.95**	0.94	**0.92**
LD4	0.82	0.83	0.80	0.82	0.80	0.80	0.83	0.80	0.75	0.81

Source: The authors.

Table 5.15 presents the F-score results of WSD, on D2 (see Section 5.1 for more details), using different strategies to use the local distance in our proposal to semantic distance evaluation. In this experiment, it is possible to note the improvement of WSD using the LD2 and LD3 strategies in comparison to the original Rada et al. (1989) measure P1. Also, all evaluated strategies had expressive results on WSD, with F-score equals or greater than 85%.

Table 5.16 presents the F-score results of WSD, on D3 (see Section 5.1 for more details), using different strategies to use the local distance in our proposal to semantic

Table 5.15: The F-score results of D2 using different local distance strategies on WSD.

	Window Size									
Semantic Measure	**1**	**2**	**3**	**4**	**5**	**6**	**7**	**8**	**9**	**AVG**
P1	0.86	0.90	**0.92**	**0.92**	0.92	**0.93**	**0.92**	0.92	**0.93**	0.91
LD1	0.88	**0.91**	**0.92**	**0.92**	0.92	0.92	**0.92**	0.92	0.92	0.91
LD2	**0.89**	**0.91**	**0.92**	**0.92**	**0.93**	**0.93**	**0.92**	**0.93**	**0.93**	**0.92**
LD3	**0.89**	**0.91**	**0.92**	**0.92**	**0.93**	**0.93**	**0.92**	**0.93**	**0.93**	**0.92**
LD4	0.86	0.89	0.89	0.88	0.90	0.90	0.91	0.90	0.91	0.89

Source: The authors.

distance evaluation. In this experiment, all local distance strategies had a better average F-score result in comparison to the original Rada et al. (1989) measure P1. Also, all evaluated strategies had expressive results on WSD, with average F-score equals or greater than 85%.

Table 5.16: The F-score results of D3 using different local distance strategies on WSD.

	Window Size									
Semantic Measure	**1**	**2**	**3**	**4**	**5**	**6**	**7**	**8**	**9**	**AVG**
P1	0.78	0.78	0.79	0.81	0.83	0.85	0.85	0.85	0.86	0.82
LD1	0.78	0.77	0.84	0.88	0.87	0.90	0.91	0.91	0.92	0.86
LD2	0.78	**0.82**	**0.89**	**0.91**	**0.91**	**0.93**	**0.93**	**0.93**	**0.93**	**0.89**
LD3	0.78	**0.82**	0.88	0.90	0.90	0.92	0.92	**0.93**	**0.93**	**0.89**
LD4	**0.81**	0.80	0.83	0.86	0.86	0.90	0.91	0.91	0.92	0.87

Source: The authors.

Table 5.17 presents the F-score results of WSD, on D4 (see Section 5.1 for more details), using different strategies to use the local distance in our proposal to semantic distance evaluation. In this experiment, the local distance strategies LD2 and LD3 had equal average F-score results in comparison to the original Rada et al. (1989) measure P1. Also, all evaluated strategies had expressive results on WSD, with average F-score equals or greater than 85%.

Table 5.17: The F-score results of D4 using different local distance strategies on WSD.

	Window Size									
Semantic Measure	**1**	**2**	**3**	**4**	**5**	**6**	**7**	**8**	**9**	**AVG**
P1	0.79	0.87	0.89	**0.91**	**0.92**	**0.93**	**0.94**	**0.94**	**0.94**	**0.90**
LD1	0.80	0.87	0.89	0.90	0.90	0.91	0.92	0.92	0.92	0.89
LD2	0.80	0.87	**0.90**	**0.91**	0.91	0.92	0.92	0.92	0.92	**0.90**
LD3	0.80	0.87	**0.90**	**0.91**	0.91	0.92	0.92	0.92	0.92	**0.90**
LD4	**0.81**	**0.88**	0.89	0.89	0.89	0.90	0.91	0.91	0.92	0.89

Source: The authors.

Overall, based on the results of the first experiment category, the local distance strategies LD2 and LD3 present better F-score results than the others. From this, for the second and third categories of the experiments, we use the LD2 strategy because it presents better F-score results though different window sizes.

In the second category of the experiments, we evaluate five different knowledge-based measures based on paths. Also, we compare these measures with their adaptations with our semantic distance approach (described in Section 4.4.3) using the LD2 strategy to local distance, and we use the operator * to differentiate the original approaches from our adaptation. Also, we keep the parameter values, presented in Table 5.12, in order not to influence one semantic evidence concerning the other.

Table 5.18 presents the F-score results of WSD using the knowledge-based measures based on paths on D1 (see Section 5.1 for more details). In this experiment, all adapted path-based measures had a better F-score result in comparison to the original approaches. This improvement is true for all tested window size variations. Also, all adapted path-based measures had very expressive results on WSD, with average F-score equals or greater than 90%, with all tested window size values.

Table 5.18: The F-score results of D1 using path-based measures on WSD.

	Window Size									
Semantic Measure	**1**	**2**	**3**	**4**	**5**	**6**	**7**	**8**	**9**	**AVG**
P1	0.80	0.81	0.82	0.83	0.82	0.82	0.84	0.83	0.80	0.82
P1*	**0.86**	**0.89**	**0.92**	**0.90**	**0.94**	**0.95**	**0.96**	**0.95**	**0.95**	**0.92**
P2	0.81	0.83	0.84	0.83	0.83	0.85	0.85	0.86	0.81	0.83
P2*	**0.86**	**0.90**	**0.92**	**0.90**	**0.93**	**0.95**	**0.96**	**0.95**	**0.93**	**0.92**
P3	0.80	0.81	0.82	0.83	0.82	0.83	0.85	0.85	0.82	0.83
P3*	**0.86**	**0.92**	**0.92**	**0.90**	**0.94**	**0.94**	**0.95**	**0.94**	**0.93**	**0.92**
P4	0.80	0.83	0.83	0.83	0.85	0.85	0.86	0.86	0.82	0.84
P4*	**0.87**	**0.90**	**0.90**	**0.90**	**0.92**	**0.92**	**0.90**	**0.90**	**0.89**	**0.90**
P5	0.82	0.81	0.82	0.82	0.82	0.80	0.82	0.77	0.76	0.80
P5*	**0.88**	**0.88**	**0.92**	**0.88**	**0.90**	**0.93**	**0.94**	**0.94**	**0.90**	**0.91**

Source: The authors.

Table 5.19 presents the F-score results of WSD using the knowledge-based measures based on paths on D2 (see Section 5.1 for more details). In this experiment, all adapted path-based measures had equal or better average F-score results in comparison to the original approaches. Also, all evaluated measures had very expressive results on WSD, with average F-score equals or greater than 90%, with all tested window size values.

Table 5.19: The F-score results of D2 using path-based measures on WSD.

	Window Size									
Semantic Measure	**1**	**2**	**3**	**4**	**5**	**6**	**7**	**8**	**9**	**AVG**
P1	0.86	0.90	**0.92**	**0.92**	0.92	**0.93**	**0.92**	0.92	**0.93**	0.91
P1*	**0.89**	**0.91**	**0.92**	**0.92**	**0.93**	**0.93**	**0.92**	**0.93**	**0.93**	**0.92**
P2	0.86	0.90	**0.92**	**0.92**	**0.93**	**0.93**	**0.93**	0.92	**0.93**	**0.92**
P2*	**0.89**	**0.92**	**0.92**	**0.92**	0.92	**0.93**	0.92	**0.93**	**0.93**	**0.92**
P3	0.86	0.91	0.92	**0.93**	**0.94**	**0.94**	**0.94**	**0.93**	**0.94**	**0.92**
P3*	**0.89**	**0.92**	**0.93**	**0.93**	0.93	0.93	0.93	**0.93**	0.93	**0.92**
P4	0.86	0.89	**0.92**	**0.92**	**0.93**	**0.93**	**0.93**	**0.93**	**0.93**	**0.92**
P4*	**0.88**	**0.90**	**0.92**	**0.92**	**0.93**	**0.93**	**0.93**	**0.93**	**0.93**	**0.92**
P5	0.86	0.88	0.91	0.91	0.91	0.92	0.91	0.92	0.92	0.90
P5*	**0.90**	**0.92**	**0.93**	**0.94**	**0.94**	**0.94**	**0.93**	**0.93**	**0.93**	**0.93**

Source: The authors.

Table 5.20 presents the F-score results of WSD using the knowledge-based measures based on paths on D3 (see Section 5.1 for more details). In this experiment, all adapted path-based measures had a better average F-score result in comparison to the original approaches. Also, all evaluated measures had very expressive results on WSD, with average F-score equals or greater than 90%, with a window size greater than 3.

Table 5.20: The F-score results of D3 using path-based measures on WSD.

	Window Size									
Semantic Measure	**1**	**2**	**3**	**4**	**5**	**6**	**7**	**8**	**9**	**AVG**
P1	**0.78**	0.78	0.79	0.81	0.83	0.85	0.85	0.85	0.86	0.82
P1*	**0.78**	**0.82**	**0.89**	**0.91**	**0.91**	**0.93**	**0.93**	**0.93**	**0.93**	**0.89**
P2	**0.78**	0.80	0.81	0.82	0.85	0.85	0.86	0.86	0.88	0.83
P2*	**0.78**	**0.84**	**0.89**	**0.91**	**0.92**	**0.93**	**0.93**	**0.93**	**0.94**	**0.90**
P3	**0.78**	0.78	0.81	0.83	0.84	0.86	0.86	0.87	0.88	0.84
P3*	**0.78**	**0.85**	**0.90**	**0.92**	**0.92**	**0.93**	**0.93**	**0.93**	**0.95**	**0.90**
P4	**0.78**	0.79	0.75	0.81	0.81	0.82	0.82	0.84	0.85	0.81
P4*	0.75	**0.84**	**0.88**	**0.90**	**0.90**	**0.92**	**0.92**	**0.93**	**0.93**	**0.88**
P5	**0.83**	0.80	0.81	0.82	0.83	0.88	0.86	0.86	0.86	0.84
P5*	0.81	**0.85**	**0.90**	**0.90**	**0.90**	**0.92**	**0.93**	**0.94**	**0.94**	**0.90**

Source: The authors.

Table 5.21 presents the F-score results of WSD using the knowledge-based measures based on paths on D4 (see Section 5.1 for more details). In this experiment, all adapted path-based measures had equals or better average F-score results in comparison to the original approaches. Also, all evaluated measures had very expressive results on WSD, with average F-score equals or greater than 90%, with a window size greater than

3, except the path-based measure P4.

Table 5.21: The F-score results of D4 using path-based measures on WSD.

	Window Size									
Semantic Measure	**1**	**2**	**3**	**4**	**5**	**6**	**7**	**8**	**9**	**AVG**
P1	0.79	**0.87**	0.89	**0.91**	**0.92**	**0.93**	**0.94**	**0.94**	**0.94**	**0.90**
P1*	**0.80**	**0.87**	**0.90**	**0.91**	0.91	0.92	0.92	0.92	0.92	**0.90**
P2	0.79	**0.88**	**0.90**	0.91	**0.93**	**0.93**	**0.93**	**0.94**	**0.94**	**0.91**
P2*	**0.80**	**0.88**	**0.90**	**0.92**	**0.92**	**0.93**	**0.93**	**0.93**	**0.93**	**0.91**
P3	0.80	**0.88**	0.88	**0.90**	**0.92**	**0.92**	**0.92**	**0.93**	**0.93**	**0.90**
P3*	**0.81**	**0.88**	**0.90**	**0.92**	**0.92**	**0.92**	**0.92**	0.92	0.92	**0.90**
P4	0.78	0.82	0.85	0.85	0.85	0.86	0.85	0.85	0.85	0.84
P4*	**0.80**	**0.86**	**0.89**	**0.90**	**0.90**	**0.91**	**0.91**	**0.92**	**0.93**	**0.89**
P5	0.81	0.88	0.90	0.90	0.91	0.92	**0.93**	**0.94**	**0.94**	0.90
P5*	**0.83**	**0.90**	**0.91**	**0.91**	**0.92**	**0.93**	0.92	0.93	0.93	**0.91**

Source: The authors.

In the third category of the experiments, we evaluate two different hybrid measures in the WSD. Also, these measures combine information content and path approaches. In the path-based part of these measures, we adapt them with our semantic distance approach (described in Section 4.4.3) with local distance strategy LD2, and we use the operator * to differentiate the original approaches from our adaptation. In addition, we keep the parameter values, presented in Table 5.22, in order not to influence one knowledge-based measure concerning the other.

Table 5.22: The evaluated hybrid knowledge-based measures

ID	**Semantic Measure**	**Parameter Values**
H1	Cai et al. (2018) + IC Cai et al. (2018)	$\alpha = 0.5, \beta = 1.0, \lambda = 0.5$
H2	Zhu and Iglesias (2017) + IC Cai et al. (2018)	$k = 0.5, \lambda = 0.5$

Source: The authors.

Table 5.23 presents the F-score results of WSD using the hybrid knowledge-based measures on D1 (see Section 5.1 for more details). In this experiment, all adapted hybrid measures had a better F-score result in comparison to the original approaches in all tested window size values. Also, the adapted measure H1* had expressive results on WSD, with average F-score equals or greater than 85%, in all tested window size values. Already, the adapted measure H2* had very expressive results on WSD, with average F-score equals or greater than 90%, with a window size greater than 1.

Table 5.24 presents the F-score results of WSD using the hybrid knowledge-based measures on D2 (see Section 5.1 for more details). In this experiment, all adapted hy-

Table 5.23: The F-score results of D1 using hybrid measures on WSD.

	Window Size									
Semantic Measure	**1**	**2**	**3**	**4**	**5**	**6**	**7**	**8**	**9**	**AVG**
H1	0.78	0.77	0.77	0.76	0.73	0.74	0.73	0.73	0.70	0.75
H1*	**0.87**	**0.90**	**0.86**	**0.86**	**0.88**	**0.90**	**0.89**	**0.89**	**0.86**	**0.88**
H2	0.81	0.82	0.82	0.83	0.82	0.80	0.82	0.80	0.79	0.81
H2*	**0.86**	**0.90**	**0.92**	**0.90**	**0.93**	**0.95**	**0.96**	**0.95**	**0.95**	**0.92**

Source: The authors.

brid measures had equals or better average F-score results in comparison to the original approaches. Also, all measures, except the H1 measure, had very expressive results on WSD, with F-score equals or greater than 90%.

Table 5.24: The F-score results of D2 using hybrid measures on WSD.

	Window Size									
Semantic Measure	**1**	**2**	**3**	**4**	**5**	**6**	**7**	**8**	**9**	**AVG**
H1	0.78	0.78	0.80	0.81	0.82	0.85	0.85	0.86	0.87	0.82
H1*	**0.88**	**0.91**	**0.93**	**0.93**	**0.93**	**0.94**	**0.94**	**0.94**	**0.94**	**0.93**
H2	0.87	0.90	**0.92**	0.92	0.92	**0.93**	0.92	**0.93**	**0.93**	**0.92**
H2*	**0.89**	**0.92**	**0.92**	**0.93**	**0.93**	**0.93**	**0.93**	0.92	**0.93**	**0.92**

Source: The authors.

Table 5.25 presents the F-score results of WSD using the hybrid knowledge-based measures on D3 (see Section 5.1 for more details). In this experiment, all adapted hybrid measures, except in H2 measure with a window size of 1, had a better F-score result in comparison to the original approaches in all tested window size values. Also, the adapted measures had very expressive results on WSD, with average F-score equals or greater than 90%, with a window size greater than 3.

Table 5.25: The F-score results of D3 using hybrid measures on WSD.

	Window Size									
Semantic Measure	**1**	**2**	**3**	**4**	**5**	**6**	**7**	**8**	**9**	**AVG**
H1	0.68	0.68	0.68	0.73	0.74	0.75	0.77	0.78	0.77	0.73
H1*	**0.83**	**0.88**	**0.91**	**0.93**	**0.92**	**0.93**	**0.94**	**0.95**	**0.96**	**0.92**
H2	**0.80**	0.78	0.81	0.81	0.83	0.85	0.85	0.85	0.86	0.83
H2*	0.78	**0.83**	**0.89**	**0.91**	**0.91**	**0.93**	**0.93**	**0.93**	**0.93**	**0.89**

Source: The authors.

Table 5.26 presents the F-score results of WSD using the hybrid knowledge-based measures on D4 (see Section 5.1 for more details). In this experiment, all adapted hybrid measures had equals or better average F-score results in comparison to the original approaches. Also, the H1*, H2, and H2* measures had very expressive results, with F-score equals or greater than 90%, with a window size greater than 1, 3, and 2, respectively.

Table 5.26: The F-score results of D4 using hybrid measures on WSD.

	Window Size									
Semantic Measure	**1**	**2**	**3**	**4**	**5**	**6**	**7**	**8**	**9**	**AVG**
H1	0.64	0.68	0.69	0.71	0.72	0.74	0.74	0.73	0.74	0.71
H1*	**0.86**	**0.90**	**0.91**	**0.93**	**0.94**	**0.94**	**0.94**	**0.95**	**0.94**	**0.93**
H2	**0.80**	**0.88**	0.89	**0.91**	**0.92**	**0.93**	**0.93**	**0.94**	**0.94**	**0.90**
H2*	**0.80**	0.87	**0.90**	**0.91**	0.91	0.92	0.92	0.92	0.92	**0.90**

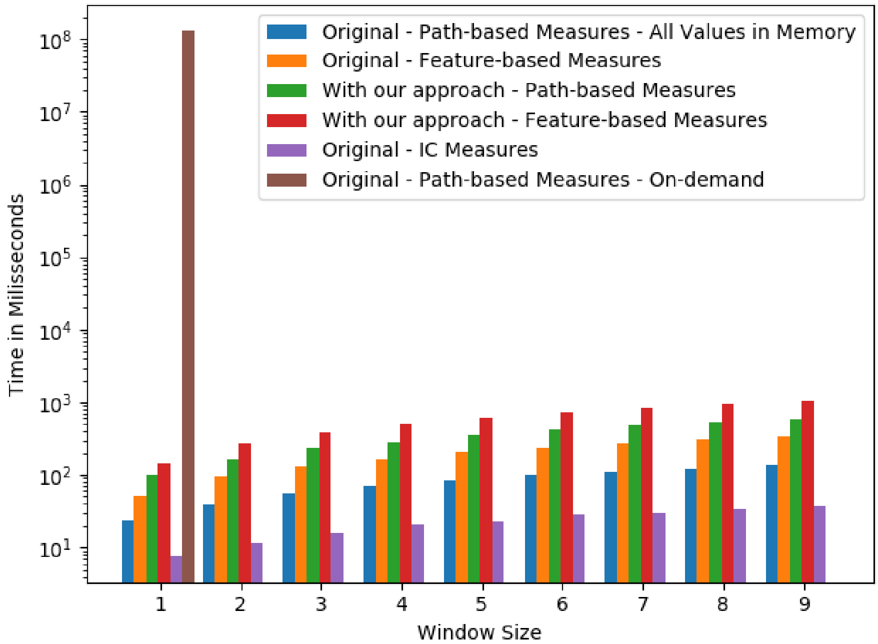
Source: The authors.

5.5 Hypothesis 3: The Traditional Path-Based Measures are Inefficient on Relatedness Evaluation

In this section, we present the experiments aiming to prove the inefficiency of the traditional path-based measures on relatedness evaluation, as described in hypothesis 3. From this, we present the experiments about the evaluation time and memory consumption of each knowledge-based semantic measure approach. In the first experiment, presented in Figure 5.1, we evaluate the average time to perform the relatedness evaluation during the WSD, for each window size value, over all datasets. The results presented in Figure 5.1 are in logarithmic scale. The average time evaluated comprises the average time of all knowledge-based measures of the same type. For example, on path-based measures, we evaluate the average time of the five path-based measures presented in Table 5.12. Also, all evaluated approaches, except for path-based measures on-demand, contain all required classes in memory. For example, the feature-based measures contain the features of the analyzed classes in memory. In this experiment, the path-based measures that evaluate the relatedness value on-demand take about 36 hours to evaluate the relatedness values on the window size of 1. With this, we hide the part of its bar in the function of the other results.

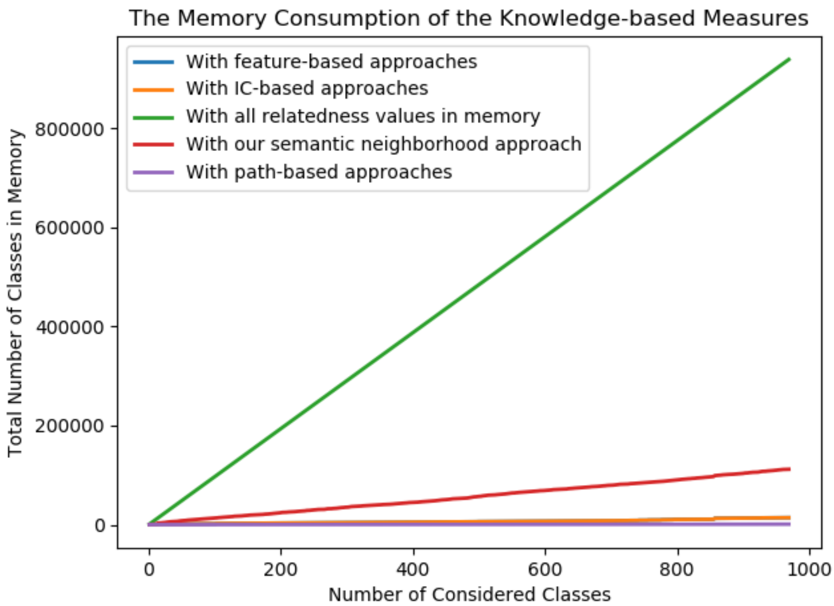
Besides the distinction performance, in this work, we use the evaluation time as another factor to choose a semantic measure. The last factor is the memory required to perform the relatedness evaluation. From this, in Figure 5.2, we present the memory consumption of each knowledge-based approach to perform the relatedness evaluation,

Figure 5.1: The evaluation time of the knowledge-based measures.



Source: The Authors.

Figure 5.2: The memory consumption of the knowledge-based measures.



Source: The Authors.

and the memory required to store the relatedness values between all ontology classes in memory. The feature and IC approaches have nearly the same memory consumption. The x-axis of this figure represents the number of classes (or values) required, in memory, for each ontology class, to make possible the relatedness analysis of a given knowledge-based approach. The y-axis of Figure 5.2 represents the total number of classes (or values) stored in memory. It is important to emphasize that storing all relatedness values for each ontology class has a quadratic memory cost.

5.6 Hypothesis 4: The Ontological Meta-Properties Improve the Distinction Performance of the Path-Based Measures

In this section, we present the experiments aiming to prove hypothesis 4. In these experiments, our objective is to show the improvement in the distinction performance by considering different weighting values for each type of existential dependence (described in Section 2.2). Thus, we do not define a general rule of how to use this meta-property, but we show that it is necessary to evaluate. Also, we evaluate the Rada et al. (1989) measure with the local distance strategy LD2 using different values to weight the types of existential dependence, as presented in Table 5.27. It is important to empathize that the strategy EDW5 have the same F-score results of the LD2 strategy of the experiments in Section 5.4, i.e., we aim to improve the P1 and EDW5 results using the existential dependence as a piece of semantic proxy in order to improve the distinction between two ontology classes. Also, the higher the weighting value, the closer the related classes are, i.e., their relatedness value is greater.

Table 5.27: Different strategies to existential dependence weights

Strategy	S	G	I
EDW1	0.9	0.5	0.1
EDW2	0.1	0.5	0.9
EDW3	0.5	0.9	0.1
EDW4	0.5	0.1	0.9
EDW5 (LD2)	0	0	0

Source: The authors.

Table 5.28 presents the F-score results of WSD, on D1 (see Section 5.1 for more details), using different values to existential dependence weights. In this experiment, all strategies had a better average F-score result in comparison to the original Rada et al. (1989) measure P1. Also, the strategies EDW1 and EDW3 had better F-score results than the strategy EDW5. In addition, all evaluated strategies had expressive results on WSD, with average F-score equals or greater than 85%.

Table 5.29 presents the F-score results of WSD, on D2 (see Section 5.1 for more details), using different values to existential dependence weights. In this experiment, all strategies had a better average F-score result in comparison to the original Rada et al.

Table 5.28: The F-score results of D1 using different existential dependence weights on WSD.

	Window Size									
Semantic Measure	**1**	**2**	**3**	**4**	**5**	**6**	**7**	**8**	**9**	**AVG**
P1	0.80	0.81	0.82	0.83	0.82	0.82	0.84	0.83	0.80	0.82
EDW1	0.85	0.91	**0.93**	**0.93**	**0.95**	**0.96**	0.95	**0.97**	**0.96**	0.93
EDW2	0.85	0.87	0.86	0.88	0.86	0.87	0.89	0.90	0.85	0.87
EDW3	**0.88**	**0.92**	**0.93**	**0.93**	0.94	**0.96**	**0.96**	**0.97**	**0.96**	**0.94**
EDW4	0.85	0.88	0.86	0.87	0.86	0.88	0.89	0.90	0.86	0.87
EDW5	0.86	0.89	0.92	0.90	0.94	0.95	0.96	0.95	0.95	0.92

Source: The authors.

(1989) measure P1. Also, the strategies EDW1, EDW2, and EDW4 had better F-score results than the strategy EDW5. The strategy EDW4 had the same F-score as strategy EDW5. In addition, all evaluated strategies had expressive results on WSD, with average F-score equals or greater than 85%.

Table 5.29: The F-score results of D2 using different existential dependence weights on WSD.

	Window Size									
Semantic Measure	**1**	**2**	**3**	**4**	**5**	**6**	**7**	**8**	**9**	**AVG**
P1	0.86	0.90	0.92	0.92	0.92	0.93	0.92	0.92	0.93	0.91
EDW1	0.88	0.91	**0.93**	**0.94**	0.93	**0.94**	**0.94**	**0.94**	**0.94**	**0.93**
EDW2	**0.89**	**0.92**	**0.93**	**0.94**	**0.94**	**0.94**	0.93	**0.94**	**0.94**	**0.93**
EDW3	**0.89**	**0.92**	0.92	0.93	0.92	0.93	0.92	0.93	0.93	0.92
EDW4	**0.89**	**0.92**	**0.93**	**0.94**	**0.94**	**0.94**	**0.94**	**0.94**	**0.94**	**0.93**
EDW5	**0.89**	0.91	0.92	0.92	0.93	0.93	0.92	0.93	0.93	0.92

Source: The authors.

Table 5.30 presents the F-score results of WSD, on D3 (see Section 5.1 for more details), using different values to existential dependence weights. In this experiment, all strategies had a better average F-score result in comparison to the original Rada et al. (1989) measure P1. Also, the strategies EDW2 and EDW4 had better average F-score results than the strategy EDW5. The strategy EDW3 had the same average F-score results as strategy EDW5. In addition, all evaluated strategies had expressive results on WSD, with average F-score equals or greater than 85%.

Table 5.31 presents the F-score results of WSD, on D4 (see Section 5.1 for more details), using different values to existential dependence weights. In this experiment, the strategies EDW2 and EDW4 had better average F-score results than the strategy EDW5 and the original Rada et al. (1989) measure P1. The strategy EDW3 had the same average

Table 5.30: The F-score results of D3 using different existential dependence weights on WSD.

	Window Size									
Semantic Measure	**1**	**2**	**3**	**4**	**5**	**6**	**7**	**8**	**9**	**AVG**
P1	0.78	0.78	0.79	0.81	0.83	0.85	0.85	0.85	0.86	0.82
EDW1	0.74	0.78	0.84	0.87	0.88	0.88	0.90	0.90	0.92	0.85
EDW2	**0.81**	**0.86**	**0.89**	**0.92**	0.91	0.92	**0.93**	**0.93**	**0.94**	**0.90**
EDW3	0.78	0.82	0.88	**0.92**	**0.92**	**0.93**	**0.93**	**0.93**	0.93	0.89
EDW4	**0.81**	0.85	0.88	0.91	0.90	0.92	**0.93**	**0.93**	**0.94**	**0.90**
EDW5	0.78	0.82	**0.89**	0.91	0.91	**0.93**	**0.93**	**0.93**	0.93	0.89

Source: The authors.

F-score results as strategy EDW5 and P1, and strategy EDW1 the worst result. In addition, all evaluated strategies had expressive results on WSD, with average F-score equals or greater than 85%.

Table 5.31: The F-score results of D4 using different existential dependence weights on WSD.

	Window Size									
Semantic Measure	**1**	**2**	**3**	**4**	**5**	**6**	**7**	**8**	**9**	**AVG**
P1	0.79	0.87	0.89	0.91	0.92	0.93	0.94	0.94	0.94	0.90
EDW1	0.76	0.84	0.87	0.88	0.89	0.90	0.90	0.90	0.91	0.87
EDW2	**0.83**	0.90	**0.92**	0.92	0.93	**0.94**	0.94	0.94	**0.95**	**0.92**
EDW3	0.81	0.87	0.89	0.90	0.91	0.92	0.92	0.92	0.93	0.90
EDW4	0.82	**0.91**	**0.92**	**0.93**	**0.94**	**0.94**	**0.95**	**0.95**	**0.95**	**0.92**
EDW5	0.80	0.87	0.90	0.91	0.91	0.92	0.92	0.92	0.92	0.90

Source: The authors.

Overall, based on the results of these experiments, the weighing strategies were able to improve the distinction of the ontology classes in comparison to the strategy EDW5 (or LD2 of the first experiment) during the word sense disambiguation (WSD) task.

6 DISCUSSION

In this chapter, we present a discussion regarding the F-score results presented in Chapter 5. Overall, these results show that the adaption of the knowledge-based measures based on features and paths, with our semantic neighborhood approach, improve the performance of these measures on word sense disambiguation (WSD). The following paragraphs discuss, in detail, the results regarding the hypothesis presented during this work.

The first hypothesis of this work is that the knowledge-based similarity measures (measures that use only the taxonomic structure of the ontology) are inefficient to distinguish two ontology classes. We prove this hypothesis by the F-score results presented in Section 5.2, where we evaluate the information content (IC) measures on WSD. In these results, all evaluated measures had poor distinction performance on the WSD task.

The second hypothesis of this work is that combining taxonomic and nontaxonomic relationships can improve the distinction of the feature-based measures. We prove this hypothesis by the F-score results presented in Section 5.3, where we evaluate the feature-based measures and their adaptation with the semantic neighbors (as presented in Section 4.3) on WSD. In these results, the adapted feature-based measures had not expressive F-score on the WSD task. However, these adapted measures had a much better performance on the distinction capability than their original versions.

The third hypothesis of this work is that the relatedness measures based on paths are effective in distinguishing two ontology classes but inefficient on-demand tasks. Based on the F-score results presented in the third category of experiments in Section 5.4, the path-based measures with and without our semantic distance approach (presented in Section 4.4.3) had expressive distinction performance on WSD, with the better results regarding the other knowledge-based measures. Also, we show that in the first category of experiments in Section 5.4, some variations on the local distance strategy can improve the distinction performance regarding the original approaches. However, as presented in Section 5.5, the Dijkstra algorithm has a low performance to compute the semantic distance between two ontology classes, in time of consultation. Also, storing all the relatedness values between all ontology classes has a quadratic cost to the memory. From this, during the relatedness evaluation, our semantic neighborhood approach is a good strategy to reduce memory consumption, and our semantic distance approach had a much better performance during on-demand relatedness evaluation. Finally, our approach proved to be an intermediate strategy that conciliates the memory consumption and evaluation time.

The fourth hypothesis of this work is that the ontological meta-properties can improve the distinction performance between two ontology classes. In this work, we use only the existential dependence meta-property of the classes to weight the relationships which start from them. Based on the experiments, presented in Section 5.6, the path-based measures, adapted with our semantic distance approach and the local distance strategy LD2, improve the distinction performance, with some weight values, regarding the original approaches as well as the adapted approaches with only the local distance strategy.

Finally, our approach proved to be a better strategy for relatedness evaluation that conciliates low memory consumption with low evaluation time, and better distinction performance regarding the knowledge-based measures present in the state-of-the-art.

7 CONCLUSION

The main objective of this work is to improve the performance of the knowledge-based measures on relatedness evaluation. To this end, we propose a novel strategy that conciliates low memory consumption with low evaluation time, and with better distinction performance regarding the knowledge-based measures present in the state-of-the-art. Firstly, we propose to store only the semantic neighbors of an ontology class in memory. These semantic neighbors are the set of related classes through direct paths. In this work, we build direct paths through the relationship categories, or a combination of them, thus performing the combination of taxonomic and non-taxonomic relationships through the path patterns. Also, we propose to use the semantic neighborhood strategy in the knowledge-based measures based on paths and features. In the feature-based measures, we use the semantic neighbors as the feature set of an ontology class. Already in path-based measures, we propose a novel strategy to compute the semantic distance between two ontology classes based on the local distance between a class and its semantic neighbors, weighted by the existential dependence type of this class.

To evaluate our approach, we perform the word sense disambiguation (WSD) using an algorithm based on structured knowledge, more specifically we use a domain ontology about core description called Strataledge® ontology. This algorithm finds in corpora the occurrences of the terms that name two or more ontology classes and try to disambiguate them based on a context window. In our experiments, we extract the context window where these terms occur from four different corpora on Oil&Gas domain using a domain ontology for core description. Also, in these experiments, we compare four different knowledge-based approaches (path-based, feature-based, information content, and hybrid approaches). In the feature, path, and hybrid approaches, we compare the WSD results with our adaptation in these measures.

As a result of our work, we prove that similarity measures are ineffective to distinguish two ontology classes. Also, we evidence that combining taxonomic and non-taxonomic relationships improve the distinction capability of feature-based measures. In addition, we show that path-based measures are the best choice in relatedness evaluation because of their distinction capability. However, we demonstrate that path-based measures are inefficient to perform, on-demand, the semantic distance function in relatedness evaluation. Furthermore, we show that the local distance strategies and the existential dependence weights can improve the adapted path-based measures on WSD. Finally, we demonstrate, with several experiments, that our semantic neighborhood approach can improve the state-of-the-art semantic measure in the evaluation time, memory consumption, and distinction capability.

In future works, we aim to explore other kinds of ontological meta-properties in relatedness evaluation. Also, we will use these meta-properties to split the horizontal category of relationships in order to analyze more precisely the different semantics of these relationships. In addition, we will explore different strategies aimed to improve the distinction capability of the feature-based and information content approaches. Other future work includes the application of our proposal in other domains such as biomedicine.

Finally, considering the increasing interest in domain ontologies in tasks such as information retrieval and the tendency of the formalization of these ontologies with top-level definitions, this work starts the discussion about how to use the ontological meta-proprieties of the top-level concepts as a new semantic proxy, besides the corpora of texts and structured proxies in domain level, where we can extract new semantic evidence to perform the similarity or relatedness evaluation on well-founded ontologies.

REFERENCES

ADHIKARI, A. et al. A novel information theoretic approach for finding semantic similarity in wordnet. In: **Proc. of IEEE International Technical Conference (TENCON-2015)**. [S.l.: s.n.], 2015. p. 1–4.

ARP, R.; SMITH, B.; SPEAR, A. D. **Building ontologies with basic formal ontology**. [S.l.]: Mit Press, 2015.

ASIM, M. N. et al. The use of ontology in retrieval: A study on textual, multilingual, and multimedia retrieval. **IEEE Access**, v. 7, p. 21662–21686, 2019. ISSN 2169-3536.

BORST, W. N. **Construction of Engineering Ontologies for Knowledge Sharing and Reuse**. Thesis (PhD) — University of Twente, Netherlands, 9 1997.

CAI, Y. et al. A hybrid approach for measuring semantic similarity based on ic-weighted path distance in wordnet. **Journal of Intelligent Information Systems**, Springer, v. 51, n. 1, p. 23–47, 2018.

CORREIA, F. Ontological dependence. **Philosophy Compass**, Wiley-Blackwell, v. 3, n. 5, p. 1013–1032, 2008.

DIEFENBACH, D. et al. A scalable approach for computing semantic relatedness using semantic web data. In: **Proceedings of the 6th International Conference on Web Intelligence, Mining and Semantics**. New York, NY, USA: ACM, 2016. (WIMS '16), p. 20:1–20:9. ISBN 978-1-4503-4056-4. Available from Internet: <http://doi.acm.org/10.1145/2912845.2912864>.

DIJKSTRA, E. W. A note on two problems in connexion with graphs. **Numerische mathematik**, Springer, v. 1, n. 1, p. 269–271, 1959.

GRUBER, T. R. A translation approach to portable ontology specifications. **Knowledge acquisition**, Elsevier, v. 5, n. 2, p. 199–220, 1993.

GUARINO, N. Semantic matching: Formal ontological distinctions for information organization, extraction, and integration. In: PAZIENZA, M. T. (Ed.). **Information Extraction A Multidisciplinary Approach to an Emerging Information Technology**. Berlin, Heidelberg: Springer Berlin Heidelberg, 1997. p. 139–170. ISBN 978-3-540-69548-6.

GUARINO, N. Formal ontology in information systems (fois). In: **Proceedings of Formal Ontology and Information Systems, (FOIS'98): IOS Press**. [S.l.: s.n.], 1998. p. 3–15.

GUARINO, N. The role of identity conditions in ontology design. In: . [S.l.: s.n.], 1999. p. 750–750.

GUARINO, N.; WELTY, C. Evaluating ontological decisions with ontoclean. **Communications of the ACM**, v. 45, p. 61–65, 02 2002.

GUARINO, N.; WELTY, C. An overview of ontoclean. In: ____. [S.l.: s.n.], 2009. p. 201–220.

GUARINO, N.; WELTY, C. A. An overview of ontoclean. In: **Handbook on ontologies**. [S.l.]: Springer, 2004. p. 151–171.

GUIZZARDI, G. **Ontological foundations for structural conceptual models**. Thesis (PhD) — University of Twente, 2005.

HAO, D. et al. An approach for calculating semantic similarity between words using wordnet. In: IEEE. **2011 Second International Conference on Digital Manufacturing & Automation**. [S.l.], 2011. p. 177–180.

HARISPE, S. et al. Semantic similarity from natural language and ontology analysis. **Synthesis Lectures on Human Language Technologies**, Morgan & Claypool Publishers, v. 8, n. 1, p. 1–254, 2015.

HIRST, G.; ST-ONGE, D. et al. Lexical chains as representations of context for the detection and correction of malapropisms. **WordNet: An electronic lexical database**, v. 305, p. 305–332, 1998.

JIANG, J. J.; CONRATH, D. W. Semantic similarity based on corpus statistics and lexical taxonomy. In: **Proceedings of the 10th Research on Computational Linguistics International Conference (ROCLING 1997)**. Taipei, Taiwan: The Association for Computational Linguistics and Chinese Language Processing (ACLCLP), 1997. p. 19–33. Available from Internet: <https://www.aclweb.org/anthology/O97-1002>.

JIN, Y. et al. An approach to measuring semantic similarity and relatedness between concepts in an ontology. In: IEEE. **2017 23rd International Conference on Automation and Computing (ICAC)**. [S.l.], 2017. p. 1–6.

LASTRA-DÍAZ, J. J.; GARCÍA-SERRANO, A. A new family of information content models with an experimental survey on wordnet. **Knowledge-Based Systems**, Elsevier, v. 89, p. 509–526, 2015.

LASTRA-DÍAZ, J. J. et al. A reproducible survey on word embeddings and ontology-based methods for word similarity: Linear combinations outperform the state of the art. **Engineering Applications of Artificial Intelligence**, v. 85, p. 645 – 665, 2019. ISSN 0952-1976. Available from Internet: <http://www.sciencedirect.com/science/article/pii/S0952197619301745>.

LEACOCK, C.; CHODOROW, M. Combining local context and wordnet similarity for word sense identification. **Wordnet: An electronic lexical database**, v. 49, n. 2, p. 265–283, 1998.

Li, Y.; Bandar, Z. A.; Mclean, D. An approach for measuring semantic similarity between words using multiple information sources. **IEEE Transactions on Knowledge and Data Engineering**, v. 15, n. 4, p. 871–882, July 2003. ISSN 1041-4347.

LIKAVEC, S.; LOMBARDI, I.; CENA, F. Sigmoid similarity - a new feature-based similarity measure. **Information Sciences**, Elsevier, v. 481, p. 203–218, 2019.

LIN, D. An information-theoretic definition of similarity. In: **Proceedings of the Fifteenth International Conference on Machine Learning**. San Francisco, CA, USA: Morgan Kaufmann Publishers Inc., 1998. (ICML '98), p. 296–304. ISBN 1-55860-556-8.

Liu, X.; Zhou, Y.; Zheng, R. Measuring semantic similarity in wordnet. In: **2007 International Conference on Machine Learning and Cybernetics**. [S.l.: s.n.], 2007. v. 6, p. 3431–3435. ISSN 2160-133X.

LORENZATTI, A. et al. Ontology for imagistic domains: Combining textual and pictorial primitives. In: HEUSER, C. A.; PERNUL, G. (Ed.). **Advances in Conceptual Modeling - Challenging Perspectives**. Berlin, Heidelberg: Springer Berlin Heidelberg, 2009. p. 169–178. ISBN 978-3-642-04947-7.

MCINNES, B. T.; PEDERSEN, T. Evaluating measures of semantic similarity and relatedness to disambiguate terms in biomedical text. **Journal of biomedical informatics**, Elsevier, v. 46, n. 6, p. 1116–1124, 2013.

MENG, L.; GU, J.; ZHOU, Z. A new model of information content based on concept's topology for measuring semantic similarity in wordnet. **International Journal of Grid and Distributed Computing**, v. 5, n. 3, p. 81–94, 2012.

MILLER, G. A. Wordnet: a lexical database for english. **Communications of the ACM**, ACM, v. 38, n. 11, p. 39–41, 1995.

MUNIR, K.; ANJUM, M. S. The use of ontologies for effective knowledge modelling and information retrieval. **Applied Computing and Informatics**, v. 14, n. 2, p. 116 – 126, 2018. ISSN 2210-8327. Available from Internet: <http://www.sciencedirect.com/science/article/pii/S2210832717300649>.

PATWARDHAN, S.; BANERJEE, S.; PEDERSEN, T. Using measures of semantic relatedness for word sense disambiguation. In: SPRINGER. **International Conference on Intelligent Text Processing and Computational Linguistics**. [S.l.], 2003. p. 241–257.

PETRAKIS, E. G. M. et al. X-similarity: Computing semantic similarity between concepts from different ontologies. **Journal of Digital Information Management (JDIM**, v. 4, 2006.

PIRRÓ, G.; EUZENAT, J. A feature and information theoretic framework for semantic similarity and relatedness. In: **The Semantic Web – ISWC 2010**. Berlin, Heidelberg: Springer Berlin Heidelberg, 2010. p. 615–630. ISBN 978-3-642-17746-0.

PIRRÓ, G.; SECO, N. Design, implementation and evaluation of a new semantic similarity metric combining features and intrinsic information content. In: SPRINGER. **OTM Confederated International Conferences" On the Move to Meaningful Internet Systems"**. [S.l.], 2008. p. 1271–1288.

QUINTERO, R. et al. Dis-c: conceptual distance in ontologies, a graph-based approach. **Knowledge and Information Systems**, Springer, p. 1–33, 2018.

RADA, R. et al. Development and application of a metric on semantic nets. **IEEE transactions on systems, man, and cybernetics**, IEEE, v. 19, n. 1, p. 17–30, 1989.

RESNIK, P. Using information content to evaluate semantic similarity in a taxonomy. In: **Proceedings of the 14th International Joint Conference on Artificial Intelligence - Volume 1**. San Francisco, CA, USA: Morgan Kaufmann Publishers Inc., 1995. (IJCAI'95), p. 448–453.

Rodriguez, M. A.; Egenhofer, M. J. Determining semantic similarity among entity classes from different ontologies. **IEEE Transactions on Knowledge and Data Engineering**, v. 15, n. 2, p. 442–456, March 2003. ISSN 1041-4347.

SÁNCHEZ, D.; BATET, M. A new model to compute the information content of concepts from taxonomic knowledge. **Int. J. Semant. Web Inf. Syst.**, IGI Global, Hershey, PA, USA, v. 8, n. 2, p. 34–50, abr. 2012. ISSN 1552-6283. Available from Internet: <http://dx.doi.org/10.4018/jswis.2012040102>.

SÁNCHEZ, D.; BATET, M.; ISERN, D. Ontology-based information content computation. **Knowledge-Based Systems**, Elsevier, v. 24, n. 2, p. 297–303, 2011.

SÁNCHEZ, D. et al. Ontology-based semantic similarity: A new feature-based approach. **Expert systems with applications**, Elsevier, v. 39, n. 9, p. 7718–7728, 2012.

SECO, N.; VEALE, T.; HAYES, J. An intrinsic information content metric for semantic similarity in wordnet. In: **ECAI**. [S.l.: s.n.], 2004. v. 16, p. 1089.

SIMOES, M. d. G. M. et al. Indexação automática e ontologias: identificação dos contributos convergentes na ciência da informação. **Ciência da Informação**, v. 46, n. 1, 2017.

STUDER, R.; BENJAMINS, V. R.; FENSEL, D. Knowledge engineering: Principles and methods. **Data Knowl. Eng.**, Elsevier Science Publishers B. V., Amsterdam, The Netherlands, The Netherlands, v. 25, n. 1-2, p. 161–197, mar. 1998. ISSN 0169-023X. Available from Internet: <http://dx.doi.org/10.1016/S0169-023X(97)00056-6>.

SáNCHEZ, D.; BATET, M. Semantic similarity estimation in the biomedical domain: An ontology-based information-theoretic perspective. **Journal of Biomedical Informatics**, v. 44, n. 5, p. 749 – 759, 2011. ISSN 1532-0464. Available from Internet: <http://www.sciencedirect.com/science/article/pii/S1532046411000645>.

TVERSKY, A. Features of similarity. **Psychological review**, American Psychological Association, v. 84, n. 4, p. 327, 1977.

WEI, T.; CHANG, H. Measuring word semantic relatedness using wordnet-based approach. **JCP**, v. 10, n. 4, p. 252–259, 2015.

WU, Z.; PALMER, M. Verbs semantics and lexical selection. In: ASSOCIATION FOR COMPUTATIONAL LINGUISTICS. **Proceedings of the 32nd annual meeting on Association for Computational Linguistics**. [S.l.], 1994. p. 133–138.

ZHANG, X.; SUN, S.; ZHANG, K. An information content-based approach for measuring concept semantic similarity in wordnet. **Wireless Personal Communications**, v. 103, n. 1, p. 117–132, Nov 2018. ISSN 1572-834X. Available from Internet: <https://doi.org/10.1007/s11277-018-5429-7>.

ZHOU, Z.; WANG, Y.; GU, J. A new model of information content for semantic similarity in wordnet. In: IEEE. **Future Generation Communication and Networking Symposia, 2008. FGCNS'08. Second International Conference on**. [S.l.], 2008. v. 3, p. 85–89.

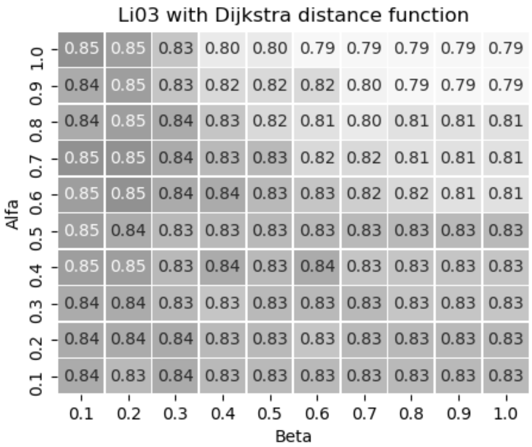
ZHU, B.; LI, X.; SANCHO, J. B. A novel asymmetric semantic similarity measurement for semantic job matching. In: IEEE. **2017 International Conference on Security, Pattern Analysis, and Cybernetics (SPAC)**. [S.l.], 2017. p. 152–157.

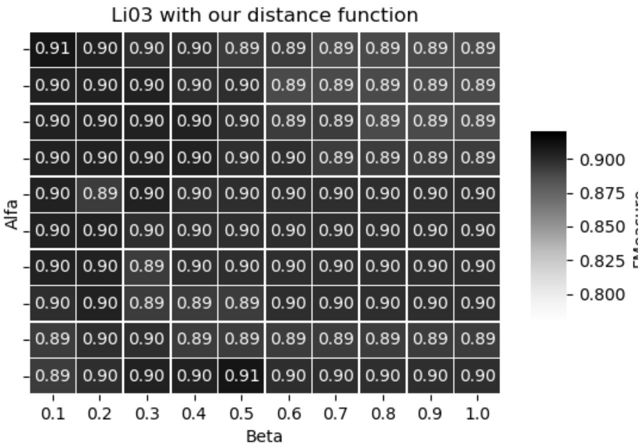
ZHU, G.; IGLESIAS, C. A. Computing semantic similarity of concepts in knowledge graphs. **IEEE Transactions on Knowledge and Data Engineering**, v. 29, n. 1, p. 72–85, Jan 2017. ISSN 1041-4347.

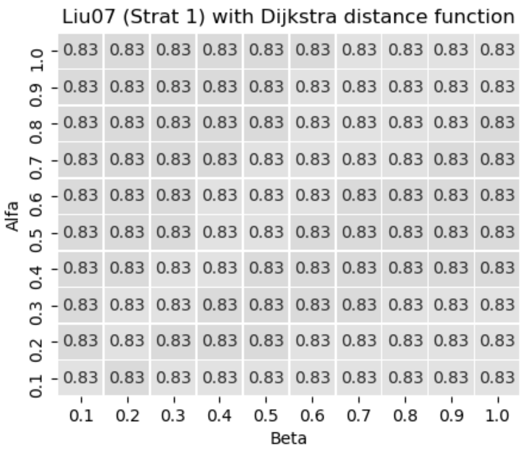
ZHU, G.; IGLESIAS, C. A. Exploiting semantic similarity for named entity disambiguation in knowledge graphs. **Expert Systems with Applications**, v. 101, p. 8 – 24, 2018. ISSN 0957-4174. Available from Internet: <http://www.sciencedirect.com/science/article/pii/S0957417418300897>.

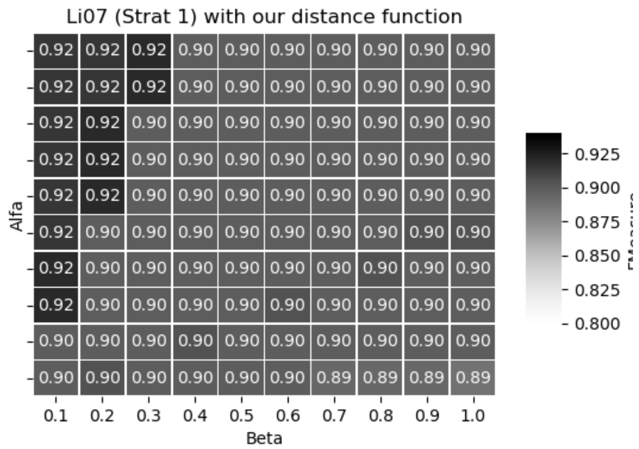
APPENDIX A - THE WSD RESULTS WITH ANOTHER PARAMETER VALUES

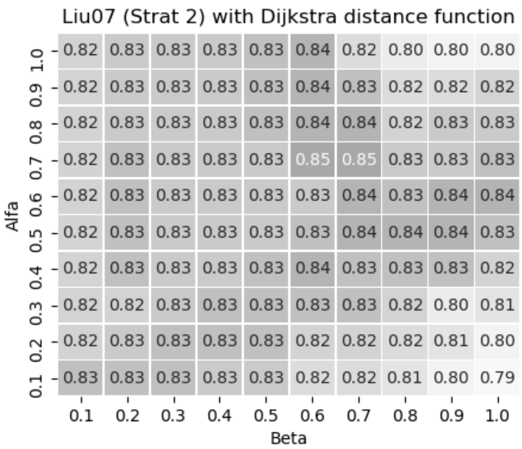
The Word Sense Disambiguation Results in Dataset D1

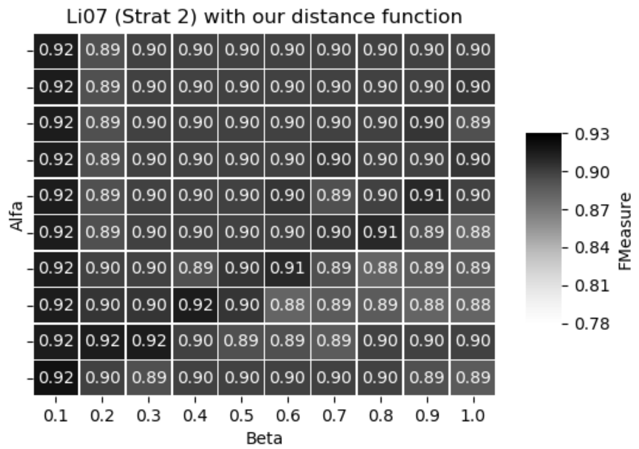


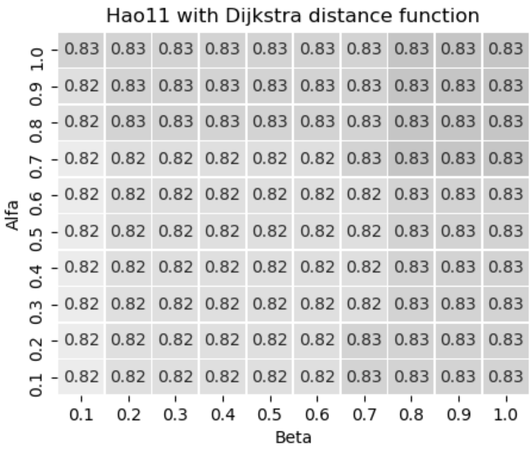


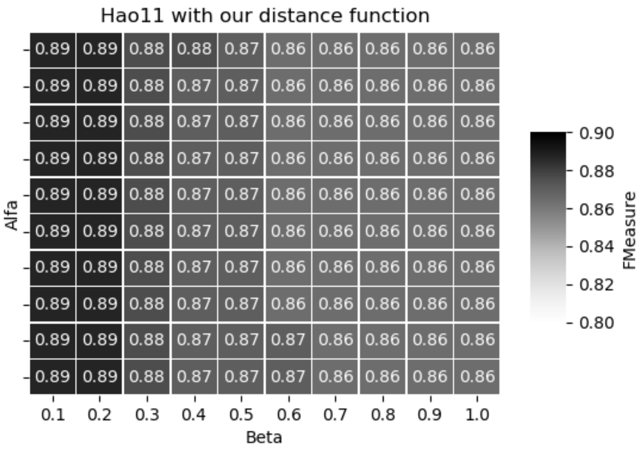


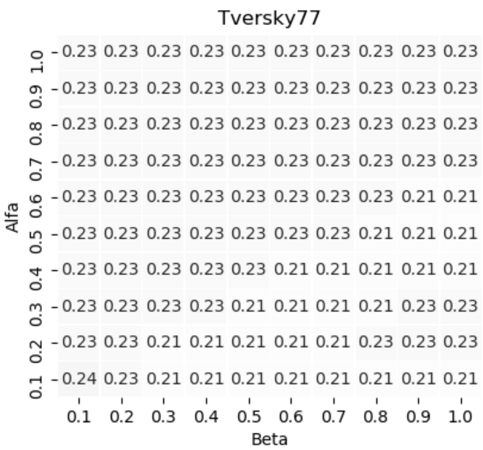


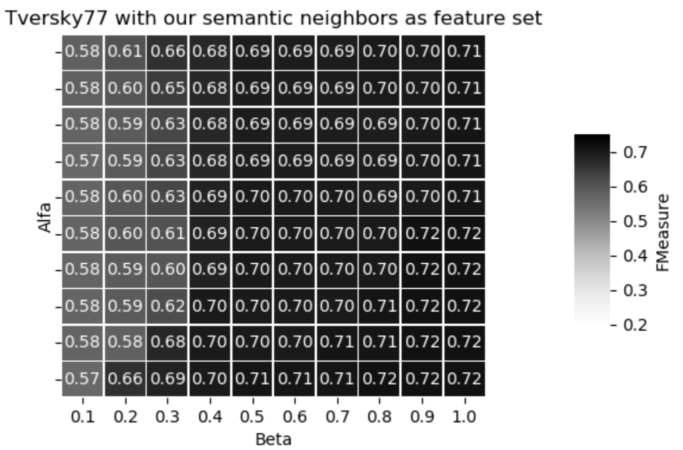


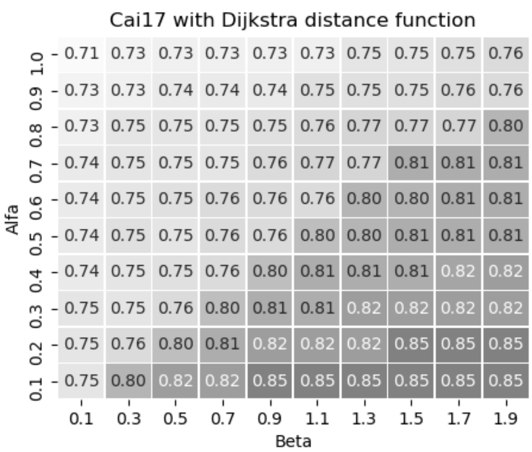


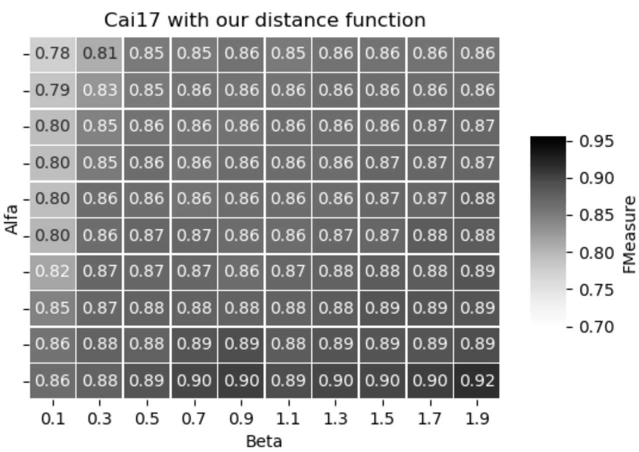

Hao11 with Dijkstra distance function
Alfa
Beta
0.1 0.2 0.3 0.4 0.5 0.6 0.7 0.8 0.9 1.0


Hao11 with our distance function
Alfa
Beta
0.1 0.2 0.3 0.4 0.5 0.6 0.7 0.8 0.9 1.0
FMeasure
0.90 0.88 0.86 0.84 0.82 0.80

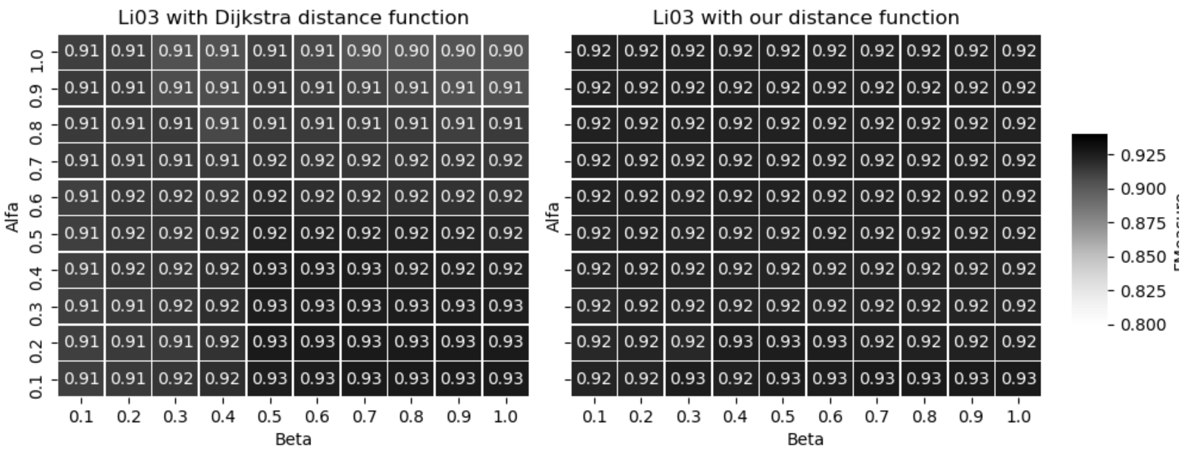

Tversky77
Alfa
Beta
0.1 0.2 0.3 0.4 0.5 0.6 0.7 0.8 0.9 1.0

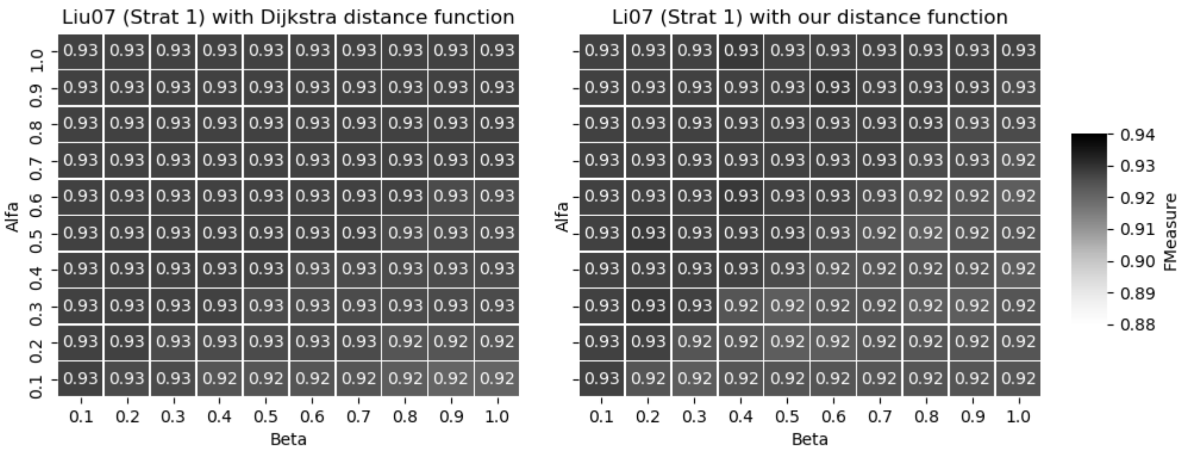

Tversky77 with our semantic neighbors as feature set
Alfa
Beta
0.1 0.2 0.3 0.4 0.5 0.6 0.7 0.8 0.9 1.0
FMeasure
0.7 0.6 0.5 0.4 0.3 0.2

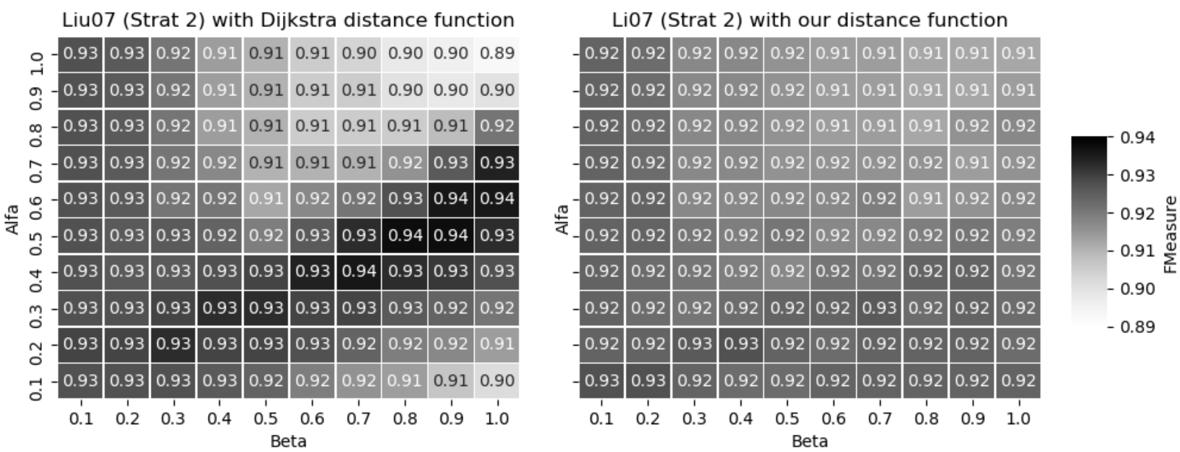

Cai17 with Dijkstra distance function
Alfa
Beta
0.1 0.3 0.5 0.7 0.9 1.1 1.3 1.5 1.7 1.9

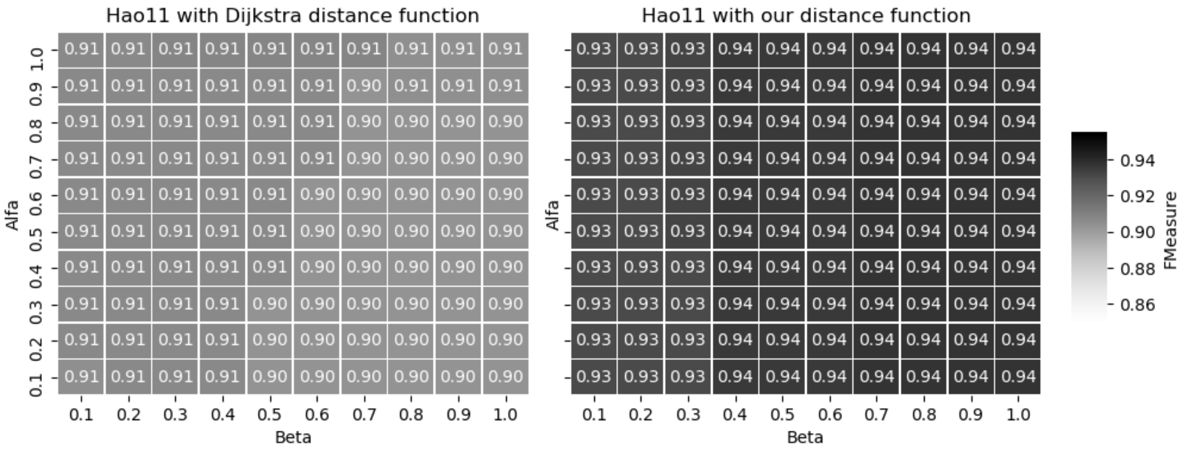

Cai17 with our distance function
Alfa
Beta
0.1 0.3 0.5 0.7 0.9 1.1 1.3 1.5 1.7 1.9
FMeasure
0.95 0.90 0.85 0.80 0.75 0.70

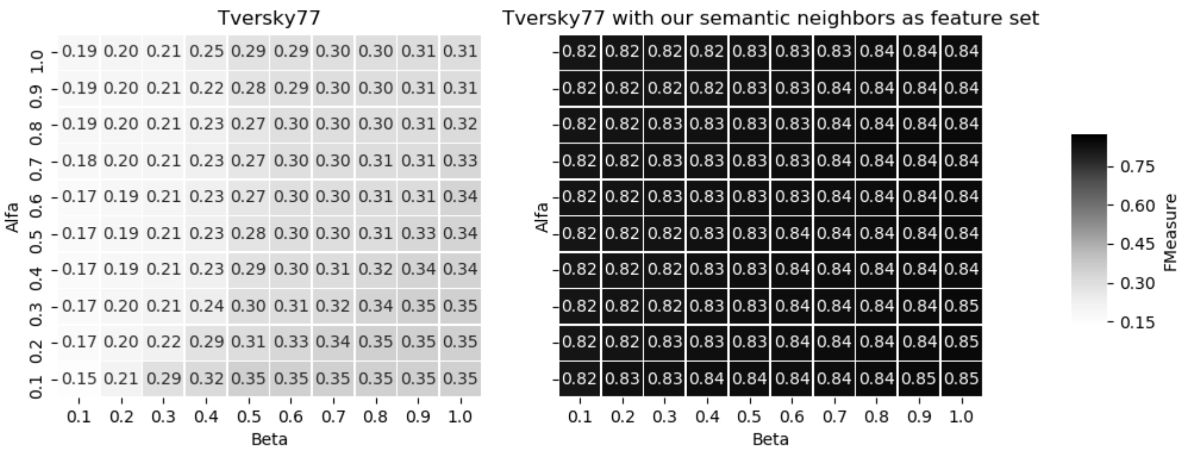
The Word Sense Disambiguation Results in Dataset D2

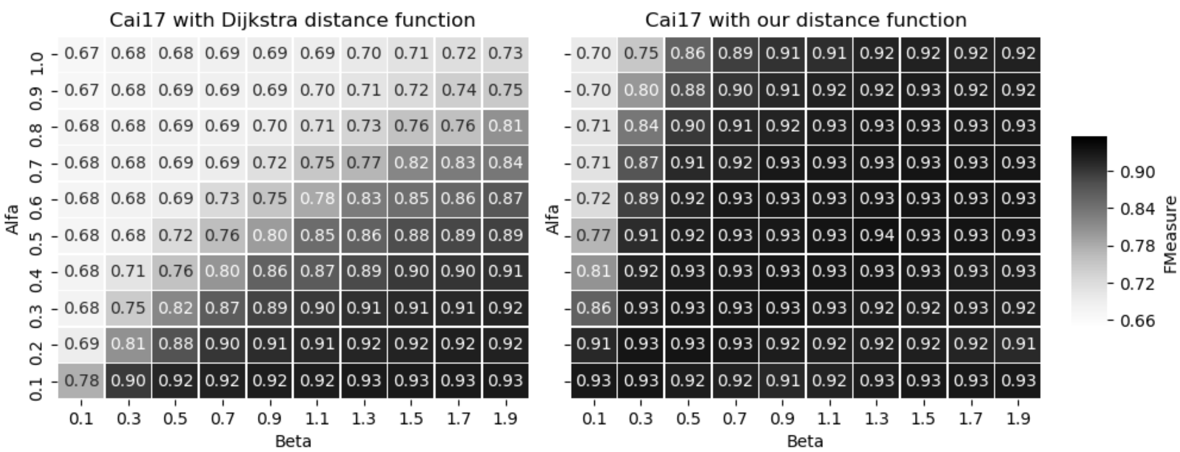




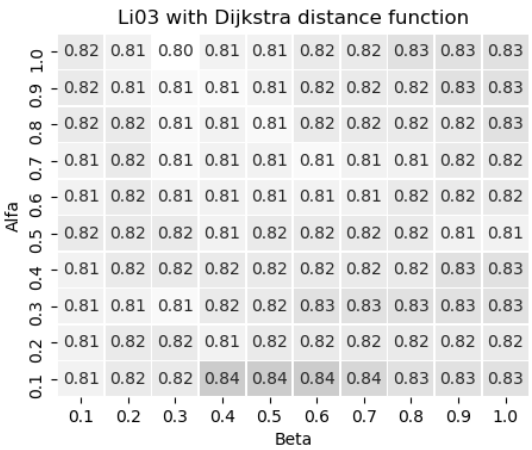


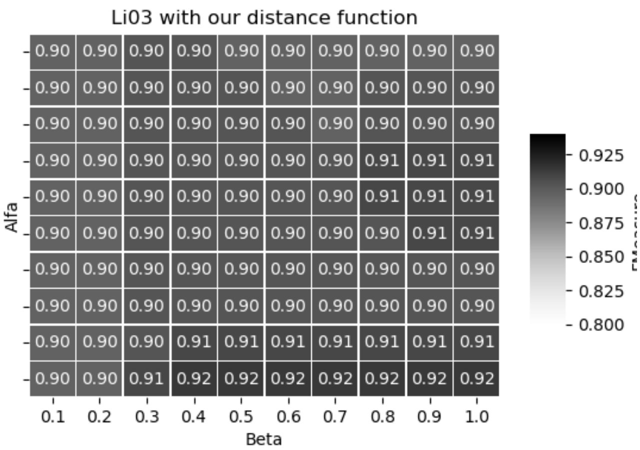

Hao11 with Dijkstra distance function
Hao11 with our distance function
Alfa
Beta
0.1 0.2 0.3 0.4 0.5 0.6 0.7 0.8 0.9 1.0
FMeasure
0.94 0.92 0.90 0.88 0.86

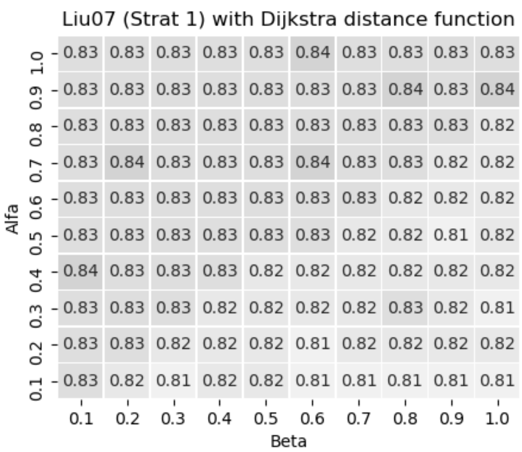

Tversky77
Tversky77 with our semantic neighbors as feature set
Alfa
Beta
0.1 0.2 0.3 0.4 0.5 0.6 0.7 0.8 0.9 1.0
FMeasure
0.75 0.60 0.45 0.30 0.15

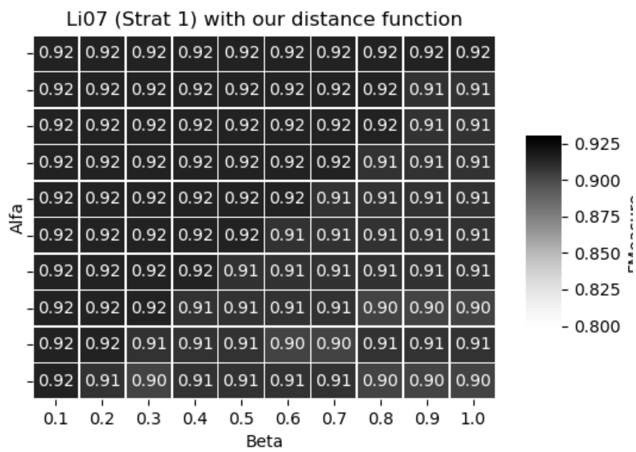

Cai17 with Dijkstra distance function
Cai17 with our distance function
Alfa
0.1 0.2 0.3 0.4 0.5 0.6 0.7 0.8 0.9 1.0
Beta
0.1 0.3 0.5 0.7 0.9 1.1 1.3 1.5 1.7 1.9
FMeasure
0.90 0.84 0.78 0.72 0.66

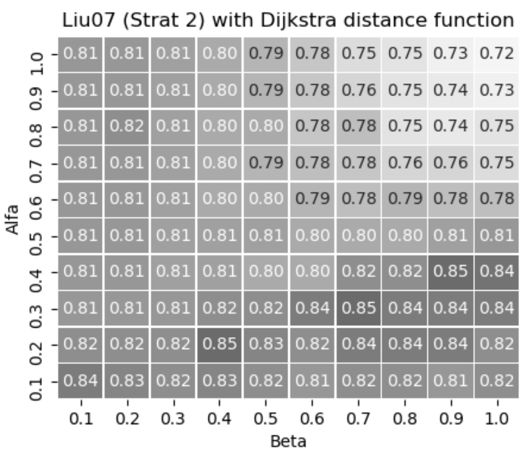
The Word Sense Disambiguation Results in Dataset D3

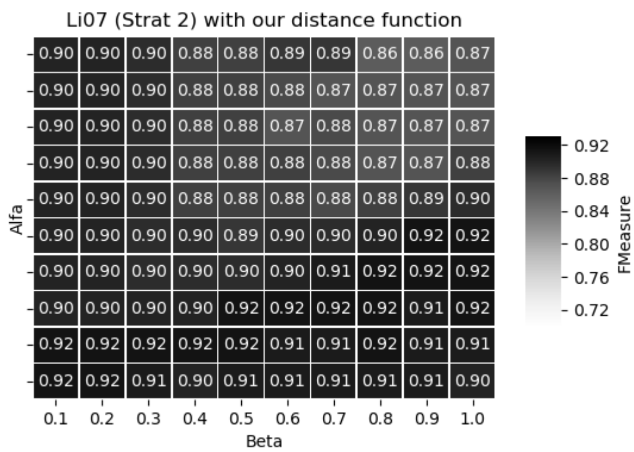


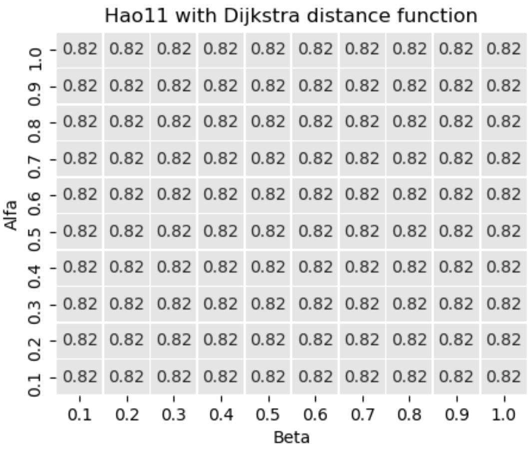


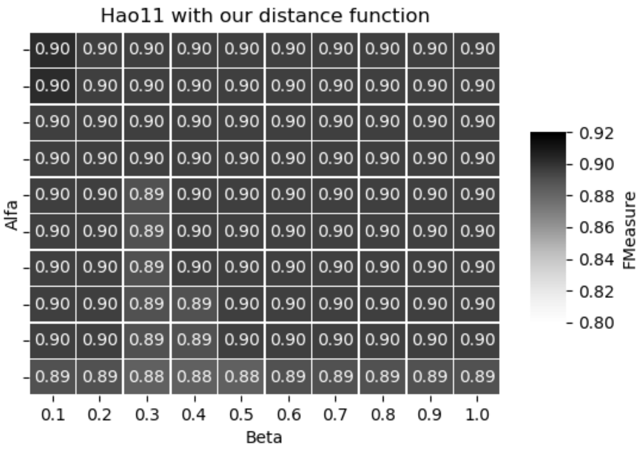


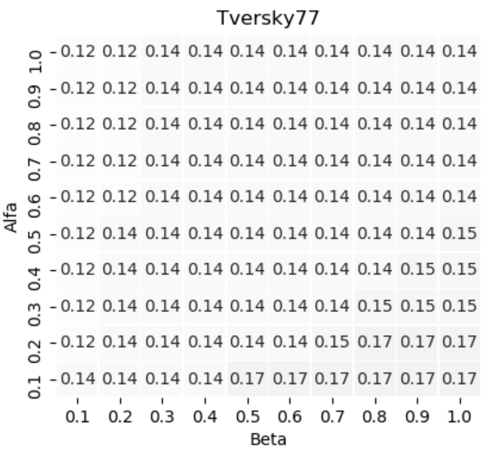


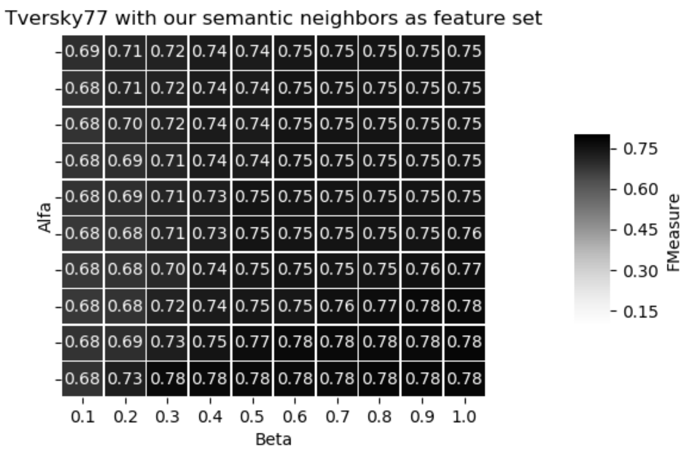


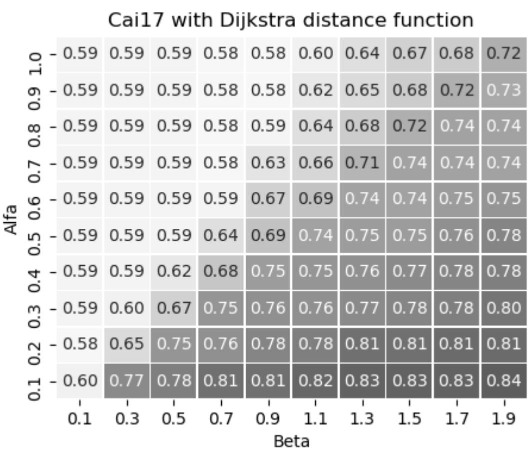


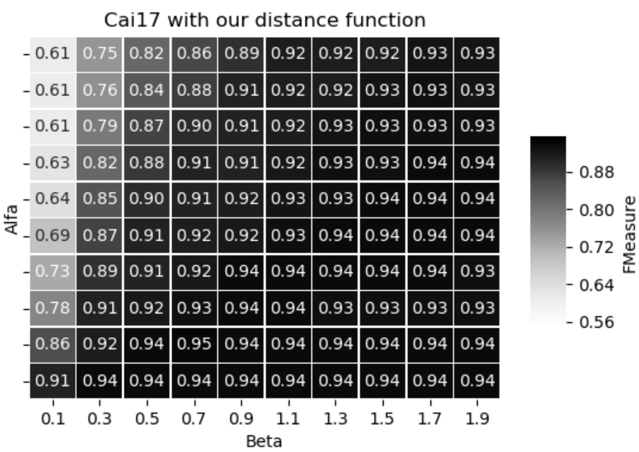

Hao11 with Dijkstra distance function
1.0 0.82 0.82 0.82 0.82 0.82 0.82 0.82 0.82 0.82 0.82
0.9 0.82 0.82 0.82 0.82 0.82 0.82 0.82 0.82 0.82 0.82
0.8 0.82 0.82 0.82 0.82 0.82 0.82 0.82 0.82 0.82 0.82
0.7 0.82 0.82 0.82 0.82 0.82 0.82 0.82 0.82 0.82 0.82
0.6 0.82 0.82 0.82 0.82 0.82 0.82 0.82 0.82 0.82 0.82
0.5 0.82 0.82 0.82 0.82 0.82 0.82 0.82 0.82 0.82 0.82
0.4 0.82 0.82 0.82 0.82 0.82 0.82 0.82 0.82 0.82 0.82
0.3 0.82 0.82 0.82 0.82 0.82 0.82 0.82 0.82 0.82 0.82
0.2 0.82 0.82 0.82 0.82 0.82 0.82 0.82 0.82 0.82 0.82
0.1 0.82 0.82 0.82 0.82 0.82 0.82 0.82 0.82 0.82 0.82
Alfa
0.1 0.2 0.3 0.4 0.5 0.6 0.7 0.8 0.9 1.0
Beta


Hao11 with our distance function
1.0 0.90 0.90 0.90 0.90 0.90 0.90 0.90 0.90 0.90 0.90
0.9 0.90 0.90 0.90 0.90 0.90 0.90 0.90 0.90 0.90 0.90
0.8 0.90 0.90 0.90 0.90 0.90 0.90 0.90 0.90 0.90 0.90
0.7 0.90 0.90 0.90 0.90 0.90 0.90 0.90 0.90 0.90 0.90
0.6 0.90 0.90 0.89 0.90 0.90 0.90 0.90 0.90 0.90 0.90
0.5 0.90 0.90 0.89 0.90 0.90 0.90 0.90 0.90 0.90 0.90
0.4 0.90 0.90 0.89 0.90 0.90 0.90 0.90 0.90 0.90 0.90
0.3 0.90 0.90 0.89 0.89 0.90 0.90 0.90 0.90 0.90 0.90
0.2 0.90 0.90 0.89 0.89 0.90 0.90 0.90 0.90 0.90 0.90
0.1 0.89 0.89 0.88 0.88 0.88 0.89 0.89 0.89 0.89 0.89
Alfa
0.1 0.2 0.3 0.4 0.5 0.6 0.7 0.8 0.9 1.0
Beta
0.92 0.90 0.88 0.86 0.84 0.82 0.80
FMeasure

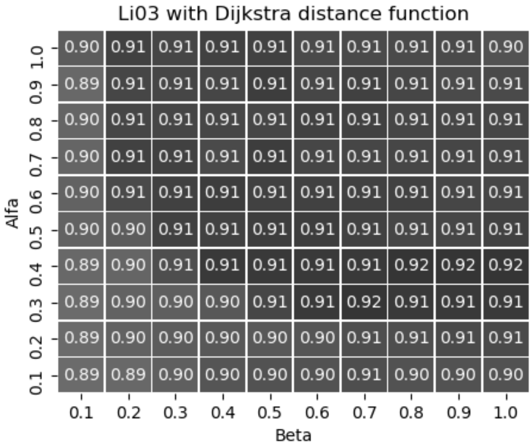

Tversky77
1.0 0.12 0.12 0.14 0.14 0.14 0.14 0.14 0.14 0.14 0.14
0.9 0.12 0.12 0.14 0.14 0.14 0.14 0.14 0.14 0.14 0.14
0.8 0.12 0.12 0.14 0.14 0.14 0.14 0.14 0.14 0.14 0.14
0.7 0.12 0.12 0.14 0.14 0.14 0.14 0.14 0.14 0.14 0.14
0.6 0.12 0.12 0.14 0.14 0.14 0.14 0.14 0.14 0.14 0.14
0.5 0.12 0.14 0.14 0.14 0.14 0.14 0.14 0.14 0.14 0.15
0.4 0.12 0.14 0.14 0.14 0.14 0.14 0.14 0.14 0.15 0.15
0.3 0.12 0.14 0.14 0.14 0.14 0.14 0.14 0.15 0.15 0.15
0.2 0.12 0.14 0.14 0.14 0.14 0.14 0.15 0.17 0.17 0.17
0.1 0.14 0.14 0.14 0.14 0.17 0.17 0.17 0.17 0.17 0.17
Alfa
0.1 0.2 0.3 0.4 0.5 0.6 0.7 0.8 0.9 1.0
Beta

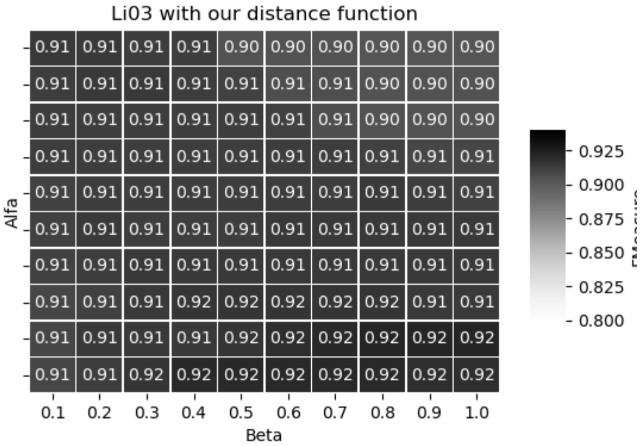

Tversky77 with our semantic neighbors as feature set
0.69 0.71 0.72 0.74 0.74 0.75 0.75 0.75 0.75 0.75
0.68 0.71 0.72 0.74 0.74 0.75 0.75 0.75 0.75 0.75
0.68 0.70 0.72 0.74 0.74 0.75 0.75 0.75 0.75 0.75
0.68 0.69 0.71 0.74 0.74 0.75 0.75 0.75 0.75 0.75
0.68 0.69 0.71 0.73 0.75 0.75 0.75 0.75 0.75 0.75
0.68 0.68 0.71 0.73 0.75 0.75 0.75 0.75 0.75 0.76
0.68 0.68 0.70 0.74 0.75 0.75 0.75 0.75 0.76 0.77
0.68 0.68 0.72 0.74 0.75 0.75 0.76 0.77 0.78 0.78
0.68 0.69 0.73 0.75 0.77 0.78 0.78 0.78 0.78 0.78
0.68 0.73 0.78 0.78 0.78 0.78 0.78 0.78 0.78 0.78
Alfa
0.1 0.2 0.3 0.4 0.5 0.6 0.7 0.8 0.9 1.0
Beta
0.75 0.60 0.45 0.30 0.15
FMeasure

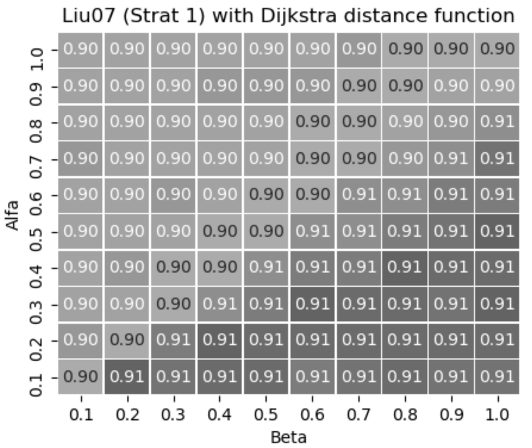

Cai17 with Dijkstra distance function
1.0 0.59 0.59 0.59 0.58 0.58 0.60 0.64 0.67 0.68 0.72
0.9 0.59 0.59 0.59 0.58 0.58 0.62 0.65 0.68 0.72 0.73
0.8 0.59 0.59 0.59 0.58 0.59 0.64 0.68 0.72 0.74 0.74
0.7 0.59 0.59 0.59 0.58 0.63 0.66 0.71 0.74 0.74 0.74
0.6 0.59 0.59 0.59 0.59 0.67 0.69 0.74 0.74 0.75 0.75
0.5 0.59 0.59 0.59 0.64 0.69 0.74 0.75 0.75 0.76 0.78
0.4 0.59 0.59 0.62 0.68 0.75 0.75 0.76 0.77 0.78 0.78
0.3 0.59 0.60 0.67 0.75 0.76 0.76 0.77 0.78 0.78 0.80
0.2 0.58 0.65 0.75 0.76 0.78 0.78 0.81 0.81 0.81 0.81
0.1 0.60 0.77 0.78 0.81 0.81 0.82 0.83 0.83 0.83 0.84
Alfa
0.1 0.3 0.5 0.7 0.9 1.1 1.3 1.5 1.7 1.9
Beta

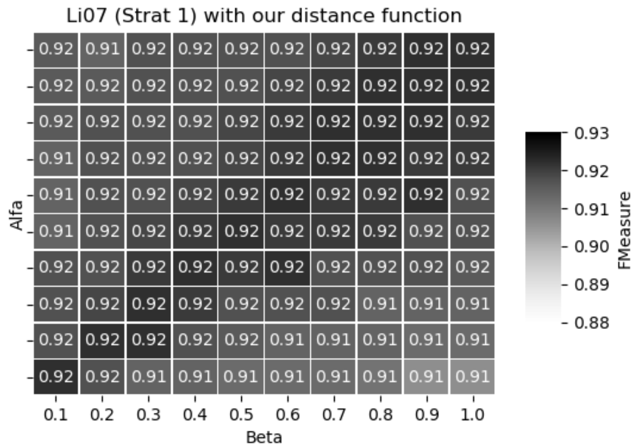

Cai17 with our distance function
0.61 0.75 0.82 0.86 0.89 0.92 0.92 0.92 0.93 0.93
0.61 0.76 0.84 0.88 0.91 0.92 0.92 0.93 0.93 0.93
0.61 0.79 0.87 0.90 0.91 0.92 0.93 0.93 0.93 0.93
0.63 0.82 0.88 0.91 0.91 0.92 0.93 0.93 0.94 0.94
0.64 0.85 0.90 0.91 0.92 0.93 0.93 0.94 0.94 0.94
0.69 0.87 0.91 0.92 0.92 0.93 0.94 0.94 0.94 0.94
0.73 0.89 0.91 0.92 0.94 0.94 0.94 0.94 0.94 0.93
0.78 0.91 0.92 0.93 0.94 0.94 0.93 0.93 0.93 0.93
0.86 0.92 0.94 0.95 0.94 0.94 0.94 0.94 0.94 0.94
0.91 0.94 0.94 0.94 0.94 0.94 0.94 0.94 0.94 0.94
Alfa
0.1 0.3 0.5 0.7 0.9 1.1 1.3 1.5 1.7 1.9
Beta
0.88 0.80 0.72 0.64 0.56
FMeasure

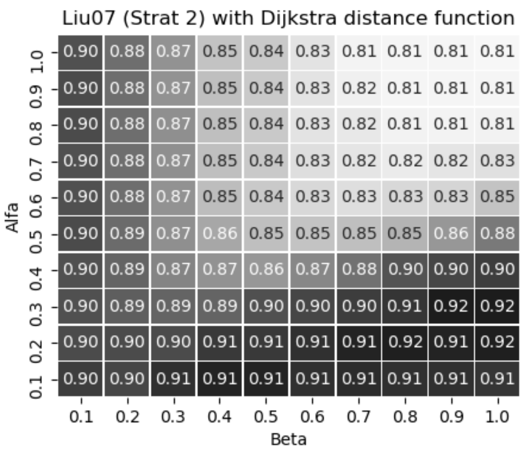
The Word Sense Disambiguation Results in Dataset D4

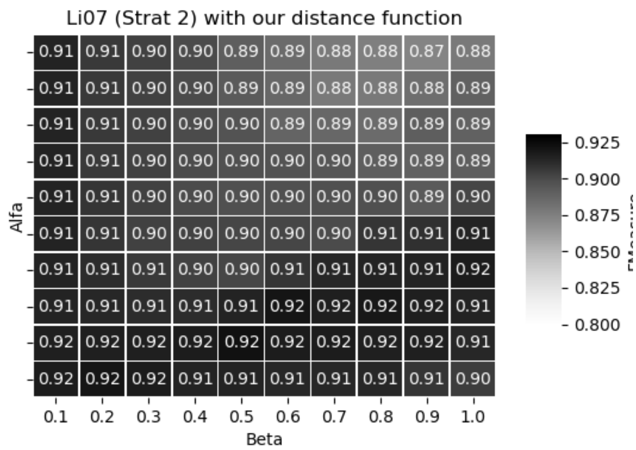


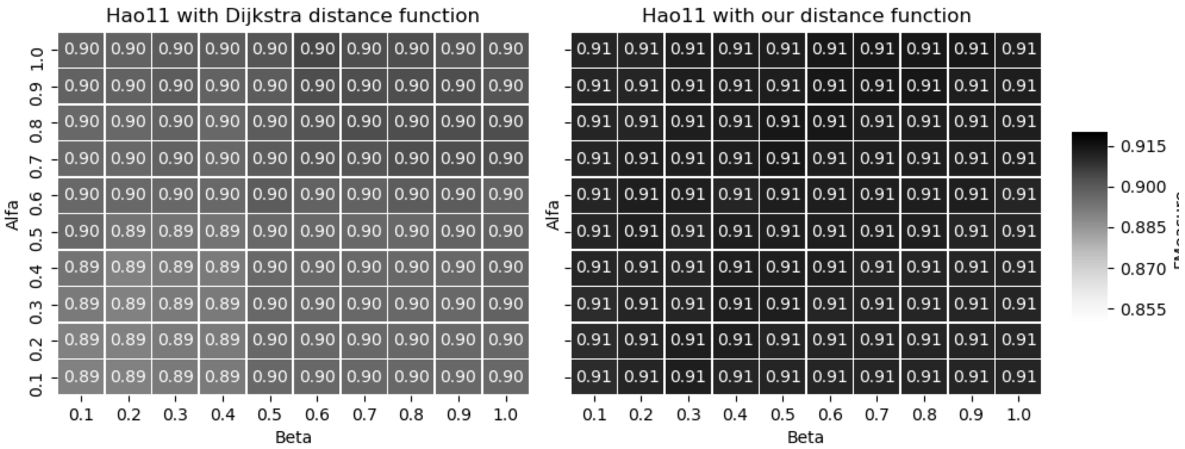


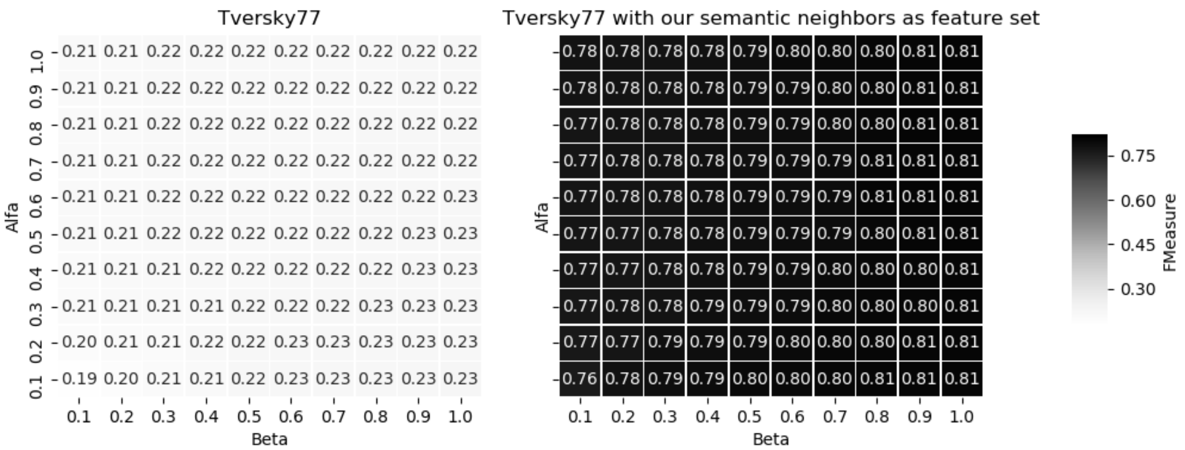


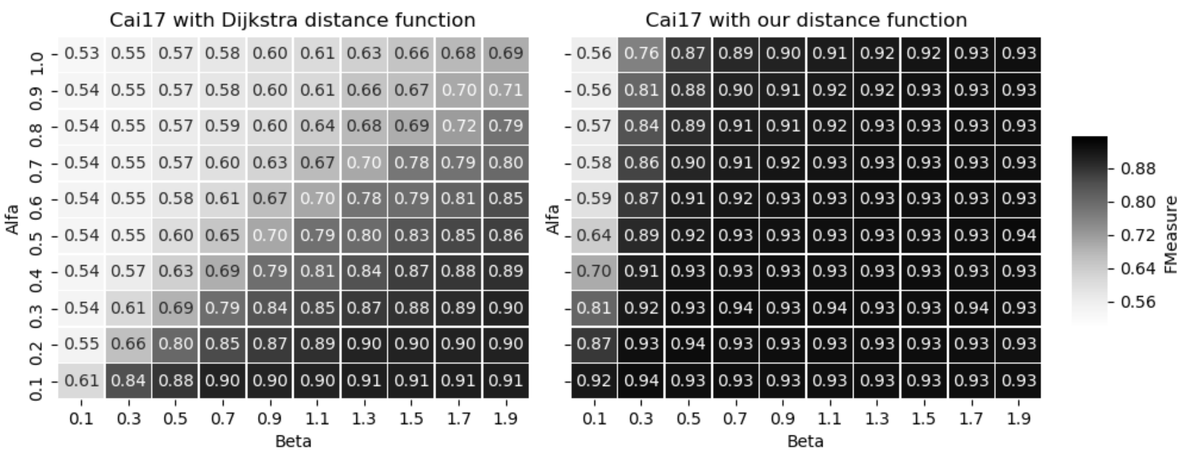







Hao11 with Dijkstra distance function
Hao11 with our distance function
Alfa
Beta
FMeasure


Tversky77
Tversky77 with our semantic neighbors as feature set
Alfa
Beta
FMeasure


Cai17 with Dijkstra distance function
Cai17 with our distance function
Alfa
Beta
FMeasure