UNIVERSIDADE DE SÃO PAULO
FFCLRP–DEPARTAMENTO DE COMPUTAÇÃO E MATEMÁTICA
PROGRAMA DE PÓS-GRADUAÇÃO EM COMPUTAÇÃO APLICADA

"Similarity algorithms for Heterogeneous Information Networks"
"Algoritmos de similaridade para Redes de Informações Heterogêneas"

Angélica Abadia Paulista Ribeiro

Dissertação apresentada à Faculdade de Filosofia, Ciências e Letras de Ribeirão Preto da USP, como parte das exigências para a obtenção do título de Mestre em Ciências, Área: Computação Aplicada.

Ribeirão Preto–SP
2018

Angélica Abadia Paulista Ribeiro

"Similarity algorithms for Heterogeneous Information Networks"
"Algoritmos de similaridade para Redes de Informações Heterogêneas"

Versão Corrigida
A Versão original se encontra disponível
na Unidade que aloja o Programa.

Modelo canônico de trabalho monográfico acadêmico em conformidade com as normas ABNT.

Supervisor: Alessandra Alaniz Macedo

Ribeirão Preto–SP

2018

Angélica Abadia Paulista Ribeiro
Similarity algorithms for Heterogeneous Information Networks. Ribeirão Preto–SP, 2018.
109p. : il.; 30 cm.

Dissertação apresentada à Faculdade de Filosofia, Ciências e Letras
de Ribeirão Preto da USP, como parte das exigências para
a obtenção do título de Mestre em Ciências,
Área: Computação Aplicada.

Supervisor: Alessandra Alaniz Macedo

1. Heterogeneous Information Networks. 2. Similarity Measures. 3. Meta-Path.

Angélica Abadia Paulista Ribeiro

"Similarity algorithms for Heterogeneous Information Networks"
"Algoritmos de similaridade para Redes de Informações Heterogêneas"

Modelo canônico de trabalho monográfico acadêmico em conformidade com as normas ABNT.

Trabalho aprovado. Ribeirão Preto–SP, 28 de janeiro de 2019:

Alessandra Alaniz Macedo
Orientadora

Professor
Dr. Alexandre Souto Martinez

Professora
Dra. Marinalva Dias Soares

Professora
Dra. Renata Pontin de Mattos Fortes

Ribeirão Preto–SP
2018

To my parents, my bothers, my family and friends.

Acknowledgements

First and foremost, I would like to pay Tribune to God and Hail Mary who gave me strength and patience in this journey. Thank you for the opportunity to live this fulfilling experience my life has been.

For my family, you all know I will never be able to thank you enough. Dad, keep in mind that I truly believe no living being could ever be more fortunate than me for having you as my papa. Mommy, you took my peace with all the pieces of advice, keeping up with your recommendation, was almost harder than doing the research itself, but again, you are truly a mom, I couldn't wish for a better one. Thank you, both for always supporting me and thank you for the education and the love you always gave me. To my siblings, Adrielle and Carlos Eduardo, Drika since we were kids you bared my mp4 and have hid and cut countless toy animals from my zoo and farm collections driving me insane, but if anyone messes with me, I could just be sorry for this poor fellow. Dudu, always riding with me wherever I needed to go, and always helping me to find the coolest tech and sports gadgets ever. However, I know better, the worst of you two, you are the best siblings anyone could ever wish for. You guys are awesome. Aunt Eni and Nona, thanks for your attention and for the support that you gave me along my entire life, aunt Eni especially when you came to stay with me in the first months in the city. I know you know how much I love you. Grandma Nenzinha you are a dream grandmother. Thank you, both a lot for the calls, for the Minas authentic sweeties and foremost, for all your love. To my first Godson Joao Pedro who is enchanting my life and making it more colorful than ever, making me discover a whole new world through a child's eyes.

I would like to thank the dearest friends who I had the honor of meeting in Ribeirão Preto, First, I would like to thank first Rafael Delalibera, for listening and talking to me, thanks for taking care of me like an older brother and of course for the support and wonderful advices regarding my work and academic life and for calming me down when times were tough. To Evandro Ruiz the kindest and the most gentle and wonderful friend ever and my brother in music, thank you for the loyal friendship, all conversations funny moments, and advice. I also would like to thank Gilberto Nakamura, for the conversations, for the scientific support and for calming me in many difficult situations. I'd like to thank Luiz Spin, the first person held out his hand to this "foreign from other Country" when I first came here. The sweetest person in the entire USP. Thanks for all great chats and relaxed moments, you really was a lifesaver during my days here. I'd like to give special thanks to Patricia and Joseph for listening to my "noias".

Also to all my predecessor professors from UFU in special ones that are my best friends, Daniel Abdala, Marcos Bueno, Daniele Oliveira and Kil Park. They are the guys who have shown me that with science we can make the world better. I remember well

when I first take their classes, and read one of their papers and have thought: “I want to be like them someday! I would like to be able to do my research this way!” In fact, they’ve thought me many things regarding the academic life, ranging from how to clearly address an idea to scientific events organization, and make a serious research. I also would like to thank you four for making me a better person not in science but as a human being. The four years of my life with you at UFU gave me to experience the prepared me for this two years at USP gave me a clear insight of how a scientific research works. Also to all the teachers who have passed through my life. To my friends, for the encouragement and constant support.

To the University of São Paulo, for the opportunity to study and develop this project. I would like to express my sincere thanks to the CAPES for the financial support). This study was financed in part by the Coordenação de Aperfeiçoamento de Pessoal de Nível Superior - Brasil (CAPES) - Finance Code 001. My research would never be possible without the continuous support provided by this Institution, especially to the Coordination from DCM 2015-2017 which provided me the scholarship. I’m honored to be one of the researchers chosen to be founded (process number 1569180), and I hope to be able to contribute to the research Brazilian community in the years to follow. Also to my advisor, Prof. Dr. Alessandra Alaniz Macedo, for her contribution to the accomplishment of this work and for having contributed to my formation.

I would like to thank my adviser, Prof. Dr. Alessandra Alaniz Macedo, for her continuous support and advisements regarding my work. I was truly an honor and a life experience to work with her. Professor Alessandra, as I usually call her, being very understandable when I had to take a short leave for health reasons, and for that, I will always be thankful to her. And also to Professor Renato Bulcão who helped me a lot during this journey.

To the secretaries and technicians of the Department of Computation and Mathematics and the Programa de Pós graduação Aplicada for support with bureaucratic and technical questions. In especially, I will never be able to thank enough Daniela for all the times she solved my bureaucratic problems here in USP. The help she dispensed with me, all the advisement’s regarding the necessary documentation to settle up in college. To the Jalmei for always helping when we needed in the lab, supplying equipment needs.

Finally, to all my colleagues in the laboratories and master course, my sincere thanks who supported me during the whole project, I leave my thanks for their support in the difficult moments of the design and laughs and chat. It was a rich experience to work with you all. In special, I would like to thank my closest colleagues, namely Luciana Almanca, Alinne Corrêa and Gilberto Nakamura, Flavio, and Ronem . I’d like to thank Luciana Almanca, Gabriel Rubio, and Gilvan, for the devices and conversations and exchange of academic experiences and supporting which others in earlier lab times.

"Não vos amoldeis às estruturas deste mundo,
mas transformai-vos pela renovação da mente,
a fim de distinguir qual é a vontade de Deus:
o que é bom, o que Lhe é agradável, o que é perfeito.
(Bíblia Sagrada, Romanos 12, 2)

Resumo

A maioria dos sistemas reais pode ser representada como um grafo de componentes multi-tipados com um grande número de interações. Redes de Informação Heterogênea (HIN) são estruturas interconectadas com dados de múltiplos tipos que suportam o rico significado semântico de tipos estruturais de nós e arestas. Nas HIN, diferentes informações podem ser apresentadas usando diferentes tipos e formas de dados, mas podem ter informações iguais ou complementares. Então, há conhecimento a ser descoberto. Estruturas de Conhecimento Terminológicos (TKS) como produtos terminológicos podem ser fontes de representações linguísticas e de conhecimento a ser usado para enriquecer a HIN e criar uma medida de similaridade para extrair os documentos similares entre si, mesmo que esses documentos sejam de tipos diferentes (por exemplo, encontrar os artigos médicos que de alguma forma estão relacionados com registros médicos). Nesse sentido, este trabalho apresenta o algoritmo NetworkCreator que cria uma Rede de Informações Heterogêneas utilizando medidas de similaridade clássicas, produtos de terminológicos e os atributos dos documentos. Nos experimentos, foram utilizados prontuários médicos e artigos científicos para construir a HIN e relacionar seus conteúdos. O algoritmo HeteSimTKSQuery também foi criado para calcular medidas de similaridade entre os documentos de diferentes tipos que se encontram na HIN. Produtos terminológicos com meta-caminhos também foram explorados. Os resultados se mostraram eficientes, alcançando em média 89% de acurácia, em alguns casos. No entanto, é importante notar que todas as HIN apresentadas na literatura pesquisada foram construídas apenas por um tipo de dados proveniente de uma única fonte. Os resultados mostram que os algoritmos são viáveis para resolver os problemas de construção de HIN e busca de similaridade. Porém, eles ainda precisam de aperfeiçoamentos. Futuramente, pode-se trabalhar na detecção da granularidade dos nós destas redes e tentar reduzir o tempo de construção da rede.

Palavras-chave: Redes de Informação Heterogêneas. Medidas de Similaridade. Meta-caminho. Produtos terminológicos.

Abstract

Most real systems can be represented as a graph of multi-typed components with a large number of interactions. Heterogeneous Information Networks (HIN) are interconnected structures with data of multiple types which support the rich semantic meaning of structural types of nodes and edges. In HIN, different information can be presented using different types and forms of data, but may have the same or complementary information. So there is knowledge to be discovered. Terminology Knowledge Structures (TKS) como terminology products can be sources of linguistic representations and knowledge to be used for enrich the HIN and create a measure of similarity to extract the documents similar to each other, even if these documents are of different types (for example, finding medical articles that are in some way related to medical records). In this sense, this work presents the creation of a Heterogeneous Information Network using classical similarity measures, terminology products and the attributes of documents by an algorithm called NetworkCreator. As a contribution, an algorithm called NetworkCreator was created that from medical records and scientific articles builds an HIN with related documents, was also created. The algorithm HeteSimTKSQuery to calculate similarity measures between documents of different types which are in HIN. Terminology products with meta-paths were also explored. The results were efficient, reaching on average 89% accuracy in some cases. However, it is important to note that all HIN presented in the researched literature were constructed only by one type of data coming from a single source. The results show that the algorithms are feasible to solve the problems of HIN construction and search for similarity. But it still needs improvement. In the future one can work on detection in the detection of node granularity of these networks and try to reduce the network construction runtime.

Keywords: Heterogeneous Information Network. Similarity Measures. Meta-Path. Terminology Products.

List of Figures

Figure 1 – The concept of network schema is proposed to describe the meta structure of a network.A meta template for an information network with the object type mapping and the link type mapping which is a directed graph defined over object types with edges as relations. 52

Figure 2 – Two objects in a heterogeneous network can be connected via different paths and these paths have different meanings this different paths are called meta paths. This paths expresses the relation between objects. This Figure represents the Meta-paths of the Authors schema. schema. 54

Figure 3 – ICD-10 has seven digits, every digit has a meaning. The first triad represents the category. The second triad represents the severity, the etiology, and the anatomic site and other vital details. The last digit represents the extension. ICD-10 code example. 57

Figure 4 – Examples of nodes of the medical record type from MTsamples. 69

Figure 5 – First example of medical article.The words in bold letters are the attributes from the node.PMID(PubMed Unique Identifier), OWN(owners), STAT(Status), LR(Date Last Revised), IS(ISSN), DP (Date of Publication), TI(Title) LID(Location Identifier), AB(Abstract), FAU(Full Author), AU(Author), AD(Affiliation) and LA(Language). 70

Figure 6 – Second Example of medical article. The words in bold letters are the attributes from the node. PMID(PubMed Unique Identifier), OWN(owners), STAT(Status), LR(Date Last Revised), IS(ISSN), DP (Date of Publication), TI(Title) LID(Location Identifier), AB(Abstract), FAU(Full Author), AU(Author), AD(Affiliation) and LA(Language). 71

Figure 7 – Network Connection. Fist of all the nodes was separated in different types the papers and the medical records. For each type of node was created a network that was connected by a classic algorithms. Than was used terminological products to build a similarity algorithm for Heterogeneous Information Network. 72

Figure 8 – HeteSimTKSQuery algorithm. In *A* first step is the HIN. Then Build a dictionary indexing for N as Locality Sensitive Hash forming a Hash Table. Than a hashing query node a to a group Gi. And then add the nodes hashed from Gi to Q in D. Then used a HeteSimTKS and the retrieved articles. 73

Figure 9 – Comparison of Accuracy, Precision, and Recall using the query "cancer or tumor or carcinoma" for vectorial models of information retrieval using classical similarity algorithms. 76

Figure 10 – Accuracy, Recall and Precision measures of the TKS algorithm with classic similarity methods using the following meta-path "Paper-Medical Record-paper" from the algorithm HeteSimTKSQuery. 80
Figure 11 – Comparison of accuracy, precision and recall for query "cancer or tumor or carcinoma" using the following meta-path "Paper-Medical Record-Paper" from the algorithm HeteSimTKSQuery. 83

List of Tables

Table 1 – Comparisons of desirable requirements of similarity measures. 44
Table 2 – Comparative Analysis of the Similarity Measures beetween Symmetry, Triangle Inequation, Approach, Path-Based and Pruning parameters. . . 47
Table 3 – Examples of the path instance from the meta-paths and the meaning of its connections. 54
Table 4 – SNOMED-CT is a multilingual and inclusive clinical health technology. ICD is an international classification list of diseases related to health problems. 59
Table 5 – The confusion matrix of Exp1. 77
Table 6 – The statistical results of the experiment 1. The following statistical measures was used: Sensitivity, Specificity, Precision, Negative Predictive Value, False Positive Rate, False Discovery Rate, False Negative Rate, Accuracy, F1 Score and Matthews Correlation Coefficient. 78
Table 7 – Confusion matrix of the Exp2. 81
Table 8 – The statistical results of the experiment 2. The following statistical measures was used: Sensitivity, Specificity, Precision, Negative Predictive Value, False Positive Rate, False Discovery Rate, False Negative Rate, Accuracy, F1 Score and Matthews Correlation Coefficient. 82
Table 9 – Confusion matrix of the Exp3. 84
Table 10 – The statistical results of the experiment 3. The following statistical measures was used: Sensitivity, Specificity, Precision, Negative Predictive Value, False Positive Rate, False Discovery Rate, False Negative Rate, Accuracy, F1 Score and Matthews Correlation Coefficient. 84
Table 11 – Comparative Analysis of the Experiments: Exp1, Exp2 and Exp3 showing its objectives the algorithms and the best results 85
Table 12 – Comparison of the Classical Similarity Algorithms runtime in milliseconds and its precision with HeteSimTKSQuery and The Hybrid HeteSimTKSQuery and its precision. 86
Table 13 – Comparison of the HeteSimTKSQuery without the Classical Similarity Measures for the path length from sizes 10, 20, 50,100 and 200. 87

List of abbreviations and acronyms

CDF	Common Data File
CLESA	Cross-Language Explicit Semantic Analysis
DISCO	Similarity through the Co-occurrence Distribution
ESA	Explicit Semantic Analysis
GLSA	Generalized Latent Semantic Analysis
HAL	Hyperspace Analogues to Language
HIN	Heterougeneous Information Networks
ICD	International Classification of Diseases
IMDb	Internet Movie Database
LSA	Latent Semantic Analysis
LSR	Latent Semantic Relation
LSH	Locality Sensitive Hash
MeDRA	Medical Dictionary for Regulatory Activities
NCBI	National Center for Biotechnology Information
NGD	Normalized Google Distance
PMI-IR	Pointwise Mutual Information - Information Retrieval
PPR	Personalized Page Rank
PRA	Path Ranking Algorithm
PS-Join	Path-based Similarity Join
PTE	Predictive Text Embedding
RWR	Random Walk with Restart
SCO-PMI	Second-order Co-occurrence - Pointwise Mutual Information
SNOMED-CT	SNOMED Clinical Terms
SVD	Singular Value Decomposition
TF-IDF	Term-Frequency and Inverse Term-Frequency

THIN	Textual Heterogeneous Information Network
TKS	Terminology and Knowledge Structures
UMLS	Unified Modelling Language System
VCM	Visualization of Concept in Medicine

Contents

1 INTRODUCTION 25
1.1 Hypothesis 28
1.2 Objective 28
1.3 Methods and Materials 28
1.4 Results and Contributions 30
1.5 Document Organization 30

2 SIMILARITY MEASURES 31
2.1 Similarity Measures for Text Document Collections 31
2.2 Knowledge-based Measures 34
2.2.1 Semantic Relatedness Measures 34
2.2.2 Semantic Based Measures 35
2.3 Networks-based Similarity Measures 37
2.4 Comparative Analysis of the Similarity Measures 42
2.5 Remarks 49

3 THEORETICAL BASIS AND TECHNOLOGIES 51
3.1 Heterogeneous Information Networks 51
3.2 Terminology Products 55
3.3 Supporting Technologies 59
3.3.1 Scipy 59
3.3.2 PyMedTermino 60
3.4 BioPython 60
3.5 Igraph 61
3.6 Remarks 61

4 THE ALGORITHMS: NETWORK CREATOR AND HETES-IMTKSQUERY 63
4.1 Creation of Heterogeneous Information Networks 63
4.2 Experiments and Results 67
4.2.1 Accuracy, Precision and Recall Measures 75
4.2.2 Runtime Measures 85
4.3 Results Discussion 88
4.4 Remarks 88

5 CONCLUSION 91

Bibliography **95**

1

Introduction

Similarity measures have long been studied due to their vast applications in Computer Science and other branches of Science, such as Mathematics, Biology and Chemistry (FRAKES; BAEZA-YATES, 1992). Similarity measures use different features and mechanisms to calculate the degree of similarity, and each of these measures generates a score with different distributions over the same domain of values. To do it, the main function receives a pair of textual objects and returns a score that indicates how similar inputs are. In the last five years, the textual similarity measures permanently overplayed in a wide range of research areas, acting on information extraction and processing, especially in tasks of information retrieval (SALTON; MCGILL, 1986), textual classification (RICHARDS; KORNAI, 2003), clustering of documents (COSTA; LIMA, 2018), topic detection (YANG et al., 2018), question-answering applications (XIONG; ZHONG; SOCHER, 2018), machine translation (HERMJAKOB et al., 2018), textual summarization (WU et al., 2018), data mining(ROCHA et al., 2018), Web search (SÁNCHEZ; MARTÍNEZ-SANAHUJA; BATET, 2018), clustering and recommender systems (SALTON; MCGILL, 1986).

Classic textual similarity measures can be classified into two approaches similarity measures based on: textual features and structural information (links, the topology of networks, etc.). The first group of textual measures calculates the similarity between objects by using their feature vectors. On the other hand, the similarities based on links define measures concerning the structural composition of a graph. The similarities based on features usually exploit textual similarity algorithms, and these similarity measures can be further classified into the following four categories: *(i)* string-based, *(ii)* term-based, *(iii)* collection-based and *(iv)* knowledge-based similarity measures (GOMAA; FAHMY, 2013).

Most similarity measures based on features may be defined in a vector space. For example, the Cosine similarity is the most traditional measure, and calculates the similarity between two non-zero vectors in an inner product space (SALTON, 1989). The Manhattan distance is the distance between two objects in a grid based on a rigidly horizontal and/or

vertical path (KRAUSE, 2012); the Euclidean distance is the distance between two points in the Euclidean space (GREUB, 1967); and Jaccard is a statistic similarity measure used for comparing the similarity and diversity of sample sets (JACCARD, 1901).

Additionally, the Dice measure is defined as twice the number of common terms in the strings under comparison, divided by the total number of terms (DICE, 1945). The Overlap Matching coefficient counts the number of similar terms considering the overlap of sets (UKKONEN, 1990). The idea of Overlap Matching is similar to the Dice coefficient: the former calculates the similarity of two strings in terms of the number of common bigrams; the latter is a similarity measure related to the Jaccard index that measures the overlap between two sets, and it is defined as the size of the intersection divided by the smallest set size between two sets (CHARIKAR, 2002). Similarity measures applied to specific domains, such as (EHSANI; DRABLØS, 2016), are not the focus of this study.

Despite the fact that the collection or knowledge-based similarity measures are not covered in this work, they are an important subject, and new knowledge-based similarity measures are constantly being proposed in other works (MAGARA; OJO; ZUVA, 2018). The trends point out to similarity measure based on the representation of text and its relationship on networks or graphs such as WordNet[1] (MILLER, 1995) and ontologies[2](GRUBER, 1993), which have shown the effectiveness of the representation of textual data as a network. These trends are supported by a whole new branch of Network Science working on large-scale graphs with non-trivial topological structures, the so-called Complex Networks(BARABÁSI et al., 2016),(SILVA; ZHAO, 2016).

Given a directed graph G, an information network is composed of textual nodes and links. A node is understood to mean: plain text (one word); tuples (rows) from a database table; long texts as structured documents (e.g., web pages) or unstructured documents (e.g., plain text, program codes, and blogs); more complex data structures such as trees and graphs; images, audios, and videos. By handling different types of information (text, audio, video, and images), the establishment of links or relationships among nodes representing such types of information allows the creation of Heterogeneous Information Networks(HIN). In other words, when the number of types of nodes or links is higher than one, the network is called Heterogeneous Information Network; and Homogeneous Information Network, otherwise. Homogeneous Networks include only nodes or links of the same type, while HIN organizes the networked data as a network including nodes and links of different types. It is an interconnected and multi-type data network, in which the semantics of nodes and edges are explicit; and therefore can be understood by

[1] WordNet is a well known lexical database of English, organized as sets of cognitive synonyms, each representing a different concept — available online at <https://wordnet.princeton.edu/>.

[2] An ontology is a knowledge representation technique that relies on explicit specification of a conceptualization. — available online at <https://protege.stanford.edu/publications/ontology_development/ontology101-noy-mcguinness.html>.

algorithms (SHI et al., 2017). For example, IMDb[3] is a network of movies with different types of nodes (actors, movies, directors) and different kinds of links or edges such as an actor connected by different movies, but from the same director. This work focuses on Heterogeneous Information Networks whose nodes and edges are text typed.

Meta path forms a natural base for network-based similarity search engines. It is a path consisting of a sequence of relations between object types (i.e., structural paths at the meta level), which defines a the new composite relation between its origin type and ending type. Consider a bibliographic network extracted from DBLP with four types of objects, namely terms (T), authors (A), venues (C) and papers (P). The meta path APA means that A writes a paper P. The meta path APC implies that an author A writes a paper P in venue C. Nevertheless, the user may prefer to explicitly define a path sequence ot determine the best path based on experimental trials.

HINs are usually rich of information[4] and semantics[5], mainly when they are the composition of data collection and data repositories(SUN et al., 2012). Applications with textual information – such as HIN – demand to identify the suitable meaning of each word or concept in a domain of interest (for disambiguation purposes). Ontologies, dictionaries and other terminology products with meta-path[6] can help this task. Furthermore, no similarity measure for HIN incorporates terminology products in their calculation.

Terminology is the study of terms and their use. Terms are phrases and words which describe industry jargon, products or services. They usually drive competitive differentiation. Most corporations use a growing number of organization-specific or industry-specific words that need to be accurately stored and shared. Terms can be anything from a commodity name to a marketing tagline.

Abbreviations, acronyms, and synonyms can cause frustration for similarity algorithms, especially when attempting to take their similarity without a clear understanding of their meaning. This cases use the terminology products to solve this problem.

Terminology products and ontology enable management to accomplish an accurate and useful similarity results by organizing these terms with a precise set of rules regarding their usage; and this ensures that the correct term is used within another term.

Terminology products use a term base that is a central repository, similar to a database, which allows for the systematic management of approved terms. The use of a term base alongside existing similarity environments ensures that more accurate and consistent similarity results are produced, and this helps make the search for similatity

[3] Internet Movie Database is the world's most famous and authoritative source for movie, TV and celebrity content. — available online at <https://www.imdb.com/>.

[4] Information is any object that gives the answer to a problem of some kind or solves uncertainty.

[5] Semantic is the linguistic study of meaning.

[6] Meta-path is a path consisting of a sequence of relations defined between different object types — available online at <https://web.stanford.edu/group/mmds/slides2012/s-han.pdf>.

easier and more productive.

1.1 Hypothesis

Similarity algorithms in networks and the classic algorithms do not work with terminology products or ontologies. HINs are rich in semantics. It is believed that when exploring meta-paths (which already have semantics) the addition of terminology products will add yet more semantics to it and result in an even more accurate result when extracting similarity.

1.2 Objective

The objective of this study is to define a novel similarity algorithm that deals with the textual HIN structure by using semantics based on meta-paths and terminology products.

1.3 Methods and Materials

To achieve the main objective, the specific objectives and a brief methodology are summarized as follows:

1. The study of the theory of Complex Networks and Similarity Measures.

2. The creation of a crawler to collect articles from PubMed. Using the Biopython[7] tool, five thousand articles were downloaded following a keyword filter by the respective specialization such as Allergy / Immunology; Bariatrics; Cardiovascular / Pulmonary; Dermatology; to name a few.

3. The creation of a crawler to collect medical records from the MTsamples public database[8], which contains medical reports from the same filter found in PubMed. It includes 4999 medical records that were downloaded and organized as aforementioned in item 2. This medical records are all from different people.

4. The creation of a computational module to preprocess the articles extracted from PubMed. After the extraction of some article attributes (e.g., abstract, authors, keywords), a set of techniques were performed as an attempt to clean article content,

[7] Biopython is a set of freely available tools for biological computation written in Python. — available online at <https://biopython.org>

[8] MTSamples is designed to give users access to a big collection of transcribed medical reports. — available online at <https://www.mtsamples.com/>

including stopwords elimination, lemmatization, and stemming. It was verified and treated the redundancy of the articles in case of multiple authors.

5. The creation of a computational module to process medical records extracted from MTsamples. After the extraction of medical records (e.g., Medications, Allergies, Objective), the same techniques were performed for article pre-processing. In both pre-processing steps, the PyMedTermino[9] tool was used.

6. The creation of an Information Network that contains PubMed articles. Every node in this network has the following structure: title, abstract and keywords. To connect the nodes (each node represents an article and its structure), similarity measures were used, including the Euclidean distance and its variants; Minkowski; Cityblock; Cosine; Correlation; Hamming, and others (DEZA; DEZA, 2009). The Information Network containing the highest similarity values among the article nodes was chosen.

7. The creation of an Information Network that contains public medical records. The structure of a medical record is variable according to the medical specialty and clinical case, which include subject, allergies, objectives, assessment, plan and keywords. To connect the nodes (each node represents a medical record and its structure), the same similarity measures were used: Euclidean distance and its variants; Minkowski; Cityblock; Cosine; Correlation; Hamming, and others. The Information Network containing the highest similarity values among the medical record nodes was chosen.

8. The creation of a novel Information Network merging the two aforementioned networks. By using the pyMedTermino tool, a pre-processing step was performed to enrich the merged Information Network with terminology products included on PyMedTermino(SNOMED CT, ICD10, MedDRA, CDF, UMLS and VCM icons). The connections between article nodes and medical record nodes were made by terms such as keywords, and linguistic terms and synonymous terms and afterwards the same similarity measures were calculated. Through these measures the different attributes are detected to connect only correlated attributes. The structure of the medical record contains the treatment and its evolution.

9. The creation of a network schema, which is a meta-template of an Information Network (SUN; HAN, 2013). For this, a random walk algorithm packed into the Igraph[10] tool was used. Every new edge traversed was saved forming the network schema. The random walk was used to extract the meta-paths in the graph.

[9] PyMedTermino is a Python module for easy access to the main medical terminologies in Python, — available online at <https://pythonhosted.org>.

[10] igraph is a library collection for creating and manipulating graphs and analyzing networks — available online at <http://igraph.org/python/>

10. The extraction of meta-paths. After the creation of the schema of the Heterogeneous information Network, the meta-paths were extracted using the Igraph tool. Meta-paths are the ways used to move on the network. Since the network schema is known, extracting the meta-paths is a consequence.

11. The definition of similarity measures according to the meta-paths by using SNOMED CT(SNOMED, 2011), ICD10(CODES; LIST,), MedDRA(BROWN; WOOD; WOOD, 1999), CDF(KRUMMENACHER; STRANG, 2007), UMLS(ARONSON, 2001), to validate each meta-path, the hierarchy and the semantic structures found in these well-established health terminologies were used.

The created similarity algorithm produces a more promising result than the use of classic similarity measures and the other networks similarity measures.

1.4 Results and Contributions

During this work, the following contributions were generated:

1. a similarity measure for Heterogeneous Information Network;

2. a methodology for the creation of a Textual Heterogeneous Information Network to explore textual semantics with the support of ontologies, meta-paths and terminology products in health data

3. the composition of a dataset for Heterogeneous Information Network in the health care area;

4. the creation of crawlers to manipulate data from PubMed and MTSamples;

1.5 Document Organization

This work is organized as follows: Chapter 2 talks about similarity measures that are somehow linked to this work. This chapter deals with the following subjects: similarity measures for text document collections, knowledge-based Measures and Networks-based Similarity Measures. Chapter 3 covers the theoretical basis and technologies focusing on the Heterogeneous Information Networks and ontology, terminology products and the supporting Technologies. Chapter 4 shows the algorithm proposed in this work and presents Experiments and Results. Chapter 5 reports final remarks and proposes future works.

2

Similarity Measures

Usually, similarity measures for text document collection are presented as semantic similarity coefficients that quantify the similarity between textual information (words, sentences, paragraphs, documents, etc) based on information obtained from corpora[1]. At concept level, the similarity measure is a type of semantic similarity used to identify the closeness between words using information extracted from semantic networks[2]. Throughout this chapter, similarity measures for text collection, knowledge-based semantic measures, and networks-based similarity measures will be presented. Usually, similarity measures for text collections are called statistical similarity measures, and knowledge-based similarity measures are called semantic measures.

2.1 Similarity Measures for Text Document Collections

a) Hyperspace Analogues to Language (HAL) creates a semantic space (represented by a matrix) of co-occurrence of words after the analysis of documents in textual collections (LUND; BURGESS; ATCHLEY, 1995; LUND; BURGESS, 1996). This semantic space is often a space with a large number of dimensions, in which words or concepts are represented by objects; the position of each object along the axis is somehow related to the meaning of the word (OSGOOD; SUCI; TANNENBAUM, 1964). To build the semantic space, first of all, it is necessary to define the meanings of a set of axes and gather information from human subjects to determine where each word in question should fall on each axis. A $N \times N$ co-occurrence matrix is composed of individual words as elements, where N is the number of words in the lexical vocabulary. The lexical co-occurrence

[1] A corpus is a large collection of textual documents which is mainly used for information extraction, information retrieval, and natural language processing (SCHÜTZE; MANNING; RAGHAVAN, 2008).

[2] A semantic network is a graph-based knowledge representation of semantic relations between concepts (nodes) (MCCRAY, 1989).

has been established by HAL as a useful basis for the construction of semantic spaces (BURGESS; LUND, 1994; LUND; BURGESS, 1996, 1996).

b) Latent Semantic Analysis (LSA) intends to overcome the main problems related to the use of lexical-based analysis: polysemy and synonymy (LANDAUER; DUMAIS, 1997). The similarities defined by LSA are based on the closeness of terms in a semantic space built according to the co-occurrence of all terms in manipulated collections of documents, instead of lexical matching. LSA exploits the *Singular Value Decomposition*, a linear algebra factorization, in which the matrix X is composed of documents, and words are decomposed into the product of three component matrices. The most important dimensions (with the highest values in the singular matrix) are selected to reduce the dimension of the working space. A semantic matrix is generated by the computation of the inner-product among each column of the matrix. Given the semantic matrix, similarities are identified by considering the highest cosine.

c) Generalized Latent Semantic Analysis (GLSA) computes the semantic relationships between terms and document, as vectors are computed as linear combinations of term vectors (MATVEEVA et al., 2005). GLSA extends LSA focusing in term vectors, instead of documents as LSA. GLSA is not based on bag-of-words[3]. It exploits pair-wise term similarities to compute a representation for terms. GLSA demands a semantic similarity measure between terms and a method for dimensionality reduction. GLSA combines any similarity measure and any reduction of dimensionality.

d) Explicit Semantic Analysis (ESA) uses a corpus of documents as a knowledge base, it represents the individual words or entire documents as vectorial representations of text documents such as HAL, LSA, and GSA (GABRILOVICH; MARKOVITCH, 2007). The ESA is a measure to calculate semantic relationships between any pair of documents, any corpora including Wikipedia articles and the Open Directory Project. Documents are represented as centroids of vectors represented its words. Words are represented as a column vector in the Term-Frequency and Inverse Term-Frequency (TF-IDF)(WU et al., 2008) array of the text corpus. The terms or texts of ESA are portrayed by vectors with high dimension. Each element of the vector represents the pair TF-IDF between terms of documents. The semantic similarity between two terms or texts is expressed by the cosine measure between the corresponding vectors. However, unlike *LSA*, *ESA* deals with human-readable labels transforming them into concepts that construct the vector space. The conversions are possible due to the use of a knowledge base (EGOZI; MARKOVITCH; GABRILOVICH, 2011; GABRILOVICH; MARKOVITCH, 2007). The scheme is extended from single words to multi-words documents by simply summing the arrays of all words in the documents (GABRILOVICH; MARKOVITCH, 2007). The semantic relatedness of

[3] Bag-of-words is a model that ignores context, semantics and order of words, simplifying computational efforts. As vocabulary may potentially run into millions of documents, this model faces scalability challenges (SCHÜTZE; MANNING; RAGHAVAN, 2008).

the words is given by a numeric estimation.

e) Cross-Language Explicit Semantic Analysis (CLESA) is a multilingual generalization of ESA (POTTHAST; STEIN; ANDERKA, 2008). *CLESA* manipulates documents aligned with a multilingual reference collection that corresponds documents as vectors of concepts regardless of the language. The relationships between two documents in different languages are calculated by the cosine considering the vector space. A document written in a specific language is represented as an ESA vector by using an index document collection in the language. The similarity between a document and another document in another language is quantified in the concept space by computing the cosine similarity between both documents.

f) Pointwise Mutual Information - Information Retrieval (PMI-IR) is a method used to calculate the similarity between pairs of words using the AltaVista's Advanced Search Query (FRIEDMAN, 2004), and to calculate the probability of similarity using the Alta Vista similarity (TURNEY, 2001). This probability is based on the proximity of the pair of words in Web pages, considering that the greater the proximity the greater the similarity. Like the LSA algorithm, the *PMI-IR* algorithm, is based on co-occurrence (MANNING et al., 1999). The core idea is that a word is characterized by its neighborhood (FIRTH, 1957). There are many different measures of the degree that two words co-occur (MANNING et al., 1999). Therefore, the ratio between one probability and another is the measure of the degree of statistical dependence.

g) Second-order Co-occurrence - Pointwise Mutual Information (SCO-PMI) is a semantic similarity measure that applies PMI-IR to sort the list of neighboring words of two target words being compared in a collection (ISLAM; INKPEN, 2008; ISLAM; INKPEN, 2006). The advantage of using *SCO-PMI* is that it calculates the similarity between two words that do not co-occur frequently, but the same neighboring words are co-occurring. The pre-processed word pairs are taken to calculate semantic word similarity using *SCO-PMI*. An evaluation of the result shows that this method outperforms several competing corpus-based methods. This method focuses on measuring the similarity between two target words. After finding the similarity among all the words in the document, the retrieval of similar information can also be performed to user query.

h) Normalized Google Distance (NGD) is a semantic similarity measure using the number of hits returned by the Google search engine for a given set of keywords (CILIBRASI; VITANYI, 2007). This algorithm returns the keywords with the same or similar meanings based on natural language processing. The words are similar in meaning if that tend to be "near" in Google distance units, while words with different meanings tend to be more distant [4].

[4] This Google distance units is not specified by the article (CILIBRASI; VITANYI, 2007).

i) Similarity through the Co-occurrence Distribution (DISCO) assumes that words with similar meanings occur in a similar context (KOLB, 2009). Large text collections are statistically analyzed to obtain similarity of distribution. The *DISCO* is a method that calculates the similarity of distribution between words using a simple context window of approximately three words for counting co-occurrences. When two words are submitted to the calculation of the exact similarity, *DISCO* brings back its vectors from the indexed data and computes the similarity according to the Lin measure (LIN, 1998), presented in 2.2.1. If the most similar word according to the distribution is requested, *DISCO* returns the second order of the word vector. *DISCO* has two main similarity measures: *DISCO1* and *DISCO2*. *DISCO1* calculates the first order similarity between two input words according to the word arrangement sets. *DISCO2* calculates the second order similarity between two input words according to the distribution of similar words.

2.2 Knowledge-based Measures

Knowledge-based measures try to identify the degree of similarity among concepts by using algorithms supported by lexical resources and/or semantic networks. The similarity measures based on knowledge can be separated into two groups: *semantic relatedness measures* and *semantic-based measures*.

2.2 Semantic Relatedness Measures

Semantic relatedness measures indicate the strength of the semantic interactions between objects, since there are no constraints on the quality of the considered semantic links. Semantic relatedness similarity measures are a category of relationships between two words, incorporating a bigger range of relationships between concepts such as “is_type_of”, “is_one_specific_example_of”, “is_part_of”, “is_the_opposite_of” (PATWARDHAN; BANERJEE; PEDERSEN, 2003). The most used examples of semantic relatedness measures are **Resnik (RES)** (RESNIK, 1995; KG; SADASIVAM, 2017), **Lin (LIN)** (LIN, 1998), **Jiang & Conrath (JCN)** (JIANG; CONRATH, 1997), **St.Onge (HSO)** (HIRST; ST-ONGE et al., 1998), **Lesk (LESK)** (BANERJEE; PEDERSEN, 2002), and **Pairs of Vectors (Vectors)** (PATWARDHAN, 2003).

a) RES is a measure of semantic similarity based on the notion of information content that considers an “is a” taxonomy. The *RES* value is the information content of the *Least Common Subsumer*[5] (RESNIK, 1995; KG; SADASIVAM, 2017).

[5] It is the most specific concept, which is an ancestor of both A and B.

b) LIN suggests the semantic similarity between two topics in a taxonomy (LIN, 1998). *LIN* is defined as a function of the meaning shared by the topics and the meaning of each individual topic. The meaning shared by two topics can be recognized by looking at the lowest common ancestor, which corresponds to the most specific common classification of the two topics. Once this common classification is identified, the meaning shared by two topics can be measured by the amount of information needed to state the commonality of the two topics. The semantic similarity is divided into the hierarchical taxonomy. The disadvantage is capturing the semantic relationships in non-hierarchical components.

c) JCN is a hybrid similarity measure that mixes words or concepts. It combines a taxonomy structure with measures based on corpus. Exploiting the best of both, the taxonomy helps to guarantee the semantics, while the statistical approach ensures the evidence of the distribution of the exploited corpus. *LIN* and *JCN* increase the information content of the *Least Common Subsumer* by considering the sum of the content of concepts. *LIN* scales the content of the *Least Common Subsumer*, whereas *JCN* assigns the difference between the sum and the information content of the *Least Common Subsumer*.

d) The **HSO** measure finds lexical chains of strings relating two meanings of a word (HIRST; ST-ONGE et al., 1998). This measure calculates the relatedness between concepts using the path distance between the concept nodes, number of changes in direction of the path connecting two concepts and the allow ableness of the path. When the relation between concepts is close, they are semantically related to each other (CHOUDHARI, 2012).

e) The **LESK** algorithm discovers overlaps in the glossary of *synsets* of WordNet (presented in section 2.2.2). The score is the sum of the squares of the overlap lengths. On the other hand, the **Vector** measure creates for each word of the specific WordNet glossary (PATWARDHAN, 2003). Afterwards, it represents each glossary/concept as a vector that is the mean of the co-occurrence vectors. *Vector* was developed as a measure of semantic relatedness that represents concepts using context vectors, and it is able to establish relatedness by measuring the angle between these vectors. This measure combines information from a dictionary with statistical information derived from large text corpora. In other words, semantic relatedness is then simply measured as the nearness of the two vectors in the multidimensional space (the cosine of two normalized vectors). One of the strengths of the Vector measure is that it can be used with any dictionary, regardless of the WordNet.

2.2 Semantic Based Measures

Semantic-based measures can be classified into four categories, in which: (i) the semantic similarity measures the taxonomic term relationships to extract the similarity; (ii) the

semantic distance is the inverse of the semantic relatedness; (iii) the semantic dissimilarity is the inverse of the semantic similarity; and (iv) the taxonomic distance is related to the dissimilarity. In the literature, the most cited examples of semantic-based measures are **Leacock & Chodorow (LCH)** (LEACOCK; CHODOROW, 1998), **Wu & Palmer (WUP)** (WU; PALMER, 1994), and **Path Length (Path)** (WU; PALMER, 1994).

a) LCH measures the length of the shortest path between two concepts using node-counting and exploiting the maximum depth of the taxonomy (LEACOCK; CHODOROW, 1998). It returns a score indicating the similarity between the different meanings of a word. It considers the shortest path connecting the meanings and the maximum depth of the taxonomy in which the meanings occur. On the other hand, the **WUP** measures the similarity between two meanings of a word considering the depth of the two meanings in the taxonomy and its *Least Common Subsumer* (WU; PALMER, 1994). *WUP* is a prototype lexical selection system called UNICON that represents the English and Chinese verbs based on a set of shared semantic domains and the selection information is also included in these representations without exact matching. The concepts are organized into hierarchical structures to form an interlanguage conceptual base. The input to the system is the source verb argument structure. **b)** The **Path** measure quantifies the similarity between two meanings of a word based on the shortest path linking between the two meanings in the taxonomy "is_a" (WU; PALMER, 1994). This measure is inversely proportional to the number of nodes along the shortest path between the concepts. The shortest possible path occurs when two concepts are the same, in which case the length is 1. Thus, the maximum similarity value is 1.

Overall, the presented approaches are distinct algorithms used to calculate semantic similarity between concepts. The semantic similarity can be improved by using human knowledge to generate a more accurate measure. Human knowledge usually is expressed in dictionaries, taxonomies, ontologies and concept networks.

c) WordNet is the most popular concept network used to measure similarity based on knowledge (MILLER et al., 1990). This concept network is a large graph, or a lexical database, where each node represents a real world concept, which are English words classified as noun, verb, adjective and adverb. All words are grouped into sets of cognitive synonyms and each expresses a distinct concept, for example, the "house" is a concept-object, whereas "teacher" is an entity, "art" is an abstract concept, and so on. Every node of the network consists of a set of synonym words called synsets. Each synset represents the real world concept associated with that node and is associated with a gloss (short definition or description of the real world concept). The synsets and the glosses are similar to the content of an ordinary dictionary such as the synonyms and the definitions, respectively. Synsets are interlinked by means of conceptual-semantic and lexical relations. Each link or edge describes a relationship between real world concepts represented by

the synsets that are linked. Types of relationships are “opposite of”, “is a member of”, “causes”, “pertains to”, “ is a kind of”, “ is a part of” and others. The network of relations between word senses present in *WordNet* encodes a vast amount of human knowledge, giving rise to a great number of possibilities of knowledge representation used for various tasks. *WordNet* has been manipulated by different approaches in order to automatically extract its association relations and to interpret these associations in terms of a set of conceptual relations, such as the DOLCE foundational ontology (GANGEMI; NAVIGLI; VELARDI, 2003). In terms of limitations, *WordNet* does not present the etymology or the pronunciation of words and it is basically composed of the everyday English word.

d) Agirre & Rigau developed a notion of conceptual density to create their algorithm for Word Sense Disambiguation (AGIRRE; RIGAU, 1997). They used the context of a given word along with the hierarchy of “is-a” relations in *WordNet* to find the exact sense of the word. It divides the network hierarchy of *WordNet* into sub-hierarchies and each of the senses of the ambiguous word belongs to one sub-hierarchy. The conceptual density for each sub-hierarchy is then calculated using a conceptual density formula which, intuitively, describes the amount of space occupied by the context words in each of the sub-hierarchies.

2.3 Networks-based Similarity Measures

Although similarity measures in homogeneous networks were extensively studied in the past decades, such as PageRank (BRIN; PAGE, 1998) and SimRank (JEH; WIDOM, 2002), research on heterogeneous networks and several measures are just beginning. PathSim (SUN et al., 2011) is proposed to measure the similarity of same-typed objects based on symmetric paths and evaluates the reachable probability along the given path. The embedding technique, which aims at learning low-dimensional vector representations for entities while preserving proximities, has recently received increasing attention due to its great performance in different types of tasks. (TANG et al., 2015)

a) LINE (TANG et al., 2015) and DeepWalk (PEROZZI; AL-RFOU; SKIENA, 2014) utilize the network link information to construct latent vectors for vertex classification and link prediction. DCA (CHO; BERGER; PENG, 2015) starts from the personalized PageRank, but does develop further decomposition to get better protein-protein interaction predictions in biology networks. However, these homogeneous models cannot capture information about the entity, type or relation across different-typed entities in HINs.

Embedding algorithms have also been developed for HINs. For example, Chang et al. propose to incorporate deep neural networks to train embedding vectors for both text and images at the same time (CHANG et al., 2015). Under a supervised setting, PTE

(TANG; QU; MEI, 2015) utilizes labels of words and constructs bipartite HINs to learn predictive embedding vectors for words. Embedding techniques have also been applied to knowledge graphs to resolve question-answering tasks (GUU; MILLER; LIANG, 2015) and retain knowledge relations between entities (XIE et al., 2016). However, these have all been specially designed for specific types of networks and tasks and thus it is diffucult to incorporate them to user guidance. Vector spaces constructed by different methods have different semantic meanings due to the statistics they emphasize.

b) AVGSim is a measure similarity with same or different-typed object pairs, evaluated through two random walk processes along a given meta-path and the reverse meta-path, respectively (MENG et al., 2014). The AVGSim value of two objects is the average of the reachable probability under a given path and the reverse path. It guarantees that AVGSim can measure the relevance of same or different-typed objects as well as satisfy the symmetric property. AVGSim disregards the length of the path and its decomposition. Thus, AVGSim can be considered simple and efficient because it can be implemented by means of dynamic programming and by means of the MapReduce parallel model that transforms the multiplication of two large matrices into several multiplications of smaller matrices. According to the authors, using MapReduce eliminates the restriction of memory size and deals with massive data more efficiently. Experiments were carried out on the DBLP dataset [6] to demonstrate the high efficiency and effectiveness of AVGSim. AVGSim is a meta-path based method which not only exploits symmetric and uniform similarity approach for heterogeneous objects, but also advocates the ability to be extended to large-scale networks.

c) ESim models network nodes as low-dimensional vector to exploit the similarity embedded in the network (SHANG et al., 2016). It handles large scale HINs due to a parallel optimization structure and a sampling method of path instances following a given meta-path. ESim accepts the meta-paths defined by users as guidance to create arrays composed of nodes in a space chosen by the user. The meta-path guidance has two possible solutions: sequential (seq) assumes that a vertex is highly relevant to its left/right neighbors in the sequence and pairwise (pair) assumes all vertexes in a path instance are highly relevant to each other. The “pairwise” solution is more effective than the “sequential” solution.

Moreover, an optimization of ESim was developed to aggregate patterns in huge HINs. ESim searches for similarity between objects of the same type. For example, given a network of social media connections between reviews, users and businesses, it is possible to find similar restaurants and to recommend potential friends. Similarity is defined by the cosine between same type nodes and has demonstrated its effectiveness, surpassing previous high-end algorithms in two large-scale real-world networks for HINs. HINs may

[6] https://old.datahub.io/dataset/dblp

contain non-directed, weighted, weighted and unweighted edges and several types of nodes.

d) Shi et al. proposed *HeteSim* for measuring the relatedness of same or different type objects in a uniform framework, considering two important properties: self-maximum and symmetric (SHI et al., 2012). The relatedness of object pairs is defined based on the search path that connects two objects by following a sequence of node types. According to the authors, symmetric property is similar to PathSim, but HeteSim has more general symmetric property not only for symmetric paths but also for asymmetric paths. In terms of self-maximum, HeteSim satisfies the identity of indiscernibles the HeteSim score for two different type objects is also 1 if the objects have the same probability distribution at the middle type object. This is reasonable, since they have similar structures based on the given path.

The disadvantages of HeteSim are (1) high computational complexity due to the adoption of path decomposition while measuring the relevance on odd-length path further increases complexity of calculation; and (2) problems when it is extended to large-scale networks with massive data, since its calculation process is based on memory computing. HeteSim takes a pair-wise random walk, to assess the relevance of different-type objects and measure the relevance of any pair of objects of an arbitrary meta-path. So the HeteSim is a uniform measure, a path-constrained and a semi-metric measure.

e) LSH-HeteSim (LI et al., 2014) is a new version of HeteSim that aims to mine drugs in biological networks considering drugs that were laid out in complicated semantic paths (LI et al., 2014). LSH (Locality Sensitive Hash) functions were introduced to solve the approximate nearest neighbor problem in high dimensional spaces (DATAR et al., 2004). It was designed in such a way that if two objects are close in the intended distance measure, the probability that they are hashed to the same value is high, and if they are far in the intended distance measure, the probability that they are hashed to the same value is low (JEGOU; DOUZE; SCHMID, 2011).

LSH-HeteSim exploits the iterations of drugs in heterogeneous biological networks where drugs are linked by edges that are connected with sophisticated semantic paths to overcome the problems of high cost of memory and computational effort. Additionally, some methods as (BU et al., 2014) and (ZHU et al., 2015) combine a meta-path based on searches for relevance considering user preference.

f) Lao et al. proposed the **Personalized Page Rank (PPR)** (AGIRRE; SOROA, 2009), which considers the notion of "learning proximity" in a labeled graph based on task- or user-specific and must be learned or engineered. There are also general-purpose graph proximity measures, such as Random Walk with Restart (RWR) (also called Personalized PageRank), which are fairly successful for many types of tasks. The authors describe a scheme for parameterizing such a measure, in which a proximity measure is defined by a weighted combination of simple "path experts", each of which corresponds to a particular

labeled path through the graph. According to the authors, PPR method outperforms untrained RWR on the majority of the tasks. It also outperforms a widely-used simpler parameterization in which a weight is associated with each label in the graph, again producing high MAP scores on all eight tasks. Another contribution of the PPR is an extension of the method to support two additional types of experts, which they call query-independent experts and popular entity experts. Query-independent experts are a rich set of query-independent ranking schemes similar to the PageRank measure[7]. Popular entity experts allow rankings to be adjusted for particular entities that are especially important. (DILIGENTI; GORI; MAGGINI, 2005; CHAKRABARTI; AGARWAL, 2006).

g) The **Path Ranking Algorithm (PRA)** (BALMIN; HRISTIDIS; PAPAKONSTANTINOU, 2004) is one-parameter-per-edge label RWR proximity measures that is limited because the context in which an edge label appears is ignored. For example, in the reference recommendation task, one of the query nodes is a year. There are two ways in which one might use a year y to find candidate papers to cite: (H1) finds papers published in year y, or (H2) finds papers frequently cited by papers published in year y. Intuitively, the second heuristic seems more conceivable than the first; however, a system that insists on using only one parameter for the "importance" of the edge label *PublishedIn* cannot easily encode this intuition.

h) NetSim (ZHANG et al., 2015) is a similarity measure that quantifies functional similarities between genes or GO terms by incorporating information from gene co-function networks in addition to using the GO structure and annotations. Given two terms t_a and t_b and their common ancestor p, the NetSim similarity between the two terms, $S\left(t_a, t_b, p\right)$, is defined as:

$$s(t_a, t_b, p) = \frac{2log|G| - 2logf(t_a, t_b, p)}{2log|G| - (log|G_a| + log|G_b|)} \times (1 - \frac{h(t_a, t_b)}{|G|} \times \frac{|G_p|}{G}) \tag{2.1}$$

whose $G_p(orG)$ is the set of items annotated to p (or the root term) and its descendants, $f(t_a, t_b, p)$ measures the importance of the path-constrained annotation, and $h(t_a, t_b)$ weights the specificity of the common parent p. It is interesting to highlight that NetSim exploits Gene Ontology to augment the analyzed information.

i) RelSim (WANG et al., 2016) investigates the problem of relation for similarity search in schema-rich HINs. Under the problem setting, users are only asked to provide some simple relation instance examples as query, and the search engine should automatically detect the Latent Semantic Relation (LSR) implied by the query. This LSR should support the search for other similar relation instances. To solve the problem, first a new meta-path-based relation similarity measure must be defined to measure the similarity

[7] PageRank (BRIN; PAGE, 1998) is an assessment of the relevance of a particular page.

between relation instances in schema-rich HINs. Then, given a query, an optimization model to efficiently learn LSR implied in the query through linear programming is built, and performs fast relation similarity searches based on the learned LSR. According to the authors, experiments on real-world datasets derived from Freebase demonstrate the effectiveness and efficiency of the approach (WANG et al., 2016). So, Wang et al. (WANG et al., 2016) defined similarity measure based on the relation of meta-path and RelSim to measure the similarity of relations between schemes and rich HINs. Other relation similarity measures can be found in (BOLLEGALA; MATSUO; ISHIZUKA, 2009; TURNEY, 2005). However, they do not distinguish the diverse and subtle semantic meanings in the relation instance, but assume there is only one general relation held in a relation instance.

The intuition behind RelSim is that if two relation instances share more heavily weighted meta-paths, they tend to be more similar. RelSim is formally defined as:

Given an LSR, denoted as $\{W_m, P_m\}_{m=1}^{M}$, the RelSim between two relation instances $e = \left\langle v^{(1)} v^{(2)} \right\rangle$ and $r0 = hv(1)0;\ v(2)0i$ is defined as:$r' = \left\langle v^{(1)'} v^{(2)'} \right\rangle$

$$RS(r, r') = \frac{2 \times \sum_m w_m min(x_m, x'_m)}{\sum_m w_m x_m \sum_m (w_m, x'_m)} \tag{2.2}$$

whose x_m is the number of path instances between $v^{(1)} and v^{(2)}$ in relation to r following meta-path P_m, and x'_m is the number of path instances between $v^{(1)'}$ and $v^{(2)'}$ in relation to r' following meta-path Pm. The use of the vector x to characterize a relation instance r, and a vector w to denote the corresponding weights. M is the number of meta-paths. In schema-rich HINs, the number of path instances between two entities following a specific meta-path is often 1 or 0, denoting whether the two entities satisfy the meta-path based relation.

RelSim can be defined in terms of two parts: (1) the semantic overlap in the numerator, which is the weighted number of overlapped meta-path based relations of r and r'; and (2) the semantic broadness in the denominator, which is the weighted number of total meta-path-based relations satisfied by r and r'. Note that, if the number of path instances for a meta-path is larger than $1, i.e., x_m > 1$, it was treated the two entities have satisfied the relation x_m times. The bigger the number of overlapped meta-path-based relations shared by the r and r', the more similar the two relation instances are, which is further normalized by the semantic broadness of r and r'.

j) LINE (ZHANG et al., 2015) is a measure of similarity used to calculate the similarity between the centers of a star network according to the similarities among the attributes of the connections between the centers of the network. LINE tries to improve PathSim manipulating the nodes and same type edges in HINs (TANG et al., 2015). It designs a HIN assuming that the meta-paths given by a user are symmetric or non-symmetric. Afterwards LINE is applied to the known edges. However, the network

projection is based on counting nodes or edges that do not preserve the underlying semantics.

k) SimRank (JEH; WIDOM, 2002) is an intuitive recursive approach that measures two objects at a time. If they are referenced by similar objects, they are similar. As the base case (reference), an object is maximally similar to itself so that it can be assigned a similarity score of 1 (JEH; WIDOM, 2002). If other objects are known to be similar a-priori, their similarities can also be preassigned. For example, a Professor a and a Professor b are similar when they are both referenced by the same university (they are co-cited by the university), and the university is (maximally) similar to itself. The similarity is denoted between objects a and b by $s(a, b)$ E [0, 1]. The recursive equation for $s(a, b)$, if $a = b$ then $s(a, b)$ is defined to be 1. Otherwise,

$$s(a,b) = \frac{C}{|I(a)||I(b)|} \sum_{i=1}^{|I(a)|} \sum_{j=1}^{|I(b)|} s(I_i(a), I_j(b)) \tag{2.3}$$

where C is a constant between 0 and 1. a or b may not have any in-neighbors. Since there is no way to infer any similarity between a and b, in this case, it should be settled $s\,(a, b) = 0$, so that the addition of the sequence may be defined (1) as 0 when $I\,(a) = 0$ or $I\,(b) = 0$.

l) Path-based Similarity Join (PS-Join) is a method that joins links to retrieve the top K-pairs of similar objects based on the specification of paths by users (XIONG; ZHU; YU, 2015). PS-join can derive several metrics of semantic similarities. These paths are *join-paths* specified by the user in a heterogeneous information network. The same authors also proposed an algorithm, the BPLSH, to prune the similarity computing of PS-Join due to the great size of the graph (XIONG; ZHU; YU, 2015).

2.4 Comparative Analysis of the Similarity Measures

There is no standard way to evaluate similarity measures without the agreement of human judgments. An example is the computational similarity measures to rate the similarity of a set of word pairs; after, it is necessary to evaluate the correlation between the computational ratings and the human ratings of the same pairs. In order to evaluate measures, it is also useful to compare the measures considering important requirements used by most applications, such as the size of documents and collections, granularity, type of matching and others (DEZA; DEZA, 2009). Moreover, assumptions can empower the scientific community making the selection of measures easier according to the usage scenario and the summarization of findings related to the measures. In this dissertation, a comparative study of the similarity measures previously mentioned is presented.

Table 1 depicts the presented *similarity measures* in the lines. The first fifteenth measures are the classical measures such as Cosine, Jaccard, Dice Measure, Euclidean distance, Overlap Coefficient, and Manhattan distance. The measures from 16 to 25 are knowledge-based measures; and the measures from 26 to 41 are the network based measures. The *main requirements* of similarity algorithms are in the columns. The *main requirements* include: 1. Changeable Granularity (the granularity of information to be manipulated such as word, paragraph or document is take into account), 2. Partial Matching (if there is partial matching), 3. Ranking Relevance Allowed (if the relevance of documents is allowed), 4. Term Weights (if there is a term-weighting scheme), 5. Easy Implementation (if it is easy to be implemented considering toolboxes that can be found in several programming languages to work with similarities in networks), 6. Size Document Dependency (if the size of the document is taken into account), 7. Dependency of Ordered Terms (if dependency of terms is necessary), 8. Semantic Sensibility (if knowledge-based resource is used), and the implemented approach exploited by the measure if it used, for example, the distance between vectors or semantic space or natural language processing. The criteria chosen to compose the Table 1 are the same criteria found in the (SUN; HAN, 2013). The are based on requirements discussed in the literature. From line 1 to line 6, the measures are the classics ones. From the line 7 to 19, the measures are based on collection of text documents. From 20 to 25, they are the knowledge-based measures, and from 26 to 41, they are the network-based ones.

Table 1 – Comparisons of desirable requirements of similarity measures.

	Characteristics	*1*	*2*	*3*	4	5	6	7	8	Approach
1	Jaccard (JACCARD, 1901)	x	x	x	x	x	-	-	-	Distance between vectors
2	Dice Measure(DICE, 1945)	x	x	x	x	x	-	-	-	Distance between vectors
3	Euclidean Distance (GREUB, 1967)	x	x	x	x	x	-	-	-	Distance between vectors
4	Cosine (SALTON, 1989)	x	x	x	x	x	-	-	-	Distance between vectors
5	Overlap Coefficient (UKKONEN, 1990)	x	x	x	x	x	-	-	-	Distance between vectors
6	Manhattan (KRAUSE, 2012)	x	x	x	x	x	-	-	-	Distance between vectors
7	HAL (LUND; BURGESS; ATCHLEY, 1995) (LUND; BURGESS, 1996)	-	x	-	-	x	x	x	x	Semantic Space: Words Co-occurrence, Similarity vector
8	LSA (LANDAUER; DUMAIS, 1997)	-	x	-	-	x	x	x	x	Singular value decomposition, cosine
9	GeLSA (MATVEEVA et al., 2005)	-	x	-	-	x	x	x	x	Terms vectors, Similarity measure and Dimensionality reduction.
10	ESA (GABRILOVICH; MARKOVITCH, 2007)	-	x	-	-	x	x	x	x	TF-IDF, Cosine, Machine Learning, and Wikipédia's data
11	CLESA (POTTHAST; STEIN; ANDERKA, 2008)	-	x	-	-	x	x	x	x	Cosine of the vectors' Document
12	PMI-IR (TURNEY, 2001)	-	x	-	-	x	x	x	x	Words Co-occurrence, Machine Learning
13	SOCPMI (ISLAM; INKPEN, 2008) (ISLAM; INKPEN, 2006)	-	x	-	-	x	x	x	x	Co-occurrence of neighboring words
14	NGD (CILIBRASI; VITANYI, 2007)	-	x	-	-	x	x	x	x	Google's Distance, Natural Language
15	DISCO (KOLB, 2009)	-	x	-	-	x	x	x	x	Words Co-occurrence Window Distribution
16	WordNet (MILLER et al., 1990)	x	x	x	x	-	x	x	x	Psycholinguistic Theories and Human lexical memory
17	WUP (WU; PALMER, 1994)	x	x	x	x	-	x	x	x	Least common Subsumer and Shortest path
18	Path Length (WU; PALMER, 1994)	x	x	x	x	-	x	x	x	Shortest path
19	RES (RESNIK, 1995)	x	x	x	x	-	x	x	x	Least common Subsumer
20	JCN (JIANG; CONRATH, 1997)	x	x	x	x	-	x	x	x	Semantic Similarity and Least common Subsumer
21	LIN (LIN, 1998)	x	x	x	x	-	x	x	x	Least common Subsumer
22	LCH (LEACOCK; CHODOROW, 1998)	x	x	x	x	-	x	x	x	Shortest path and Taxonomic Score
23	HSO (HIRST; ST-ONGE et al., 1998)	x	x	x	x	-	x	x	x	Semantic Affinity and Lexical Chains Pairs
24	LESK (BANERJEE; PEDERSEN, 2002)	x	x	x	x	-	x	x	x	Semantic Affinity and Lesk's Dictionary with WordNet
25	Vector (PATWARDHAN, 2003)	x	x	x	x	-	x	x	x	Semantic Affinity and Co-occurrence Matrix with Wordnet.
26	PageRank (BRIN; PAGE, 1998)	x	x	x	x	x	x	x	x	Inner degree and Outter degree.
27	SimRank (JEH; WIDOM, 2002)	x	x	x	x	x	x	x	x	Meta-path
28	PageRank Personalizado (JEH; WIDOM, 2003)	x	x	x	x	x	x	x	x	random walk probability
29	Lao et al (LAO; COHEN, 2010)	x	x	x	x	x	x	x	x	Edges' weighting
30	Wang et al (WANG et al., 2011)	x	x	x	x	x	x	x	x	Machine Learning
31	PathSim (SUN et al., 2011)	x	x	x	x	x	x	x	x	Meta-path, network scheme and Machine learning
32	HeteSim (SHI et al., 2012)	x	x	x	x	x	x	x	x	Meta-path.
33	Wang et al (WANG; HU; YU, 2012)	x	x	x	x	x	x	x	x	Similarity Networks and information tunnels
34	Yu et al. (YU et al., 2012)	x	x	x	x	x	x	x	x	Meta-path
35	LSH-HeteSim (LI et al., 2014)	x	x	x	x	x	x	x	x	HeteSim.
36	AVGSim (MENG et al., 2014)	x	x	x	x	x	x	x	x	Random walk with meta-path and reverse meta-path.
37	PS-join (XIONG; ZHU; YU, 2015)	x	x	x	x	x	x	x	x	Meta-path.
38	NetSim (ZHANG et al., 2015)	x	x	x	x	x	x	x	x	Network Structures Measurement, probability over attribute network, Pruning Algorithm
39	Path-based similarity join (WANG et al., 2016)	x	x	x	x	x	x	x	x	Meta-path.
40	ESim (SHANG et al., 2016)	x	x	x	x	x	x	x	x	Meta-path
41	RelSim (WANG et al., 2016)	x	x	x	x	x	x	x	x	Meta-path

The following requirements of similarity measures: **1.** Changeable Granularity; **2.** Partial Matching; **3.** Ranking Relevance Allowed; **4.** Terms Weights; **5.** Easy Implementation; **6.** Size Document Dependency; **7.** Dependency of Terms; **8.** Semantic Sensibility.
Source: Own authorship.

From this comparative analysis it is possible to conclude that:

1. The statistical and semantic similarity measures (from line 7 to line 19) hold a process which aggregates semantics.

2. The statistical measures do not allow the change of level of granularity of the manipulated information.

3. The classical measures and the semantic measures allow the change of level of granularity of the manipulated information.

4. The classical measures are dependent on the size and terms. The statistical measures are not so dependent on the size and terms. The statistical measures usually have a higher computational cost and the semantic is focused on the collection.

5. Finally, the semantic or knowledge-based measures are full of specified requirements except for size document dependency, terms dependence, and semantic sensibility, and they can be a laborious task of implementation. In many cases, they are domain-specific.

6. Network measures are dynamic and more realistic since relationships are expressed in network connections.

7. Heterogeneous Information Networks(HIN) have been given more attention because they are a newly developed network model. The search for top k pairs of same objects is required in many real-life applications. Multiple tasks of data mining are being explored in HIN, which includes clustering of objects, search for similar objects and classification. Searching for similarity is a basic and an important task which is required in many applications. Various similarity measures along with their respective detailed methodologies and algorithms are discussed in this dissertation.

8. In the network similarity measures(from line 26 onwards), PathSim computes the similarity score and cluster results based on pruning method. The pruning method was proposed to obtain the promising candidate objects to be pruned. The HeteSim measure is based on the number of incoming links of target nodes and outgoing links of source nodes. When the idea (if they are related to similar objects these two objects are similar) used in the SimRank measure is applied to the HIN), it presents challenges.

9. Similarity measures based on paths are recommended first because a relevant path captures the semantic information necessary to the walk path. Second, because for symmetric or asymmetric paths, the measure should be able to calculate a heterogeneous object pair in accordance with a single score (SHI et al., 2014).

10. NetSim uses "meeting probability" to calculate the similarity among attributes (ZHANG et al., 2015).

11. The similarity is computed over the network characteristic which is extracted from the entire structure of x-star network[8]. In this technique, the characteristics of similarity is calculated by the SimRank measure. The similarity between centers is computed as the similarity of attributes based on the basic idea that centers are similar to each other if they are likely to be linked with similar characteristics attributes. The importance of link and the relation between the nodes is achieved considering the relationship between centers. The online query processing should be carried out very fast. To support it, in this experiment, an algorithm for pruning was used. To create a pruning algorithm, it was necessary to build the pruning index. The candidate centers which are not useful are pruned by using the pruning algorithm(ZHANG et al., 2015).

12. The PS-Join stores the dataset size to calculate the corresponding cost of every hash table. The hash table uses nearby buckets. PS-Join abandons pairs of objects placed in buckets that are far from each other. Then, it cuts down many similarity calculations. LSH is improved by BPLSH based on PS-join. PSjoin is based on LSH with Buckets Pruning method because it may be possible that some pairs within two nearby buckets mayor are not included in the final result. To efficiently take a similar pair of objects that need to be hashed into separate buckets, an LSH table is provided with an extra bucket array. The Bucket array stocks a set of values for upper bounds and lower bounds for every bucket depending on the distance between its object members and the random hyperplanes which form LSH. This information is used to prune the buckets that are not needed in the comparison. PS-Join helps to provide a better result to calculate the top-k similarity (XIONG; ZHU; YU, 2015).

13. Network similarities are better than the other sorts of similarities. When network similarities model the similarity problem assuming that the granularity can change, it allows partial matching and ranking relevance. The terms weights are also very well handled. Network similarities are easy to implement because they have several toolboxes that can deal with them, they depend on document size, and it handles the dependency of terms. Moreover, they have semantic sensibility. Network similarities are better than any other methods described in this work.

Table 2 describes a comparative analysis of the similarity measures discussed in Table 1 based in the following parameters to evaluate the similarity algorithms. These parameters are mathematically proved in (DEZA; DEZA, 2009), these parameters are

[8] An x-star network is an information network which consists of centers with connections among themselves, and different type attributes linked to these centers.

the symmetry (conformity, measure, shape and relative position), path-based (distance measures by the path length in a network), triangle inequality (lastly, the length of one side is less than the sum of the lengths of the other two sides), and Pruning(which are the pruning algorithms). Table 2 compares HeteSim , PathSim, PCWR, SimRank, RolESim, P-PageRank, NetSim and JoinSim because of their importance.

Table 2 – Comparative Analysis of the Similarity Measures beetween Symmetry, Triangle Inequation, Approach, Path-Based and Pruning parameters.

Similarity Name	***Symmetry***	***Triangle Inequation***	***Approach***	***Based***	***Pruning***
HeteSim (SHI et al., 2014)	yes	no	Relecance Based	PRW	N.A
PathSim(SUN et al., 2011)	yes	no	meta-path Based	Path Count	Clustering Algorithm
PCWR(LAO; COHEN, 2010)	no	no	N.A	RW	N.A
SimRank(JEH; WIDOM, 2002)	yes	no	N.A	PRW	N.A
RolESim(JIN; LEE; HONG, 2011)	yes	yes	N.A	PRW	N.A
P-PageRank(JEH; WIDOM, 2003)	no	no	N.A	RW	N.A
NetSim (PENG et al., 2015)	no	no	N.A	RW	By NetSim Pruming algorithm
JoinSim(XIONG; ZHU; YU, 2015)	yes	yes	Join Path Based	RW	By bucket Pruning algoritm

Source: Own authorship.

From this comparative analysis, according to Table 2, it can be concluded that:

1. The JoinSim method is the best if it is used to find the top k similar pairs of objects based on the join path specified by a user in a HIN. Because JoinSim satisfies the Triangle Inequality Property, it can be used to find out similarity join in large networks. A triangle inequality can find most interesting structures on a metric space via convergence. For example any convergent sequence in a metric space is a 'Cauchy Sequence' that is a direct consequence of the triangle inequality.

2. HeteSim has symmetry; it has no triangle inequality. HeteSim works with Pairwise Random Walk, and it has no pruning techniques.

3. PathSim has symmetry, has no triangle inequation, uses the approach with meta-path Based that uses Path count, has pruning and has clustering algorithms.

4. PCWR has no symmetry, no triangle inequation, no pruning algorithm and has Random Walk.

5. SimRank has symmetry, no triangle inequation, and is based on Pairwise Random Walk.

6. RolESim has symmetry, triangle inequation, and is based on Pairwise Random Walk.

7. P-PageRank has no triangle inequation and uses just Random Walk.

8. NetSim has no triangle inequation and is based on Random Walk; the pruning is the NetSim pruning algorithm.

9. JoinSim has symmetry, triangle inequation , is based on join path, Random Walk, and bucket pruning algorithm.

10. The PathSim method can capture the semantic meaning of the meta-path, but it is not able to handle similarity join in a large network as it does not satisfy the triangle inequality property. However, The PathSim measure does not support LSH, which is satisfied by the similarity measure, and this drawback prohibits its application for similarity join in multidimensional spaces (SUN et al., 2011). Achievements of the efficiency of the PathSim method have limited behavior to the meta-path with infinite length, but PathSim is very good for shorter meta-paths (SUN et al., 2011). Less time is required for the NetSim measure to be computed. The scale and size of the attribute network is smaller than the actual x-star network (ZHANG et al., 2015). The attribute network is assembled from the entire x-star network.

11. The following methods are designed to work just with Homogeneous Networks: SimRank (JEH; WIDOM, 2002), PageRank (BRIN; PAGE, 1998), PageRank Personalizado (JEH; WIDOM, 2003), Lao et al (LAO; COHEN, 2010) ,Wang et al (WANG et al., 2011) and NetSim (ZHANG et al., 2015).

12. The other similarity measures are works with heterogeneous networks, such as PathSim (SUN et al., 2011) HeteSim (SHI et al., 2012),Wang et al (WANG; HU; YU, 2012) ,Yu et al. (YU et al., 2012),LSH-HeteSim citeli2014efficient,AVGSim (MENG et al., 2014),PS-join (XIONG; ZHU; YU, 2015), Path-based similarity join (WANG et al., 2016),ESim (SHANG et al., 2016) and RelSim (WANG et al., 2016).

The use of each similarity measure has been discussed. The comparative analysis of all network measures presented in this work was done with the help of parameters like Path-Based, Symmetry, Triangle Inequality and Pruning. Based on the analysis of the cited parameters, we advocate that the JoinSim method is the best for finding the top k similar pair of objects in HIN. Although similarity measures belonging to the same family share the same characteristics, these measures adopt different approaches to solve the same problem. For example, similarity measures considered to be classical (vectorial model), although all these measures use vector arithmetic, the range used by each of the changes, as well as the operations used between the matrices. For other categories or groups of network, similarities also follow the same pattern, deal with and solve the same problem through different approaches, some more efficient than others and at higher cost.

2.5 Remarks

The algorithms described and the similarity measures for text document collections have few semantic analysis. They manipulate relationships among a set of terms and documents. They do not use concepts related to the documents and terms or terminology products.

Different from similarity measures for text document collections, knowledge-based measures try to identify the similarity degree between concepts by using algorithms supported by lexical resources and/or semantic networks, so they are kind of a hybrid method that uses the best of the two worlds.

The Network-based Similarity Measures obtain the similarity according to the links between the nodes. Getting similarity through a network-based similarity measure has more semantic than other similarity measures because of the relations among nodes. The edges between the nodes can have weights that increase semantics compared to the other kinds of similarity measures. But it is more expensive to maintain the network than to use the other methods. The network measures are the related efforts of this work.

The similarity measures discussed here for Homogeneous Information Network, and the similarities for Heterogeneous Information Networks do not use terminology products. The proposed algorithm aims to fill this gap extracting similarity from the Heterogeneous Information Network using meta paths since the most state of the art have shown to be effective. It also adds terminology products to aggregate semantics and obtains more accurate similarity results.

3

Theoretical Basis and Technologies

This chapter presents the theoretical basis and technologies used in the development of this work. The theoretical basis of information network, which includes meta-path and ontologies, are important in this work because they help to connect documents whose words have semantic relationship. Ordinary technologies used in this work, such as programming libraries, are also presented in this chapter.

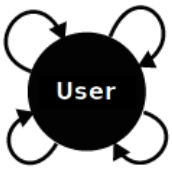
3.1 Heterogeneous Information Networks

Most real-world systems consist of a great number of interacting, multi-typed components (HAN, 2009), such as communication, human social activities, biological networks, and computer systems. In such systems, the interacting components constitute interconnected networks, which can be called information networks without loss of generalization (SUN; HAN, 2013). It is clear that Information Networks are everywhere and form a crucial component of modern information infrastructure. Information network analysis has received great attention from researchers from many areas of study, such as Social Science, Computer Science, Physics, and so on. Especially, the information network analysis has become an exciting research theme in data mining and information retrieval fields in the past decades. The central model is to mine hidden patterns in link relations from networked data. The analysis of information network is related to works in social network analysis (WASSERMAN; FAUST, 1994), (OTTE; ROUSSEAU, 2002), analysis and link mining (GETOOR; DIEHL, 2005), (JENSEN; GOLDBERG, 1998), (FELDMAN, 2002), hypertext and web mining (CHAKRABARTI et al., 2002), graph mining (HOLDER; COOK, 2005) and network science (LEWIS, 2011).

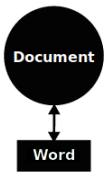
Researchers have begun to investigate the relationship between different types of data to capture the semantics between them. As a result, Heterogeneous Information Networks (HIN) have been proposed, because they have multiple types of nodes and various types of edges. Heterogeneous Information Networks often imply multiple semantic

Figure 1 – The concept of network schema is proposed to describe the meta structure of a network.A meta template for an information network with the object type mapping and the link type mapping which is a directed graph defined over object types with edges as relations.

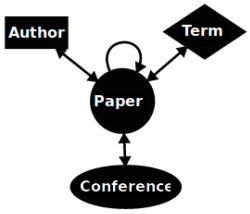
(a) Multi-Relation Heterogeneous Information Network Schema.



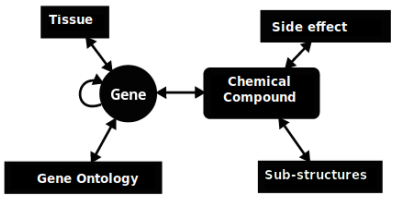
(b) Bipartite Heterogeneous Information Network Schema.



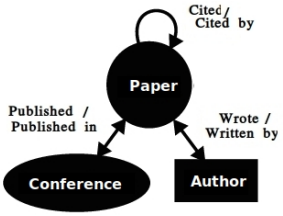
(c) Star Heterogeneous Information Network Schema.



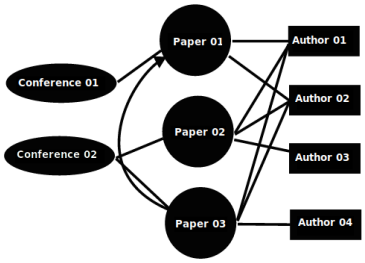
(d) Multiple hubs Heterogeneous Information Network Schema.



(e) Authors Heterogeneous Information Network Schema.



(f) Network Heterogeneous Information Network Authors Instance.



Source: (SUN; HAN, 2013).

structures different from homogeneous networks, which have only one type of relationship between only one type of vertex. In HIN, the edges indicate interaction among several types of objects in the network, and generally imply similarity or influence between these objects. However, it may be difficult to represent the characteristics of HIN. To facilitate navigation between the different types of nodes and edges, network schemes are proposed. The schema of a HIN specifies constraints on the sets of objects and the relations between objects. These constraints make a network of semi-structured heterogeneous information, guiding the exploration of network semantics. Figures 1a (one type of node but many types of edges), 1b (two types of nodes and one type of edge), 1c (similar to a star with a center node and different kinds of nodes and edges) and 1d (multiple hubs) show examples of HIN schemes. Schemes of different types of heterogeneous networks were adapted from (SUN; HAN, 2013) in Figure 1e. It is a schema of authors that publish papers in conferences, and Figure 1f is the instance of the schema of Figure 1e.

Most real systems have a large number of interactions with different types of components (HAN, 2009), such as human social activities, communication, computer systems, co-authoring networks and biological networks.

An information network represents an abstraction from the real world and focuses

on the objects and the interactions between these objects. It turns out that this level of abstraction has great power, not only representing and storing essential information about the real world, but also providing a useful tool for mining knowledge from this abstraction, exploring the power of links. Formally, information networks are defined in the literature according to the definition:

An information network is an oriented graph $G = (V, E)$ with a mapping function in an object type $\varphi : V \rightarrow A$ and a mapping function on a link $\psi : E \rightarrow R$ (SUN; HAN, 2013), (SUN; YU; HAN, 2009). Each object $v \in V$ belongs to a particular type of object in a set with various object types $A : \varphi(v) \in A$, and each link $e \in E$ belongs to a particular type of relationship in the set of relationship types $R : \psi(e)\ inR$. If two links belong to the same relationship type, the two links share the same output object and the same input object. Unlike the definition of traditional networks, it is possible to explicitly distinguish the types of objects and relationships in information networks.

The information network is called a Heterogeneous Information Network if the types of objects $|A| > 1$ or types of relationships $|R| > 1$; otherwise, it is a Homogeneous Information Network. The network schema represented in Figures 1e,1f, denoted as $T_G = (A, R)$, is a model goal for an information network $G = (V, E)$ with object type mapping $\varphi : V \rightarrow A$ and the edge type mapping $\psi : E \rightarrow R$, which is a directed graph defined on the types of objects A, as edges as relations of R (SUN; HAN, 2013), (SUN; YU; HAN, 2009).

The scheme and instance of a network of co-authors was adapted from (SUN; HAN, 2013) , shown in figures 1e and 1f. The network scheme helps to describe the meta structure of a network. An information network following a network scheme is then called a network instance of the network scheme.

The *meta-path* (SUN et al., 2011) is shown in Figures 2a,2b,2c. A meta-path P is a path defined in a schema $S = (A, R)$, and is indicated in the form of $A_1 \stackrel{R_1}{\rightarrow} A_2 \stackrel{R_2}{\rightarrow} ... \stackrel{R_l}{\rightarrow} A_{l+1}$, which defines a composite relationship $R = R_1 \circ R_2 \circ ... \circ R_l$ between objects $A_1, A_2, ..., A_{l+1}$,where $\circ$ denotes the composition operated on the relations. The *meta-path* is an interesting proposal to ensure the rich semantics of heterogeneous networks, based on different *meta-paths*. The objects have different connection relations with several semantic paths; which can have an effect on many data mining tasks. For example, the similarity between authors can be assessed on the basis of different *meta-paths*.

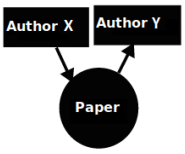
Figures 2a , 2b and 2c represent a meta-path of a co-authorship network adapted from(SUN; HAN, 2013). As shown in Figure 2a,2b,2c, authors can be linked by means of the *meta-path* "Author-Paper-Author" (APA), shown in Figure 2a. The importance of authors under path APA seeks authors who write many works with other authors. While the authors publish articles in the same conferences, the *meta-path* to be followed is "Author-Paper-Conference-Paper-Author" ($APCPA$), shown in Figure 2b. Whereas the

meta-path "Author-Paper-Conference" (APC) shows the authors that have their articles published in a conference, shown in Figure 2c.

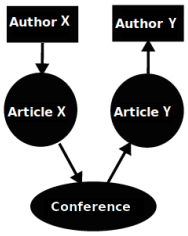
In addition, Table 3 shows paths and semantic examples of these meta-paths. That the semantics under these paths are different. Path APA means authors who collaborate in the same articles, for example, the co-authorship relationship, while path $APCPA$ means authors who publish articles in the same conference. With *meta-paths*, different types of objects can be connected. For example, authors and conferences can be connected with path APC, which means the publications of articles with authors in certain conferences.

Figure 2 – Two objects in a heterogeneous network can be connected via different paths and these paths have different meanings this different paths are called meta paths. This paths expresses the relation between objects. This Figure represents the Meta-paths of the Authors schema. schema.

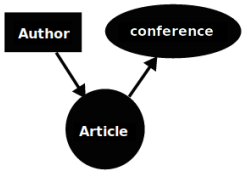
(a) Author-Paper-Author Meta-Path.



(b) Author-Article-Conference-Article-Author Meta-Path.



(c) Author-Article-Conference Meta-Path.



Source: (SUN; HAN, 2013).

Table 3 – Examples of the path instance from the meta-paths and the meaning of its connections.

Path instance	***Meta-Path***	***Intended semantics***
Angélica-RI-Alessandra	*Author-Paper-Author*	*Author collaborated with the same article*
Mateus-EST- Evandro		
Angélica-RI$_x$-SBC-RI$_y$-Alessandra	*Autor-Paper-Conference-Paper-Author*	*Authors published in the same conference*
Mateus-EST$_x$-SBC-EST$_y$- Evandro		
Alessandra-RI-KDD	*Author-Paper-conference*	*Authors who published in a conference*
Evandro-EST-PLN		

Source: Own authorship.

Most of the existing similarity measures are defined for Homogeneous Networks. The semantic meanings between paths are not taken into account in Homogeneous Networks. Thus, the methods used in Homogeneous Information Networks cannot be directly applied to Heterogeneous Information Networks. As the HIN nodes and edges cannot be dealt with as being of the same type, the semantic information contained in the network is lost.

On the other hand, treating each node as a distinct type can also disperse the information in the network. Most studies that manipulate HIN use pre-labels to let the heterogeneous network semi-structured so that it is easier to navigate between the desired nodes and capture the essential semantics of the real world. It is easier to work with heterogeneous semi-structured networks. As a unique and effective feature capture tool, the *meta-path* has been widely used in many data mining tasks in HIN, such as similarity measure, grouping, and classification.

3.2 Terminology Products

The use of dictionaries for extracting information in documents exploits data models of the domains of interest that enable the search for occurrences in the text. Some disadvantages are the restriction of dictionary terms and the lack of recognition of some terms due to the spelling variation (TSURUOKA; TSUJII, 2004). An ontology describes a collection of representational primitives with which to model a domain of knowledge or discourse. (GRUBER, 1993).

An ontology consists of precise definitions and a formal collection of axioms, delimiting the interpretation and the use of terms present in the vocabulary (GRUBER, 1993). Ontology is a formal, clear description of concepts in a domain of discourse that is the concepts, properties of each concept describing various features and attributes of the concept, and restrictions on slots.

Every field creates ontologies to restrict complexity and organize information into data and knowledge. As new ontologies are created, their use hopefully improves problem-solving within that domain.

The selection of ontologies supports causality mining in pharma by categorizing recognized specific relationships to a causality connection ontology. Ontologies also enrich mining health records, semantic web mining, insights, semantic publishing, and fraud detection.

Ontologies provide users with the required structure to link one piece of information to others on the Linked Data network. Because they are used to define common modeling representations of data from heterogeneous systems and distributed and databases, ontologies allow database interoperability, smooth knowledge management, and cross-database search.

The main characteristics of ontologies is that they allow automated reasoning on data because their concepts have fundamental relationships. The afore-mentioned reasoning is easy to implement in semantic network databases that use ontologies as their semantic schema.

Ontologies are frameworks that represent shareable and reusable knowledge across domains. Their capacity to describe relationships and their high interconnectedness make them the bases for modeling high-quality, connected and coherent data.

For Example, translating research papers from every field is a problem made easier when specialists from different nations keep a controlled vocabulary of jargons between each of their languages. An advantage of its use in visual data repositories is the possibility of searching for images by searching the ontological annotations instead of using visual attributes of the image (LOPES et al., 2007). Another data model widely used by researchers in the field of health and medicine is the UMLS (Unified Medical Language System)(BODENREIDER, 2004).

UMLS is a type of file and software that opposes many biomedical and health vocabularies and standards turnable interoperability between computer systems. UMLS is used to develop applications, like classification tools, dictionaries, electronic health records, and language translators. This data model is known to facilitate the creation of information systems and services with a high degree of effectiveness and interoperability, through the distribution of terminological keys and classification and codification of biomedical standards (BODENREIDER, 2004). UMLS is a system composed of ontological knowledge and terminology accumulated over the years, used in automatic approaches and semi-automatic textual analysis (BRIN, 1998)

UMLS links health information, medical terms, drug names, and billing codes across different computer systems and so is a very powerful tool for semantics. An example is linking linking terms and codes between the doctor, the pharmacy, and the insurance company. UMLS has many other uses, including search engine retrieval, data mining, public health statistics broadcasting, and terminology research (CHEN et al., 2018).

UMLS has three tools, called the Knowledge Sources:

- Metathesaurus: consists of codes and terms from many vocabularies, including ICD-10, MeSH, and SNOMED CT.
- Semantic Network: consists of semantic types (different kinds of vertices) and their semantic relationships (different kinds of edges).
- consists of specialist Lexicon and Lexical Tools: Natural language processing tools.

ICD(International Classification of Diseases) is made of a medical classification list made by the World Health Organization (WHO). ICD-10 is the 10th revision of the International Statistical Classification of Diseases and Related Health Problems (ICD). It contains complaints, signs, codes for diseases,abnormal findings, social circumstances, symptoms, and diseases or external causes of injury.

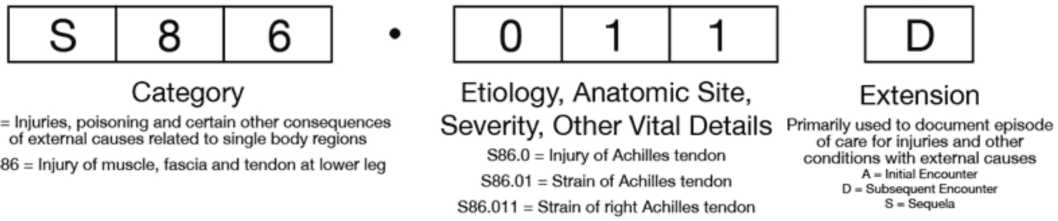
ICD is the settlement for the identification of globally health direction and statistics. It is also the international standard for broadcasting diseases and health (HELLMAN et al., 2018). ICD is the diagnostic classification standard for all clinical and research objectives. It defines the set of disorders, diseases, injuries and other related health states of affairs in one's life listed in a comprehensive hierarchical form that permits the analysis retrieval and easy storage of health information for evidence-based decision-making.

ICD also distributes and correlates health information among hospitals, settings, regions and countries, apart from contrasting data in identical locations across different lengths of time.

Some ICD applications contain prevalence of diseases, and monitoring of the incidence observing reimbursements and keeping track of safety and resource allocation direction, and feature directions. They also contain death estimates, as well as diseases, reasons for encounter, symptoms, injuries, determinants that affect the health status, and extrinsic causes of diseases. ICD-10 was approved in May 1990 by the Forty-third World Health Assembly. It has been applied in more than 20,000 scientific articles and used by more than 100 countries. The new ICD is an ontology[1].

Each ICD-10 number code represents something. Code sets can have from three to seven characters and numbers. Many three-character codes are used as headings for categories of codes; and can expand to four, five, or six characters to enrich specific details concerning the diagnosis, as shown in Figure 3

Figure 3 – ICD-10 has seven digits, every digit has a meaning. The first triad represents the category. The second triad represents the severity, the etiology, and the anatomic site and other vital details. The last digit represents the extension. ICD-10 code example.



Source: Available online at <https://www.webpt.com/sites/default/files/icd-10-cmcodeachillesstrainexample.gif>

The first three characters of an ICD-10 code show the category of the diagnosis. For example, the letter "S" designates that the diagnosis relates to "Injuries, poisoning and certain other consequences of external causes related to single body regions." "S" used in association with the numerals "8" and "6" means that the diagnosis falls into the category of "Injury of fascia, muscle and tendon at lower leg." A three-character category

[1] — available online at <https://dkm.fbk.eu/technologies/icd-10-ontology>.

that has no more subdivisions can stand alone as a code. In this case, nonetheless, higher specificities are possible, and they should fill in as many "blanks" as possible.

The next three characters (characters four to six) resemble the similar etiology (i.e., set of causes, cause, or condition or manner of causation of a disease), severity, anatomic site, or other vital clinical details. So, numbers "0," "1," and "1" mean a diagnosis of "Strain of the right Achilles tendon."

SNOMED-CT (SNOMED Clinical Terms)(WILLETT et al., 2018) is an inclusive, multilingual clinical healthcare terminology. It is a resource with inclusive and scientifically approved clinical content that allows the consistent and processable description of the clinical content in electronic health records. It is used in more than fifty countries because it can be used to represent clinically important information in a consistent, dependable and inclusive way as an integral part of generating electronic health information. Also it manages the advancement of general high-quality clinical content in health records. It provides a standardized way to represent clinical phrases captured by the clinician and allows the mechanical interpretation of such phrases. It is a clinically approved controlled vocabulary that is semantically rich and that simplifies developmental growth in order to meet requirements. It has located clinical information that benefits clinicians, individual and patients, as well as populations and it, manage evidence-based care. The use of an Electronic Health Record (EHR) increases communication and improves the availability of valuable information. There are many advantages to saving clinical information in ways that permit meaning-based retrieval. Advantages ranges from expanded opportunities for real-time decision support to more accurate retrospective broadcasting for research and management.

SNOMED-CT is a based clinical information that benefits patients and clinicians as well as inhabitants. It also manages evidence-based care. It enables clinical health records to help people by allowing very important medical data and clinical information to be documented using a constant and common characterization during examinations. This enables direction and decision support systems to analyse records and provide actual-time advice, for instance, over clinical alerts. It removes language barriers permitting multilingual use and supporting the distribution of suitable information with other systems in delivering care to a patient through data catching. That allows comprehension and interpretation of common information from providers. It permits inclusive searches that recognize patients who need to be followed up or have their treatments switched based on corrected guidance. It enables clinical health records that facilitate the early identification of come out health issues by monitoring the health of the population and the responses to changing clinical practices. It allows accurate and targeted access to valuable information, reducing costly duplication and errors. It also enables the delivery of useful data to support clinical research and contributes with evidence for future improvements in treatments.

When investigating outliers and exceptions it enhances the audits of healthcare delivery providing options for a detailed analysis of clinical records. It allows the feature of care knowledgeable by individuals. Health records based on healcare enable decision-making by allowing links among clinical enhanced clinical directions and records and protocols. As a consequence, this lowers penalty of treatment, duplicate and inappropriate tests and restricts the impact and frequency of adverse healthcare events. It also increases the cost-effectiveness and quality of care brought to users. The Figure 4 shows the difference between SNOMED-CT and ICD.

Table 4 – SNOMED-CT is a multilingual and inclusive clinical health technology. ICD is an international classification list of diseases related to health problems.

	SNOMED CT	**ICD**
Type	Terminology system	Classification system
Purpose	Information input	Information output
Function	Describes and defines clinical information for primary data purposes	Add and categorize clinical information for secondary data purposes

Source: Own authorship.

3.3 Supporting Technologies

The technologies used in this work are presented in this section. Scipy was used as it has a user-friendly interface and efficient numerical routines such as the classic similarity measures adopted in this work, and PyMedTermino was used because it is a module for easy access to the main medical terminologies in Python. It facilitates the access to terminologies.

3.3 Scipy

SciPy is a Python-based ecosystem for open-source software for mathematics, science, and engineering. It is a module for easy access to the main medical terminologies in Python. These are some of the main packages:

Scipy library, Ipython, Matplotlib, NumPy, Sympy, and Pandas. The SciPy library is one of the core packages that make up the SciPy stack. It provides many user-friendly and efficient numerical routines such as routines for numerical integration and optimization.

The following similarity measures provided by the Scipy library in python were used used (JONES; OLIPHANT; PETERSON, 2014): Bradycurtis, Canberra, Chebyshev, Cityblock, Correlation, Cosine, Dice, Euclidean, Hamming, Jaccard, Kulsinski, Maha-

lanobis, Minkowski, Rogerstanimoto, Russellrao, Seuclidean, Sqeuclidean, Sokalmichener, Sokalsneath, Yule and Wminkowski.

3.3 PyMedTermino

PyMedTermino (Medical Terminologies for Python) is a Python module for accessible access to the main Medical Terminologies in Python (LAMY; VENOT; DUCLOS, 2015). The following terminologies are supported: MedDRA (BROWN; WOOD; WOOD, 1999), UMLS (BODENREIDER, 2004), SNOMED CT (DONNELLY, 2006) and ICD10 (ORGANIZATION et al., 1993) icons. PyMedTermino facilitates the connection to terminologies; terminology contents should be obtained separately (e.g., downloaded from UMLS).

SNOMED CT is broadly used as it is the most comprehensive and precise clinical health terminology product available. It is distributed around the globe by SNOMED International. SNOMED CT has been developed to assure it meets the diverse needs and expectations of clinicians worldwide and is now accepted as a common global language for health terms. Patients and healthcare professionals benefit from improved health records, clinical decisions, and analysis, leading to higher quality, consistency and safety in health care delivery (DONNELLY, 2006).

ICD10 is the Tenth revision of the International Statistical Classification of Diseases and Related Health[2].

MedDRA is a highly specific standardized medical terminology , which is rich to aid sharing of regulatory information internationally for medical products used by humans[3].

UMLS integrates and distributes key terminology, classification, coding standards and associated resources to promote the creation of more effective and interoperable biomedical information systems and services, including electronic health records (BODENREIDER, 2004).

3.4 BioPython

Biopython is a set of openly available tools for biological computation written in Python by a worldwide team of developers.

It is a distributed collaborative energy to develop Python libraries and applications that address the requirements of current and future work in bioinformatics. The source code is made available under the Biopython License, which is very liberal and compatible with practically every license in the world.

[2] http://apps.who.int/classifications/icd10/browse/2010/en
[3] https://www.meddra.org/

They are a member project of the Open Bioinformatics Foundation (OBF), who care of their domain name and hosting for our mailing list etc. OBF used to host our development repository, issue tracker and website but now they are hosted by GitHub.

3.5 Igraph

Igraph is a collection of network analysis tools with emphasis on ease of use, portability, and efficiency. It is open source and free, and can be programmed in Python, C/C++, and R.

It is a package that makes working with graphs easier, and the build process simpler. It has methods to work with clustering, centrality, connectivity, agglomeration coefficient, degrees distribution, resistance, pattern mixing, degrees correlation, network statistics, and topologies.

3.6 Remarks

This chapter shows the theoretical basis such as Heterogeneous Information Networks and terminology products (UMLS, ICD and SNOMED-CT), and and technologies such as Scipy, PyMedTermino, BioPython, and IGraph. They helped the construction of a similarity measure using terminology products. These technologies were exploited as follows.

All distances found in Scipy were used for the construction phase of the network of scientific articles and the network of medical records. A threshold was established to make the connection between these documents and the connection was given according to its attributes. On the other hand, PyMedTermino was used to access the SNOMED-CT, ICD 10, MedDra and UMLS to extract the terminologies synonyms and the list of diseases, signs and symptoms, abnormal findings, external causes of lesions. Moreover, Biopython was used to retrieve the scientific articles by using the Entrez module. It enabled the retrieval of articles by keywords, CUI, DOI, titles, abstracts, by words similar to UMLS among others. Finaly, the Igraph was used to create the graph and its structure. It calculated coefficient of grouping, minimum paths, related components among others.

4

The Algorithms: Network Creator and HeteSimTKSQuery

This chapter describes the Heterogeneous Network Creation which presents the Network Work Creator algorithm that creates the nodes and edges of the Heterogeneous Information Network. Also describes the algorithm HeteSimTKSQuery, which uses the HeteSimTKS similarity measure proposed in this work the three experiments: the first exploits five classic similarity measures; the second exploits the hybrid approach mixing, the classical algorithms with the HeteSimTKSQuery; and the third used the HeteSimTKSQuery that creates the network similarity method with medical terminologies using PyMedTermino. The results of the experiments are presented in this chapter that treats the accuracy, the precision and the recall measures. The runtime of the three experiments are shown according to their precision.

Many real systems can be represented as a graph or a network of multi-typed components with a large number of nodes and diverse types of relationships. Heterogeneous Information Networks (HIN) are structures interconnected with multi-typed data supporting the rich semantics of structural types of objects and links. Different information can be presented using different types of data, and there may be complementary information linked in to an HIN. So, there is knowledge to be discovered. Terminology and Knowledge Structures (TKS) such as ontologies, thesaurus, and dictionaries, can be sources of linguistic representations and knowledge to be used to create and enrich HIN.

4.1 Creation of Heterogeneous Information Networks

Algorithm 1, called **Network Creator**, creates a network (line 4 to 27) whose input data is a list of documents (line 2). The similarities between each document are calculated to establish connections among the documents (line 8). The similarities are measured by means of similarity algorithms and measures cited in Section 3.3. A threshold = 0.5 was considered based on empirical experiments in order to increase the accuracy of the created

connections (line 9 to 11).

Algorithm 1: Network Creator

```
1: Input: Medical Records/Document collection C
2: Output: Network N
3: for each collection in C do
4:     for each documentA in collection do
5:         for each documentB in collection do
6:             for each attributesA in documentA do
7:                 for each attributesB in documentB do
8:                     Sim = similarity(attributesA, attributesB, similarity algorithm)
9:                     if Sim >= 0.5 then                                  ▷ Threshold
10:                        N = connect(documentA, documentB)
11:                    end if
12:                    if HasSynonyms(attributesA, attributesB) then
13:                        N = connectByattributes(documentA,documentB)
14:                    end if
15:                    if HasConcepts(attributesA, attributesB) then
16:                        N = connectByattributes(documentA,documentB)
17:                    end if
18:                    if HasTerms(attributesA, attributesB) then
19:                        N = connectByattributes(documentA,documentB)
20:                    end if
21:                end for
22:            end for
23:        end for
24:    end for
25: end for
26: return N    ▷ The Network whose documents are connected through the attributes.
```

Afterwards, the attributes of the network are enriched by the TKS cited in Section 3.2(line 12 to 20). TKS provide access to synonyms and translations, and management of concepts and relations between them. For instance, **Network Creator** specifically uses the UMLS Metathesaurus to discover and label concepts pertaining to the document. The result provided by **Network Creator** is an HIN constructed by both classical similarity measures and TKS turning it into a semantically rich network. The network represents documents as nodes, and each node has attributes. These attributes establish the connections or links. By running the **Network Creator algorithm** with a collection of medical records, a network of medical records is created. It can distinguish the different contexts to know whether or not to connect a document with another. When the input data is a collection of scientific papers, an information networks of papers is established.

The algorithm is independent of the type of input, in this case, medical records, or scientific articles are used that means the algorithm is generic. For each similarity distance used, **Network Creator** can define a different network. The best network is taken using **Algoritm 2**, called **HeteSimTKSQuery**, whose input data are:

- N: HIN created by **Network Creator**;
- node a: the query which is the node to measure the similarity with other nodes in the network;
- m: dimension of a hash vector for the pruning;
- t: number of hash table necessary for the pruning;
- node p: nodes p (the starting node), and
- P: the meta-paths (the meta-path to be covered).

Initially, **Algorithm 2** creates an indexing structure for HIN N (line 3); and produces t hash tables G, cyclically presenting the index of meta-paths (lines 4 to 7). Each hash table G is created with each collection of information to be linked. G has the set of candidate nodes Q, which have their interconnections checked by **HeteSimTKS** function (line 9). **HeteSimTKS** is a similarity algorithm defines HIN by using TKS. The **HeteSimTKS** uses meta-paths , and the classic similarity measures (see Appendix B) and the following terminology products on PyMedTermino: SNOMED CT, ICD10, MedDRA, CDF, and UMLS. **HeteSimTKS** calculates the similarity between the query node a and each target node in the candidate set Q by using p and the TKS(o). Given a meta-path p, **Algorithm 2** calculates the value of HeteSimTKS between the x and y nodes of different collections of information following Equation 4.1:

$$\begin{aligned} HeteSimTKS(x, y, FO) = 2 \times |\{p_{x \rightsquigarrow y} : p_{x \rightsquigarrow y} \in P\}| + \sum |FOx \leftrightsquigarrow FOy| \\ -(|\{p_{x \rightsquigarrow x} \in P\}| + |\{p_{y \rightsquigarrow y} \in P\}|) \end{aligned} \tag{4.1}$$

in which $p_{x \rightsquigarrow y}$ is a path instance between x and y; $p_{x \rightsquigarrow x}$ is between x and x $p_{y \rightsquigarrow y}$ is between y and y; p is a meta-path, and $FOx \leftrightsquigarrow FOy$ is the TKS.

Algorithm 2: HeteSimTKSQuery

Input: data set N,m,t, node p, node a,meta-path p
Output: a set of nodes pairs with their similarity values
Build Dictionary Indexing for N as Locality Sensitive Hash▷ creates a LSH indexing structure for given hash dictionary set N
for each hash Table $N_{H_m^i}(1 \leq i \leq t)$ **do** ▷ produces t hash tables cyclically, and eventually gets a collection of all nodes which are mapped to the same Group, as the candidate set Q
    Hashing query nodes a to a group G_i
    add nodes hashed to target group G_i to set Q
end for
for each node $p \in Q$ **do** ▷ it calculates the similarity between the query node p and each target node in candidate set Q using HeteSimTKS, and returns a result set of all objects with a similarity value.
    HeteSimTKS(a,p,o)
    R = add all(a,p) ▷ add the pairs and their similarity values into the set R
end for
return R ▷ a set of object pairs with their similarity values

HeteSimTKS returns a set of all nodes R with a similarity value to p_x and p_y. The output result is a collection of nodes R, which form the candidate set with a similarity value enriched by **HeteSimTKS** by using the TKS o.

Pruning can reduce a lot of similarity measure computation considering that using a candidate subset generated by hashing minimizes computation times.

For example, Figure 4 depicts medical records which would be connected by the attributes “Sample Type / Medical Specialty”, “Sample Name”, “Description”, and “Keywords”. The similarity was calculated and if it was higher than the threshold, a connection between these medical records was created. Figure 4a depicts a case of a medical record where the attributes Sample Type / Medical Specialty, Sample Name, 2-D M-Mode, Doppler and Keywords exist. Each medical record is different, and contains different content in the attributes as well as different attributes from each other. As shown in Figure 4b, there are the attributes: Sample Type/ Medical Specialty, Sample Name, Description, Keywords, and the attribute Comments that was nonexistent on the medical record represented by Figure 4a, that increases the degree of complexity of the construction of the network because there are attributes that exists in some documents that does not exists in others.

Figures 5 and 6 depict medical scientific articles. **Network Creator algorithm** was used to connect the documents, and check the concepts, terms, and synonyms for each attribute. Figure 5 depicts one case of a scientific paper that presents the following

attributes: PMID(PubMed Unique Identifier), OWN(owners), STAT(Status), LR(Date Last Revised), IS(ISSN), DP (Date of Publication), TI(Title) LID(Location Identifier), AB(Abstract), FAU(Full Author), AU(Author), AD(Affiliation) and LA(Language). Each scientific paper is different from each other as they have different content in their attributes and different attributes from each other. As shown in Figure 6 attributes are the same, except for CI(Copyright Information), which was nonexistent on the scientific paper represented by Figure 5, and which increases the degree of complexity of the construction of the network because some attributes exist in some documents and not in others.

The scientific articles were semantically enriched using the dictionaries on Biopython. The metadata of articles and medical records are attributes of the nodes. For each classic similarity, a network was created. Moreover, for each similarity algorithm, a network was built considering a threshold = 0.5; dictionaries followed the same procedure. For each network the similarity algorithm HeteSimTKS had three versions: a pure meta-path similarity, the classic meta-path with a dictionary to aggregate semantics, and a version using only TKS. The meta-paths were formed by the links between each attributes see Figures 4,5 and 6 of each document. This meta-paths are the way that the algorithm HeteSimTKSQuery finds the path between one medical record and one medical article.

4.2 Experiments and Results

Initially the proposed algorithms related medical record and scientific articles. The medical records (See Figure 4) were extracted from the MTSamples Collection[1]. The scientific papers were collected from the PubMed Digital Library[2].The medical record samples transcript reports for many specialties and different types of expertise. It has 4,999 items of different fields of Medicine such as Dermatology, Cardiovascular, Pulmonary and others. The papers were retrieved according to the keywords and the areas of Medicine listed in the MTSamples Collection. Biopython was used to download the medical articles. It has a module, the **Bio.SeqIO module**, which allows one to obtain articles from the National Center for Biotechnology Information (NCBI) databases sorted by disease. The International Classification of Diseases (ICD) identifies diseases by using a unique identifier cited in the articles. **Bio.SeqIO** can also use a general similarity called NCBI.ELink similarity measure, which is calculated by considering the **PubMed ID** attribute of the whole document. The Biopython library uses UMLS and other medical knowledge tools to calculate the similarity measure.

After finding the similar articles, the articles were connected by using the following attributes: (i) NCBI.ELink similarity, (ii) title of articles, (iii) abstracts, (iv) keywords,

[1] http://mtsamples.com/
[2] https://www.ncbi.nlm.nih.gov/pubmed/

(v) authors etc[3]. The similarities were calculated using the classical similarity measures.

The connection of the medical records was conducted using different attributes, such as: (i)subjective, (ii) medications, (iii) allergies, (iv)objectives, (v) assessment, (vi) plan, and (vii) keywords.

Figure 7 depicts how the network of articles and network of medical records were created following **Algorithm 1**. First, the same types of documents (articles or records) were connected and represented by their different attributes, in such a way that they could have different connections between them. Afterwards, the two networks created in the first step were linked in order to compose a merged network, a Heterogeneous Information Network. Figure 7 depicts how edges varying the line traces representing different connections indicate the number of connections established between the nodes. The similarity algorithms proposed, **Algorithms 1** and **2**, manipulate meta-path and terminology products while the classic similarity measures apply the vectorial model. The vectorial measures like Cosine, Euclidean and the HeteSimTKS accuracy (part of **Algorithm 2**) have independent variables, but the **hybrid approach** HeteSimTKSQuery with classical similarity measures are dependent on each other. The possible values are normalized, so the possible values for each variable vary from 0 to 1. The closer to 1, the more similar they are.

[3] The full atributes can be found in <https://www.nlm.nih.gov/bsd/mms/medlineelements.html> but they can vary from document to document

Figure 4 – Examples of nodes of the medical record type from MTsamples.

(a) First example of medical record.The words in bold letters are the attributes from the node.

Sample Type / Medical Specialty: Cardiovascular / Pulmonary
Sample Name: 2-D Echocardiogram - 1
Description: 2-D M-Mode. Doppler.
2-D M-MODE:
1. Left atrial enlargement with left atrial diameter of 4.7 cm.
2. Normal size right and left ventricle.
3. Normal LV systolic function with left ventricular ejection fraction of 51%.
4. Normal LV diastolic function.
5. No pericardial effusion.
6. Normal morphology of aortic valve, mitral valve, tricuspid valve, and pulmonary valve.
7. PA systolic pressure is 36 mmHg.
DOPPLER:
1. Mild mitral and tricuspid regurgitation.
2. Trace aortic and pulmonary regurgitation.
Keywords: cardiovascular / pulmonary, 2-d m-mode, doppler, aortic valve, atrial enlargement, diastolic function, ejection fraction, mitral, mitral valve, pericardial effusion, pulmonary valve, regurgitation, systolic function, tricuspid, tricuspid valve, normal lv,

(b) Second example of medical record. The words in bold letters are the attributes from the node.

Sample Type / Medical Specialty: Cardiovascular / Pulmonary
Sample Name: 2-D Echocardiogram - 2
Description: 2-D Echocardiogram
COMMENTS:
1. The left ventricular cavity size and wall thickness appear normal. The wall motion and left ventricular systolic function appears hyperdynamic with estimated ejection fraction of 70% to 75%. There is near-cavity obliteration seen. There also appears to be increased left ventricular outflow tract gradient at the mid cavity level consistent with hyperdynamic left ventricular systolic function. There is abnormal left ventricular relaxation pattern seen as well as elevated left atrial pressures seen by Doppler examination.
2. The left atrium appears mildly dilated.
3. The right atrium and right ventricle appear normal.
4. The aortic root appears normal.
5. The aortic valve appears calcified with mild aortic valve stenosis, calculated aortic valve area is 1.3 cm square with a maximum instantaneous gradient of 34 and a mean gradient of 19 mm.
6. There is mitral annular calcification extending to leaflets and supportive structures with thickening of mitral valve leaflets with mild mitral regurgitation.
7. The tricuspid valve appears normal with trace tricuspid regurgitation with moderate pulmonary artery hypertension. Estimated pulmonary artery systolic pressure is 49 mmHg. Estimated right atrial pressure of 10 mmHg.
8. The pulmonary valve appears normal with trace pulmonary insufficiency.
9. There is no pericardial effusion or intracardiac mass seen.
10. There is a color Doppler suggestive of a patent foramen ovale with lipomatous hypertrophy of the interatrial septum.
11. The study was somewhat technically limited and hence subtle abnormalities could be missed from the study.
Keywords: cardiovascular / pulmonary, 2-d, doppler, echocardiogram, annular, aortic root, aortic valve, atrial, atrium, calcification, cavity, ejection fraction, mitral, obliteration, outflow, regurgitation, relaxation pattern, stenosis, systolic function, tricuspid, valve, ventricular, ventricular cavity, wall motion, pulmonary artery.

Source: Own authorship.

Figure 5 – First example of medical article.The words in bold letters are the attributes from the node.PMID(PubMed Unique Identifier), OWN(owners), STAT(Status), LR(Date Last Revised), IS(ISSN), DP (Date of Publication), TI(Title) LID(Location Identifier), AB(Abstract), FAU(Full Author), AU(Author), AD(Affiliation) and LA(Language).

PMID - 30030490
OWN - NLM
STAT - Publisher
LR - 20180721
IS - 1759-4820 (Electronic)
IS - 1759-4812 (Linking)
DP - 2015 Jul 20
TI - Epigenetic modifiers: activities in renal cell carcinoma.
LID - 10.1038/s41585-018-0052-7 [doi]
AB - Renal cell carcinomas (RCCs) are a diverse set of malignancies that have recently been shown to harbour mutations in a number of chromatin modifier genes - including PBRM1, SETD2, BAP1, KDM5C, KDM6A, and MLL2 - through high-throughput sequencing efforts. Current research focuses on understanding the biological activities that chromatin modifiers employ to suppress tumorigenesis and on developing clinical approaches that take advantage of this knowledge. Unsurprisingly, several common themes unify the functions of these epigenetic modifiers, particularly regulation of histone post-translational modifications and nucleosome organization. Furthermore, chromatin modifiers also govern processes crucial for DNA repair and maintenance of genomic integrity as well as the regulation of splicing and other key processes. Many chromatin modifiers have additional non-canonical roles in cytoskeletal regulation, which further contribute to genomic stability, expanding the repertoire of functions that might be essential in tumorigenesis. Our understanding of how mutations in chromatin modifiers contribute to tumorigenesis in RCC is improving but remains an area of intense investigation. Importantly, elucidating the activities of chromatin modifiers offers intriguing opportunities for the development of new therapeutic interventions in RCC.
FAU - de Cubas, Aguirre A
AU - de Cubas AA
AD - Department of Medicine, Division of Hematology and Oncology, Vanderbilt University Medical Center, Nashville, TN, USA.
FAU - Rathmell, W Kimryn
AU - Rathmell WK
AD - Department of Medicine, Division of Hematology and Oncology, Vanderbilt University Medical Center, Nashville, TN, USA. Kimryn.rathmell@vanderbilt.edu.
LA - eng

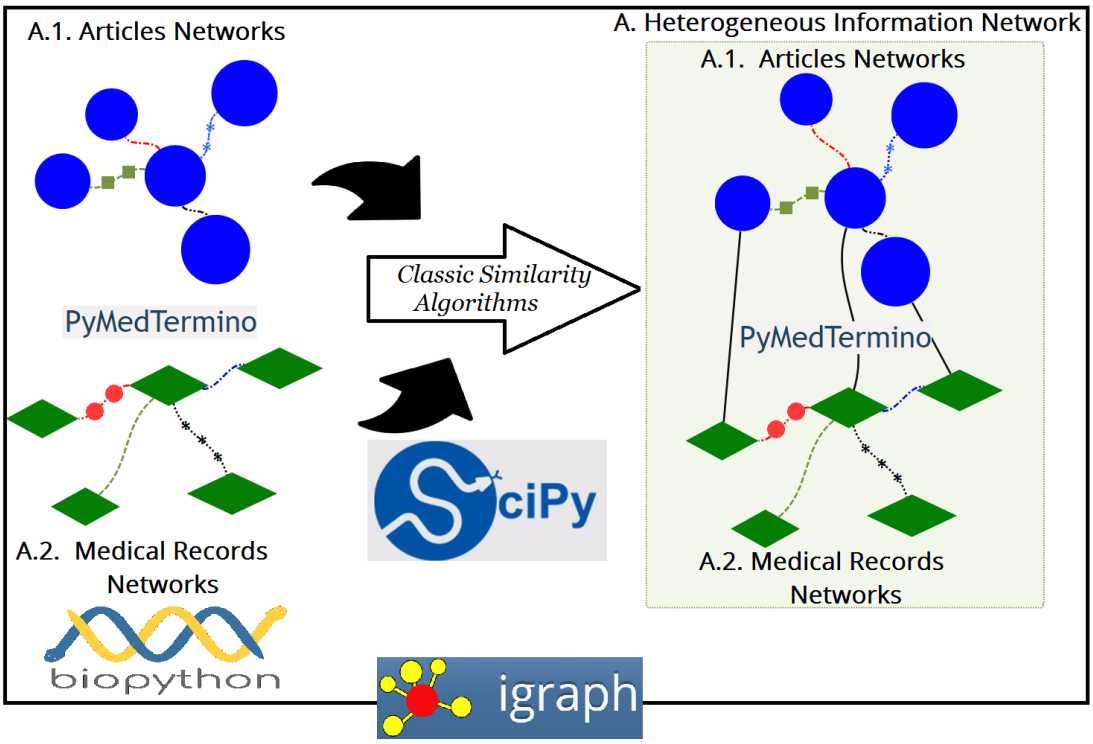
Source: Own authorship.

Figure 6 – Second Example of medical article. The words in bold letters are the attributes from the node. PMID(PubMed Unique Identifier), OWN(owners), STAT(Status), LR(Date Last Revised), IS(ISSN), DP (Date of Publication), TI(Title) LID(Location Identifier), AB(Abstract), FAU(Full Author), AU(Author), AD(Affiliation) and LA(Language).

PMID - 30030093
OWN - NLM
STAT - Publisher
LR - 20180827
IS - 1527-9995 (Electronic)
IS - 0090-4295 (Linking)
DP - 2015 Jul 17
TI - A Bilateral Metachronous Mesothelioma of the Tunica Vaginalis.
LID - S0090-4295(18)30721-0 [pii]
LID - 10.1016/j.urology.2015.07.003 [doi]
AB - This is a unique case of bilateral metachronous testicular mesothelioma of the tunica vaginalis. Testicular mesothelioma is a rare entity found in patients with or without asbestos occupational exposure. The tumor most commonly presents as a unilateral testicular mass. More rare presentations include bilateral synchronous or metachronous tumors. Treatment is with surgical resection and prognosis is not generally favorable. The benefits of adjuvant therapy with radiation or chemotherapy remain unknown and further studies are needed.
CI - Copyright (c) 2018 Elsevier Inc. All rights reserved.
FAU - Abello, Alejandro
AU - Abello A
AD - Beth Israel Deaconess Medical Center/Harvard Medical School, Urology Division, Boston, MA.
FAU - Steinkeler, Jennifer
AU - Steinkeler J
AD - Beth Israel Deaconess Medical Center/Harvard Medical School, Urology Division, Boston, MA.
FAU - Das, Anurag K
AU - Das AK
AD - Beth Israel Deaconess Medical Center/Harvard Medical School, Urology Division, Boston, MA. Electronic address: adas@bidmc.harvard.edu.
LA - eng

Source: Own authorship.

Figure 7 – Network Connection. Fist of all the nodes was separated in different types the papers and the medical records. For each type of node was created a network that was connected by a classic algorithms. Than was used terminological products to build a similarity algorithm for Heterogeneous Information Network.

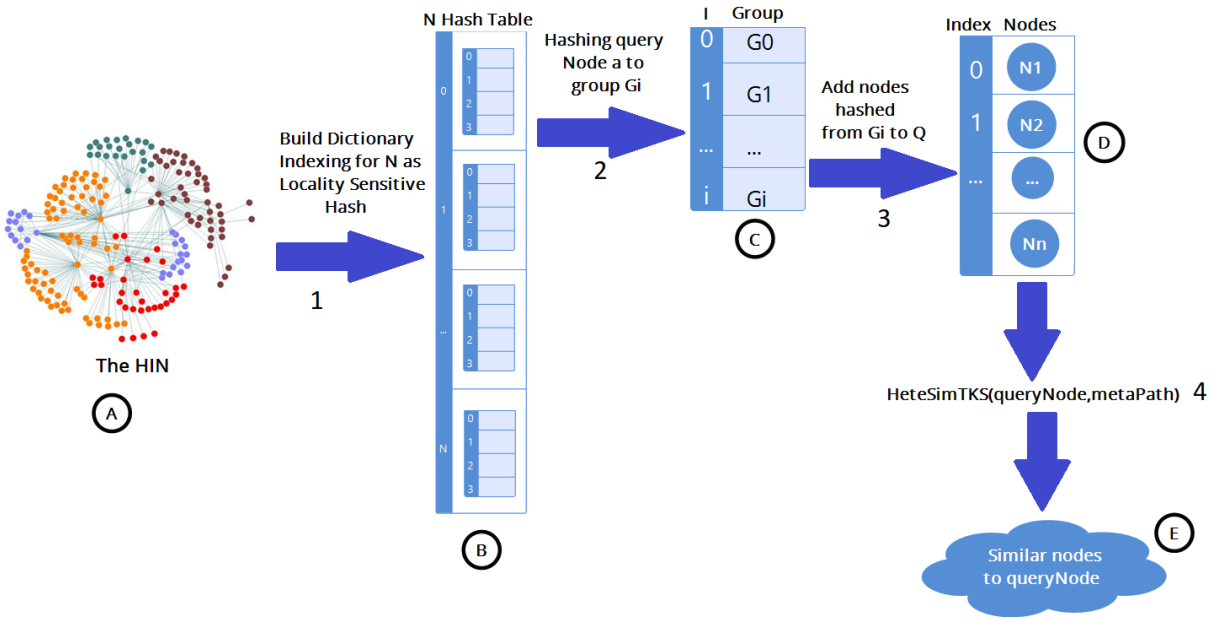


Source: Own authorship.

Figure 8 shows the second part of **Algorithm 2**, the *HeteSimTKSQuery*. By considering the Heterogeneous Information Network as input (step 1), a dictionary of indexing for N(The Network) was built as Locality Sensitive Hash. Figure 8.A the Hash Table is a structure for a given hash dictionary set N in B (Figure 8.B.) In Step 2, we hash the query node forming the Gi structure in 8.C. Nodes hashed from Gi are added to Q, Step 3, forming the candidate nodes in 8.D. Finally, the HeteSimTKS similarity function, which searches a similar document using the query node; using the meta-path and the TKS. The output is similar nodes to the query node.

To perform the three experiments, **Algorithm 2** searched for papers that were more similar to the electronic medical records using the keywords of the medical records and the articles as a ground truth, since it had passed by specialists (doctors, nurses etc) who had previously selected them before putting them on the site. The query used in all experiments was "cancer or tumor or carcinoma" since cancer is a broadly studied disease that retrieves plenty of material.

Figure 8 – HeteSimTKSQuery algorithm. In *A* first step is the HIN. Then Build a dictionary indexing for N as Locality Sensitive Hash forming a Hash Table. Than a hashing query node a to a group Gi. And then add the nodes hashed from Gi to Q in D. Then used a HeteSimTKS and the retrieved articles.



Source: Own authorship.

The goal of the experiments was to measure the *Accuracy*[4], *Precision*[5], and *Recall*[6] of the similarity measure and of the two proposed algorithms created by using the list of

[4] The weighted arithmetic mean of precision and inverse precision.
[5] The fraction of relevant instances among the retrieved instances.
[6] The fraction of relevant instances retrieved over the total amount of relevant instances.

keywords previously labeled by specialists (doctors, nurses etc).

Three experiments were conducted. The idea was to investigate to what extent the classical measures can influence HeteSimTKSQuery. The experiments were:

1. **Exp1**, exploits five classic similarity measures (Cosine, Euclidian, Minkowski, Hamming and Jaccard) to create a bag of words and compare them to the other experiments.

   a) **Input data:** Medical records and medical articles.

   b) **Variables:** Cosine, Euclidian, Minkowski, Hamming and Jaccard similarities.

   c) **Metrics Involved:** Accuracy, Precision, and Recall.

   d) **Obtaining Results:** In order of best precision, the results were as follows: Cosine, Euclidian, Minkowiski, Hamming and Jaccard. In order of best accuracy: Cosine, Minkowski, Jaccard, Euclidean and Hamming. And in order of best recall: Cosine, Euclidian, Jaccard, Minkowski and Hamming.

   e) **Result Analysis:** The Cosine similarity measure showed the best similarity measure for the vectorial models of information retrieval using similarity algorithms.

2. **Exp2** is the hybrid approach that mixes the classical algorithms with the HeteSimTKSQuery.

   a) **Input data:** Heterogeneous Information Network.

   b) **Variables:** HeteSimTKSQuery algorithm, and the Cosine, Euclidian, Minkowski similarity measures.

   c) **Metrics Involved:** Accuracy, Precision, and Recall.

   d) **Obtaining Results:** Considering all the intersections between the Cosine, Euclidean and Minkowski similarities measures and the the HeteSimTKSQuery path length (10, 20, 50, 100 and 200), the best results were when the path lenght was 10 and was obtained with Minkowski, Euclidean and Cosine in this order.

   e) **Result Analysis:** The best performance was Minkowski obtained with HetSimTKS when the path instances were 10 because there were fewer paths to walk.

3. and **Exp3** uses the proposed HeteSimTKSQuery, which creates the network similarity method with Medical Terminologies using PyMedTermino.

   a) **Input data:** Network

b) **Variables:** The path length, which were 10, 20, 50, 100 and 200.

c) **Metrics Involved:** Accuracy, Precision, and Recall.

d) **Obtaining Results:** The results increased as the path instance decreased.

e) **Result Analysis:** The greater the path instances the worse the results. The best performance in HeteSimTKSQuery was obtained when there were 10 path instances because there were fewer paths to walk, implying less semantics to deal with.

These experiments were carried out to verify the accuracy of each experiment as well as if each version could be used alone. The next subsections present the Accuracy, Precision and Recall measures of Exp1, Exp2 and Exp3, as well as their runtime measures. The latter is the length of time a certain algorithm takes to run.

4.2 Accuracy, Precision and Recall Measures

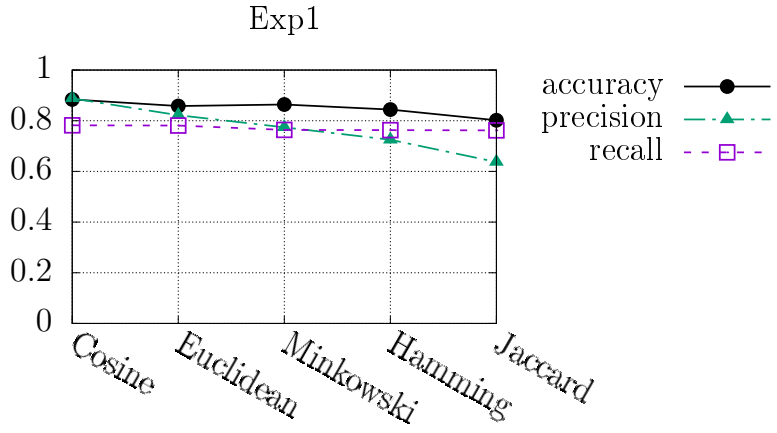
The **first experiment, Exp1**, applies classical similarity algorithms to the bag of words. In order to measure the accuracy, precision, and recall, a vectorial model of information retrieval was implemented using classical similarity algorithms. Using the query "cancer or tumor or carcinoma" to search paper, the accuracy, precision, and recall of the vectorial method (see Appendix C) are shown in Figure 9. The continuous line with balls represents accuracy. The dashed line with triangles represents precision, and the dashed line with squares represents recall. Axis x presents the classical similarity measures, and axis y presents the hit scale. The best results among all classical similarity measures are shown in Figure 9.

The Cosine measure has the accuracy with 0.88, the precision with 0.89 and the recall with 0.78. Cosine has the best precision and recall because it is a classical similarity measure for measuring distance when the magnitude of the vectors does not matter. The Euclidian has the accuracy with 0.86, the precision with 0.82, and recall with 0.78. The Euclidian similarity measure is the second best similarity approach. Where the precision is higher, the accuracy is higher than the recall. The Minkowski has the accuracy with 0.86, the precision with 0.77, and the recall with 0.76. The Hamming has the accuracy with 0.84, precision with 0.72, and recall with 0.76. Hamming has the worse accuracy and Hamming and almost the worse precision. The Jaccard has accuracy with 0.80, the precision with 0.64, and the recall 0.76. The Jaccard similarity has the highest accuracy and almost the same precision and recall. The results are similar because weights are associated with the same variables. Similarity measures require normalization to deal with varying magnitudes, scaling, distribution or measurement units. The choice of which one to use depends on both the task and the input data.

The best similarity recall is the Cosine, with 0.73 of recall followed by Euclidian, Jaccard with 0.63 and Minkowski and Hamming in this order. The best precision is the Cosine followed by the Euclidean, Minkowski, Hamming and Jaccard measures. The best accuracy is the Jaccard followed by the Minkowski, the Cosine, the Euclidian and then the Hamming measures, in this order. It is interesting to note that the accuracy line is between the precision and recall, which is higher than the recall.

The Cosine similarity measure had the best result among all classical similarity measures in Exp1. It reached the 0.88 for accuracy, which means, in general, the degree to which the result of a calculation, measurement, or specification conforms to a standard or the correct value. As for recall, it reached 0.78, which represents the fraction of relevant instances that were retrieved over the total amount of relevant instances of the similarity measure. Moreover, the Cosine with precision with 0.89 that returns the most similar medical records with the Pubmed articles. The cosine measure supports the use of term-weighting schemes for terms and the ranking of results.Table 5 shows the confusion matrix for Exp1. This matrix does not change very much from measure to measure; the only significant difference is in the nonrelevant documents retrieved. Table 6 shows the statistics measures for the Exp1. It shows that the Cosine is the best classical measure considering its low levels of False Negative Rate, False Discovery Rate, False Positive Rate, and its best performance among the other measures in the Recall, Specificity, Accuracy, Matthews Correlation Coefficient, Efficiency and Precision.

Figure 9 – Comparison of Accuracy, Precision, and Recall using the query "cancer or tumor or carcinoma" for vectorial models of information retrieval using classical similarity algorithms.



Source: Own authorship.

Table 5 – The confusion matrix of Exp1.

Measure	**Document Status**	**Relevant**	**Nonrelevant**	**Total**
Cosine	Retrieved	2851	616	3467
	Not Retrieved	800	5732	6532
	Total	3651	6348	9999
Euclidean	Retrieved	2851	616	3467
	Not Retrieved	800	5732	6532
	Total	3651	6348	9999
Minkowski	Retrieved	2251	656	2907
	Not Retrieved	700	6392	7092
	Total	2951	7048	9999
Hamming	Retrieved	2251	856	3107
	Not Retrieved	700	6192	6892
	Total	2951	7048	9999
Jaccard	Retrieved	2251	1280	3531
	Not Retrieved	700	5768	6468
	Total	2951	7048	9999

Source: Own authorship.

Table 6 – The statistical results of the experiment 1. The following statistical measures was used: Sensitivity, Specificity, Precision, Negative Predictive Value, False Positive Rate, False Discovery Rate, False Negative Rate, Accuracy, F1 Score and Matthews Correlation Coefficient.

	Cosine	**Euclidean**	**Minkowski**	**Hamming**	**Jaccard**
Recall	78.2%	78.1%	76.4%	76.3%	76.3%
Specificity	94.4%	90.3%	90.7%	87.9%	81.8%
Accuracy	88.4%	85.8%	86.4%	84.4%	80.2%
Matthews Correlation Coefficient	75%	69%	67%	63%	55%
Efficiency	86.2%	84.2%	83.5%	82.1%	79.1%
Precision	88.8%	82.2%	77.4%	72.5%	63.7%
Negative Predictive Value	88.2%	87.8%	90.1%	89.8%	89.2%
F1-Measure	83%	80.1%	76.8%	74.3%	69.4%
False Positive Rate	7%	9.7%	9.3%	12.1%	18.2%
False Discovery Rate	11%	18.8%	22.6%	27.5%	36.2%
False Negative Rate	22%	22%	24.7%	23.7%	23.7%

Source: Own authorship.

The **second experiment, Exp2**, used the classical similarity algorithms with the HeteSimTKSQuery. The classical similarity measures were merged HeteSimTKSQuery in order to bring together the speed of the classical algorithms and the semantics of the HeteSimTKSQuery. Figure 10a shows the accuracy for the Cosine, Euclidean and Minkowski by using the HeteSimTKSQuery with the query "cancer or tumor or carcinoma." This experiment uses the HeteSimTKSQuery with the classical similarity measures to find the similarity. Axis y shows the accuracy and axis x shows HeteSimTKSQuery with the classics measures with the length of the meta-path (the number of vertices in the path instance). The number of nodes was changed (10, 20, 50,100 and 200) to analyze the computational performance. The accuracy with the best outcome is the Euclidian Similarity Measure with HeteSimTKSQuery with path instance with ten nodes with 0.99.

The recall for the Cosine, Euclidean and Minkowski methods with the HeteSimTK-SQuery with the query is shown in Figure10b. Axis y shows the recall and axis x the Cosine, the Euclidean and the Minkowiski measures with the HeteSimTKSQuery. The length of the meta-path is 200, 100, 50, 20, and 10 nodes. The Euclidean measure with ten vertices was the best recall result 0.97. The numbers were chosen by empirical results since: the more the number of path instances increases, the more the recall decreases; and the more the number of path instances increases, the more the computational cost increases. So, the best recall was obtained by the Euclidean measure when the number of instance nodes was 10.

From all classical similarity measures presented in Appendix B, the best precision results were: the Cosine, Euclidean and Minkowski methods using the HeteSimTKSQuery with the query, a comparison between them is shown in Figure 10c.The best precision was with Euclidean distance and the HeteSimTKSQuery length with ten vertices, 0.99, followed by Minkowski, 0.98, and Cosine, 0.98. As the number of vertices of the path instance increases, the similarity value decreases.

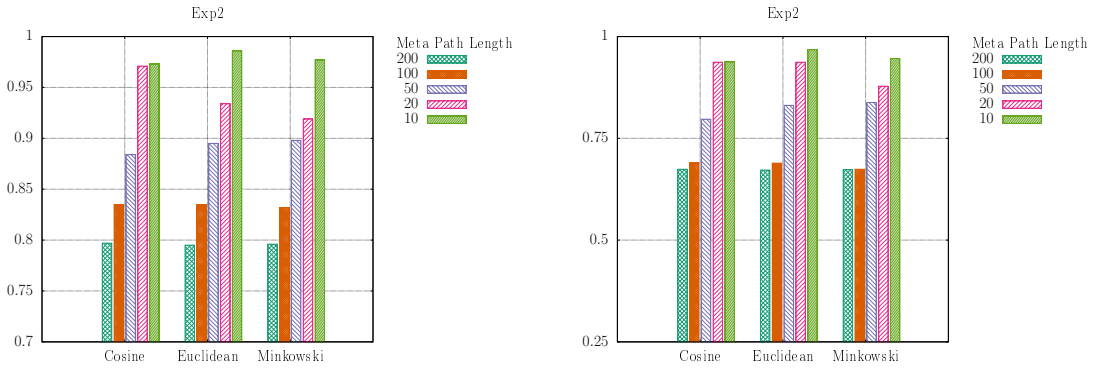
In Exp2, the Euclidian similarity measure blended with the HeteSimTKSQuery with a 10-node meta-path was the best similarity measure for Accuracy and Precision. It reached 0.93 for accuracy. From recall, the best result was achieved with the Minkowski measure, and the configuration of the HeteSimTKSQuery remained the same, 10 node meta-path. It obtained 0.79, which represent the fraction of relevant instances that were retrieved over the total amount of relevant instances of the similarity measure. Moreover, the precision with 0.99 with the Euclidian similarity measure blended with HeteSimTKSQuery with the length of the meta-path with ten nodes, which represents the fraction of relevant instances among the retrieved instances. This Exp2 was proposed to compare the improvement that the HeteSimTKSQuery with classical similarity algorithms has on the HeteSimTKSQuery alone. Table 7 presents the confusion matrix for the Exp2. The more the number of nodes on the path higher was the number of nonrelevant retrieved documents. The opposite

is true: the lower the number of nodes on the path higher was the number of relevant retrieved documents. Table 8 shows the statistics measures for Exp2. It shows that the Euclidean is the best classical measure since its low levels of False Negative Rate, False Discovery Rate,False Positive Rate, and its best performance among the other measures in the Recall, Specificity, Accuracy, Matthews Correlation Coefficient, Efficiency, Precision, Negative Predictive Value and F1-Measure.

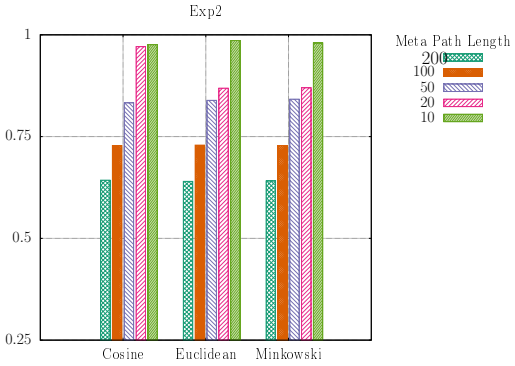
Figure 10 – Accuracy, Recall and Precision measures of the TKS algorithm with classic similarity methods using the following meta-path "Paper-Medical Record-paper" from the algorithm HeteSimTKSQuery.

(a) Accuracy comparison for the Cosine, Euclidean and Minkowiski methods using the HeteSimTKSQuery with the following query "cancer or tumor or carcinoma".

(b) Recall comparison for the Cosine, Euclidean and Minkowiski methods using the HeteSimTKSQuery with the following query "cancer or tumor or carcinoma".



(c) Precision comparison for the Cosine, Euclidean and Minkowiski methods using the HeteSimTKSQuery with the following query "cancer or tumor or carcinoma".



Source: Own authorship.

Table 7 – Confusion matrix of the Exp2.

Measure	Length of the Meta-path	Document Status	Relevant	Nonrelevant	Total
Cosine	10	Retrieved	3000	74	3074
		Not Retrieved	200	6725	6925
		Total	3200	6799	9999
	20	Retrieved	2998	90	3088
		Not Retrieved	202	6709	6911
		Total	3200	6799	9999
	50	Retrieved	2550	513	3063
		Not Retrieved	650	6286	6936
		Total	3200	6799	9999
	100	Retrieved	2000	751	2751
		Not Retrieved	900	6348	7248
		Total	2900	7099	9999
	200	Retrieved	1954	1084	3038
		Not Retrieved	946	6015	6961
		Total	2900	7099	9999
Euclidean	10	Retrieved	3070	44	3114
		Not Retrieved	100	6785	6885
		Total	3170	6829	9999
	20	Retrieved	2998	457	3455
		Not Retrieved	202	6342	6544
		Total	3200	6799	9999
	50	Retrieved	2660	510	3170
		Not Retrieved	540	6289	6829
		Total	3200	6799	9999
	100	Retrieved	1998	744	2742
		Not Retrieved	902	6355	7257
		Total	2900	7099	9999
	200	Retrieved	1950	1097	3047
		Not Retrieved	950	6002	6952
		Total	2900	7099	9999
Minkowski	10	Retrieved	3000	60	3060
		Not Retrieved	170	6769	6939
		Total	3170	6829	9999
	20	Retrieved	2808	413	3221
		Not Retrieved	392	6386	6778
		Total	3200	6799	9999
	50	Retrieved	2680	504	3184
		Not Retrieved	520	6295	6815
		Total	3200	6799	9999
	100	Retrieved	1954	732	2686
		Not Retrieved	946	6367	7313
		Total	2900	7099	9999
	200	Retrieved	1949	1093	3042
		Not Retrieved	949	6008	6957
		Total	2898	7101	9999

Source: Own authorship.

Table 8 – The statistical results of the experiment 2. The following statistical measures was used: Sensitivity, Specificity, Precision, Negative Predictive Value, False Positive Rate, False Discovery Rate, False Negative Rate, Accuracy, F1 Score and Matthews Correlation Coefficient.

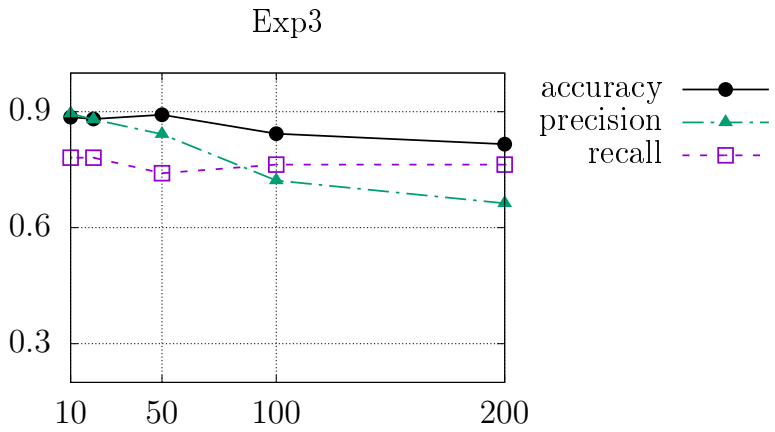
Classic Distance	Statistical Measure	Length of the Meta-path				
		10	20	50	100	200
Cosseno	Recall	93.8%	93.7%	79.7%	69.0%	67.4%
	Specificity	98.9%	98.7%	92.5%	89.4%	84.7%
	Accuracy	97.3%	97.1%	88.4%	83.5%	79.70%
	Matthews Correlation Coefficient	94%	93%	73%	59%	51%
	Efficiency	96.3%	96.2%	86.1%	79.2%	76.1%
	Precision	97.6%	97.1%	83.3%	72.8%	64.3%
	Negative Predictive Value	97.1%	97.1%	90.6%	87.6%	86.4%
	F1-Measure	95.6%	95.3%	81.4%	70.7%	65.8%
	False Positive Rate	1%	1.3%	7.5%	10.5%	15.3%
	False Discovery Rate	2.4%	2.9%	16.7%	27.3%	35.7%
	False Negative Rate	6.2%	6.3%	20.3%	31.0%	32.6%
Euclidean	Recall	96.80%	93.7%	83.1%	68.9%	67.2%
	Specificity	99.4%	93.3%	92.5%	89.5%	84.5%
	Accuracy	98.6%	93.4%	89.5%	83.5%	79.5%
	Matthews Correlation Coefficient	97%	85%	76%	59%	51%
	Efficiency	98.1%	93.5%	87.8%	79.2%	75.9%
	Precision	98.6%	86.9%	83.9%	72.9%	64%
	Negative Predictive Value	98.5%	96.9%	92.1%	87.6%	86.3%
	F1-Measure	97.7%	90.1%	83.5%	70.8%	65.6%
	False Positive Rate	0.6%	6.7%	7.5%	10.5%	15.4%
	False Discovery Rate	1.4%	13.2%	16%	27.1%	36%
	False Negative Rate	3.1%	6.3%	16.9%	31.1%	32.7%
Minkowski	Recall	94.6%	87.8%	83.8%	67.4%	67.3%
	Specificity	99.1%	93.9%	92.6%	89.7%	84.6%
	Accuracy	97.7%	91.9%	89.8%	83.2%	79.6%
	Matthews Correlation Coefficient	95%	82%	76%	58%	51%
	Efficiency	96.9%	90.8%	88.2%	78.5%	75.9%
	Precision	98%	87%	84.2%	72.8%	64.1%
	Negative Predictive Value	97.6%	94.2%	92.4%	87.1%	86.4%
	F1-Measure	96.3%	87.4%	84%	69.9%	65.6%
	False Positive Rate	0.9%	6%	0.7%	10.3%	15.4%
	False Discovery Rate	2%	12.8%	15.8%	27.2%	35.9%
	False Negative Rate	5.4%	12.2%	16.2%	32.6%	32.7%

Source: Own authorship.

The **third experiment, Exp3,** used only the HeteSimTKSQuery algorithm. Exp3 compares the accuracy, precision and recall for query "cancer or tumor or carcinoma" by using the following meta-path "Article-Medical-Record-Article"(meta path that means articles that are similar to medical records)to compare the influence of meta-path without classical similarity measures. The result of this experiment is shown in Figure 11.

The best accuracy, precision, and recall values (0.89, 0.89, and 0.78, respectively) were obtained when there were ten vertices in the path instance, followed by 20, 50, 100, and 200 nodes. HeteSimTKSQuery returned the most similar medical records with PubMed articles. It is observed that as the length of the path increases, precision worsens because the longer the path, the more semantic information there is in the path. Therefore, HeteSimTKSQuery returned the most similar medical records with PubMed articles. Table 9 shows the confusion matrix for the Exp3. The higher the number of nodes in the path, higher was the number of nonrelevant retrieved documents, and the relevant not retrieved decreses. Moreover, the lower was the number of nodes on the path higher was the number of relevant retrieved documents. Table 10 shows the statistics measures for the Exp3. It shows that the number of nodes in the paths isthe best classical measure since its low levels of False Negative Rate, False Discovery Rate, and its best performance among the other measures in the Recall, Matthews Correlation Coefficient, Efficiency, Precision, and F1-Measure.

Figure 11 – Comparison of accuracy, precision and recall for query "cancer or tumor or carcinoma" using the following meta-path "Paper-Medical Record-Paper" from the algorithm HeteSimTKSQuery.



Source: Own authorship.

Table 9 – Confusion matrix of the Exp3.

Length of the Meta-path	Document Status	Relevant	Nonrelevant	Total
10	Retrieved	2851	337	3188
	Not Retrieved	800	6011	6811
	Total	3651	6348	9999
20	Retrieved	2851	388	3239
	Not Retrieved	800	5960	6760
	Total	3651	6348	9999
50	Retrieved	2000	379	2379
	Not Retrieved	700	6920	7620
	Total	2700	7299	9999
100	Retrieved	2250	868	3118
	Not Retrieved	700	6181	6881
	Total	2950	7049	9999
200	Retrieved	2251	1144	3395
	Not Retrieved	700	5904	6604
	Total	2951	7048	9999

Source: Own authorship.

Table 10 – The statistical results of the experiment 3. The following statistical measures was used: Sensitivity, Specificity, Precision, Negative Predictive Value, False Positive Rate, False Discovery Rate, False Negative Rate, Accuracy, F1 Score and Matthews Correlation Coefficient.

Statistical Measure	10	20	50	100	200
Recall	78.1%	78.1%	74.1%	76.3%	76.3%
Specificity	94.7%	93.9%	94.8%	87.7%	83.8%
Accuracy	88.6%	88.1%	89.2%	84.3%	81.6%
Matthews Correlation Coefficient	75%	74%	0.72	63%	58%
Efficiency	86.4%	86.0%	84.4%	82.0%	80.0%
Precision	89.5%	88%	84.2%	72.2%	66.3%
Negative Predictive Value	88.3%	88.2%	90.8%	89.8%	89.4%
F1-Measure	83.4%	82.8%	78.8%	74.1%	70.9%
False Positive Rate	5.3%	6.1%	5.2%	12.3%	16.2%
False Discovery Rate	10.6%	12%	15.9%	27.8%	33.7%
False Negative Rate	21.9%	22%	25.9%	23.7%	23.7%

Source: Own authorship.

In **Exp1** the classical similarity measure that showed the best performance was the Cosine measure because it had more valuable accuracy, precision and recall. The Exp2, which blends the HeteSimTKSQuery with the classical similarity algorithms with the best performance, was the Minkowski measure. The Exp3 used the HeteSimTKSQuery with path instances 200, 100, 50, 20, and 10 sizes. The best result of the Exp3 was obtained when the path instances was 10.

Table 11 – Comparative Analysis of the Experiments: Exp1, Exp2 and Exp3 showing its objectives the algorithms and the best results

	Exp1	**Exp2**	**Exp3**
Objective	Exploits five classic similarity measures (Cosine, Euclidian, Minkowski, Hamming and Jaccard) to create a bag of words and compare to the other experiments.	The hybrid approach, mixing the classical algorithms with the HeteSimTKSQuery. Search the similarity measure by an HIN.	Search the similarity measure by an HIN.
Algorithms	Cosine, Euclidian, Minkowski, Hamming and Jaccard similarities	HeteSimTKSQuery algorithm, and the Cosine, Euclidian, Minkowski similarity measures	HeteSimTKSQuery
Best Results	Cosine	HeteSimTKSQuery with 10 path instances with Euclidian	HeteSimTKSQuery with 10 path instances

Source: Own authorship.

4.2 Runtime Measures

Tables 12 and 13 present the runtime[7] for the searching query “cancer or tumor or carcinoma”on the corpus by using twenty-one similarities presented in Appendix B and the constructed network by using the HeteSimTKSQuery algorithm. The runtime measure is important because similarity algorithms can be used in many embedded systems that require hard or soft real-time execution under rigid timing constraints. The machine configuration is on the Appendix A.

[7] Runtime is the length of time taken for a program to run. It begins when a program is executed and ends when the program is closed or quit.

Table 12 – Comparison of the Classical Similarity Algorithms runtime in milliseconds and its precision with HeteSimTKSQuery and The Hybrid HeteSimTKSQuery and its precision.

	Classical Similarity Measures Runtime (ms)	**Precision**	**HeteSimTKS Runtime (ms)**	**Precision**
Bray-Curtis	333.78	0.19	1.67E-05	0.87
Canberra	691.41	0.44	3.46E-05	0.48
Chebyshev	205.03	0.29	1.03E-05	0.86
City-Block	166.89	0.63	8.34E-06	0.96
Correlation	1177.78	0.08	5.89E-05	0.86
Cosine	710.48	0.88	3.55E-05	0.97
Dice	929.83	0.46	4.65E-05	0.60
Euclidean	314.71	0.82	1.57E-05	0.98
Hamming	324.24	0.72	1.62E-05	0.95
Jaccard	314.71	0.63	1.57E-05	0.87
Kulsinski	658.03	0.36	3.29E-05	0.69
Mahalanobis	925.06	0.39	4.63E-05	0.62
Minkowski	267.02	0.77	1.34E-05	0.98
Rogers-Tanimoto	681.87	0.57	3.41E-05	0.66
Russell-Rao	243.18	0.37	1.22E-05	0.55
Seuclidean	977.51	0.12	4.89E-05	0.31
Sokal-Michener	734.32	0.51	3.67E-05	0.29
Sokal-Sneath	710.48	0.28	3.55E-05	0.03
Sq-Euclidean	319.48	0.58	1.60E-05	0.75
Wminkowski	1072.88	0.50	5.36E-05	0.60
Yule	753.40	0.46	3.77E-05	0.46

Source: Own authorship.

In Table 12, the smallest runtime using only the classical similarity measures is the City-Block similarity measure with 166.89ms followed by Chebyshev, Russell-Rao, Minkowski, and the Euclidean in this order. The highest precision of the classical similarity measures was obtained with the Cosine , which reached 0.88, followed by the Euclidean, Minkowski, Hamming and Jaccard measures.

On the other hand, the highest runtime of the classical similarity measures was the Correlation with 1177.79ms followed by Wminkowski, Seuclidean, Dice, and the Mahalanobis measures. The lowest precision was the Correlation 0,08 followed by Seuclidean, Bray-Curtis, Sokal-Sneath and Chebyshev.

Table 12 shows the runtime of the hybrid approach HeteSimTKSQuery running with the classical similarity algorithms. The shortest runtime of the HeteSimTKSQuery with classical similarity algorithms was obtained by HeteSimTKSQuery with the City-Block similarity measure with 8.34E-06ms followed by Chebyshev, Russell-Rao, Minkowski and

Jaccard respectively. Moreover, the highest precision was achieved by HeteSimTKSQuery running with the Euclidean similarity with 0.99 followed by Minkowski, Cosine, City-Block, and Hamming respectively.

Differently, the highest runtime of the HeteSimTKSQuery with these classical similarity algorithms was obtained by the HeteSimTKSQuery with Correlation 5.89E-05ms and then by Wminkowski, Seuclidean, Dice, and Mahalanobis respectively. The smallest precision was obtained by the Sokal-Sneath with 0.03 and then by the Sokal-Michener, Seuclidean, Yule and Canberra respectively.

The best results in the second experiment was obtained by the hybrid method that blends the HeteSimTKSQuery with the classical similarity measures. Amont all runtimes of HeteSimTKSQuery with the classic similarities algorithms, the smallest runtime was obtained with the City-Block similarity measure of 8.34E-06ms and the **highest runtime** with HeteSimTKSQuery running with the Correlation similarity of 5.89E-05ms. The smallest precision of the HeteSimTKSQuery with the classic similarity algorithms was obtained by the Sokal-Sneath with 0.03 and the **highest** was obtained by the Euclidean with 0.99.

The **smallest runtime** using only the classical similarity measures was achieved by the City-Block similarity measure with 166.89ms and the **highest runtime** with HeteSimTKSQuery running with the Correlation similarity with 1177.78ms. The **smallest precision** of the classic similarity measures was the Correlation similarity measure with 0.09 and the **highest** was the cosine with 0.88.

Table 13 shows the third approach, the network approach with only TKS. The more vertices there are in the path instance the longer the retrieval time. The smallest the path of path instance the better the result. Consequently, the best runtime was when the path instance was 10 with 3083.54ms and the highest precision was also when the path instance was 10 with 0.89. The lowest results was when the path instance was 200 with 3925.95ms and the lowest precision was also when the path instance was 200 with 0.66.

Table 13 – Comparison of the HeteSimTKSQuery without the Classical Similarity Measures for the path length from sizes 10, 20, 50,100 and 200.

	200	***100***	***50***	***20***	***10***
HeteSimTKS	**3925.96**	3576.28	3925.96	3099.44	**3083.55**
Precision	**0,66**	0.72	0.84	0.88	**0.89**

Source: Own authorship.

4.3 Results Discussion

Exp1 the best result was the Cosine, and the best precision was obtained with 0.88 and runtime 710.49ms. The differences were pretty much the same between the classical similarity algorithms. That is explained by the fact that some classical similarity algorithms in Table 12 are better for some types of data and others are better for other types.

As a result, for **Exp2** and **Exp3** there is a trade off among the runtime, the implementing complexity of the algorithm, and the complexity of the algorithms: The faster the algorithm the more difficult it is to implement. And the lower complexity of the algorithm the best the runtime.

From, **Exp2**, the classical similarity measures with the HeteSimTKSQuery, has a tradeoff between runtime and precision. **Exp2** and **Exp3** it was when the path length was 10, and they increase as the path lenght increases.

For **Exp3**, which only had the HeteSimTKSQuery, the best precision was 0.89 and the runtime was 3083.55ms.

Exp2, the hybrid approach, the HeteSimTKSQuery blended with Euclidean had a precision of 0.99 and 1.57E-05ms; and the second best was the one blended with the Cosine with precision 0.88. The HeteSimTKSQuery had the precision of 0.89 and runtime 3925.96ms. It is possible to conclude that the hybrid approach is the best one as it has the best tradeoff between runtime and the algorithm complexity.

4.4 Remarks

The empirical work involved the development of three experiments (**Exp1**, **Exp2**, and **Exp3**) in the sense of objectives. The first exploited five classic similarity measures: Cosine, Euclidian, Minkowski, Hamming and Jaccard to create a bag of words to compare to the other experiments. The second was the hybrid approach, mixing the classical algorithms with the HeteSimTKSQuery. And the third created the network similarity method with Medical Terminologies using PyMedTermino.

The terminology products in the network improved the quality of the similarity but increased the runtime. Among the classical similarity measures, the hybrid approach, i.e. the classical similarity measures with the HeteSimTKSQuery, and only the HeteSimTK-SQuery, the best runtime from the of approach was 5.89E-05ms. The hybrid approach with the Euclidean had the best precision with 0.98. It can be concluded that the hybrid method is the best one, as it has the best tradeoff between runtime and the algorithm complexity.

The results were better than expected, surpassing expectations on everything in Exp2 which is the hybrid models with the HeteSimTKSQuery and measures of classic similarity.

Precision measure concerns the "quality" of the response, it must be something close to the real solution. Runtime concerns the time to compute the response, it must be feasible, that is to say that in obtaining the answer, it is still useful. This project is more focused in the quality of answers since (i) the runtime can be softened by parallel computing, increasing processing capacity of the computer, and (ii) the documents can be used in a selective indexing, that can be retrieved in a general or specific way.

5

Conclusion

This work created two novel algorithms: a Heterogeneous Information Network Creator Algorithm and a Similarity one. The latter deals with the textual HIN structured by using semantics based on meta-paths and terminology products.

To achieve the main goal proposed in this work, a study of the theory of Complex Networks and Similarity Measures to retrieve the main Similarity Measures were carried out. A crawler was created to collect five thousand articles from PubMed. The articles were used in the network and another crawler was created to collect medical records from the MTsamples, a public database that contains medical reports from the same filter found in PubMed. It includes 4999 medical records that were downloaded and organized. The crawlers were essential on the acquisition of data for the construction of the Heterogeneous Information Network. Every node in this network had attributes. To connect the nodes (each node represents an article and its structure), similarity measures were used, including the Euclidean distance and its variants; Minkowski; Cityblock; Cosine; Correlation; Hamming, and other measures. Further, a computational module was created to preprocess the articles extracting some attributes. A set of techniques was performed as an attempt to clean articles content, including stopwords of elimination, lemmatization, and stemming. The idea was to make the manipulation of the data in the network easier. Moreover, it a computational module was created to process medical records extracted from MTsamples. After the extraction of medical records, the same techniques were performed for articles pre-processing. In both pre-processing steps, the PyMedTermino tool was used.

The Information Network that contains the highest similarity values among the article nodes was chosen. This Heterogeneous Information Network created with scientific papers is connected with another Information Network created that contains public medical records. The structure of a medical record varies according to the medical specialty and clinical case, and it has different attributes. To connect the nodes (each node represents a medical record and its structure), the same similarity measures were used. The Information

Network containing the highest similarity values among the medical record nodes was chosen. Finally a Heterogeneous Information Network merging the two aforementioned networks was created. By using the pyMedTermino tool, a pre-processing step was performed to enrich the joined Information Network with terminology products included on PyMedTermino. Terms such as keywords connect the article nodes and the medical record nodes, and linguistic terms and synonymous terms after that the same similarity measures were calculated. Further, a network schema was created, using a random walk algorithm packed into the Igraph tool. The random walk was used to extract the meta-paths in the graph. After the creation of the schema of the Heterogeneous Information Network, the meta-paths were extracted using the Igraph tool. The meta-paths are the way one uses use to walk on the network.Finally, the similarity measure according to the meta-paths was created by using terminology products to validate each meta-path. For this, the hierarchy and the semantic structures found in these well-established health terminologies were used.

The Heterogeneous Information Network Created only allows two types of nodes. More types of node could not be allowed. The time cost to create the network and to update it is high. The similarity measures also have very high time costs.

The problem of the Heterogenous Information Networks construction and the search for similarity of related but different kinds of data were treated here as shown in Section 4, that two algorithms developed deal with the mentioned situations. The proposed NetworkCreator Algorithm creates a Heterogeneous Information Network only from raw textual data by using meta-path and terminology products, and the similarity algorithm HeteSimTKSQuery together with the classical measures produces a better result than other networks similarity measures.

A comparative analysis of all similarity measures is done with support parameters like precision, recall, and accuracy. It can be concluded from this work that HeteSimTKS-Query similarity measure merged with classic similarity measures are better to find the similar pairs of different objects but complementary to one another (medical records and papers) and objects that of the same type (medical records with medical records or papers with papers). The proposed algorithms showed to be efficient and accurate using the Euclidean distance.

Moreover, a paper called “A Review of Text-Based and Knowledge-Based Semantic Similarity Measures” was published in the Proceedings of the 5th Symposium on Knowledge discovery, mining and learning (RIBEIRO; ZHAO; MACEDO, 2017).

For future works, we intend to propose a faster Heterogeneous Information Network creator by identifying the granularity of the nodes. This work does not consider the dynamical granularity of the nodes. It is also necessary to reduce the runtime of the HeteSimTKSQuery and Network Creator. The resulting network can be used as the basis

of machine learning. Finally, another future work is to parallelize the algorithms in a high performance cluster and to verify robustness of the algorithms created by considering different machine configurations and databases.

Bibliography

AGIRRE, E.; RIGAU, G. A proposal for word sense disambiguation using conceptual distance. *AMSTERDAM STUDIES IN THE THEORY AND HISTORY OF LINGUISTIC SCIENCE SERIES 4*, JOHN BENJAMINS BV, p. 161–172, 1997.

AGIRRE, E.; SOROA, A. Personalizing pagerank for word sense disambiguation. In: ASSOCIATION FOR COMPUTATIONAL LINGUISTICS. *Proceedings of the 12th Conference of the European Chapter of the Association for Computational Linguistics*. [S.l.], 2009. p. 33–41.

ARONSON, A. R. Effective mapping of biomedical text to the umls metathesaurus: the metamap program. In: AMERICAN MEDICAL INFORMATICS ASSOCIATION. *Proceedings of the AMIA Symposium*. [S.l.], 2001. p. 17.

BALMIN, A.; HRISTIDIS, V.; PAPAKONSTANTINOU, Y. Objectrank: Authority-based keyword search in databases. In: VLDB ENDOWMENT. *Proceedings of the Thirtieth international conference on Very large data bases-Volume 30*. [S.l.], 2004. p. 564–575.

BANERJEE, S.; PEDERSEN, T. An adapted lesk algorithm for word sense disambiguation using wordnet. In: SPRINGER. *International Conference on Intelligent Text Processing and Computational Linguistics*. [S.l.], 2002. p. 136–145.

BARABÁSI, A.-L. et al. *Network science*. [S.l.]: Cambridge university press, 2016.

BODENREIDER, O. The unified medical language system (umls): integrating biomedical terminology. *Nucleic acids research*, Oxford Univ Press, v. 32, n. suppl 1, p. D267–D270, 2004.

BOLLEGALA, D. T.; MATSUO, Y.; ISHIZUKA, M. Measuring the similarity between implicit semantic relations from the web. In: ACM. *Proceedings of the 18th international conference on World wide web*. [S.l.], 2009. p. 651–660.

BRIN, S. Extracting patterns and relations from the world wide web. In: SPRINGER. *International Workshop on The World Wide Web and Databases*. [S.l.], 1998. p. 172–183.

BRIN, S.; PAGE, L. Proceedings of the seventh international world wide web conference the anatomy of a large-scale hypertextual web search engine. *Computer Networks and ISDN Systems*, v. 30, n. 1, p. 107 – 117, 1998. ISSN 0169-7552. Disponível em: <http://www.sciencedirect.com/science/article/pii/S016975529800110X>.

BROWN, E. G.; WOOD, L.; WOOD, S. The medical dictionary for regulatory activities (meddra). *Drug safety*, Springer, v. 20, n. 2, p. 109–117, 1999.

BU, S. et al. Integrating meta-path selection with user-preference for top-k relevant search in heterogeneous information networks. In: IEEE. *Computer Supported Cooperative Work in Design (CSCWD), Proceedings of the 2014 IEEE 18th International Conference on*. [S.l.], 2014. p. 301–306.

BURGESS, C.; LUND, K. Multiple constraints in syntactic ambiguity resolution: A connectionist account of psycholinguistic data. *COGSCI-94, Atlanta, GA*, 1994.

CHAKRABARTI, S.; AGARWAL, A. Learning parameters in entity relationship graphs from ranking preferences. In: SPRINGER. *European Conference on Principles of Data Mining and Knowledge Discovery*. [S.l.], 2006. p. 91–102.

CHAKRABARTI, S. et al. *Mining the Web: Analysis of hypertext and semi structured data*. [S.l.]: Morgan Kaufmann San Francisco, 2002.

CHANG, S. et al. Heterogeneous network embedding via deep architectures. In: ACM. *Proceedings of the 21th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*. [S.l.], 2015. p. 119–128.

CHARIKAR, M. S. Similarity estimation techniques from rounding algorithms. In: ACM. *Proceedings of the thiry-fourth annual ACM symposium on Theory of computing*. [S.l.], 2002. p. 380–388.

CHEN, D. et al. Knowledge discovery from posts in online health communities using unified medical language system. *International journal of environmental research and public health*, Multidisciplinary Digital Publishing Institute, v. 15, n. 6, p. 1291, 2018.

CHO, H.; BERGER, B.; PENG, J. Diffusion component analysis: unraveling functional topology in biological networks. In: SPRINGER. *International Conference on Research in Computational Molecular Biology*. [S.l.], 2015. p. 62–64.

CHOUDHARI, M. *Extending the hirst and st-onge measure of semantic relatedness for the unified medical language system*. Tese (Doutorado) — University of Minnesota, 2012.

CILIBRASI, R. L.; VITANYI, P. M. The google similarity distance. *IEEE Transactions on knowledge and data engineering*, IEEE, v. 19, n. 3, p. 370–383, 2007.

CODES, D. D. I.; LIST, A. C. S. I. Icd-10 code. *Alcohol*, v. 10, p. 239.

COSTA, R.; LIMA, C. Document clustering using an ontology-based vector space model. In: *Information Retrieval and Management: Concepts, Methodologies, Tools, and Applications*. [S.l.]: IGI Global, 2018. p. 1860–1883.

DATAR, M. et al. Locality-sensitive hashing scheme based on p-stable distributions. In: ACM. *Proceedings of the twentieth annual symposium on Computational geometry*. [S.l.], 2004. p. 253–262.

DEZA, M. M.; DEZA, E. Encyclopedia of distances. In: *Encyclopedia of Distances*. [S.l.]: Springer, 2009. p. 1–583.

DICE, L. R. Measures of the amount of ecologic association between species. *Ecology*, Wiley Online Library, v. 26, n. 3, p. 297–302, 1945.

DILIGENTI, M.; GORI, M.; MAGGINI, M. Learning web page scores by error back-propagation. In: *IJCAI*. [S.l.: s.n.], 2005. p. 684–689.

DONNELLY, K. Snomed-ct: The advanced terminology and coding system for ehealth. *Studies in health technology and informatics*, IOS Press; 1999, v. 121, p. 279, 2006.

EGOZI, O.; MARKOVITCH, S.; GABRILOVICH, E. Concept-based information retrieval using explicit semantic analysis. *ACM Transactions on Information Systems (TOIS)*, ACM, v. 29, n. 2, p. 8, 2011.

EHSANI, R.; DRABLØS, F. Topoicsim: a new semantic similarity measure based on gene ontology. *BMC bioinformatics*, BioMed Central, v. 17, n. 1, p. 296, 2016.

FELDMAN, R. Link analysis: Current state of the art. *Tutorial at the KDD*, v. 2, 2002.

FIRTH, J. R. A synopsis of linguistic theory, 1930-1955. Blackwell, 1957.

FRAKES, W. B.; BAEZA-YATES, R. *Information retrieval: Data structures & algorithms*. [S.l.]: prentice Hall Englewood Cliffs, NJ, 1992. v. 331.

FRIEDMAN, B. G. *Web search savvy: Strategies and shortcuts for online research*. [S.l.]: Psychology Press, 2004.

GABRILOVICH, E.; MARKOVITCH, S. Computing semantic relatedness using wikipedia-based explicit semantic analysis. In: *IJcAI*. [S.l.: s.n.], 2007. v. 7, p. 1606–1611.

GANGEMI, A.; NAVIGLI, R.; VELARDI, P. The ontowordnet project: extension and axiomatization of conceptual relations in wordnet. In: SPRINGER. *OTM Confederated International Conferences" On the Move to Meaningful Internet Systems"*. [S.l.], 2003. p. 820–838.

GETOOR, L.; DIEHL, C. P. Link mining: a survey. *ACM SIGKDD Explorations Newsletter*, ACM, v. 7, n. 2, p. 3–12, 2005.

GOMAA, W. H.; FAHMY, A. A. A survey of text similarity approaches. *International Journal of Computer Applications*, Citeseer, v. 68, n. 13, p. 13–18, 2013.

GREUB, W. H. *Linear Algebra: 3d Ed*. [S.l.]: Springer, 1967.

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

GUU, K.; MILLER, J.; LIANG, P. Traversing knowledge graphs in vector space. *arXiv preprint arXiv:1506.01094*, 2015.

HAN, J. Mining heterogeneous information networks by exploring the power of links. In: SPRINGER. *Discovery Science*. [S.l.], 2009. p. 13–30.

HELLMAN, J. et al. The impact of international classification of diseases, 10th revision (icd-10) after the centers for medicare and medicaid services (cms) grace period. *Investigative Ophthalmology & Visual Science*, The Association for Research in Vision and Ophthalmology, v. 59, n. 9, p. 4153–4153, 2018.

HERMJAKOB, U. et al. Incident-driven machine translation and name tagging for low-resource languages. *Machine Translation*, Springer, v. 32, n. 1-2, p. 59–89, 2018.

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.

HOLDER, L. B.; COOK, D. J. Graph-based data mining. In: *Encyclopedia of data warehousing and mining*. [S.l.]: IGI Global, 2005. p. 540–545.

ISLAM, A.; INKPEN, D. Second order co-occurrence pmi for determining the semantic similarity of words. In: *Proc.of the International Conference on Language Resources and Evaluation, Genoa, Italy*. [S.l.: s.n.], 2006. p. 1033–1038.

ISLAM, A.; INKPEN, D. Semantic text similarity using corpus-based word similarity and string similarity. *ACM Transactions on Knowledge Discovery from Data (TKDD)*, ACM, v. 2, n. 2, p. 10, 2008.

JACCARD, P. *Etude comparative de la distribution florale dans une portion des Alpes et du Jura*. [S.l.]: Impr. Corbaz, 1901.

JEGOU, H.; DOUZE, M.; SCHMID, C. Product quantization for nearest neighbor search. *IEEE transactions on pattern analysis and machine intelligence*, IEEE, v. 33, n. 1, p. 117–128, 2011.

JEH, G.; WIDOM, J. Simrank: a measure of structural-context similarity. In: ACM. *Proceedings of the eighth ACM SIGKDD international conference on Knowledge discovery and data mining*. [S.l.], 2002. p. 538–543.

JEH, G.; WIDOM, J. Scaling personalized web search. In: ACM. *Proceedings of the 12th international conference on World Wide Web*. [S.l.], 2003. p. 271–279.

JENSEN, D.; GOLDBERG, H. Aaai fall symposium on ai and link analysis. *AAAI, Menlo Park, CA*, v. 3, 1998.

JIANG, J. J.; CONRATH, D. W. Semantic similarity based on corpus statistics and lexical taxonomy. *arXiv preprint cmp-lg/9709008*, 1997.

JIN, R.; LEE, V. E.; HONG, H. Axiomatic ranking of network role similarity. In: ACM. *Proceedings of the 17th ACM SIGKDD international conference on Knowledge discovery and data mining*. [S.l.], 2011. p. 922–930.

JONES, E.; OLIPHANT, T.; PETERSON, P. {SciPy}: open source scientific tools for {Python}. 2014.

KG, S.; SADASIVAM, G. S. Modified heuristic similarity measure for personalization using collaborative filtering technique. *Appl. Math*, v. 11, n. 1, p. 307–315, 2017.

KOLB, P. Experiments on the difference between semantic similarity and relatedness. In: *Proceedings of the 17th Nordic Conference on Computational Linguistics-NODALIDAâ^{TM}09*. [S.l.: s.n.], 2009.

KRAUSE, E. F. *Taxicab geometry: An adventure in non-Euclidean geometry*. [S.l.]: Courier Corporation, 2012.

KRUMMENACHER, R.; STRANG, T. Ontology-based context modeling. In: *Proceedings Third Workshop on Context-Aware Proactive Systems (CAPS 2007)(June 2007)*. [S.l.: s.n.], 2007. p. 22.

LAMY, J.-B.; VENOT, A.; DUCLOS, C. Pymedtermino: an open-source generic api for advanced terminology services. In: *MIE*. [S.l.: s.n.], 2015. p. 924–928.

LANDAUER, T. K.; DUMAIS, S. T. A solution to plato's problem: The latent semantic analysis theory of acquisition, induction, and representation of knowledge. *Psychological review*, American Psychological Association, v. 104, n. 2, p. 211, 1997.

LAO, N.; COHEN, W. W. Relational retrieval using a combination of path-constrained random walks. *Machine learning*, Springer, v. 81, n. 1, p. 53–67, 2010.

LEACOCK, C.; CHODOROW, M. *Combining local context and WordNet sense similarity for word sense identification. WordNet, An Electronic Lexical Database*. [S.l.]: The MIT Press Cambridge, 1998.

LEWIS, T. G. *Network science: Theory and applications*. [S.l.]: John Wiley & Sons, 2011.

LI, C. et al. An efficient drug-target interaction mining algorithm in heterogeneous biological networks. In: SPRINGER. *Pacific-Asia Conference on Knowledge Discovery and Data Mining*. [S.l.], 2014. p. 65–76.

LIN, D. Extracting collocations from text corpora. In: CITESEER. *First workshop on computational terminology*. [S.l.], 1998. p. 57–63.

LOPES, J. L. et al. Uma abordagem baseada em ontologias para sensibilidade ao contexto na computação pervasiva. In: *Anais do I Workshop on Pervasive and Ubiquitous Computing (WPUC). Citado na pág*. [S.l.: s.n.], 2007. v. 15.

LUND, K.; BURGESS, C. Producing high-dimensional semantic spaces from lexical co-occurrence. *Behavior Research Methods, Instruments, & Computers*, Springer, v. 28, n. 2, p. 203–208, 1996.

LUND, K.; BURGESS, C.; ATCHLEY, R. A. Semantic and associative priming in high-dimensional semantic space. In: *Proceedings of the 17th annual conference of the Cognitive Science Society*. [S.l.: s.n.], 1995. v. 17, p. 660–665.

MAGARA, M. B.; OJO, S. O.; ZUVA, T. A comparative analysis of text similarity measures and algorithms in research paper recommender systems. In: IEEE. *Information Communications Technology and Society (ICTAS), 2018 Conference on*. [S.l.], 2018. p. 1–5.

MANNING, C. et al. *Foundations of statistical natural language processing*. [S.l.]: MIT Press, 1999. v. 999.

MATVEEVA, I. et al. Generalized latent semantic analysis for term representation. In: *Proc. of RANLP*. [S.l.: s.n.], 2005.

MCCRAY, A. T. The umls semantic network. In: AMERICAN MEDICAL INFORMATICS ASSOCIATION. *Proceedings. Symposium on Computer Applications in Medical Care*. [S.l.], 1989. p. 503–507.

MENG, X. et al. Relevance measure in large-scale heterogeneous networks. In: SPRINGER. *Asia-Pacific Web Conference*. [S.l.], 2014. p. 636–643.

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

MILLER, G. A. et al. Introduction to wordnet: An on-line lexical database. *International journal of lexicography*, Oxford Univ Press, v. 3, n. 4, p. 235–244, 1990.

ORGANIZATION, W. H. et al. Icd-10: The icd-10 classification of mental and behavioural disorders: diagnostic criteria for research. In: *ICD-10: the ICD-10 classification of mental and behavioural disorders: diagnostic criteria for research*. [S.l.: s.n.], 1993.

OSGOOD, C.; SUCI, G.; TANNENBAUM, P. *The measurement of meaning*. [S.l.]: University of Illinois Press, 1964.

OTTE, E.; ROUSSEAU, R. Social network analysis: a powerful strategy, also for the information sciences. *Journal of information Science*, Sage Publications, v. 28, n. 6, p. 441–453, 2002.

PATWARDHAN, S. *Incorporating dictionary and corpus information into a context vector measure of semantic relatedness*. Tese (Doutorado) — University of Minnesota, Duluth, 2003.

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.

PENG, J. et al. Measuring semantic similarities by combining gene ontology annotations and gene co-function networks. *BMC bioinformatics*, BioMed Central, v. 16, n. 1, p. 44, 2015.

PEROZZI, B.; AL-RFOU, R.; SKIENA, S. Deepwalk: Online learning of social representations. In: ACM. *Proceedings of the 20th ACM SIGKDD international conference on Knowledge discovery and data mining*. [S.l.], 2014. p. 701–710.

POTTHAST, M.; STEIN, B.; ANDERKA, M. A wikipedia-based multilingual retrieval model. In: SPRINGER. *European Conference on Information Retrieval*. [S.l.], 2008. p. 522–530.

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. ISBN 1-55860-363-8, 978-1-558-60363-9. Disponível em: <http://dl.acm.org/citation.cfm?id=1625855.1625914>.

RIBEIRO, A. A. P.; ZHAO, L.; MACEDO, A. A. A review of text-based and knowledge-based semantic similarity measures. In: BRAZILIAN SYMPOSIUM ON DATABASES. *Proceedings of the 5th Symposium on Knowledge Discovery, Mining and Learning*. [S.l.], 2017. p. 19–26.

RICHARDS, J. M.; KORNAI, A. *Textual data classification method and apparatus*. [S.l.]: Google Patents, 2003. US Patent 6,507,829.

ROCHA, B. A. et al. Advanced data mining approaches in the assessment of urinary concentrations of bisphenols, chlorophenols, parabens and benzophenones in brazilian children and their association to dna damage. *Environment international*, Elsevier, v. 116, p. 269–277, 2018.

SALTON, G. Automatic text processing. addison welsley. *Reading, Massachusetts*, v. 4, 1989.

SALTON, G.; MCGILL, M. J. Introduction to modern information retrieval. McGraw-Hill, Inc., 1986.

SÁNCHEZ, D.; MARTÍNEZ-SANAHUJA, L.; BATET, M. Survey and evaluation of web search engine hit counts as research tools in computational linguistics. *Information Systems*, Elsevier, v. 73, p. 50–60, 2018.

SCHÜTZE, H.; MANNING, C. D.; RAGHAVAN, P. *Introduction to information retrieval*. [S.l.]: Cambridge University Press, 2008. v. 39.

SHANG, J. et al. Meta-path guided embedding for similarity search in large-scale heterogeneous information networks. *arXiv preprint arXiv:1610.09769*, 2016.

SHI, C. et al. Hetesim: A general framework for relevance measure in heterogeneous networks. *IEEE Transactions on Knowledge and Data Engineering*, IEEE, v. 26, n. 10, p. 2479–2492, 2014.

SHI, C. et al. Relevance search in heterogeneous networks. In: ACM. *Proceedings of the 15th international conference on extending database technology*. [S.l.], 2012. p. 180–191.

SHI, C. et al. A survey of heterogeneous information network analysis. *IEEE Transactions on Knowledge and Data Engineering*, IEEE, v. 29, n. 1, p. 17–37, 2017.

SILVA, T. C.; ZHAO, L. *Machine learning in complex networks*. [S.l.]: Springer, 2016. v. 1.

SNOMED, C. Systematized nomenclature of medicine-clinical terms. *International Health Terminology Standards Development Organisation*, 2011.

SUN, Y.; HAN, J. Mining heterogeneous information networks: a structural analysis approach. *ACM SIGKDD Explorations Newsletter*, ACM, v. 14, n. 2, p. 20–28, 2013.

SUN, Y. et al. Pathsim: Meta path-based top-k similarity search in heterogeneous information networks. *Proceedings of the VLDB Endowment*, Citeseer, v. 4, n. 11, p. 992–1003, 2011.

SUN, Y. et al. Integrating meta-path selection with user-guided object clustering in heterogeneous information networks. In: ACM. *Proceedings of the 18th ACM SIGKDD international conference on Knowledge discovery and data mining*. [S.l.], 2012. p. 1348–1356.

SUN, Y.; YU, Y.; HAN, J. Ranking-based clustering of heterogeneous information networks with star network schema. In: ACM. *Proceedings of the 15th ACM SIGKDD international conference on Knowledge discovery and data mining*. [S.l.], 2009. p. 797–806.

TANG, J.; QU, M.; MEI, Q. Pte: Predictive text embedding through large-scale heterogeneous text networks. In: ACM. *Proceedings of the 21th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*. [S.l.], 2015. p. 1165–1174.

TANG, J. et al. Line: Large-scale information network embedding. In: ACM. *Proceedings of the 24th International Conference on World Wide Web*. [S.l.], 2015. p. 1067–1077.

TSURUOKA, Y.; TSUJII, J. Improving the performance of dictionary-based approaches in protein name recognition. *Journal of biomedical informatics*, Elsevier, v. 37, n. 6, p. 461–470, 2004.

TURNEY, P. Mining the web for synonyms: Pmi-ir versus lsa on toefl. 2001.

TURNEY, P. D. Measuring semantic similarity by latent relational analysis. *arXiv preprint cs/0508053*, 2005.

UKKONEN, E. A linear-time algorithm for finding approximate shortest common superstrings. *Algorithmica*, Springer, v. 5, n. 1, p. 313–323, 1990.

WANG, C. et al. Learning relevance from heterogeneous social network and its application in online targeting. In: ACM. *Proceedings of the 34th international ACM SIGIR conference on Research and development in Information Retrieval*. [S.l.], 2011. p. 655–664.

WANG, C. et al. Relsim: Relation similarity search in schema-rich heterogeneous information networks. In: SIAM. *Proceedings of the 2016 SIAM International Conference on Data Mining*. [S.l.], 2016. p. 621–629.

WANG, G.; HU, Q.; YU, P. S. Influence and similarity on heterogeneous networks. In: ACM. *Proceedings of the 21st ACM international conference on Information and knowledge management*. [S.l.], 2012. p. 1462–1466.

WANG, Y. et al. Flickr group recommendation with auxiliary information in heterogeneous information networks. *Multimedia Systems*, Springer, p. 1–10, 2016.

WASSERMAN, S.; FAUST, K. *Social network analysis: Methods and applications*. [S.l.]: Cambridge university press, 1994. v. 8.

WILLETT, D. L. et al. Snomed ct concept hierarchies for sharing definitions of clinical conditions using electronic health record data. *Applied clinical informatics*, Georg Thieme Verlag KG, v. 9, n. 03, p. 667–682, 2018.

WU, B. et al. *Flexible summarization of textual content*. [S.l.]: Google Patents, 2018. US Patent App. 15/221,367.

WU, H. C. et al. Interpreting tf-idf term weights as making relevance decisions. *ACM Transactions on Information Systems (TOIS)*, ACM, v. 26, n. 3, p. 13, 2008.

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.

XIE, R. et al. Representation learning of knowledge graphs with entity descriptions. In: *AAAI*. [S.l.: s.n.], 2016. p. 2659–2665.

XIONG, C.; ZHONG, V.; SOCHER, R. *Dynamic coattention network for question answering*. [S.l.]: Google Patents, 2018. US Patent App. 15/421,193.

XIONG, Y.; ZHU, Y.; YU, P. S. Top-k similarity join in heterogeneous information networks. *Knowledge and Data Engineering, IEEE Transactions on*, IEEE, v. 27, n. 6, p. 1710–1723, 2015.

YANG, S. et al. A topic detection method based on keygraph and community partition. In: ACM. *Proceedings of the 2018 International Conference on Computing and Artificial Intelligence*. [S.l.], 2018. p. 30–34.

YU, X. et al. User guided entity similarity search using meta-path selection in heterogeneous information networks. In: ACM. *Proceedings of the 21st ACM international conference on Information and knowledge management*. [S.l.], 2012. p. 2025–2029.

ZHANG, M. et al. Top-k similarity search in heterogeneous information networks with x-star network schema. *Expert Systems with Applications*, Elsevier, v. 42, n. 2, p. 699–712, 2015.

ZHU, M. et al. Relevance search on signed heterogeneous information network based on meta-path factorization. In: SPRINGER. *International Conference on Web-Age Information Management*. [S.l.], 2015. p. 181–192.

Appendix A: Machine Configuration

The machine configuration is the following:

```
OS Configuration:          Standalone Workstation
OS Build Type:             Multiprocessor Free
System Manufacturer:       HP
System Model:              HP ENVY Notebook
System Type:               x64-based PC
Processor(s):              1 Processor(s) Installed.
                           [01]: Intel64 Family 6 Model 78 Stepping
                           3 GenuineIntel ~2601 Mhz
BIOS Version:              Insyde F.32, 1/19/2016
Total Physical Memory:     16,266 MB
Available Physical Memory: 10,763 MB
Virtual Memory: Max Size:  18,698 MB
Virtual Memory: Available: 12,490 MB
Virtual Memory: In Use:    6,208 MB
Network Card(s):           4 NIC(s) Installed.
                           [01]: Realtek PCIe GBE Family Controller
                                 Connection Name: Ethernet
                                 Status:          Media disconnected
                           [02]: Intel(R) Dual Band Wireless-AC 7265
                                 Connection Name: Wi-Fi
                                 DHCP Enabled:    Yes
                                 DHCP Server:     192.168.1.253
                                 IP address(es)
                                 [01]: 192.168.1.105
                                 [02]: fe80::14d6:c152:2cdd:ed74
                           [03]: Bluetooth Device (Personal Area Network)
                                 Connection Name: Conexão de Rede Bluetooth
                                 Status:          Media disconnected
                           [04]: TeamViewer VPN Adapter
```

```
                    Connection Name: Ethernet 2
                    Status:          Media disconnected
Hyper-V Requirements:      VM Monitor Mode Extensions: Yes
                           Virtualization Enabled In Firmware: No
                           Second Level Address Translation: Yes
                           Data Execution Prevention Available: Yes
```

Appendix B: Classics Similarity Measures

Similarity Algorithm	Formula Used
Bray-Curtis	BrayCurtis $\left(\overrightarrow{A},\overrightarrow{B}\right)=\sum_{i=1}^{n}\frac{\left\|a_i-b_i\right\|}{(a_i+b_i)}$ (1)
Canberra	Canberra $\left(\overrightarrow{A},\overrightarrow{B}\right)=\sum_{i=1}^{n}\frac{\left\|a_i+b_i\right\|}{\left\|a_i\right\|+\left\|b_i\right\|}$ (2)
Chebyshev	Chebyshev $\left(\overrightarrow{A},\overrightarrow{B}\right)=max_i\left\|a_i+b_i\right\|$ (3)
City-Block	CityBlock $\left(\overrightarrow{A},\overrightarrow{B}\right)=\sum_{i=1}^{n}a_i+b_i\vert$ (4)
Correlation	correlation $\left(\overrightarrow{A},\overrightarrow{B}\right)=1-\frac{\left(A-\overline{A}\right)\cdot\left(B-\overline{B}\right)}{\left\|\left(A-\overline{A}\right)\right\|_2\left\|\left(B-\overline{B}\right)\right\|_2}$ (5)
Cosine	cosine $\left(\overrightarrow{A},\overrightarrow{B}\right)=\left\|\overrightarrow{A}\right\|\left\|\overrightarrow{B}\right\|cos\theta$ (6)
Dice	CityBlock $\left(\overrightarrow{A},\overrightarrow{B}\right)=\frac{2\left\|\overrightarrow{A}\cap\overrightarrow{B}\right\|}{\left\|\overrightarrow{A}\right\|+\left\|\overrightarrow{B}\right\|}$ (7)
Euclidean	Euclidean $\left(\overrightarrow{A},\overrightarrow{B}\right)=\sqrt{\sum_{i=1}^{n}(a_i+b_i)}$ (8)
Hamming	Hamming $\left(\overrightarrow{C}_{\left(\overrightarrow{A},\overrightarrow{B}\right)}\right)=\sum_{i=1}^{n}c_i,c_i\begin{cases}a_i=b_i & c_i=0\\ a_i\neq b_i & c_i=1\end{cases}$ (9)
Jaccard	Jaccard $\left(\overrightarrow{A},\overrightarrow{B}\right)=\frac{\left\|\overrightarrow{A}\cap\overrightarrow{B}\right\|}{\left\|\overrightarrow{A}\cup\overrightarrow{B}\right\|}$ (10)
Kulsinski	Kulsinski $\left(\overrightarrow{A},\overrightarrow{B}\right)=1-\frac{1}{2}\left(\frac{\left\|\overrightarrow{A}\cap\overrightarrow{B}\right\|}{\left\|\overrightarrow{A}\right\|}+\frac{\left\|\overrightarrow{A}\cap\overrightarrow{B}\right\|}{\left\|\overrightarrow{B}\right\|}\right)$ (11)
Mahalanobis	mahalanobis $\left(\overrightarrow{A},\overrightarrow{B}\right)=\sqrt{\left(\overrightarrow{A}-\overrightarrow{B}\right)^T S^{-1}\left(\overrightarrow{A}-\overrightarrow{B}\right)}$ (12)
Minkowski	Minkowski $\left(\overrightarrow{A},\overrightarrow{B}\right)=\left(\sum_{i=1}^{n}\left\|a_i-b_i\right\|^p\right)^{\frac{1}{p}}$ (13)
Rogers-Tanimoto	RogersTanimoto $\left(\overrightarrow{A},\overrightarrow{B}\right)=\frac{\left\|\overrightarrow{A}\cap\overrightarrow{B}\right\|+\left\|\overrightarrow{C}\right\|}{\left\|\overrightarrow{A}\cap\overrightarrow{B}\right\|+2\left(\left\|\overrightarrow{A}-\overrightarrow{B}\right\|+\left\|\overrightarrow{B}-\overrightarrow{A}\right\|\right)+\left\|\overrightarrow{C}\right\|};\overrightarrow{C}\notin\left\{\overrightarrow{A}\cup\overrightarrow{B}\right\}$ (14)
Russell-Rao	RusselRao $\left(\overrightarrow{A},\overrightarrow{B}\right)=\frac{\left\|\overrightarrow{A}\cap\overrightarrow{B}\right\|}{\left\|\overrightarrow{A}\cap\overrightarrow{B}\right\|+\left\|\overrightarrow{A}-\overrightarrow{B}\right\|+\left\|\overrightarrow{B}-\overrightarrow{A}\right\|+\left\|\overrightarrow{C}\right\|};\overrightarrow{C}\notin\left\{\overrightarrow{A}\cup\overrightarrow{B}\right\}$ (15)
Seuclidean	Euclidean $\left(\overrightarrow{A},\overrightarrow{B}\right)=\sqrt{\sum_{i=1}^{n}(a_i+b_i)}$ (16)
Sokal-Michener	SokalMichener $\left(\overrightarrow{A},\overrightarrow{B}\right)=\frac{\left\|\overrightarrow{A}\cap\overrightarrow{B}\right\|+\left\|\overrightarrow{C}\right\|}{\left\|\overrightarrow{A}\cap\overrightarrow{B}\right\|+\left\|\overrightarrow{A}-\overrightarrow{B}\right\|+\left\|\overrightarrow{B}-\overrightarrow{A}\right\|+\left\|\overrightarrow{C}\right\|};\overrightarrow{C}\notin\left\{\overrightarrow{A}\cup\overrightarrow{B}\right\}$ (17)
Sokal-Sneath	SokalSneath $\left(\overrightarrow{A},\overrightarrow{B}\right)=\frac{\left\|\overrightarrow{A}\cap\overrightarrow{B}\right\|}{\left\|\overrightarrow{A}\cap\overrightarrow{B}\right\|+2\left(\left\|\overrightarrow{A}-\overrightarrow{B}\right\|+\left\|\overrightarrow{B}-\overrightarrow{A}\right\|\right)}$ (18)
Sq-Euclidean	SqEuclidean $\left(\overrightarrow{A},\overrightarrow{B}\right)=\sqrt{\sum_{i=1}^{n}(a_i-b_i)^2}$ (19)
Wminkowski	wminkowski $\left(\overrightarrow{A},\overrightarrow{B}\right)=\left(\sum_{i=1}^{n}\left(\left\|w_i\left(a_i-b_i\right)\right\|^p\right)\right)^{\frac{1}{p}}$ (20)
Yule	Yule $\left(\overrightarrow{A},\overrightarrow{B}\right)=1\frac{\overrightarrow{A}-\overrightarrow{B}}{\overrightarrow{A}+\overrightarrow{B}}$ (21)

Appendix C: Statistical Formulas

Measure	Derivations
Sensitivity	$TPR = \frac{TP}{(TP+FN)}$
Specificity	$SPC = \frac{TN}{(FP+TN)}$
Precision	$PPV = \frac{TP}{(TP+FP)}$
Negative Predictive Value	$NPV = \frac{TN}{(TN+FN)}$
False Positive Rate	$FPR = \frac{FP}{(FP+TN)}$
False Discovery Rate	$FDR = \frac{FP}{(FP+TP)}$
False Negative Rate	$FNR = \frac{FN}{(FN+TP)}$
Accuracy	$ACC = \frac{(TP+TN)}{(P+N)}$
F1 Score	$F1 = \frac{2TP}{(2TP+FP+FN)}$
Matthews Correlation Coefficient	$\frac{TP*TN-FP*FN}{\sqrt{(TP+FP)*(TP+FN)*(TN+FP)*(TN+FN)}}$