FASSI 2021

The Seventh International Conference on Fundamentals and Advances in Software Systems Integration

ISBN: 978-1-61208-923-2

November 14 - 18, 2021

Athens, Greece

FASSI 2021 Editors

Petre Dini, IARIA, USA/EU

FASSI 2021

Forward

The Seventh International Conference on Fundamentals and Advances in Software Systems Integration (FASSI 2021), held on November 14-18, 2021, continued a series of events started in 2015 and covering research in the field of software system integration.

On the surface the question of how to integrate two software systems appears to be a technical concern, one that involves addressing issues, such as how to exchange data (Hohpe 2012), and which software systems are responsible for which part of a business process. Furthermore, because we can build interfaces between software systems we might therefore believe that the problems of software integration have been solved. But those responsible for the design of a software system face a number of trade-offs. For example the decoupling of software components is one way to reduce assumptions, such as those about where code is executed and when it is executed (Hohpe 2012). However, decoupling introduces other problems because it leads to an increase in the number of connections and introduces issues of availability, responsiveness and synchronicity of changes (Hohpe 2012).

The objective of this conference is to work toward on understanding of these issues, the trade-offs and the problems of software integration and to explore strategies for dealing with them. We are interested to receive paper from researchers working in the field of software system integration.

We take here the opportunity to warmly thank all the members of the FASSI 2021 technical program committee, as well as all the reviewers. The creation of such a high quality conference program would not have been possible without their involvement. We also kindly thank all the authors who dedicated much of their time and effort to contribute to FASSI 2021. We truly believe that, thanks to all these efforts, the final conference program consisted of top quality contributions.

We also thank the members of the FASSI 2021 organizing committee for their help in handling the logistics and for their work that made this professional meeting a success.

We hope that FASSI 2021 was a successful international forum for the exchange of ideas and results between academia and industry and to promote further progress in the area of software systems integration.

FASSI 2021 Chairs

FASSI 2021 Steering Committee
Chris Ireland, The Open University, UK
Mihaela Iridon, Candea LLC, USA

FASSI 2021 Publicity Chair
Mar Parra Boronat, Universitat Politecnica de Valencia, Spain
Sandra Viciano Tudela, Universitat Politecnica de Valencia, Spain

FASSI 2021

Committee

FASSI 2021 Steering Committee
Christopher Ireland, OnMove, UK
Mihaela Iridon, Candea LLC, USA

FASSI 2021 Publicity Chair
Mar Parra, Universitat Politecnica de Valencia, Spain
Sandra Viciano Tudela, Universitat Politecnica de Valencia, Spain

FASSI 2021 Technical Program Committee

Ghaleb M. Abdulla, Lawrence Livermore National Laboratory, USA
Lavanya Addepalli, Universitat Politecnica de Valencia, Spain
Isabela Alves Marques, Universidade Federal de Uberlândia, Minas Gerais, Brazil
Mariia Andriyiivna Nazarkevych, Lviv Polytechnic National University, Ukraine
Vu Nguyen Huynh Anh, Center in Management Information Systems - Université catholique de Louvain, Belgium
Pablo O. Antonino, Fraunhofer IESE, Germany
Imen Ben Mansour, University of Manouba, Tunisia
Dhouha Ben Noureddine, University of Carthage/ University of El Manar, Tunisia
Silvia Bonfanti, University of Bergamo, Italy
Michael Franklin Bosu, Waikato Institute of Technology, New Zealand
Nitish Devadiga, Datarista Inc. / Carnegie Mellon University, USA
Michal Doležel, Prague University of Economics and Business, Czech Republic
Ivanna Dronyuk, Lviv Polytechnic National University, Ukraine
Imane El Alaoui, University of Ibn Tofail, Kénitra, Morocco
Aziz Fellah, Northwest Missouri State University, USA
Ali Mohsen Frihida, University of Tunis El Manar, Tunisia
Youssef Gahi, Ibn Tofail University, Kenitra, Morocco
Svetlana Gerbel, Hannover Medical School / Medical Data Integration Center / Hannover University of Applied Sciences and Arts, Germany
Alan Hayes, University of Bath, UK
Samedi Heng, HEC Liège - Université de Liège, Belgium
Sebastian Herold, Karlstad University, Sweden
Anca Daniela Ionita, University Politehnica of Bucharest, Romania
Christopher Ireland, OnMove, UK
Mihaela Iridon, Candea LLC, USA
Yassine Issaoui, University Casablanca, Morocco
Ivan Izonin, Lviv Polytechnic National University, Ukraine
Asharul I. Khan, Sultan Qaboos University, Muscat, Oman
Elmar Krainz, FH JOANNEUM University of applied Sciences, Austria
Cristiane Lana, University of São Paulo, Brazil
Damian Lyons, Fordham University, USA

Copyright Information

For your reference, this is the text governing the copyright release for material published by IARIA.

The copyright release is a transfer of publication rights, which allows IARIA and its partners to drive the dissemination of the published material. This allows IARIA to give articles increased visibility via distribution, inclusion in libraries, and arrangements for submission to indexes.

I, the undersigned, declare that the article is original, and that I represent the authors of this article in the copyright release matters. If this work has been done as work-for-hire, I have obtained all necessary clearances to execute a copyright release. I hereby irrevocably transfer exclusive copyright for this material to IARIA. I give IARIA permission or reproduce the work in any media format such as, but not limited to, print, digital, or electronic. I give IARIA permission to distribute the materials without restriction to any institutions or individuals. I give IARIA permission to submit the work for inclusion in article repositories as IARIA sees fit.

I, the undersigned, declare that to the best of my knowledge, the article is does not contain libelous or otherwise unlawful contents or invading the right of privacy or infringing on a proprietary right.

Following the copyright release, any circulated version of the article must bear the copyright notice and any header and footer information that IARIA applies to the published article.

IARIA grants royalty-free permission to the authors to disseminate the work, under the above provisions, for any academic, commercial, or industrial use. IARIA grants royalty-free permission to any individuals or institutions to make the article available electronically, online, or in print.

IARIA acknowledges that rights to any algorithm, process, procedure, apparatus, or articles of manufacture remain with the authors and their employers.

I, the undersigned, understand that IARIA will not be liable, in contract, tort (including, without limitation, negligence), pre-contract or other representations (other than fraudulent misrepresentations) or otherwise in connection with the publication of my work.

Exception to the above is made for work-for-hire performed while employed by the government. In that case, copyright to the material remains with the said government. The rightful owners (authors and government entity) grant unlimited and unrestricted permission to IARIA, IARIA's contractors, and IARIA's partners to further distribute the work.

Table of Contents

Analyzing Hyperonyms of Stack Overflow Posts

László Tóth
Software Engineering Department
University of Szeged
Szeged, Hungary
email: premissa@inf.u-szeged.hu

László Vidács
MTA-SZTE Research Group on Artificial Intelligence
Department of Software Engineering
University of Szeged
Szeged, Hungary
email: lac@inf.u-szeged.hu

***Abstract*—Communication among people is often a challenging task due to the different interpretations of the terms they use. The way people interpret the terms highly depends on the semantic context, where the notions were acquired. The different contexts provide somewhat distinct meanings to the terms used. In software development and integration, requirements engineering and customer support are primarily affected by the difficulties stemming from communication obstacles. The necessary information is often inadequately forwarded to developers resulting in poorly specified software requirements or misinterpreted user feedback. The communication difficulties mentioned can be solved by clarifying the meanings of the concepts used. Semantic networks built on different contexts are suitable tools for this purpose. This paper presents a formal description of the semantic network and the semantic space needed for the algorithmic treatment of the concepts. It provides a model for extracting hyperonymy and hyponymy relations from text corpora created in specific semantic domains. The model was applied on a corpus acquired from Stack Overflow containing conversations among the software developers to solve programming issues.**

Keywords-semantic network; semantic space; lexico-syntactic patterns; noun phrases; NLP; Stack Overflow.

I. INTRODUCTION

Communication among people is, in general, often fraught with difficulties and misunderstandings. Information exchange is influenced by factors such as cultural background, social environment, the available communication channels, the personality and mental state of the participants, and their communication intention, even when a common language is used. The cultural background and the social environment have paramount importance because they can affect the actual meanings of the words used in the communication. However, the participants can only understand each other if they use communication elements in the same sense.

Although communication disruptions can cause problems in everyday life, they may also have unforeseeable consequences in business life, especially if the communication among the participants does not go smoothly. Primarily in larger or sometimes in medium-sized companies, there is a remarkable diversification between the individual departments in the organizational culture, often reflected in their language usage. This phenomenon, called the communication silo [1], is increasingly present in communication between IT and the business area [2][3]. The success of a software project depends significantly on the fact that requirements and business rules are communicated to developers accurately. Although accumulated experiences exist to support the requirements engineering process, the collection of requirements is usually incomplete [4]. The principal cause of this deficiency is the same silo phenomena that generate impediments in exchanging information between units within a given organization.

The difference in the meaning of terms used in various semantic contexts plays a vital role in the silo phenomenon. Semantic networks [5] built on different contexts can provide significant support in solving this problem. This paper presents research that proposes an expandable semantic network based on discourses in the software development domain, relying on a chosen type of association called the hyperonymy-hyponymy relationship. In addition to the graph-based semantic database, the contributions of our research are the formal definitions of the concepts of semantic network and semantic space, as well as the development of a novel phrase structure grammar along with the automation supporting the recognition of noun phrases needed to build the database.

The paper is structured as follows. Section II introduces the theoretical background behind this study. Section III presents the elements participating in our research, such as the Stack Overflow, noun phrases, and the lexico-syntactic patterns. In Section IV, we introduce the technical details about the extraction process, along with the pre- and post-processing phases. In Section V, the results of the experiment are presented and discussed. A few aspects of threats to validity are examined in Section VI. Section VII provides a literature review of related works. Finally, Section VIII concludes the paper.

II. THEORETICAL BACKGROUND

The formation of the conceptual system of humans depends on several factors. One such important factor is the presence of common characteristics of perception. These properties influence the order of language acquisition and also have effects on the usage of various languages [5]. The very first linguistic elements acquired by young children relate to nouns denoting objects and verbs connecting to simple movements. These lexical elements carry the same meaning for everyone within a given language; using them in communication does not cause misunderstandings. The more abstract concepts are built from already known lexical elements, and many of them are context-dependent. Similarly, the actual meaning of multi-meaning words is determined by their context. When

 ISBN: 978-1-61208-923-2

the context is asymmetric in the communication, context-dependent words can introduce misunderstandings.

The meaning of an abstract concept in a given context marked by a particular term is determined by its relation to other concepts valid in the same context. Processing concepts marked with their terms by a computer program requires awareness of these relationships, which, in turn, can help clarify the actual meaning of a given term.

Concepts are abstract elements referring to the things of the world and their relationships. Concepts are mental objects, and also, according to Frege [6], they are abstract objects and can be organized in a hierarchical structure. According to our knowledge about the functioning of the human mind, each concept is stored in our memory with its relationships, forming a structure called a *semantic network* [5].

Definition 1: A semantic network is a

$$G = (C, R, \Sigma_1, \Sigma_2, s, d, l_1, l_2)$$

labeled, directed multigraph, where C is the set of nodes representing the concepts of a given domain, and R is the set of edges representing the relationships between the elements of C. Σ_1 and Σ_2 are the alphabets of the labels corresponding to the nodes and edges, respectively. The $s, d : R \to C$ are the source and destination functions:

$$\forall r \in R,\ \exists (X, Y \in C) : r = (X, Y) \wedge X = s(r) \wedge Y = d(r).$$

Similarly, $l_1 : \Sigma_1 \to C$ and $l_2 : \Sigma_2 \to R$ are the two labeling functions for the nodes and the edges, respectively.

The connection between the world and the concepts has a third component, without which communication would be impossible. This part is the marker associated with concepts, most often a linguistic phrase. Ideally, the relationship between concepts and linguistic markers in a given language is injective, but this is not the case in reality. Injection exists only in a narrower context called *semantic space*.

Definition 2: Let C be an n-element set of concepts and $C_k \subseteq C$ its k-element subset $(k \leq n)$. Let

$$S = (c_1, f_1), (c_2, f_2), ..., (c_k, f_k)$$

$c_i \in C_k$, $f_i \in F (i = 1, ..., k \leq n)$ be a series, where

$$F := \{f | f : C \times C_k \to [0, 1]\}$$

is a set of membership functions designating those $c_j \in C_k$ $(j = 1, ..., k)$ concepts that participate in defining a particular $c_i \in C$ concept $(i = 1, .., .n)$. Let $V = \mathbf{R}^k$ be a vector space over $\mathbf{R}$ and $\Psi : C \to V$ surjective mapping as follows:

1) $\forall c \in C : \Psi(c) = \mathbf{v}(v_1, v_2, ..., v_k) \in V$.
2) $v_i = f_i(c, c_i) \in [0, 1]$, $(i = 1, ..., k)$, where $(c_i, f_i) \in S$ and $(c, c_i) \in C \times C_k$, a fuzzy relationship between c and c_i, $(i = 1, ..., k)$.

Given S, the vector space V is called the semantic space of C if the mapping Ψ is bijective.

Semantic networks restricted to various semantic spaces provide a comprehensive and tractable way for examining the semantic relationships among the concepts denoted by various word phrases. These relationships can be defined in various ways. We focus on generalization and specialization, which are called *hypernymy* and *hyponymy* relations in linguistics.

Definition 3: Let C be the set of concepts and let L be a natural language with the alphabet Σ. Let Σ^* be the constraint of the set of words over L for the valid words and word phrases of L. Let $X, Y \in \Sigma^*$, and let $M : \Sigma^* \to C$ be a mapping from the word phrases to the set of concepts. Let P be the set of properties describing the objects of the world, and let $\Pi : C \to P$ be a mapping from the set of concepts to the set of properties. *Hyponymy* relation between X and Y is defined as

$$Hyponym(X, Y) \iff \Pi(M(Y)) \subset \Pi(M(X)).$$

Hyperonymy relation, denoted as $Hyperonym(X, Y)$, is the inverse relation of the *hyponymy*.

Note: Other linguistic relationships, such as *meronymy*, *holonymy*, *synonymy*, and *antonymy*, can be defined similarly.

III. Mining linguistic relationships from Stack Overflow posts

Stack Overflow is an extensive knowledge repository in the software development and engineering field. The phrases found in its posts might serve as a base to build a semantic network for the software development domain. Nevertheless, some critical issues need to be considered upon using this dataset. The posts often contain code fragments or special character strings used in programming. Besides, many posts are written by non-English speakers; therefore, they might contain smaller or bigger grammatical errors.

In terms of content value, tacit knowledge in Stack Overflow posts can be trusted to be high quality because of the site's strict community control. Posts that fail to meet requirements established by the SO community are to be closed and eventually deleted [7]. The SO community has been making a significant effort to maintain the quality and the professionality of the site [8]; therefore, the information in posts with positive scores can be considered trustworthy.

Hyponymy and hyperonymy relations can be extracted from free texts using lexico-syntactic patterns proposed by Hearst [9]. These relationships represent the canonical is-a relationship, which can also be interpreted as the specialization and the generalization in object-oriented modeling. The collection of lexico-syntactic patterns have been gradually expanded since Hearst established the base models. The momentum of this expansion comes from the rapid spread of web-based text mining. Our work adopted the patterns introduced in [10]. These patterns were slightly modified to avoid collision with the extraction process of noun phrases from the text and write more compact code.

In computer-based language processing, we need to give a formal definition to the examined linguistic structures, which approximates the set of the structures used in reality. For this purpose, the phrase structure grammar [11] is a suitable choice due to its algorithmic manageability.

For matching lexico-syntactic patterns on the input text identifying noun phrases [12] in the original text is required.

 ISBN: 978-1-61208-923-2

TABLE 1
GRAMMAR DEFINED FOR NP RECOGNITION

⟨NP⟩	⊨	(PDT) (DET (CD) | PRP$) ((⟨ADJPS⟩) ⟨HEAD⟩ ((⟨GERS⟩) ((⟨PPS⟩) ((⟨REL⟩)
⟨ADJPS⟩	⊨	((⟨ADVS⟩) ⟨ADJ⟩ ((⟨CONJ⟩ ⟨ADJ⟩)*
⟨ADVS⟩	⊨	⟨ADV⟩ ((⟨CONJ⟩ ⟨ADV⟩)*
⟨ADJ⟩	⊨	JJ | JJR | JJS
⟨ADV⟩	⊨	RB | RBR | RBS
⟨HEAD⟩	⊨	NN(POS) | NNP(POS) | NNPS(POS) | NNS(POS) | PRP | SYM | FW
⟨GERS⟩	⊨	⟨GER⟩ ((⟨CONJ⟩ ⟨GER⟩)*
⟨GER⟩	⊨	VBG (⟨NP⟩)
⟨PPS⟩	⊨	IN ⟨NP⟩ | TO VB ((⟨CONJ⟩ VB)*
⟨CONJ⟩	⊨	, | CC
⟨REL⟩	⊨	WDT | WP | WP$ | WRB ⟨VP⟩

For this purpose, a phrase structure grammar was developed, which is shown in Table 1. The ⟨.⟩ symbols denote the non-terminals, whereas the symbols without brackets denote the terminals based on the POS *(parts-of-speech)* tags defined in The Penn Treebank [13]. Parentheses denote the optional elements.
Note: Although the relative clauses are presented in the grammar, they are omitted from the current implementation.

IV. TECHNOLOGY USED IN MINING

This section describes the technical background of the mining process applied to the Stack Overflow posts presented in Figure 1.

The preprocessing phase follows the same steps as defined in our previous work [14]. The dataset used is the Stack Overflow dump was created on March 4, 2019, and migrated locally to a PostgreSQL 10.10 database. From this dataset, posts that received non-negative scores from the Stack Overflow community were selected.

The HTML tags were removed from the selected posts, and the code blocks and hyperlinks were replaced with `code example` and `link` strings, respectively. Some characters, such as the +, |, and \ were replaced with a single space character, but punctuation characters were retained. `C++, C#, F#`, and their lowercase counterparts were replaced with `cplusplus, csharp`, and `fsharp`, respectively. The resulting text was split into sentences.

The sentences need further processing steps before the NP (*noun phrase*) recognition can be performed. All characters were converted into lowercase. Non-alphabetic characters were transformed into a single space, but those characters that play a vital role in recognizing the patterns, or have a distinctive role in the text, such as comma, apostrophe, colon, were retained. Additionally, the recognition process needs the parts-of-speech tags; therefore, the sentences were converted into a series of *(word, POS tag)* tuple pairs.

The NP parser is based on the automation presented in Figure 2. The automation utilizes the POS tags to compute

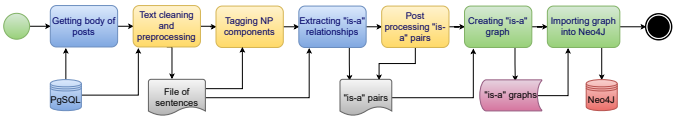

Figure 1. Creating semantic network

the proper transition. Although recognizing NPs and lexico-syntactic patterns is a separate procedure, the lexical elements *(such as some, any, kind, sort)* used in lexico-syntactic patterns should be considered when recognizing NPs.

The POS tagging does not always produce the appropriate tag, which requires manual work to correct it. In our experiments, two important cases were detected and corrected. In the expressions *sound/s like*, the POS tag of *sound* was corrected to VB (*verb*), and in the case of the phrase *operating system*, the POS tag of *operating* was corrected to JJ (*adjective*).

By running the procedure described above, the noun phrases recognized by the algorithm are tagged using <NP > and </NP > tags in the string containing the original tokens. For example, the sentence *"Flour or any other grain can be found in the kitchen"* is transformed to *"<NP > flour </NP > or any other <NP > grain </NP > can be found in <NP > the kitchen </NP >"* series of tokens.

The token series obtained in the NP recognition process was the input for the extraction of relations fitted to lexico-syntactic patterns presented in Table 2 . The extractor was written using regular expressions carefully designed to identify the patterns.

From the recognized terms, the pairs of related NPs were written in a CSV file. The first part of the pair is the more specific NP *(hyponym part)*, and the second part is the more general one *(hyperonym part)*. The pairs were only saved if neither of the two parts of the pairs was a single pronoun.

The lexico-syntactic patterns used in the extraction process assume an input text written with proper English grammar. However, Stack Overflow posts are often written by non-English speaking users who sometimes make grammatical errors. These errors might result in a wrong relation extracted from the text. Therefore, a few post-processing steps have been introduced to reduce the number of mismatched pairs.

During the post-processing, pairs in which one part is empty or contains a single character other than *c*, *f*, and *b*, and those in which both parts are the same, have been deleted. If only one part of the relation is a proper noun, that part was considered the hyponymy part. If both parts of the relationship are proper nouns, or neither of them, the relationship was checked against WordNet [15] and was corrected according to it.

After the post-processing phase, the extracted pairs are ready for graph building. These pairs provide the set of edges of the semantic graph. Unfortunately, duplications can also occur among these pairs, which are to be removed during the graph building process. For building the graph, the `networkx` Python package was utilized. The resulting graph was then imported into a Neo4J [16] database.

 ISBN: 978-1-61208-923-2

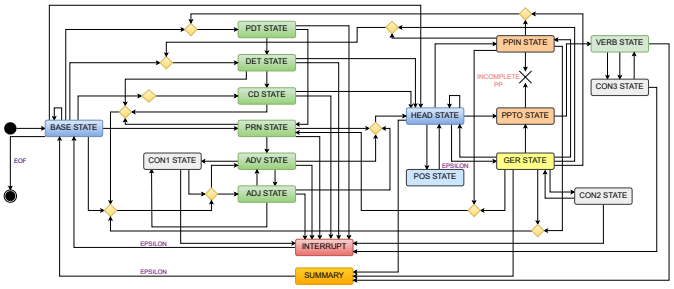

Figure 2. NP Extractor Automation

V. Examination of the Extracted Semantic Net

This section provides the investigation results of the extracted semantic network and some statistical data describing the various outcomes of the whole process.

The total number of imported posts from the dump created on March 4, 2019, is 43,872,992. Only posts with non-negative scores were considered for further processing, the number of which is 42,160,482; this is 96.1 % of the original posts. From this dataset, 137,440,998 sentences were created during the sentence-based tokenization process; 7,583,195 hyponymy and hyperonymy relations were extracted. The ratio is quite small; approximately only 5.5 % of the sentences contain the studied relations. It is important to note that the amount of relations imported into the semantic graph is smaller due to the filtering and merging method applied during the graph creation. Table 2 presents the distribution of the extracted relations from this set. According to the distribution, the following three lexico-syntactic patterns are used the most dominantly in Stack Overflow posts: NP_t *is|are|was|were* $(a|an)$ NP_h, NP_t *as* NP_h and NP_h *like* NP_t. In turn, the occurrence of the following patterns is marginal: NP_h *example of this is|are* NP_t and NP_t *or the many* NP_h.

The statistical results were compared to those of Seitner et al [10]. They applied a similar lexico-syntactic pattern-based mining on the dataset obtained from CommonCrawl [17] using a slightly different grammar for NP identification and, therefore, a slightly different set of patterns. Despite the differences, we found that the occurrence of patterns followed a similar trend. Interestingly, according to Seitner et al., the most commonly occurring pattern is a sub-pattern of the most frequent pattern found in our study (NP_t *is|are|was|were* $(a|an)$ NP_h). The incidence of the other patterns is similar, although there are some differences as well. For example, the frequency of the pattern NP_t *as* NP_h is significant in our case, while in Seitner's study, this is not typical.

Our semantic graph contains 3,926,617 nodes and 7,413,639 relationships. The considered is-a relationship is directed. The natural direction is defined by the order of the input pairs and points from the more specific term to the more general one. The natural degree of a given node indicates how many examples of different higher-order concepts can be represented with that same node. In the opposite direction, the degree of a node indicates the number of possible subtypes related to the generality or specificity of the concept denoted by that node.

TABLE 2
Distribution of the Patterns in the Extraction Results

Lexico-syntactic pattern	Number of cases
NP_t *is|are|was|were* $(a\|an)$ NP_h	3,490,476
NP_t *as* NP_h	2,143,334
NP_h *like* NP_t	1,033,497
NP_h *such as* NP_t	278,078
NP_t *and|or* $(any\|some)$ *other* NP_h	207,079
NP_h *especially|esp(.)|including|inc(.)* NP_t	117,151
NP_h *for example* NP_t	66,968
NP_h *except* NP_t	44,766
NP_h *e.g.|i.e.* NP_t	44,295
NP_h *other than* NP_t	33,956
NP_t (is) *one of the|these|those|this|that* NP_h	30,996
NP_t *which look(s)|sound(s) like* NP_h	28,589
such NP_h *as* NP_t	14,969
compare NP_t *with* NP_h	10,992
NP_h *compared to* NP_t	8,106
NP_h *which is|are similar to* NP_t	6,385
NP_h *in particular* NP_t	5,671
NP_h *mainly|mostly|notably* NP_t	3,866
NP_h *particularly|principally* NP_t	3,035
NP_h *which is called|named* NP_t	2,924
NP_h *whether* NP_t *or*	1,957
NP_t *is|was|are|were a kind of|kinds of* NP_h	1,936
NP_t *like other* NP_h	1,813
NP_t *is|was|are|were a form of|forms of* NP_h	799
NP_t *is|are example(s) of* NP_h	623
NP_t *is|was|are|were a sort of|sorts of* NP_h	620
examples of NP_h *is|are* NP_t	276
NP_t *or the many* NP_h	21
NP_h *example of this is|are* NP_t	17

The distribution of the degree in the normal direction is presented in Figure 3. Looking at the relationship in the opposite direction, we get a similar distribution. It can be seen in the figure that the degree of the node *code example* excels from the distributions. The difference between the degree of this node and that of the second-highest is 106,653 in the normal (*hyponymy* $\rightarrow$ *hyperonymy*) direction and 208,263 in reverse (*hyperonymy* $\rightarrow$ *hyponymy*) direction. The node with the second highest degree in the normal direction is the node *below*, with a score of 148,435. This result seems to be a mistake because the word *below* can be a preposition or an adverb. The expected results, however, should only be NPs. However, this particular word, *below*, can sometimes behave like a noun as well as an adjective [18]. Consider the following example from a Stack Overflow post: *"and below is the python interpreter setting on pycharm..."*. In this case, the NLTK POS tagger recognizes *below* as a noun; therefore, the NP extractor also recognizes it as a noun phrase. The extracted relationship is *"the python interpreter setting on pycharm | below"* in this case. The culprit is the phrase *"below is."* The tagger recognizes all similar occurrences as nouns.

The node *below* and similar adverbs, such as *before, after, following, next*, and *above*, are not interpretable without their actual context. Therefore, nodes representing these words can be removed securely from the semantic network, including their counterparts with either definite or indefinite articles.

 ISBN: 978-1-61208-923-2

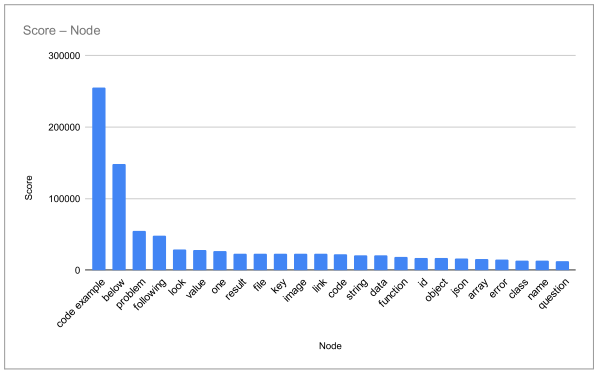

Figure 3. Distribution of degree in hyponym-hyperonym direction

Similarly, placeholders, such as the phrase *code example* or the word *link*, are replacing specific objects – code snippets or hyperlinks, respectively – and they neither provide a refinement nor a more general meaning of any terms. These nodes can also be discarded from the semantic network. After discarding the nodes mentioned above, 3,926,609 nodes and 6,059,017 relationships remained in the resulting graph.

The degree distribution of the resulting graph was also examined. Considering the distance in the square of skewness and kurtosis space of the actual distribution from some theoretical distributions indicates that a specialized gamma distribution can be fitted. The analysis results show that in the case of the hyponymy $\rightarrow$ hyperonymy direction, the distribution follows the Box-Cox Power Exponential distribution [19], whereas in the case of the opposite direction, the distribution follows the power-law distribution.

The average clustering coefficient is 0.017, meaning that only 1.7% of the concepts tend to form triadic closures. A possible explanation of this small number can be that large degree nodes connect the communities formed in the knowledge graph. This phenomenon can be interpreted as the concepts being defined based on a few core concepts. Further studies are needed to confirm this conjecture.

VI. Threats to validity

The methods presented in this paper, along with the results of the evaluation, have some limitations:

- The automation that recognizes noun phrases is not complete. Relative clauses are omitted from the automation. These parts are built on additional structures like verb phrases, allowing for the recognition of more complex noun phrases. Building the knowledge graph, we do not need such complex terms; therefore, omitting this part of the grammar does not cause any loss regarding the results.
- Lexico-syntactic patterns used in the extraction process do not fully cover possible relationships, and collision with the NP grammar might occur. During the implementation, we considered the structures found in the literature, and we avoided those that arise in other relation-type in terms of the substructure. Our tool considers the special terms used in the lexico-syntactic patterns and reduces errors resulting from collisions with grammar.
- In some special cases, the NLTK POS tagger assigns an incorrect tag to some terms. These mistakes can be corrected using manual inspection or applying other taggers and examining the differences. We focused only on crucial cases and corrected the mistagged terms.
- The patterns used in the implementation are based on text with proper English grammar. However, grammatical errors are common, which has a detrimental effect on the extraction process. We have effectively mitigated this effect using various post-processing steps.

VII. Related work

The graph structure of semantic databases was exploited after the 2000s, thanks to increased storage and computational capacities. Steyvers and Tenebaum [20] investigated the structure of two famous semantic networks, WordNet and Roget's Thesaurus. They found that these databases have a small world structure and the distributions of the number of connections follow the power-law distribution. Seitner et al. [10] have extracted hyperonym relations from the CommonCrawl web corpus. Itto et al. [21] focused on subtracting meronymy relationships from texts created in product development and customer service relations, whereas Yildiz and his workmates studied the meronym relationships in Turkish raw text, applying lexico-syntactic patterns [22]. Vizcaino et al. [23] focused on establishing a standard vocabulary among the participants of global software development to overcome the communication barriers. Futia and his workmates have developed a tool called SeMi to build large-scale Knowledge Graphs from structured sources semi-automatically [24].

NLP methods, such as the usage of ontologies and machine learning in requirements engineering, are also intensively examined. Falessi et al. [25] investigated the usage of NLP techniques in requirements classification. Holter [26] developed methods based on NLP techniques to translate requirements given in natural language into a structured semantic database. A systematic literature review of using machine learning methods and NLP in requirements engineering is presented in the paper of Ahmad et al. [27]. The ontology-based approach of extracting functional or non-functional requirements from the original texts has also been applied successfully. Li and Chen [28] and Alrumaih et al. [29] applied an ontology-based approach for classifying requirements.

VIII. Conclusion

A significant proportion of communication difficulties among people and organizations stem from distinct interpretations of the linguistic elements used in communication. The only way to deal with semantic differences caused by different contexts is to clarify the meanings of the terms used, for which semantic networks are suitable tools.

 ISBN: 978-1-61208-923-2

In this paper, we have examined the structure of semantic networks and given a precise definition of both networks and the concept of semantic space. We have provided a grammar that can recognize complex noun phrases and presented an expandable process based on lexico-syntactic patterns to extract hyperonym relations from written texts. The process was applied on the posts of StackOverflow, which plays the role of semantic space of the software development.

The investigation of the yielded semantic graph shows that relatively few sentences from the available textual corpus contained structures fitting the relationships examined. The distribution of patterns also shows that the users prefer the usage of some particular patterns to others. The semantic network structure suggests that the concepts in the studied environment are built from a small number of basic concepts. This phenomenon can be seen on the graph; several nodes are connected to these nodes representing the basic concepts. This phenomenon, however, needs further investigation.

As a continuation of the research, our goals include extracting other relationships, such as meronymy-holonymy and incorporating them into the structure of the existing semantic network. Our further aim is to investigate learning methods in extracting these relationships and applying the methods studied to build semantic networks in different semantic domains.

ACKNOWLEDGMENT

The research presented in this paper, carried out by University of Szeged was supported by the Ministry of Innovation and the National Research, Development and Innovation Office within the framework of the Artificial Intelligence National Laboratory Programme. This research was supported by grant NKFIH-1279-2/2020 of the Ministry for Innovation and Technology, Hungary.

REFERENCES

[1] S. Bundred, "Solutions to silos: joining up knowledge," *Public Money & Management*, vol. 26, no. 2, pp. 125–130, 2016, DOI: 10.1111/j.1467-9302.2006.00511.x.

[2] H. Enquist and N. Makrygiannis, "Understanding misunderstandings [in complex information systems development]," in *Proceedings of the thirty-first Hawaii International Conference on System Sciences*, vol. 6. Los Alamitos, CA, USA: IEEE Computer Society, 1998, pp. 83–92, DOI: 10.1109/HICSS.1998.654762.

[3] T. Mohapeloa, "Effects of silo mentality on corporate ITC's business model," in *Proceedings of the International Conference on Business Excellence*, vol. 11, no. 1, 2017, pp. 1009–1019, DOI: 10.1515/picbe-2017-0105.

[4] D. Firesmith, "Common requirements problems, their negative consequences, and the industry best practices to help solve them." *Journal of Object Technology*, vol. 6, no. 1, pp. 17–33, 2007, DOI: 10.5381/jot.2007.6.1.c2.

[5] P. Csaba and L. Ágnes, *Pszicholingvisztika I-II.* Budapest: Akadémiai kiadó, 2014, ISBN: 978-963-05-9499-8.

[6] G. Frege, P. T. Geach, and M. Black, "On concept and object," *Mind*, vol. 60, no. 238, pp. 168–180, 1951, ISSN: 0026-4423.

[7] L. Tóth, B. Nagy, T. Gyimóthy, and L. Vidács, "Why will my question be closed? NLP-based pre-submission predictions of question closing reasons on stack overflow," in *Proceedings of the ACM/IEEE 42nd International Conference on Software Engineering: New Ideas and Emerging Results*. New York, NY, USA: Association for Computing Machinery, 2020, p. 45–48, DOI: 10.1145/3377816.3381733.

[8] D. Correa and A. Sureka, "Fit or unfit: Analysis and prediction of 'closed questions' on stack overflow," in *Proceedings of the First ACM Conference on Online Social Networks*, ser. COSN '13. New York, NY, USA: ACM, 2013, pp. 201–212, DOI: 10.1145/2512938.2512954.

[9] M. A. Hearst, "Automatic acquisition of hyponyms from large text corpora," in *COOLING '92: The 14th International Conference on Computational Linguistics*. USA: Association for Computational Linguistics, 1992, DOI: 10.5555/992133.

[10] J. Seitner, C. Bizer, K. Eckert, S. Faralli, R. Meusel, H. Paulheim, and S. P. Ponzetto, "A large database of hypernymy relations extracted from the web," in *Proceedings of the Tenth International Conference on Language Resources and Evaluation*. Portorož, Slovenia: European Language Resources Association (ELRA), 2016, pp. 360–367.

[11] N. Chomsky and D. W. Lightfoot, *Syntactic Structures*. Berlin, Boston: Walter de Gruyter, 2009, ISBN: 978-3-11-021832-9.

[12] D. Crystal, *A Dictionary of Linguistics and Phonetics, Sixth Edition*. Wiley-Blackwell, 2008, ISBN: 9781444302776.

[13] A. Taylor, M. Marcus, and B. Santorini, "The Penn Treebank: An Overview," in *Treebanks: Building and Using Parsed Corpora*, ser. Text, Speech and Language Technology, Dordrecht, 2003, pp. 5–22, DOI: 10.1007/978-94-010-0201-1_1.

[14] L. Tóth, B. Nagy, T. Gyimóthy, and L. Vidács, "Mining hypernyms semantic relations from stack overflow," in *Proceedings of the IEEE/ACM 42nd International Conference on Software Engineering Workshops*. New York, NY, USA: Association for Computing Machinery, 2020, p. 360–366, DOI: 10.1145/3387940.3392160.

[15] G. A. Miller. Wordnet: A lexical database for english. [Online]. Available: https://wordnet.princeton.edu/ 2021.10.26.

[16] Graph modeling guidelines - developer guides. [Online]. Available: http://neo4j.com/developer/guide-data-modeling/ 2021.10.26.

[17] CommonCrawl website. [Online]. Available: https://commoncrawl.org/ 2021.10.26.

[18] Merriam-Webster dictionary: bellow. [Online]. Available: https://www.merriam-webster.com/dictionary/below/ 2021.10.26.

[19] R. A. Rigby and D. M. Stasinopoulos, "Smooth centile curves for skew and kurtotic data modelled using the Box-Cox power exponential distribution," *Statistics in Medicine*, vol. 23, no. 19, pp. 3053–3076, 2004, DOI: 10.1002/sim.1861.

[20] M. Steyvers and J. B. Tenenbaum, "The large-scale structure of semantic networks: statistical analyses and a model of semantic growth," *Cognitive Science*, vol. 29, no. 1, pp. 41–78, 2005, DOI: 10.1207/s15516709cog2901_3.

[21] A. Ittoo, G. Bouma, L. Maruster, and H. Wortmann, "Extracting meronymy relationships from domain–specific, textual corporate databases," in *Lecture Notes in Computer Sciences*, vol. 6177. Berlin, Heidelberg: Springer, 2010, pp. 48–59, DOI: 10.1007/978-3-642-13881-2.

[22] T. Yıldız, S. Yıldırım, and B. Diri, "A study on turkish meronym extraction using a variety of lexico–syntactic patterns," in *Human Language Technology. Challenges for Computer Science and Linguistics*. Cham: Springer International Publishing, 2016, pp. 386–394, DOI: 10.1007/978-3-319-43808-5_29.

[23] A. Vizcaíno, F. García, M. Piattini, and S. Beecham, "A validated ontology for global software development," *Computer Standards & Interfaces*, vol. 46, no. C, pp. 66–78, may 2016, DOI: 10.1016/j.csi.2016.02.004.

[24] G. Futia, A. Vetrò, and J. C. De Martin, "SeMi: A semantic modeling machine to build knowledge graphs with graph neural networks," *SoftwareX*, vol. 12, p. 100516, Jul. 2020, DOI: 10.1016/j.softx.2020.100516.

[25] D. Falessi and G. Cantone, "The effort savings from using nlp to classify equivalent requirements," *IEEE Software*, vol. 36, no. 1, pp. 48–55, 2019, DOI: 10.1109/MS.2018.2874620.

[26] O. M. Holter, "Semantic parsing of textual requirements," in *The Semantic Web: ESWC 2020 Satellite Events*. Cham: Springer International Publishing, 2020, pp. 240–249, DOI: 10.1007/978-3-030-62327-2_39.

[27] A. Ahmad, C. Feng, M. Khan, A. Khan, A. Ullah, S. Nazir, and A. Tahir, "A systematic literature review on using machine learning algorithms for software requirements identification on Stack Overflow," *Security and Communication Networks*, 2020, dOI: 10.1155/2020/8830683.

[28] T. Li and Z. Chen, "An ontology-based learning approach for automatically classifying security requirements," *Journal of Systems and Software*, vol. 165, p. 110566, 2020, DOI: 10.1016/j.jss.2020.110566.

[29] H. Alrumaih, A. Mirza, and H. Alsalamah, "Domain ontology for requirements classification in requirements engineering context," *IEEE Access*, vol. 8, pp. 89 899–89 908, 2020, DOI: 10.1109/ACCESS.2020.2993838.

 ISBN: 978-1-61208-923-2