future internet

MDPI

Article

Graph-Based Taxonomic Semantic Class Labeling

Tajana Ban Kirigin [1,*], Sanda Bujačić Babić [1] and Benedikt Perak [2]

[1] Faculty of Mathematics, University of Rijeka, R. Matejčić 2, 51000 Rijeka, Croatia
[2] Faculty of Humanities and Social Sciences, University of Rijeka, Sveučilišna avenija 4, 51000 Rijeka, Croatia
* Correspondence: bank@uniri.hr

Abstract: We present a graph-based method for the lexical task of labeling senses of polysemous lexemes. The labeling task aims at generalizing sense features of a lexical item in a corpus using more abstract concepts. In this method, a coordination dependency-based lexical graph is first constructed with clusters of conceptually associated lexemes representing related senses and conceptual domains of a source lexeme. The label abstraction is based on the syntactic patterns of the *x is_a y* dependency relation. For each sense cluster, an additional lexical graph is constructed by extracting label candidates from a corpus and selecting the most prominent *is_a* collocates in the constructed label graph. The obtained label lexemes represent the sense abstraction of the cluster of conceptually associated lexemes. In a similar graph-based procedure, the semantic class representation is validated by constructing a WordNet hypernym relation graph. These additional labels indicate the most appropriate hypernym category of a lexical sense community. The proposed labeling method extracts hierarchically abstract conceptual content and the sense semantic features of the polysemous source lexeme, which can facilitate lexical understanding and build corpus-based taxonomies.

Keywords: lexical graph analysis; corpus; syntactic–semantic dependency; word sense; hypernym; knowledge representation and reasoning

Citation: Ban Kirigin, T.; Bujačić Babić, S.; Perak, B. Graph-Based Taxonomic Semantic Class Labeling. *Future Internet* **2022**, *14*, 383. https://doi.org/10.3390/fi14120383

Academic Editor: Leonilde Varela and Goran D. Putnik

Received: 21 October 2022
Accepted: 14 December 2022
Published: 19 December 2022

1. Introduction

Taxonomic semantic class labeling is a procedure for extracting a common semantic label for a set of conceptually related lexemes. The automation of taxonomic labeling can be exploited in many applications, as it can represent tacit knowledge and infer ontological relations, which is important for various subsequent natural language processing (NLP) tasks and applications [1]. In lexicology, taxonomic semantic labels enable better standardization of lexical descriptions and allow the construction of "pseudo-definitions"—compact descriptions of the lexical conceptual content.

However, the automatic creation of taxonomic labels for a lexeme or a set of lexemes is a non-trivial task with multi-class classification and multi-label classification procedures. One of the major multi-class classification problems of this task is related to the polysemous structure of a lexeme. Namely, an ambiguous word can have multiple senses or acquire new meanings that are often not even semantically related. For instance, the noun *bass* can be labeled as a species of fish or as an instrument.

Moreover, for a set of lexemes representing a semantic class, taxonomic labeling derives a target semantic class label as a result of the categorial relation *is_a* shared with all semantic class members [2]. For instance, the set of source lexemes *banana*, *apple*, *mango*, and *pear* could be labeled as *fruit* or *plants*, while the community of lexemes *banana*, *apple*, *cookie*, *icecream*, and *sandwich* should be labeled as *snack*. Taxonomic semantic class labeling, in this sense, is a kind of generalization of a semantic field determined by the semantic scope of associated lexical units using a more abstract term that can taxonomically represent its semantic class.

From the perspective of cognitive linguistics [3] and cognitive grammar [4], these two lexical phenomena, word disambiguation and taxonomic labeling, can be seen as mutually

supportive tasks. The meaning of a word is always profiled in relation to other words through a process of semantic framing and syntactic–semantic construal [3] (Chapter 2). This means that conceptually similar words tend to form a radial network [3] (Chapter 3), invoked by a certain semantic frame [3] (Chapter 10). Using this theoretical background, we can assume that a source lexeme can be disambiguated by identifying the associated lexical networks that form a semantic field [5]. We can hypothesize that the clustered lexemes can be utilized to label related prototypical semantic frames as categorizations. Additionally, we can suppose that these semantic processes employ particular syntactic constructions.

The Construction Grammar Conceptual Network (ConGraCNet) [5] semantic class labeling method proposed in this paper introduces the procedure to identify clustered lexical fields of a source lexeme node as a semantic class and reveal their taxonomic labels. It is based on the use of syntactic dependencies to analyze information structures in a text. The use of syntactic dependencies is an established approach that has been shown to be effective in a wide range of applications. Syntactic dependencies refer to the relationships between words in a sentence, such as the subject–verb relationship or the direct object–verb relationship. These relationships can be represented using a graph, where each word is represented as a node and the dependencies between words are represented as edges between nodes. One of the main advantages of using syntactic dependencies is that they provide a structured representation of the conceptual relationships between words in a sentence, which can be used to extract valuable information and make inferences about the meaning of the sentence. This is particularly useful in natural language processing tasks such as summarization and information extraction, which involve analyzing the structure of a sentence and identifying the main subjects and objects in a sentence, the manner, time, aspect of the process, etc. From the perspective of lexical semantics, syntactic dependencies can help in disambiguating words and phrases, as we have shown in our previous work on ConGraCNet, making it easier to determine the semantic potential of a word and its categorical structures. Although similar in function, knowledge databases and ontologies are a type of knowledge representation that contains top-down organized logical descriptions of the concepts and relationships in a given domain. Ontologies are typically used in the semantic web to provide a standardized way of representing and reasoning about the meaning of concepts and relationships. However, they do not represent the synchronic or diachronic dynamic knowledge description extracted from the usage. Thus, while ontologies can be used to infer the labeling of concepts and sense communities, syntactic dependency graphs may be more effective for analyzing dynamic, cross-cultural knowledge.

In this paper, we introduce a sequential lexical graph analysis procedure that is employed both for (a) extracting the most fitting conceptual generalizations of a group of related lexemes and for (b) representing the taxonomic label for a semantic class of a given lexeme. In particular, the proposed method exploits the syntactic–semantic dependency relations coordination *and* | *or* and categorization *is_a* and projects their network structure into conceptual similarity and taxonomic relation embedding graph layers, respectively. The proposed approach capitalizes on syntactic–semantic dependency relations and projects them into graph layers to achieve its desired outcomes.

The ConGraCNet coordination-based graph method for semantic association analysis [5] underlying the labeling task, reveals the polysemous nature of lexical concepts by using coordination *and/or* syntactic construction. The obtained graph of associative lexemes for a source lexeme is clustered into subgraphs that reveal the semantic classes of the source lexeme in a corpus. Building from this informational structure of lexical subgraphs, two additional graphs are constructed to capture the taxonomic relations of the subgraph semantic domains: one based on the collocates of the *is_a* dependency relation and the other based on the hypernym relation from available commonsense knowledge bases such as Wordnet [6]. Finally, graph analysis is performed to identify a set of corresponding label candidates representing the abstract semantic features of the associated lexemes in the semantic class. The proposed ConGraCNet multilayered method is an effective way of

semantic analysis that reveals the polysemous nature of lexical concepts and provides a cognitive linguistic approach to understanding the semantic meanings of lexical concepts and their taxonomic relations.

We present examples from different corpora and describe the construction of lexical graphs and the process of candidate selection based on collocation and centrality measures, particularly the Semi-Local Integration (SLI) centrality [7]. The accompanying ConGraCNet web app [8] integrates manually annotated lexicons and semi-automatic corpus resources and methods. The app is available as a semantic resource in the EmoCNet project [9].

The aim of this paper is to (i) present a graph-based method for labeling the taxonomic semantic class of associative concepts; (ii) present a semantic resource for web applications that integrates manually annotated lexicons and semi-automatic corpus techniques. We make the following contributions:

- A graph-based approach to the lexical task of labeling semantic classes, which are formed as associative lexical communities related to a polysemous lexeme, using a combination of coordination and *is_a* syntactic–semantic lexical relations;
- A related graph-based method that incorporates the knowledge base with a built-in hierarchical structure, such as the WordNet hypernym relation, to assign hypernym labels to semantic classes;
- A web app implementation of the graph-based labeling algorithm.

Relying on the graph clustering and centrality measures from multiple syntactic embedding layers, we demonstrate the relevance of the proposed method, showing how and why it can be used to modify the taxonomic labeling outcome. The "why question" is especially relevant in the context of understanding the underlying cognitive processes that take place while creating abstract categorization of conceptual content. Additionally, the explainable nature of the procedure is in contrast to the growing number of computationally superior black-box models that produce non-transparent results without the possibility to explain the complex algorithmic processes. That being said, we also envisage the possibility of engaging the syntactic-dependency graph procedures in computationally extensive methods such as graph neural networks (GNN), with the possible benefit of creating semantically more transparent and controllable procedures for taxonomic labeling.

Moreover, our method can be produced on a relatively smaller set of corpora, as it is specially designed to be able to infer and represent the specificity of the corpora. We also show how the proposed method can be used to compare corpora and identify the differences in their taxonomic labeling, as well as to analyze the relative composition of the taxonomic labels for each lexeme in a given corpus. The results demonstrate that the proposed method is able to identify the most fitting taxonomic labels for a given lexeme and, more importantly, to identify the relevance of a given label in a given context. The paper begins with a description of related work, Section 2. The labeling method is described in Section 3 and then discussed in Section 4, where we also define future research directions. Conclusions are drawn in Section 5.

Relation to Our Previous Work

Some of the early research on labeling associative lexical communities was covered in [10] and in the conference paper [11]. Label graph construction and label assignment have been considerably elaborated by the selection of dependencies and the application of the SLI measure [7]. As explained in detail in Section 3, both the weights assigned to the edges in the label graph and the identification of the most appropriate label candidates reflect multiple underlying graph-theoretic features, including the collocation measures of the syntactic–semantic dependency relations *and/or* and *is_a*, as well as the centrality properties of the graphs modeling the lexical sense communities.

2. Related Work

Models for the computational representation of lexical–semantic knowledge have their origins in methods and resources that can be broadly divided into two categories.

Top-down curated knowledge databases such as WordNet [6] and its counterparts in other languages [12] form the basis for computational lexicons and the contextualization of paradigmatic hypernymy relations. WordNet lexical synsets, and later VerbNet [13], PropBank [14], BabelNet [15], and VerbAtlas [16] encode lexical semantic knowledge using word senses as units of meaning. A major problem with Wordnet is the top-down structure of curated resource creation, which inevitably leads to less granularity and the static nature of the inventories.

This class of resources also includes Common-Sense Knowledge (CSK) databases, which store descriptions of a set of common and generic facts or views about a set of concepts, including *is_a* relations. They describe the general information that people use to describe, differentiate, and reason about concepts. ConceptNet [17,18] is one of the largest such resources, incorporating data from the original MIT Open Mind Common Sense project. Unlike WordNet, which distinguishes the meanings of a given lemma, the terms in ConceptNet are ambiguous, which can lead to confusion in lexical–semantic hypernym relationships for concepts denoted by ambiguous words (e.g., *bass* as an instrument vs. a species of fish).

On the other hand, bottom-up approaches to semantic labeling rely on the extraction of semantic features from the idea that conceptually similar words are used in syntagmatic similar contexts [19] and the use of corpus-based syntactic pattern analysis [20,21].

The underlying idea is to analyze the prototypical syntagmatic patterns of words used in large corpora and assign meaning on a contextual basis through prototypical sentence patterns. Automatic approaches such as those of [22–24] use syntactic patterns for automatic extraction of concept descriptions.

A radically different bottom-up approach is based on vector space models of lexical representations that view concepts as geometric vectors whose dimensions are qualitative features [25] and other similar methods such as Latent Semantic Analysis [26], Latent Dirichlet Allocation [27], embeddings of words [28–30], and word senses [31–33]. Most of the recent approaches have been modified with the introduction of the open-source bidirectional machine learning framework that uses the surrounding text to determine the context of words. These models allow for direct similarity computation, but the knowledge does not explicitly define the concepts and the relations between vector representations are not ontologically organized. Moreover, there is growing concern about the capacity of neural networks to be understood in a transparent way. As automated machine learning models become increasingly complex, many people are anxious about their use and believe it is necessary to create models that can easily be interpreted or explained [34–36]. In this respect, there are also some efforts to extract tacit human knowledge [37–41].

Finally, there are mixed methods that use resources and methods from different approaches, such as the semagram-based knowledge model, which consists of 26 semantic relations and integrates features from different sources [42], or the multilingual label propagation scheme introduced in [43], which leverages word embeddings and the multilingual information from a knowledge database. Additionally, an interesting unsupervised procedure with the aim to reconstruct associative knowledge and emotional profiles in the text is the Textual Forma Mentis Network (TFMN), which utilizes syntactic parsing and psychological cognitive reflection to uncover how people structure and perceive knowledge with a combination of network science, psycholinguistics, and Big Data [44].

In our earlier work [5,10], we introduced the graph method for distinguishing lexical senses, theoretically based on the notions of cognitive construction grammar that syntactic constructions construe a semantic value invoked by a semantic frame, and practically designed using a graph-based analysis of syntactic dependencies within a large corpus. As part of the ConGraCNet application developed to integrate data from various NLP pipelines, lexical dictionaries, and sentiment dictionaries, the method has shown perspective results, for example in the study of linguistic emotion expressions and the conceptual analysis of cultural framings [10,45–48]. This procedure is the basis of the taxonomic lexical labeling procedure introduced in this article.

3. Graph-Based Taxonomic Semantic Associative Community Labeling

In this section, the graph-based method for taxonomic labeling of semantic classes that represent lexical senses is described in more detail. The selection of label candidates representing the abstract taxonomic lexical sense is based on two specific syntactic patterns: (1) associative semantic features of the *and/or* coordination to identify the related semantic class members of the source lexeme, and (2) taxonomic categorial features of the *is_a* relation to identify taxonomic semantic label candidates. In addition, the hypernym relation can be included from available commonsense knowledge bases such as Wordnet to select the best hypernym candidates for each associated sense. The dependency graph structures are analyzed to identify the relevant senses with corresponding taxonomic semantic classes and hypernym labels.

3.1. Graph-Theoretical Foundations

The sequential procedure for identifying the relevant semantic classes of a source lexeme starts with the ConGraCNet method to obtain the lexical coordination graph with clusters and computed SLI centrality measures, followed by the construction of the taxonomic label candidate graph for each of the clusters in the ConGraCNet graph. The concepts of SLI measure and the ConGraCNet method are briefly presented below.

3.1.1. Semi-Local Integration Measure

For the task of analyzing the common semantic features of lexeme nodes in a lexical graph, the most important and representative nodes are those that are strongly integrated into the graph clusters, i.e., those that are closely related to many other nodes in the network structure. Among numerous centrality measures that compute various network properties, the measure that best evaluates the above graph features is the SLI measure introduced in [7]. The SLI node importance effectively identifies and evaluates the degree of integration of a node with other nodes in the clusters of a complex network. Hence, the SLI node centrality values for nodes in the subgraphs are integrated into the procedure of semantic class label propagation.

For better readability, we provide a brief explanation of the SLI definition for undirected and weighted graphs. Given a graph $G = (V, E)$ with the set of nodes V and the set of edges E, let E_a denote the set of all immediate neighbours of a,

$$E_a := \{x \mid e_{ax} \in E\}.$$

Both the *weighted degree* (*strength*) of the nodes, denoted with $d^w(x)$, and the weight of edges, denoted with $w(e)$, play a part in the SLI calculation. Another important piece of information participating in the SLI calculation is the cycle basis of the graph G, denoted by $P(G)$. The number of simple cycles containing the edge e, denoted by $p(e)$, contributes to the importance of nodes incident to e. The higher the number of cycles in the local subgraph that include the edge, the higher the integration in the cluster.

The assignment of the SLI importance of a node $a \in V$ starts with the calculation of its importance score $I(a)$:

$$I(a) := d^w(a) + \sum_{e_{ab} \in E_a} I(e_{ab}), \tag{1}$$

where $I(e_{ab})$ is the importance score of the edge between nodes a and b. That is, to emphasize the interconnection of the node a in the local subgraph, the *weighted degree* of a is boosted by the edge importance score of all of its incident edges, i.e., by the strength of its connection to the other nodes in the neighborhood. The edge importance score, $I(e_{ab})$, is defined as:

$$I(e_{ab}) := \lambda(e) \cdot \left(d^w(a) + d^w(b) - 2w(e)\right) \cdot \frac{w(e) \cdot d^w(a)}{d^w(a) + d^w(b)}, \tag{2}$$

where the edge cycle factor of the edge $e \in E$ is given by:

$$\lambda(e) := p(e) + 1. \tag{3}$$

Finally, for easier interpretability, the node importance score $I(a)$ is normalized, representing the corresponding percentile of the overall node importance in the graph, i.e.,

$$SLI\,(a) := \frac{I(a)}{S_G} \cdot 100, \tag{4}$$

where S_G is the sum of the importance scores of all nodes of G:

$$S_G := \sum_{b \in V} I(b).$$

For more details and a computational representation of the SLI formula, see [7]. Additionally, the Python function that implements the SLI measure is open-source and available in the GitHub repository [49].

3.1.2. ConGraCNet Method

The ConGraCNet method [5] relies on a set of syntactic relations used to construct dependency-based multilayer lexical networks. Each layer is constructed from lexemes collocated in a syntactic dependency that can be harnessed for their semantic potential and function [45,50]. The basic coordination network layer is constructed from collocated lexemes in the syntactic dependency *and/or*, which typically associates two ontologically related entities, attributes, or processes. The highest-ranked co-occurrences of the seed lexeme in the coordinated second-order construction [LEXEME_A and/or LEXEME_B] are used to build and analyze a graph of syntactically collocated lexemes. They form conceptually associated local clusters, or a second-order friend-of-a-friend, lexical graph with subgraph communities of conceptually associated lexemes, FoF$[and|or]_s$, representing the conceptual domains related to the source lexeme s. The semantically coherent lexical clusters form the basis for dependency-based labeling. More details on the method, including the relevant theoretical grounding, can be found in [5].

For an example of FoF$[and|or]_{ethics}$ graph see Figure 1. Note the various identified subgraphs exhibiting various conceptual domains closely related to the meaning of the lexeme *ethics* as expressed in the enTenTen13 corpus. In particular, the community of famous philosophers is clustered around the lexemes *Aristotele* and *aristotle*.

For a selected polysemous source lexeme s, the ConGraCNet method first constructs a weighted undirected clustered graph FoF$[and|or]_s$ and computes the centrality of its lexeme nodes. Once the algorithm provides the graph structure with subgraph communities, the ground has been set for the task of assigning labels to each subgraph community.

An important step in the labeling process is to identify the most important nodes of each subnetwork FoF$[and|or]_i^s$, as these nodes carry the most representative community features. Since the SLI measure detects the level of node integration in the network, it plays an important role in the subsequent task of weighting semantic class labels, both for syntactic *is_a* dependency labeling and hypernym labeling.

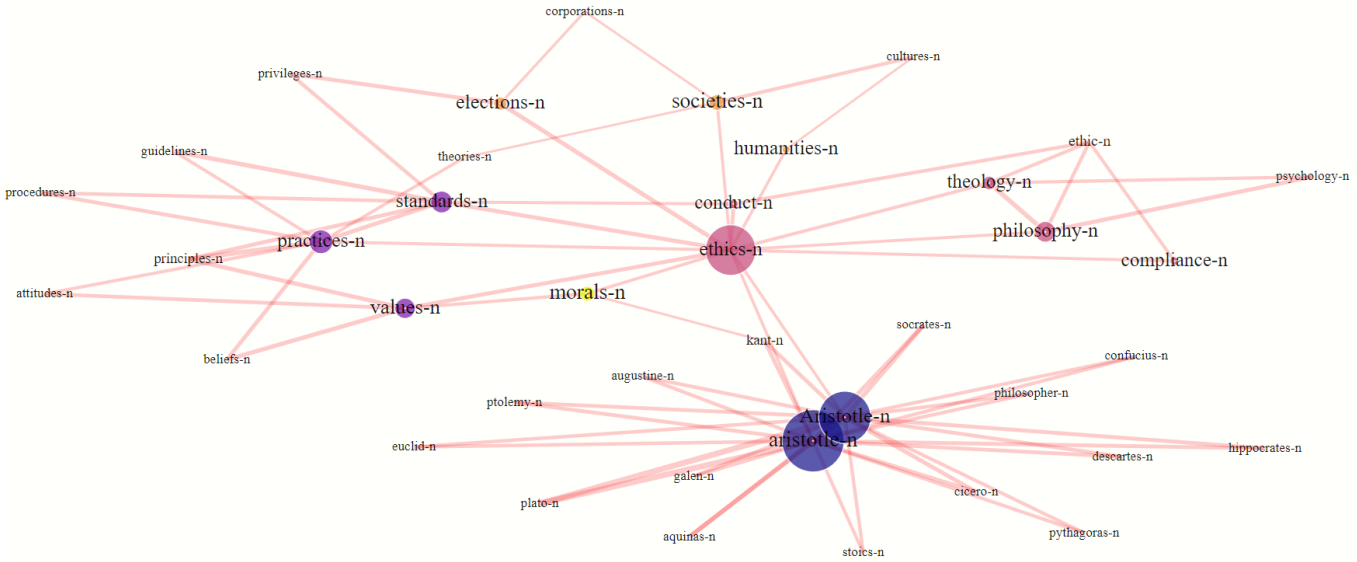

Figure 1. FoF$[and|or]_{ethics}$: ConGraCNet graph representation of the sense structure of a source noun lexeme *ethics* in enTenTen13.

3.2. Dependency-Based Taxonomic Semantic Class Labeling

The taxonomic labeling method described in this subsection does not rely on a commonsense knowledge database as an outside lexical resource. It is based on a corpus dependency-based graph computational linguistic approach. This graph method labels different semantic classes of a source lexeme based on the evaluation of syntactic–semantic collocations of the *is_a* relation in a corpus.

The idea behind our approach is that the conceptual taxonomic structure, i.e., the paradigmatic lexical relation hypernym/hyponym, is expressed in the linguistic usage by nominal predicates and copula structures, for example "car is a vehicle" or "bass is a fish". Conversely, we can implore unsupervised graph-based algorithms to extract the taxonomic hyponym–hypernym candidates from the syntactically tagged corpus analysis of a *is_a* syntactic dependency.

Taxonomic semantic label candidates for a community of associated lexemes are selected based on syntactic–semantic *is_a* dependency relations extracted from a syntactically tagged corpus. The highest-ranked lexical collocates in the community are prototypical taxonomic representatives of the semantic information people use to describe, differentiate, and reason about these associated concepts.

The syntactic dependency *is_a* typically establishes a semantic association between two lexemes, one being a taxonomic subclass of the other. This type of association has a categorical semantic function. However, *is_a* is also used in metaphorical expressions, such as "Love is a fire that consumes lovers", "Reading is a way to spend the time", or "She is an object of desire", where the target argument metaphorically maps the content of the conceptual domain onto the source domain. This metaphorical type of association produces noise in the taxonomic labeling task. The proposed procedure to extract the metaphorical from categorical taxonomic collocates uses the fact that the source and target domain in the metaphorical mapping have larger ontological differences than the categorical type of association [51]. In this way, we propose an algorithm that not only identifies the taxonomic labels for the source lexeme, but also serves to filter the lexemes used for the prototypical metaphorical mappings.

The *is_a* relation can be expressed by a copula relation, which is used to link a subject to a nonverbal predicate. The copula is often a verb, but nonverbal (pronominal) copulas

are also common in many languages of the world. In this study, we rely on a set of predetermined syntactic patterns that use the *is_a* syntactic–semantic lexical relations defined by the Word Sketch grammars, classified as "w% is a…" and "… is a w%" [52].

The data for the construction of the *is_a* network represented in this article are obtained from the English Web corpora [53] with preprocessed "[word] *is_a*" dependency.

The extraction of the *is_a* collocates is performed in both directions to capture the widest array of taxonomic relations, such as the relation of *banana–fruit* in the examples "A banana is a fruit" and "This fruit is a banana". That is, for a given lexeme $_X$ we extract its [LEXEME$_X$ is_a LEXEME$_Y$] and its [LEXEME$_Y$ is_a LEXEME$_X$] collocates from the corpus. The construction of the directed lexical network based on the *is_a* collocations allows us to formalize a syntactically based categorical labeling method that takes into account a broader range of semantic features of the *is_a* patterns, including the hypernym and hyponym aspects of the relations.

3.2.1. Dependency-Based Labeling: Procedure Description

The ConGraCNet graph construction procedure of the *is_a*-type label graph for an associated lexical cluster consisting of the following steps:

(i) For each lexeme node c_j in the clustered coordination community FoF$[and|or]_i^s$ with lexemes $c_1, \ldots, c_n$, a number, k, of the top-rated collocates in the
[LEXEME$_{c_j}$ is_a LEXEME$_h$] and [LEXEME$_h$ is_a LEXEME$_{c_j}$]
relations is identified from a selected corpus;
(ii) A weighted, first-order (source-friend) directed graph $F[is\_a]_i^s$ of identified lexical *is_a* collocates of the clustered coordination community FoF$[and|or]_i^s$ is constructed;
(iii) The most prominent in nodes in the $F[is\_a]_i^s$ graph are identified.

The obtained graph is used to predict the categorically abstract label of the corresponding lexical cluster from the list of most central lexical nodes in the label graph. We now describe the procedure in detail.

Step (i) Given a coordination lexical graph FoF$[and|or]_s$ and its clustered coordination community FoF$[and|or]_i^s$ with lexemes $c_1, \ldots, c_n$, a corpus C and a parameter k, we extract the collocates, h, of each of the lexemes c_j on the basis of the *is_a* syntactic–semantic dependency:

$$H_j := \{\, h \mid [c_j \text{ is\_a } h] \text{ appears in } C\} \cup \{\, h \mid [h \text{ is\_a } c_j] \text{ appears in } C\},\ j \in \{1, \ldots, n\}.$$

That is, to each lexeme c_j from the lexical subgraph FoF$[and|or]_i^s$, whose semantic class is to be labeled, we associate the set of its syntactic–semantic *is_a*-collocates, H_j. The elements of the set H_j are determined using both directions of the *is_a* relation, as justified in the discussion above. The selection of elements of H_j depends on the chosen corpus C since they are extracted as collocates of c_j from the text of C.

Then, from the set H_j, a number of most frequent collocates of c_j occurring in the text of C are selected as elements of H_j^k. The natural number k is a parameter of the procedure, providing a potentially different graph structure for label selection. However, since we weight the label network, transferring the graph-structure properties from the underlying FoF$[and|or]^s$ graph onto the label graph, the most relevant nodes will emerge as the best candidates already for a reasonably low parameter k, such as, e.g., $k = 15$.

Using a standard collocation measure, f, e.g., *logDice* or *frequency*, we filter the k highest-ranked elements of each set H_j, to obtain $H_j^k \subseteq H_j$, the sets of k strongest *is_a*-syntactic–semantic dependency collocates of each of c_j in corpus C. The selected lexemes that form the sets H_j^k, $j \in \{1, \ldots, n\}$, are the candidates for labels of the community FoF$[and|or]_i^s$. Namely, they will feature as nodes in the label graph constructed and analyzed in the next steps.

Step (ii) We construct a label graph that incorporates the nodes of the clustered coordination community FoF$[and|or]_i^s$ along with the strongest corresponding *is_a* collocates obtained in the previous step. The label graph $F[is\_a]_i^s = G_L = (V_L, E_L)$ is then defined as a weighted directed graph. More precisely, the set of its nodes is the set of community lexeme nodes $c_1, \ldots, c_n$, and the elements of each H_j^k, i.e., k top-ranked *is_a* collocates of each lexeme c_j in the lexical subgraph:

$$V_L := \{c_1, \ldots, c_n\} \cup H_1^k \cup \ldots \cup H_n^k.$$

The edges of G_L and their weights are defined according to the chosen collocation measure f and the selection performed:

$$E_L := \bigcup_{j=1}^{n} \{ (c_j, h) \mid h \in H_j^k \}, \quad w(e) := f(e),\ e \in E_L.$$

The set of edges E_L is obtained as the union of the edges incident to each of the lexemes c_j, from the clustered coordination community FoF$[and|or]_j^s$. In this way, for each lexeme c_j in the lexical subgraph, we include the edges (c_j, h) which point from the lexeme c_j to its collocate h, according to the direction of the *is_a* syntactic–semantic relation. In addition, the edge weights reflect the lexical graph properties and indicate which links between nodes in the graph G_L are more important than others according to the collocation measure f.

Step (iii) Community labels are selected among the in nodes of the label graph G_L. For this purpose, the label importance score, I_L, is defined for the in nodes of G_L, h, taking into account their weighted in degree and the SLI centrality score of the corresponding out node in the FoF$[and|or]_s$ graph:

$$I_L(h) := \sum_{\substack{e \in E_L \\ e=(c,h)}} w(c,h) \cdot SLI(c)^2 = \sum_{(c,h) \in E_L} f(h) \cdot SLI(c)^2.$$

According to the above definition, all directed edges $e = (c, h)$ from the graph G_L, $e \in E_L$, pointing to the lexeme h, contribute to the label importance score of the lexeme node h. This is achieved through the corresponding edge weights. The higher the total weight of the incoming edges, the higher the importance of the label node. Furthermore, we take into account the centrality of the corresponding outgoing nodes. For each lexeme c incident to the h we multiply the edge weight $w(c, h)$ with $SLI(c)^2$, strongly emphasizing the importance of the lexeme node c in the graph representation of the lexical community FoF$[and|or]^s$.

Since SLI indicates the strength of integration of the nodes into the graph clusters, we use it as a crucial tool to highlight the best label candidates in our labeling task. To strongly emphasize the influence of the most integrated nodes in the community, the squared values of the SLI scores feature in the related label importance score. In this way, the collocated nodes that are highly integrated into the subnetwork structure, receive high label importance scores and are therefore the best candidates for label selection.

The lexeme nodes with the highest label importance score, I_L, typically one of two lexemes, are selected as labels of the given community.

Note that, as shown in Figure 2, some of the lexemes c_j from the coordination community may share some of the selected *is_a* collocates, i.e., target the same lexeme node. Therefore, the sets H_j^k are not necessarily pair-wise disjoint. Note also that some of the lexemes c_j may themselves appear as elements of H_j^k sets, i.e., lexeme nodes from the coordination community may appear as label candidates.

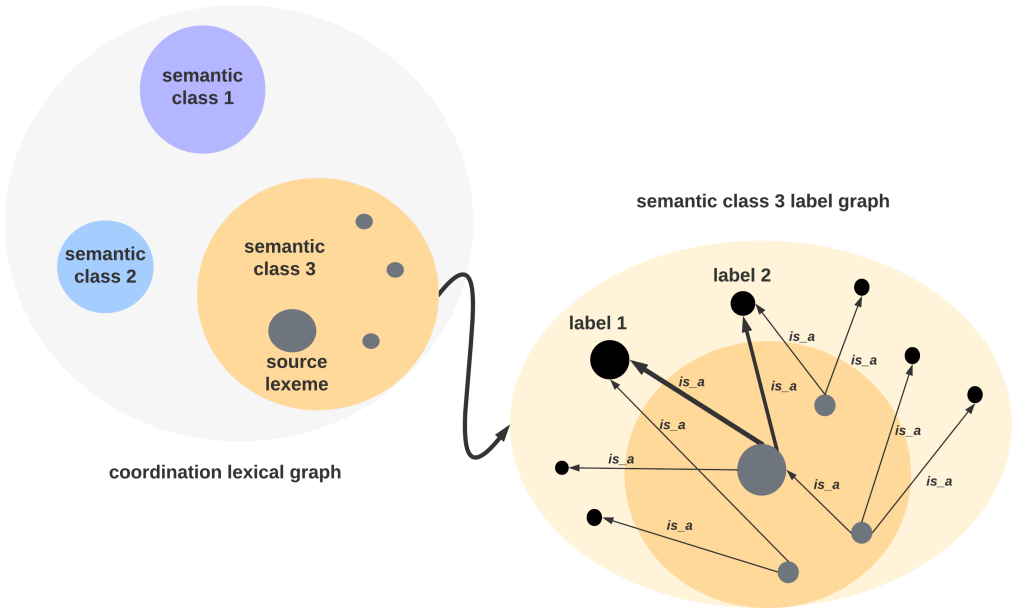

Figure 2. Schematic illustration of the taxonomic semantic class labeling propagation procedure associated with the coordination lexical graph.

The sequential nature of the procedure allows a parametrization of the label selection with regard to the lexeme nodes from the underlying semantic fields. One of the options is to exclude lexemes that are not in the FoF$[and|or]_s$ coordination graph. This restriction allows a more meronymic (part–whole) relationship between the labels. Another option is to exclude lexemes of a coordination cluster FoF$[and|or]_i^s$, designated for taxonomic labeling, to appear as label candidates. In this way, the label is not chosen from a set of expressed lexemes in a semantic field. A discussion of the theoretical justifications for these two parameters may be found in Section 4.

We point out that, compared to the dependency-based labeling introduced in our earlier work [11], additional graph-theoretic tools are used to map the graph-structure properties of the lexical networks onto the semantic features of the corresponding sense clusters. In addition to the corpus *is_a*-collocation measure, we also carry over the integration SLI centrality from the coordination lexical networks representing the sense communities when constructing the label graph. Such a stronger projection of the lexical graph structure, highlighting the importance of the most integrated nodes in the graph and, thus, their corresponding label candidates, enhances the quality of the semantic label extraction.

3.2.2. Dependency-Based Labeling: Implementation

The above procedure of labeling is implemented in ConGraCNet web app [8] with the following default parameters: We set the default value to $k = 50$ and $frequency$ as the default corpus collocation measure. The first two labels are selected. Only lexemes that occur in the FoF$[and|or]_s$ graph are taken as label candidates.

The second-degree coordination-based graph FoF$[and|or]_s$ is constructed using the default settings: enTenTen13 corpus, n = 15 best-ranked coordination collocations in the first- and second-degree networks, and with pruning removing nodes with degree less than 2 and with betweenness centrality in the bottom 50%. Clustering is performed using the Leiden community algorithm [54,54] with mvp partition type clustering.

Graph calculations are performed using the Python library iGraph [55]. For text input, we used the pipeline that collects syntactic dependency data from the morpho-syntactically tagged corpora Sketch Engine API [56,57]. The Sketch Engine API was used to extract a summary of various syntactic dependency co-occurrence data for each lemma.

To illustrate dependency-based taxonomic semantic class labeling, consider the example analysis of the lexeme *ethics*. The label graph of the source lexeme *ethics-n* $F[is\_a]_i^{ethics}$, constructed using the large morpho-syntactically tagged English corpus enTenTen13, shows five associated lexical communities $\text{FoF}[and|or]_i^{ethics}$, as illustrated in Figure 1.

The source lexeme *ethics* is clustered in the third semantic community $\text{FoF}[and|or]_3^{ethics}$: *ethics-n, conduct-n, theology-n, compliance-n, philosophy-n, ethic-n, psychology-n*. The first cluster of associated lexemes $\text{FoF}[and|or]_1^{ethics}$ refers to the semantic class of famous philosophers. The corresponding *is_a* dependency label graph $F[is\_a]_1^{ethics}$ is shown in Figure 3.

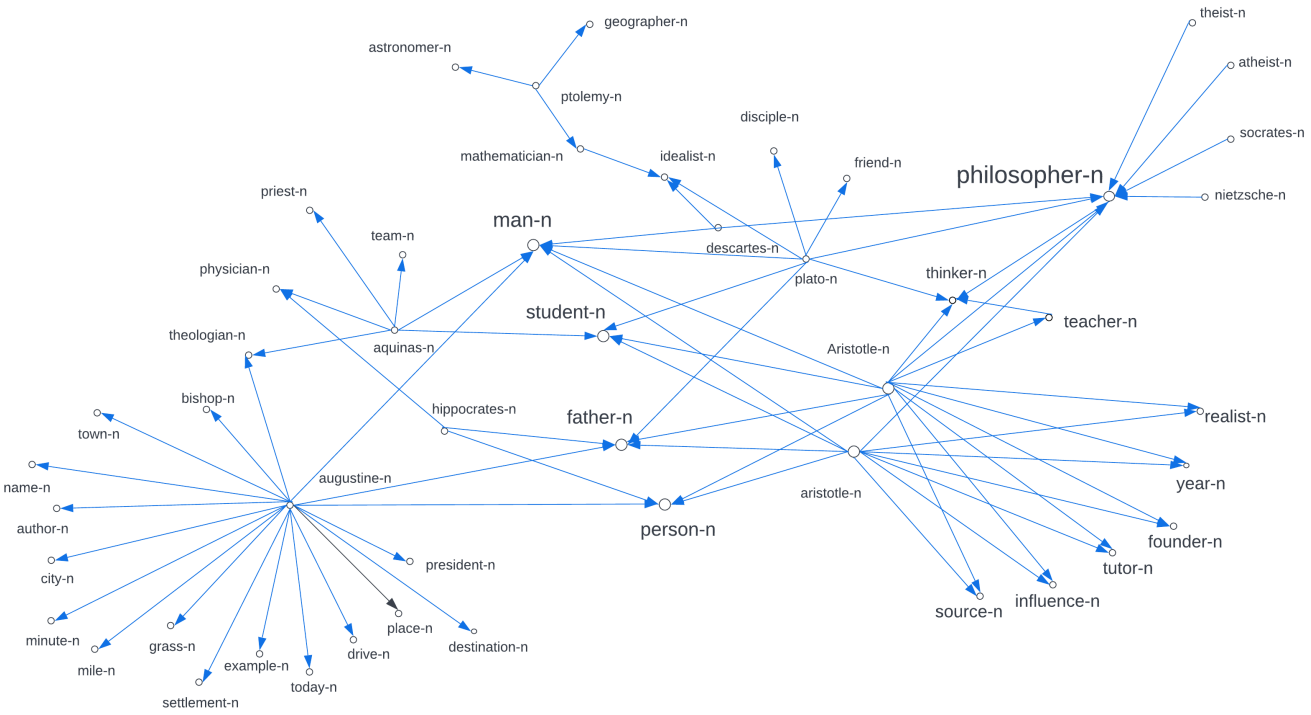

Figure 3. Semantic class labeling of $\text{FoF}[and|or]_1^{ethics}$: *is_a* dependency label graph $F[is\_a]_1^{ethics}$ for associated community 1 of the ConGraCNet graph of the source lexeme *ethics-n*, FoF_{ethics}, in enTenTen13.

The label graph $F[is\_a]_3^{ethics}$ is populated with lexemes that are *is_a* collocates of lexemes from the third community, e.g., *plato, Aristotle, aristotle, theist-n, atheist-n, nietzche-n* all point to the lexeme *philosopher-n*. The leaf nodes in the graph are populated with lexemes from diverse conceptual domains, e.g., *priest, grass, town, year, friend*. These lexemes will have less impact on the label importance score.

For simplicity, in the illustration in Figure 3, neither the weights of the edges representing *is_a* relation strength nor the out node *SLI* measure are displayed. However, the out nodes that are more important according to the *SLI* scores will project more priority to their collocates in the label selection summarization. Since, for example, the lexeme node *Aristotle* has the highest *SLI* score among all the nodes in the corresponding community, its collocates will gain high priority as label candidates. Among them, some will gain additional weight from other important lexeme nodes and high weight from their respective edges.

Since the labeling procedure is based on the presence of categorical relations in a corpus, if a lexeme from the semantic class does not have the relation *is_a*, the node can be omitted during processing. For example: *cicero-n, galen-n, euclid-n, pythagoras-n, confucius-n*, and *stoics-n* are not represented in the label graph.

All identified graph communities representing semantic classes related to the lexeme *ethics* and the corresponding labels are listed in Table 1. Each of the associated communities representing semantic fields of a source lexeme *ethics-n* is abstracted with two taxonomic labels. The column containing only candidates from FoF_{ethics} nodes is the most conservative, providing the kind of taxonomic labels with meronymic relation identified in the FoF_{ethics} network. The column without FoF_{ethics} membership restriction shows a slightly more abstract type of label. For example, the label candidates for the $\text{FoF}[and|or]_5^{ethics}$ semantic community are *philosopher* and *rationalist*, as opposed to the label *kant-n* suggested by the FoF membership enforcement labeling procedure.

Table 1. Best-ranked *is_a* dependency labels and WordNet hypernym labels for associated communities $\text{FoF}[and|or]_i^{ethics}$ of the source lexeme *ethics-n* in enTenTen13.

	Associated Community	***is_a* Labels from $\text{FoF}_{\text{ethics}}$**	***is_a* Labels**	**WordNet Hypernym Labels**
1	aristotle-n, Aristotle-n, plato-n, aquinas-n, socrates-n, cicero-n, hippocrates-n, descartes-n, ptolemy-n, augustine-n, galen-n, euclid-n, pythagoras-n, confucius-n, philosopher-n, stoics-n	philosopher-n, socrates-n	philosopher-n, man-n	linear_unit.n.01, dynasty.n.01
2	standards-n, values-n, practices-n, guidelines-n, principles-n, procedures-n, beliefs-n, attitudes-n	values-n, standards-n	set-n, standard-n	belief.n.01, values.n.01
3	ethics-n, conduct-n, theology-n, compliance-n, philosophy-n, ethic-n, psychology-n	science-n, philosophy-n	study-n, science-n	motivation.n.01, philosophy.n.01
4	elections-n, societies-n, humanities-n, privileges-n, corporations-n, cultures-n	societies-n, cultures-n	member-n, organization-n	discipline.n.01, humanistic_discipline.n.01
5	morals-n, kant-n	philosopher-n, kant-n	philosopher-n, rationalist-n	motivation.n.01, ethical_motive.n.01

3.3. WordNet Hypernym Labeling

Another approach to identify taxonomic label candidates for a semantic class is to use a hypernym relation from a lexical knowledge database such as WordNet. The proposed WordNet hypernym labeling is based on the construction of a hypernym graph using WordNet synsets and hypernym relation:

$$\text{(lexeme)-[has\_synset]} \rightarrow \text{(synset)-[has\_hypernym]} \rightarrow \text{(hypernym\_synset)},$$

as described in [10].

In order to label a polysemous lexeme s, the most appropriate hypernym synset labels can be constructed for each lexical cluster in the $\text{FoF}[and|or]$ graph of the source lexeme.

3.3.1. WordNet Hypernym Labeling: Procedure Description

Starting from the clustered coordination graph $\text{FoF}[and|or]_i^s$ of the source lexeme s, a WordNet hypernym label graph $F[hyper]_i^s$ is constructed for each of its subgraphs. The procedure for the construction of the hypernym graph consists of the following steps:

(i) For each lexeme node c_j in the clustered coordination community $\text{FoF}[and|or]_i^s$ with lexemes $c_1, \ldots, c_n$, synsets t are identified on the basis of the WordNet synset relation:

$$(\text{lexeme}_{c_j})\text{-[has\_synset]} \rightarrow (\text{synset}_t);$$

(ii) For each of the above synsets t, WordNet hypernyms h are extracted through the WordNet hypernym relation

$$(\text{synset}_t)\text{-[has\_hypernym]} \rightarrow (\text{hypernym\_synset}_h);$$

(iii) A weighted, first-order (source-friend) directed graph $F[hyper]_i^s$ of identified lexical WordNet synset/hypernym collocates of the clustered coordination community FoF$[and|or]_i^s$ is constructed;

(iv) The most prominent in nodes in the hypernym graph $F[hyper]_i^s$ are identified.

More precisely, the hypernym label graph $F[hyper]_i^s$ is constructed as follows.

Given a clustered coordination community FoF$[and|or]_i^s$ with lexemes $c_1, \ldots, c_n$ from the coordination lexical graph FoF$[and|or]_s$, the corresponding synsets t are extracted from WordNet. Together with the lexemes and the corresponding synsets from the first step, a set of WordNet hypernym synsets is assembled.

The hypernym label graph $F[hyper]_i^s = G_H = (V_H, E_H)$ is then defined as the following weighted directed graph. The set of nodes V_H is the set of the community lexeme nodes and the corresponding hypernym lexemes extracted from WordNet:

$$\begin{aligned} V_H &:= \{c_1, \ldots, c_n\} \cup \{h \mid \exists (x,h) \in H\}, \qquad \text{where} \\ H &:= \bigcup_{j=1}^{n} \{\, (c_j, h) \mid \exists t \,(t \text{ is a synset of } c_j, h \text{ is a hypernym of } t)\}. \end{aligned}$$

The set H is a set of directed edges, pointing from the lexeme nodes to their hypernyms, identified through the corresponding synsets t are extracted from WordNet through its (lexeme_{c_j})-[has_synset] $\rightarrow (\text{synset}_t)$ and (synset_t)-[has_hypernym] $\rightarrow (\text{hypernym\_synset}_h)$ relations. This identifies the hypernyms corresponding to the lexemes from the sense community represented with FoF$[and|or]_i^s$, as the in nodes of the edges from H. In this way, the corresponding WordNet hypernyms are identified compositionally through WordNet synset categories of each of the lexemes c_j based on the

$$(\text{lexeme}_{c_j})\text{-[has\_synset]} \rightarrow \text{is}\,(\text{synset}_t) \rightarrow (\text{hypernym\_synset}_h)$$

WordNet relation.

The edges of G_H and their weights are defined according to the SLI measure of the out nodes in the FoF$_s$ graph:

$$E_H := H, \quad w(c,h) := SLI(c)^2,\ (c,h) \in E_H.$$

The *weighted in degree* of the hypernym nodes provides the summarization of the integration projection from the nodes of community FoF$[and|or]_i^s$. The in nodes with the highest *weighted in degree* are selected as hypernym labels, typically one or two labels.

Note that, differently from the approach in [10], the hypernym graph $F[hyper]_i^s$ is weighted according to the SLI scores of the lexemes from the coordination community FoF$[and|or]_i^s$. Similarly to the dependency-based labeling described in Section 3.2, the integration level of the nodes in the subgraph is projected to label candidates so that the labels of the more integrated nodes receive higher priority.

3.3.2. WordNet Hypernym Labeling: Implementation

The procedure of WordNet hypernym labeling is implemented in ConGraCNet web app [8]. According to the *weighted in degree*, the first two labels are selected.

The assignment of hypernym labels for the related senses of the source lexeme *ethics-n*, is listed in Table 1. For example, lexemes nodes in the hypernym graph for community FoF$[and|or]_3^{ethics}$ of the source lexeme *ethics*, $F[hyper]_3^{ethics}$, include the lexemes *motivation.n.01, philosophy.n.01, belief.n.01, system.n.04, learned_profession.n.01, discipline.n.01*.

Since the lexeme *ethics* has the highest *SLI* centrality score among the lexemes in the community, its synset hypernyms, *motivation.n.01, philosophy.n.01*, have received the most significant label importance through the corresponding *weighted in degree*, and have, therefore, been recognized as labels of the corresponding semantic class.

4. Results and Discussion

In the previous sections, we have shown that dependency-based lexical graph analysis can be used to construct a taxonomic structure of polysemous lexemes. This unsupervised graph method depends on a number of corpus linguistic and graph analysis parameters, including the selection of a corpus, the extraction of syntactic dependencies, and the measures used for ranking label candidates.

In particular, the proposed method exploits the syntactic–semantic dependency relations coordination *and* | *or* and categorization *is_a* and projects their network structure into conceptual similarity and taxonomic relation embedding graph layers, respectively.

The results show that the proposed method is able to identify the most appropriate taxonomic labels for a given lexeme and, more importantly, to determine the relevance of a given label in a given context. We demonstrate that the proposed method is not a black-box technique, as we have shown how graph clustering and centrality measures from multiple embedding layers can be used to change the result of taxonomic labeling in restrictive and non-restrictive settings. Moreover, our method can be produced on a relatively smaller set of corpora, as it is specifically designed to infer and represent the specificity of the corpora. In this section, we also show how the proposed method can be used to compare two corpora and identify the differences in their taxonomic labeling, as well as analyze the relative composition of taxonomic labels for each lexeme in a given corpus.

Expanding our previous work on taxonomic labeling with the enhanced multilayered graph procedure, we demonstrate how the *SLI* centrality measure can be used to emphasize the integration of nodes in the graph for the propagation of the most appropriate labels.

This approach can be very useful in avoiding the propagation of erroneous labels from noisy data. Additionally, the graph-based method provides an efficient solution for the identification of the most suitable labels for an entire taxonomic tree, which is often a complex task. This method can also be used to identify the most appropriate labels for a given set of taxonomic data, which can be used to generate a more accurate and comprehensive taxonomic map. The representation of the results can be obtained using the web application that currently supports concept analysis in three languages (English, Italian, Croatian) and several corpora.

An important feature of this procedure is the possibility to control the amount of semantic meronymy (part–whole) of the taxonomic candidates by using the process of co-mapping the cluster relationships between label graph in the $F[is\_a]_i^s$ layer and the underlying $F[and|or]_i^s$ embedding layer. Namely, if we want the labels to reflect less of the meronymic feature relative to the source lexeme s, we include label candidates that are not necessarily members of the FoF$[and|or]_s$ graph. In this way, we weaken the similarity and partonomy relationships of the semantic field used to identify conceptual abstraction used for labeling. For instance, Table 2 presents top-ranked label candidates, constructed without the restrictions to match with FoF$[and|or]_{ethics}$ lexical members. Two best label candidates, in this case, are *philosopher-n* and *man-n*. Alternatively, if only labels included in the FoF$[and|or]_{ethics}$ are permitted, we get the community labels *philosopher-n* and *socrates-n*, as shown in Table 3. This difference shows that a more abstract lexeme *man-n* is presented in a more permissive label graph, as expected. By weakening the similarity and partonomy relationships of the semantic field used to identify conceptual abstractions for labeling, we can control the amount of semantic abstractness ascribed to the construction of the label graph.

By using the non-restrictive settings we can also notice the smaller relative score difference between the top candidates, in Table 2, as opposed to the restrictive procedure shown in Table 3 where the label candidate *philosopher-n* is many times more prominent

than other candidates in a restrictive setting. In conclusion, the non-restrictive setting should be chosen for open taxonomic labeling, while the restrictive setting produces a clearer candidate for the taxonomic representation of a semantic class, as it widens the score difference between the top candidates.

Table 2. Label candidates for community FoF$[and|or]_1^{ethics}$ (*aristotle-n, Aristotle-n, plato-n, aquinas-n, socrates-n, cicero-n, hippocrates-n, descartes-n, ptolemy-n, augustine-n, galen-n, euclid-n, pythagoras-n, confucius-n, philosopher-n, stoics-n*) extracted through the $F[is\_a]_1^{ethics}$ label graph.

Label	Label Importance Score I_L
philosopher-n	18,344.7
man-n	9679.79
student-n	8151.67
person-n	7640.74
teacher-n	5094.46
year-n	4584.21
source-n	4584.21
father-n	4076.75

Table 3. Label candidates for community FoF$[and|or]_1^{ethics}$ (*aristotle-n, Aristotle-n, plato-n, aquinas-n, socrates-n, cicero-n, hippocrates-n, descartes-n, ptolemy-n, augustine-n, galen-n, euclid-n, pythagoras-n, confucius-n, philosopher-n, stoics-n*) extracted through the $F[is\_a]_1^{ethics}$ label graph and restricted to FoF_{ethics} graph membership.

Label	Label Importance Score I_L
philosopher-n	18,344.7
socrates-n	1.08
plato-n	0.94
nietzsche-n	0.76
ptolemy-n	0.0
hippocrates-n	0.0
descartes-n	0.0
augustine-n	0.0

Moreover, the candidate labels for abstract source concepts of a semantic class do not always exactly match the part–whole pattern, and some of them express the conventional metaphorical patterns of conceptual mapping with candidates being ontologically more distant. For example, the label candidates for the first coordination cluster of the source lexeme *love-n*, FoF$[and|or]_1^{love}$, consisting of lexemes: *respect-n*, *affection-n*, *friendship-n*, *admiration-n*, *trust-n*, *appreciation-n*, *loyalty-n*, and *companionship-n*, are shown in Tables 4 and 5.

Table 4. Label candidates for community FoF$[and|or]_1^{love}$ (*respect-n*, *affection-n*, *friendship-n*, *admiration-n*, *trust-n*, *appreciation-n*, *loyalty-n*, *companionship-n*) extracted through the $F[is\_a]_1^{love}$ label graph in non restrictive setting.

Label	Label Importance Score I_L
key-n	18,835.2
lot-n	16,558.7
street-n	15,694.8
magic-n	15,497.6
thing-n	15,328.6
value-n	14,603.6

Table 5. Label candidates for community FoF$[and|or]_1^{love}$ (*respect-n*, *affection-n*, *friendship-n*, *admiration-n*, *trust-n*, *appreciation-n*, *loyalty-n*, *companionship-n*) extracted through the $F[is\_a]_1^{love}$ label graph and restricted to FoF$[and|or]_{love}$ graph membership.

Label	Label Importance Score I_L
love-n	13,217.1
relationship-n	12,777.1
respect-n	4454.4
life-n	4320.1
friendship-n	4062.1
marriage-n	2672.6

It is clear that the non-restrictive option in Table 4 produces candidates that match the metaphorical definition of *love-n*. For instance, "love is a key" is a conceptual metaphor used in expressions such as "love is a key (that can open many doors)" [51]. However, if we want a formal classification, we would opt for the restrictive procedure with the results shown in Table 5, such as "love is a relationship". Finally, we can also exclude lexical items in the community FoF$[and|or]_1^{love}$ to get the labels shown in Table 6.

Table 6. Label candidates for community FoF$[and|or]_1^{love}$ (*respect-n*, *affection-n*, *friendship-n*, *admiration-n*, *trust-n*, *appreciation-n*, *loyalty-n*, *companionship-n*) extracted through the $F[is\_a]_1^{love}$ label graph excluding the lexemes of the community.

Label	Label Importance Score I_L
love-n	13,217.1
relationship-n	12,777.1
life-n	4454.4
marriage-n	2672.6
work-n	2033.7
culture-n	2033.5

The results of these tables can be seen to represent different levels of abstraction. Table 4 is the most general and abstract, while Table 6 is the most specific and precise. The choice of which level to use will depend on the purpose it is being used for. For example, if one were trying to classify a sentiment as either positive or negative they would likely choose the more precise labels from Table 6. On the other hand, if someone was trying to give an overview of a concept, then they may opt for less precise labels such as those found in Table 4.

This procedure, coupled with an additional investigation of ontological similarity performed at the level of the FoF$[and|or]_s$ network, can be used to extract the metaphorical collocations with the *is_a* constructions.

This raises a discussion about the qualitative difference between part–whole patterns of taxonomic labeling and conventional metaphorical patterns used as a proxy for labeling a semantic domain. The part–whole pattern implies that one concept is directly derived from another concept. However, conventional metaphorical patterns suggest that two concepts are related in some way, but the relationship is not necessarily literal. For example, with the label candidates for *love-n*, *respect-n* can be seen as a part of love, as it is typically included in an understanding of what love means, while *heart-n* may be seen more as a metaphor for love rather than being a direct part of it. Therefore, when considering the labels for abstract source concepts of a semantic class, it is important to consider the question of whether the conventional metaphorical patterns of conceptual mapping can be used as labels for a semantic class, even when they may seem ontologically distant. Especially, if it could be argued that these labels reflect our culture's use of metaphors to explain complex concepts such as love.

Furthermore, it is important to consider the context of the source concept when choosing the most appropriate candidate labels for semantic classes. This can help in understanding how metaphors are used to convey meaning and how they are expressed in language. For example, a metaphor such as "love is a key" implies that love opens up opportunities and possibilities, while "love is a relationship" expresses a more intimate connection between two people. Considering this context can help to identify which of these two relations is used to convey the semantic domain of a source concept.

The discussion of the efficacy of these two approaches to the construction of label graphs for semantic classes is ongoing. It is clear that the restrictive approach produces more accurate labels, as it makes use of a taxonomically defined and controllable set of lexemes that reduce ambiguity. On the other hand, the value of a non-restrictive approach is that it allows for greater flexibility, reflecting construal nuances in meaning between different text corpora. The measure-dependent propagation features need to be further analyzed in our future work, including the possibility to extract metaphoric mapping using the syntactic dependency *is_a*.

In our future work, we plan to implement the syntactic *is_a* relation for other languages. In particular, we are developing a Universal Dependency based tagging construction adapted to the language-specific grammatical expression of the *is_a* relation. An important aspect of developing a Universal Dependency based tagging construction is the consideration of language-specific rules. For example, some languages may require an additional level of specificity when it comes to specifying which elements are in the *is_a* relation. This could be done through specific verb conjugations or other syntactic cues. Furthermore, depending on the language, different kinds of syntactic constructions may be needed for expressing the *is_a* relation accurately. It is essential to consider these nuances when developing a Universal Dependency-based tagging construction. Additionally, it is important to keep in mind that there may be cultural and regional differences in how certain languages express particular concepts and ideas, so this should also be taken into account when creating a tagging system.

The informational advantage of corpus-based taxonomic semantic class labeling is the representation of corpus-specific taxonomic structures for a set of associated lexemes. However, this can also be a disadvantage for smaller corpora with a sparse set of syntactic patterns forming an *is_a* representation. One way to address this issue is to use a thesaurus-based approach, which involves using a pre-existing database of word relationships. These databases can provide a richer set of syntactic patterns and enable us to better analyze the measure-dependent propagation features.

On the other hand, there are some drawbacks associated with this enrichment. For one, the knowledge databases store the prototypical knowledge, while, in a constantly changing cultural environment, new ideas are emerging and their links to established concepts are liable to be reorganized. In practice, WordNet might be unable to capture or show all the subtle aspects of conceptual associations that could be found in a specific corpus-based semantic class labeling. Additionally, if the thesaurus does not contain all relevant words related to a particular concept, then it will not be able to accurately represent that concept in relation to other concepts.

For the comparison of *is_a* dependency labels and WordNet hypernym labels, see Table 1, which shows the prediction of hypernym labels for the related senses of the source lexeme *ethics-n* for both methods of label propagation based on the same FoF$[and|or]_{ethics}$ graph. The example shows the typical results obtained with this procedure, with a high level of semantic class abstraction provided by the synset candidates. For instance, the first semantic class with various philosophers is labeled dynasty.n.01, while the very abstract semantic class *standards-n, values-n*, etc. yields the even more abstract synset label *belief.n.01*. This exemplifies the overall validity of providing hypernym synsets for taxonomic semantic labeling.

One of the main advantages of the WordNet hypernym graph algorithm is the symbolic categorical assignment of lexical nodes to a class within a structured taxonomy. This allows

semantic enrichment of the associated lexical communities obtained by unsupervised bottom-up graph classification method and results in a set of synsets with well-defined and curated top-down knowledge relations. The hypernym graph $F[hyper]_i^s$ abstracts the categories of lexical communities using WordNet dictionary knowledge relative to the data provided by a large web corpus. This results in a comparable corpus-based representation of lexical usage given the same set of graph parameters.

For example, Table 7 shows the labeling of communities based on the English Timestamped newsfeed 2014–2019 corpus. Both corpora yield a set of sense clusters that abstract *anger-n* in a comparatively similar sense of distinct strong feeling *emotion.n.01*, associated in particular with *violence, insecurity, intolerance, resentment*, and *sadness*.

Table 7. Best-ranked WordNet hypernym labels for associated communities FoF$[and|or]_i^{anger}$ of the source lexeme *anger-n* in English Timestamped newsfeed 2014–2019 corpus extracted through the $F[hyper]_{anger}$ hypernym graph.

	Associated Community	**WordNet Hypernym Labels**
1	disappointment-n, disgust-n, shock-n, surprise-n, dismay-n, outrage-n, disbelief-n, horror-n	collapse.n.01, reflex.n.01, fight.n.02, stupefaction.n.01, surprise.n.02
2	anger-n, frustration-n, fear-n, confusion-n, anxiety-n, panic-n, uncertainty-n, doubt-n	emotion.n.01, mortal_sin.n.01, emotianal_arousal.n.01,
3	hatred-n, hate-n, bigotry-n, intolerance-n, contempt-n, prejudice-n, racism-n	emotion.n.01, intolerance.n.02, impatience.n.03
4	resentment-n, bitterness-n, regret-n, jealousy-n, envy-n	hostility.n.03, taste_property.n.01, taste.n.01, disagreeableness.n.02
5	sadness-n, grief-n, rage-n, despair-n, sorrow-n	feeling.n.01, unhappiness.n.02, uncheerfulness.n.02

Another advantage of using WordNet is the ability to find corresponding hypernym structures in many languages via the Open Multilingual WordNet library [12]. The results of the comparative translation equivalent of the concept *anger-n* in Croatian, concept *ljutnja-n*, calculated on the basis of the Croatian hrWac corpus (hrWac) [58,59] are presented in Table 8.

The cross-linguistic comparison yields a commensurable and yet culturally specific insight into the associative conceptual matrix of the lexeme *ljutnja*, which indicates a somewhat different picture than its English translation equivalent. The lexeme *ljutnja* in hrWac also displays the aggressive features of anger, but is more abstractly associated with states and emotions experienced when one is not well or does not achieve the desired goal, as well as with the quality of having no strength or power.

We can argue that corpus-based taxonomic labeling graph procedures highlight usage-based and cultural differences in the semantic processing of the same lexical concept. These features provide a transparent and consistent approach to intra-and cross-cultural analysis of the semantic lexical potential for a given source word.

As a drawback of the method, it should be noted that the lexical sparseness of WordNet hypernym relations hinders the full scope of mapping. Nevertheless, the structure of the coordination layer subgraphs can be compensated to some extent by the association of more frequent noun lexemes, which provide a more conventional abstract categorical label for an associated sense of a source lexeme.

Table 8. Best-ranked WordNet hypernym labels for associated communities FoF$[and|or]_i^{ljutnja}$ of the source lexeme *ljutnja-n* in Croatian hrWac corpus extracted through the $F[hyper]_{ljutnja}$ hypernym graph.

	Associated Community	**WordNet Hypernym Labels**
1	ljutnja-n, frustracija-n, nezadovoljstvo-n, nervoza-n, stres-n, strah-n, napetost-n, nemir-n	emotion.n.01, emotianal_arousal.n.01, hostility.n.03
2	razočaranje-n, ogorčenje-n, gnjev-n, nevjerica-n, revolt-n, zabrinutost-n, gnušanje-n	dissatisfaction.n.01, annoyance.n.02
3	ogorčenost-n, povrijeđenost-n, nemoć-n, cinizam-n, osvetoljubivost-n, zamjeranje-n, zgražanje-n	quality.n.01, property.n.01, powerlessness.n.01
4	srdžba-n, ljubomora-n, jal-n, razdraženost-n, zavist-n, osveta-n, nesigurnost-n	emotion.n.01, anger.n.01, envy.n.01
5	bijes-n, gorčina-n, tuga-n, mržnja-n, očaj-n, jad-n	anger.n.01, sadness.n.01

In terms of extending the knowledge-base approach to labeling, although the WordNet labeling results show perspective results, in our future work we plan to integrate other knowledge databases with similar semantic relations, such as Conceptnet and Wikipedia, and compare the results with corpus-based word *is_a* category and category *is_a* word syntax dependency. This is an interesting prospect and could potentially lead to a more comprehensive approach to labeling. For example, integrating Conceptnet could provide the ability to link words with more nuanced relationships than WordNet, such as "related concepts". This could be useful for creating more complex labels that capture the nuances in language.

Additionally, Wikipedia integration could add another layer of contextual information, allowing for labels that are based on topics rather than just individual words. Ultimately, this type of knowledge-based approach could lead to better performance in downstream tasks such as text classification and sentiment analysis.

Finally, in light of the new state-of-the-art methodologies, we plan to introduce the described dependency layers within a graph-to-graph Transformer [60], which has shown possibilities for integration of dependency layers into sequence-based language modeling, as well as experiment with building graph neural network language models [61]. The integration of dependency layers into sequence-based language modeling is a promising area for further exploration. This could enable the development of models that are more capable of capturing the syntactic and semantic nuances of natural language, as well as better represent structures such as long-distance dependencies. Additionally, building graph neural network language models has been shown to improve the performance of natural language understanding tasks by leveraging structural information from input lexical graphs. Such models could potentially provide even greater flexibility in terms of integrating both syntactic and semantic information. Finally, it would be interesting to explore how these new state-of-the-art methods can be used to create systems that are able to generate natural language outputs that accurately reflect the underlying structure of an input lexical graph.

5. Conclusions

The article describes the dependency-based graph procedure for taxonomic semantic class labeling using the *is_a* syntactic dependency layer of a morpho-syntactically tagged corpus, implemented in the ConGraCNet application. The graph-based method for taxo-

nomic labeling is a powerful and efficient tool for the extraction of semantic classes and their associated labels. This method utilizes the semantic potential of multilayered graph embeddings constructed from syntactic dependencies in a large tagged corpus to identify the associated semantic class of a lexeme. The labeling graph is then constructed to assign taxonomic lexical labels to each lexical sense in the cluster.

This procedure provides a conceptual abstraction of the sense clusters associated with the source lexeme and can be used to differentiate taxonomic meaning in a conceptually rich, ontologically transparent, and computationally efficient manner. In a similar graph-based procedure, we demonstrated the construction of a hypernym relation-based graph for the identification of taxonomic semantic class labels using the WordNet synset relations. The candidate labels, extracted from the *is_a* relation and hypernym relation and identified using centrality measures, abstract the semantic class of a particular cluster and provide a means of differentiating taxonomic meaning in a conceptually rich, ontologically transparent, and computationally efficient manner.

Author Contributions: Conceptualization, B.P., S.B.B. and T.B.K.; methodology, T.B.K., S.B.B. and B.P.; software, B.P. and S.B.B.; validation, T.B.K., S.B.B. and B.P.; formal analysis, T.B.K., S.B.B. and B.P.; investigation, T.B.K., S.B.B. and B.P.; resources, B.P. and S.B.B.; data curation, B.P. and S.B.B.; writing—original draft preparation, T.B.K., S.B.B. and B.P.; writing—review and editing, T.B.K., S.B.B. and B.P.; visualization, B.P., S.B.B. and T.B.K.; supervision, B.P., S.B.B. and T.B.K.; project administration, B.P. and T.B.K.; funding acquisition, B.P. and T.B.K. All authors have read and agreed to the published version of the manuscript.

Funding: This work has been supported in part by the Croatian Science Foundation under the project UIP-05-2017-9219 and the University of Rijeka under the project UNIRI-human-18-243.

Conflicts of Interest: The authors declare no conflict of interest.

Abbreviations

The following abbreviations are used in this manuscript:

NLP	Natural Language Processing
ConGraCNet	Construction Grammar Conceptual Network
SLI	Semi-Local Integration
GNN	Graph Neural Networks
CSK	Common-Sense Knowledge
TFMN	Textual Forma Mentis Networks

References

1. Hovy, E.; Navigli, R.; Ponzetto, S.P. Collaboratively built semi-structured content and Artificial Intelligence: The story so far. *Artif. Intell.* **2013**, *194*, 2–27.
2. Mel'čuk, I.; Milićević, J. *An Advanced Introduction to Semantics: A Meaning-Text Approach*; Cambridge University Press: Cambridge, UK, 2020.
3. Geeraerts, D. *Cognitive Linguistics: Basic Readings*; Walter de Gruyter: Berlin, Germany, 2006; Volume 34.
4. Langacker, R.W. *Cognitive Grammar: A Basic Introduction*; Oxford University Press: New York, NY, USA, 2008.
5. Perak, B.; Ban Kirigin, T. Construction Grammar Conceptual Network: Coordination-based graph method for semantic association analysis. *Nat. Lang. Eng.* **2022**, 1–31. https://doi.org/10.1017/S1351324922000274.
6. Miller, G.A. WordNet: A lexical database for English. *Commun. ACM* **1995**, *38*, 39–41.
7. Ban Kirigin, T.; Bujačić Babić, S.; Perak, B. Semi-Local Integration Measure of Node Importance. *Mathematics* **2021**, *10*, 405.
8. ConGraCNet Application. Available online: https://github.com/bperak/ConGraCNet (accessed on 16 December 2022).
9. EmoCNet Project. Available online: emocnet.uniri.hr (accessed on 16 December 2022).
10. Ban Kirigin, T.; Bujačić Babić, S.; Perak, B. Lexical Sense Labeling and Sentiment Potential Analysis Using Corpus-Based Dependency Graph. *Mathematics* **2021**, *9*, 1449.
11. Perak, B.; Ban Kirigin, T. Dependency-based Labeling of Associative Lexical Communities. In Proceedings of the Central European Conference on Information and Intelligent Systems (CECIIS 2021), Varaždin, Croatia, 13–15 October 2021; pp. 34–42.
12. Bond, F.; Foster, R. Linking and extending an open multilingual wordnet. In Proceedings of the 51st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers), Sofia, Bulgaria, 4–9 August 2013; pp. 1352–1362.
13. Schuler, K.K. *VerbNet: A Broad-Coverage, Comprehensive Verb Lexicon*; University of Pennsylvania: Philadelphia, PA, USA, 2005.

14. Kingsbury, P.R.; Palmer, M. From TreeBank to PropBank. In Proceedings of the LREC, Las Palmas, Canary Islands, Spain, 29–31 May 2002; pp. 1989–1993.
15. Navigli, R.; Ponzetto, S.P. BabelNet: Building a very large multilingual semantic network. In Proceedings of the 48th Annual Meeting of the Association for Computational Linguistics, Uppsala, Sweden, 11–16 July 2010; pp. 216–225.
16. Di Fabio, A.; Conia, S.; Navigli, R. VerbAtlas: A novel large-scale verbal semantic resource and its application to semantic role labeling. In Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP), Hong Kong, China, 3–7 November 2019; pp. 627–637.
17. Speer, R.; Havasi, C. Representing general relational knowledge in conceptnet 5. In Proceedings of the LREC, Istanbul, Turkey, 21–27 May 2012; Volume 2012, pp. 3679–86.
18. Speer, R.; Chin, J.; Havasi, C. Conceptnet 5.5: An open multilingual graph of general knowledge. In Proceedings of the Thirty-First AAAI Conference on Artificial Intelligence, San Francisco, CA, USA, 4–9 February 2017.
19. Harris, Z. Distributional structure. *Word* **1954**, *10*, 146–162.
20. Hanks, P. Corpus pattern analysis. In *Proceedings of the Euralex*; Université de Bretagne-Sud Lorient: Lorient, France, 2004; Volume 1, pp. 87–98.
21. Hanks, P. *Lexical Analysis: Norms and Exploitations*; MIT Press: Cambridge, MA, USA, 2013.
22. Baroni, M.; Murphy, B.; Barbu, E.; Poesio, M. Strudel: A corpus-based semantic model based on properties and types. *Cogn. Sci.* **2010**, *34*, 222–254.
23. Navigli, R.; Velardi, P. Learning word-class lattices for definition and hypernym extraction. In Proceedings of the 48th Annual Meeting of the Association for Computational Linguistics, Uppsala Sweden, 11–16 July 2010; pp. 1318–1327.
24. Boella, G.; Di Caro, L. Extracting definitions and hypernym relations relying on syntactic dependencies and support vector machines. In Proceedings of the 51st Annual Meeting of the Association for Computational Linguistics, ACL 2013, Sofia, Bulgaria, 4–9 August 2013; Association for Computational Linguistics (ACL): Stroudsburg, PA, USA, 2013; Volume 2, pp. 532–537.
25. Gardenfors, P. *Conceptual Spaces: The Geometry of Thought*; MIT Press: Cambridge, MA, USA, 2004.
26. Dumais, S.T. Latent semantic analysis. *Annu. Rev. Inf. Sci. Technol.* **2004**, *38*, 188–230.
27. Blei, D.M.; Ng, A.Y.; Jordan, M.I. Latent dirichlet allocation. *J. Mach. Learn. Res.* **2003**, *3*, 993–1022.
28. Mikolov, T.; Sutskever, I.; Chen, K.; Corrado, G.S.; Dean, J. Distributed representations of words and phrases and their compositionality. *Adv. Neural Inf. Process. Syst.* **2013**, *2*, 3111–3119.
29. Pennington, J.; Socher, R.; Manning, C. Glove: Global vectors for word representation. In Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing (EMNLP), Doha, Qatar, 26–28 October 2014; pp. 1532–1543.
30. Bojanowski, P.; Grave, E.; Joulin, A.; Mikolov, T. Enriching word vectors with subword information. *Trans. Assoc. Comput. Linguist.* **2017**, *5*, 135–146.
31. Huang, E.H.; Socher, R.; Manning, C.D.; Ng, A.Y. Improving word representations via global context and multiple word prototypes. In Proceedings of the 50th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers), Jeju Island, Republic of Korea, 8—14 July 2012; pp. 873–882.
32. Iacobacci, I.; Pilehvar, M.T.; Navigli, R. Sensembed: Learning sense embeddings for word and relational similarity. In Proceedings of the 53rd Annual Meeting of the Association for Computational Linguistics and the 7th International Joint Conference on Natural Language Processing (Volume 1: Long Papers), Beijing, China, 26–31 July 2015; pp. 95–105.
33. Scarlini, B.; Pasini, T.; Navigli, R. Sensembert: Context-enhanced sense embeddings for multilingual word sense disambiguation. In Proceedings of the AAAI Conference on Artificial Intelligence, New York, NY, USA, 7–12 February 2020; Volume 34, pp. 8758–8765.
34. Rudin, C. Stop explaining black box machine learning models for high stakes decisions and use interpretable models instead. *Nat. Mach. Intell.* **2019**, *1*, 206–215.
35. Loyola-Gonzalez, O. Black-box vs. white-box: Understanding their advantages and weaknesses from a practical point of view. *IEEE Access* **2019**, *7*, 154096–154113. https://doi.org/10.1109/ACCESS.2019.2949286.
36. Molnar, C. Interpretable Machine Learning; Available online: https://github.com/christophM/interpretable-ml-book (accessed on 16 December 2022).
37. Petroni, F.; Lewis, P.; Piktus, A.; Rocktäschel, T.; Wu, Y.; Miller, A.H.; Riedel, S. How context affects language models' factual predictions. *arXiv* **2020**, arXiv:2005.04611.
38. Kavumba, P.; Heinzerling, B.; Brassard, A.; Inui, K. Learning to Learn to be Right for the Right Reasons. *arXiv* **2021**, arXiv:2104.11514.
39. Weir, N.; Poliak, A.; Van Durme, B. Probing neural language models for human tacit assumptions. *arXiv* **2020**, arXiv:2004.04877.
40. Roberts, A.; Raffel, C.; Shazeer, N. How much knowledge can you pack into the parameters of a language model? *arXiv* **2020**, arXiv:2002.08910.
41. Paranyushkin, D. InfraNodus: Generating insight using text network analysis. In Proceedings of the World Wide Web Conference, San Francisco, CA, USA, 13–17 May 2019; pp. 3584–3589.
42. Leone, V.; Siragusa, G.; Di Caro, L.; Navigli, R. Building semantic grams of human knowledge. In Proceedings of the 12th Language Resources and Evaluation Conference, Marseille, France, 11–16 May 2020; pp. 2991–3000.

43. Barba, E.; Procopio, L.; Campolungo, N.; Pasini, T.; Navigli, R. Mulan: Multilingual label propagation for word sense disambiguation. In Proceedings of the Twenty-Ninth International Conference on International Joint Conferences on Artificial Intelligence, Yokohama, Japan, 7–15 January 2021; pp. 3837–3844.
44. Stella, M. Text-mining forma mentis networks reconstruct public perception of the STEM gender gap in social media. *PeerJ Comput. Sci.* **2020**, *6*, e295.
45. Perak, B. Conceptualisation of the Emotion Terms: Structuring, Categorisation, Metonymic and Metaphoric Processes within Multi-layered Graph Representation of the Syntactic and Semantic Analysis of Corpus Data. In *Cognitive Modelling in Language and Discourse across Cultures*; Cambridge Scholars Publishing: Newcastle upon Tyne, UK, 2017; pp. 299–319.
46. Perak, B. An ontological and constructional approach to the discourse analysis of commemorative speeches in Croatia. In *Framing the Nation and Collective Identities Political Rituals and Cultural Memory of the Twentieth-Century Traumas in Croatia*; Pavlaković, V., Pauković, D., Eds.; Memory Studies: Global Constellations; Routledge: London, UK, 2019; pp. 63–100.
47. Perak, B. Emocije u korpusima: Konstrukcijska gramatika i graf metode analize izražavanja emotivnih kategorija. In Proceedings of the Zagrebačka slavistička škola-48. hrvatski seminar za strane slaviste, Dubrovnik, Croatia, 19–30 August 2019; pp. 100–120.
48. Perak, B.; Ban Kirigin, T. Corpus-Based Syntactic-Semantic Graph Analysis: Semantic Domains of the Concept Feeling. *Raspr. Časopis Instituta Za Hrvat. Jez. I Jezikoslovlje* **2020**, *46*, 493–532.
49. Semi-Local Intregation Measure. Available online: https://github.com/sbujacic/SLI-Node-Importance-Measure (accessed on 16 December 2022).
50. Ban Kirigin, T.; Meštrović, A.; Martinčić-Ipšić, S. Towards a formal model of language networks. In Proceedings of the International Conference on Information and Software Technologies, Druskininkai, Lithuania, 15–16 October 2015; Springer: Cham, Switzerland, 2015, pp. 469–479.
51. Brdar, M.; Brdar-Szabó, R.; Perak, B. Metaphor repositories and cross-linguistic comparison. *Metaphor Metonymy Digit. Age Theory Methods Build. Repos. Fig. Lang.* **2019**, *8*, 64.
52. Thomas, J. *Discovering English with Sketch Engine Workbook*; Lulu.com: Morrisville, NC, USA, 2016.
53. Sketch Engine. Available online: https://bonito.sketchengine.eu/corpus/wsdef?corpname=preloaded/ententen13_tt2_1 (accessed on 16 December 2022).
54. Traag, V.A.; Waltman, L.; Van Eck, N.J. From Louvain to Leiden: Guaranteeing well-connected communities. *Sci. Rep.* **2019**, *9*, 5233.
55. Csardi, G.; Nepusz, T. The igraph software package for complex network research. *InterJournal Complex Syst.* **2006**, *1695*, 1–9.
56. Kilgarriff, A.; Baisa, V.; Bušta, J.; Jakubíček, M.; Kovář, V.; Michelfeit, J.; Rychlỳ, P.; Suchomel, V. The Sketch Engine: Ten years on. *Lexicography* **2014**, *1*, 7–36.
57. Sketch Engine. Available online: https://www.sketchengine.eu/ (accessed on 16 December 2022).
58. hrWac22. Available online: https://app.sketchengine.eu/#dashboard?corpname=preloaded%2Fhrwac22_ws (accessed on 16 December 2022).
59. Sketch Engine. Available online: https://bonito.sketchengine.eu/corpus/wsdef?corpname=preloaded/hrwac22_ws (accessed on 16 December 2022).
60. Mohammadshahi, A.; Henderson, J. Graph-to-graph transformer for transition-based dependency parsing. *arXiv* **2019**, arXiv:1911.03561.
61. Wu, Z.; Pan, S.; Chen, F.; Long, G.; Zhang, C.; Philip, S.Y. A comprehensive survey on graph neural networks. *IEEE Trans. Neural Netw. Learn. Syst.* **2020**, *32*, 4–24.