The Ontology of Natural Language Processing

Razieh Adelkhah
Faculty of Computer Science and Engineering
Shahid Beheshti University
Tehran, Iran
r.adelkhah@mail.sbu.ac.ir

Mehrnoush Shamsfard
Faculty of Computer Science and Engineering
Shahid Beheshti University
Tehran, Iran
m-shams@sbu.ac.ir

Niloofar Naderian
Faculty of Computer Science and Engineering
Shahid Beheshti University
Tehran, Iran
n.naderian@mail.sbu.ac.ir

***Abstract*— In this paper, we describe our proposed methodology for constructing an ontology of natural language processing (NLP). We use a semi-automatic method; a combination of rule-based and machine learning techniques; to construct and populate an ontology with bilingual (English-Persian) concept labels (lexicon) and evaluate it manually. This methodology results in a complete ontology in the natural language processing domain with 887 concepts, 88 relations, and 71 features. The built ontology is populated with near 36000 NLP related papers and 32000 authors, and about 201000 "is_Related_to", 83500 "is_Author_of", and 29000 "Presented_in" relations. The instantiation is done to enable applications find experts, publications and institutions related to various topics in NLP field.**

Keywords*— Domain Ontology; Ontology Construction; NLP Ontology;

I. INTRODUCTION

Ontologies, which are abstract models of a world and specify concepts and relationships between them, can be used to access information appropriately and provide accurate access to information based on meaning [1].

Information access is one of the main requirements for people and organizations. Nowadays, the world faces with the rapid growth in the number and diversity of research activities, scientific resources, publications and experts. Without automatic methods and systems for information access including search engines, expert finders, summarizers, translators, and knowledge extractors accessing and using this huge amount of information is rather impossible. Domain-specific ontologies are one of the essential resources for such systems. They can help us to resolve knowledge-based queries.

In this paper, we focus on the construction of a bilingual ontology for Natural Language Processing (NLP) domain. The ontology is constructed to be used in an expert finding system.

To construct and populate the NLP ontology, we employed a semi-automatic method, which will be discussed. The employed method is language and domain independent so can be applied to any domain or language as well.

The resulted ontology is revised manually and is going to be used in an expert finding system.

There are some datasets and ontologies in different domains. General Ontology for Linguistic Description (GOLD) ([2]) is the most significant model for the scientific description of human languages. Reference [3] have been construct a thematic ontology while representing a method for the automated thesauri development. Reference [4] have been proposed a method for integrating of multiple domain taxonomies to build a reference ontology to be used in profiling scholars' background knowledge. Olia ontologies serve as a reference hub for annotation terminology for linguistic phenomena on a great bandwidth of language within the Linguistic Linked Open Data (LLOD) cloud [5]. OnLit is a data model, which can be used to represent linguistic terms and concepts in a semantically interrelated data structure [6]. Despite all of the mentioned works, to the best of our knowledge, there is no NLP ontology with such a wide coverage as our ontology. The NLP ontology that we provide includes many of the related terms to the domain, that a researcher or author could possibly use or mention in research works. The ontology classifies the domain terms from an academic and technical point of view, which not only can be used in different kinds of applications, but also demonstrate the domain by a complete categorical glossary of the related terms.

For the development of the ontology, we use a revised version of the ontology design and evaluation methodology of [7]. We specify the ontology in 6 steps which are specified in the next sections: determine the scope and provide competency questions in IIII, extract concepts in III, define class hierarchies and relations in IV, completion and integration in VV, final review in VIVI, and population in VIIVII. The architecture of the system is shown in Fig 1.

II. COMPETENCY QUESTIONS

The first step in ontology construction methodology is determining the domain, scope, application of the ontology and the competency questions it should be able to answer. As it was discussed, although the built ontology can be used in many applications, our aim is to use it in an expert finding application. The application is going to be used for finding reviewers for journals and conferences for a given manuscript, finding supervisor, advisor or consultant in a specific sub-domain of NLP and also finding relevant publications in a domain of interest.

Some of the competency questions the ontology should answer are as following:

- What are the main concepts and topics in natural language processing domain?
- What resources could be used in different applications of natural language processing?
- What topics are needed in applications of the domain?
- What approaches are used for solving different problems of the domain?
- What are the expected skills of the expert in each field?

978-1-7281-1431-6/19/$31.00 ©2019 IEEE

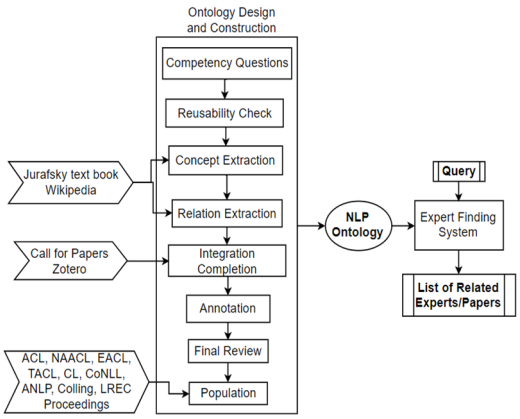

Fig. 1. The Architecture of the System

- What is the expertise of an individual?
- What are the expected skills of the author of a publication?
- What is the topic of an application if it uses each tool?
- What are the resources used in a specific approach or a specific piece of work?
- Who is the expert person in each field of the domain?
- What are the most relevant publications in a specific topic?

III. CONCEPT EXTRACTION

For building the ontology, first, we construct an initial core from scratch manually. Then, we use NLP textbook [8] to extract key phrases of the domain. After that, we use information about NLP terms available at Wikipedia to complete the list of extracted concepts.

The initial core is constructed by listing 150 most known terms of natural language processing domain, and categorizing them under four categories of concepts, relations, properties, and instances manually. We also create a hierarchy between concepts. The core had 98 concepts, 16 relations, 20 properties and 16 instances.

At the next step, on the extracted text of the NLP textbook [8], first we remove all stop-words, lone characters, numbers, adverbs and verbs. Then, we extract the most frequent words and phrases (1-gram and 2-grams) of the text as candidate phrases, by calculating the tf-idf and consider a threshold for its values. So the most frequent words and phrases of the domain are extracted as candidate concepts.

Finally, we manually review the results and remove superficial ones according to application requirements. As the result, 120 out of 180 concepts, with 67.2% precision are accepted.

Then, to extract concepts from Wikipedia, we use the structured data available at the page entitled "outline of Natural Language Processing"[1], which has a proper category of concepts of NLP. Furthermore, we create a small-scale ontology semi-automatically for each main part of the Wiki page. Then we merge these ontologies with our NLP ontology to extract new concepts. For this purpose, for each concept pair (C_1, C_2) in which C_1 is from the main NLP ontology and C_2 is from the small-scale ontology of Wiki page, we measure the similarity between two phrases by computing syntactic similarity according to (1) presented by [4].

$$SSM\ (c_1.c_2) = \max\left(0.\frac{min\big(Len(c_1).Len(c_2)\big) - EditDistance(c_1.c_2)}{min\big(Len(c_1).Len(c_2)\big)}\right) (1)$$

Where EditDistance (C_1, C_2) estimates the number of edits required to change C_1 to C_2. Having similarity more than a predefined threshold of 0.7 tags the pair as similar concepts. Concepts that do not have any similar class in the main ontology considered as new concepts to be added to the ontology. In the process of comparison, we compare each of the concepts of wiki-pedia ontology (from leaf to the root) to NLP ontology classes and find the potential similar ones. Also, we do the same process to find the possible super-class for each new concept to determine the position of them in the ontology.

There is an example of "Applications" part from Wikipedia page shown in Fig 2 The initial core is constructed by listing 150 most known terms of natural language processing domain, and categorizing them under four categories of concepts, relations, properties, and instances manually. We also create a hierarchy between concepts. The core had 98 concepts, 16 relations, 20 properties and 16 instances.

In this example, the "Open Domain Question Answering" didn't have any similar one in the NLP ontology, but its super-class "Question Answering" is matched with the exact same named class in the Ontology and the next level class, their main headline, "Applications" is matched with the "Application" class of the NLP ontology. These comparisons result that the "Open Domain Question Answering" should be added to the NLP ontology, and its suggested super class is "Question Answering".

The "outline of Natural Language Processing" page of Wikipedia, added about 290 new concepts to our ontology.

After these three steps, we performed the following tasks to extract more concepts for the ontology. We try to add any concepts related to NLP domain and also handle semantically similar concepts.

The method of finding existing similar concepts is same as what we explained for Wiki pages, but we do not have the hierarchical structure in these resources, so placing new concepts in the main ontology cannot be handled automatically.

- We employ subject categories available in call for papers (CFPs) of related conferences and journals. Then we revise the ontology and review and complete its concept and subtrees.
- We Extract the n-grams in web pages related to NLP groups and laboratories and preprocess them. Thenafter eliminating the existing ones, we add new ones to the ontology as concepts.
- We use the list of resources available in the Zotero program[2] and the items introduced by the expert to complete the "Resource" subtree of the ontology.

[1]https://en.wikipedia.org/wiki/Outline_of_natural_language_processing.
[2] https://www.zotero.org/

New Concept	Matched with	Syntactic Similarity
Open Domain Question Answering	-	-
Question Answering (Super-Class of Open Domain Question Answering)	Question Answering	1.0
Applications (Super-Class of Question Answering)	Application	0.90

Fig. 2. Syntactic Similarity between "Processes of NLP" and the NLP ontology

> Resources included a variety of data sets, applications, corporations, and middle-ware related to different areas of NLP.

To enter the new extracted terms into the ontology, we do use lexical similarity and semantic similarity between concepts to compare them and find similar ones to be merged. We also use the similarities to find the right place for concepts that are not under the main hierarchy of the ontology if available.

To calculate lexical similarity, as we did for Wikipedia pages, we use the Levenshtein distance. But for these resources which do not have the hierarchy of terms, we compute the similarity of a term not only with concepts' labels, but also with their sub-classes, super-classes, data properties, and object properties. Also, we assign a weight to each of the similarities according to their importance. As an example properties are less important than super-class so it is assigned a lower weight.

To calculate semantic similarity we use WordNet to find all synonyms of a concept. Then we calculate the Levenshtein distance between all pairs of synonyms of two concepts.

Finally, we compare the sum of calculated distances of all concepts and find similar ones. Then we merge the similar concepts and all of their sub-classes.

At the end of these steps and by using all of these resources, about 700 concepts was placed in the ontology.

IV. Relation Extraction

To find hierarchical and non-hierarchical relations between concepts we use non-structured data of NLP textbook [8].

To extract hierarchical relations, we employed a template driven method using Hearst patterns (1992).

- NPy including NPx and|or|, NPx
- NPy such as NPx and|or|, NPx
- NPy like NPx and|or|, NPx
- NPx is a/an NPy
- NPx and other Npy
- NPx or other Npy

In multi-word noun phrases (NPs) we consider the extracted relation for the head of the NP as well as the NP. Also, we consider a hierarchical relation between the heads of noun phrases and the whole noun phrase.

After creating a thousand of candidate relations, 305 proper relations were accepted manually.

[3]www.lrec conf.org/proceedings/lrec2016/topics.html

To extract non-hierarchical relations, we select four most frequent verbs in NLP domain (Generate/Produce, Use, and Have) which we have defined their corresponding relations in the ontology. We extract all sentences containing these verbs and use Stanford parser [9] to get their dependency tree and extract dependencies between tokens of sentences. Then we select the object and subject argument of verbs as concepts with verb corresponding relation. Some of these extracted relations are as follows:

- Grammar ~ have ~ rules
- Word ~ have ~ tag
- Word ~ have ~ morphemes
- Synthesis ~ produce ~ speech
- Grammars ~ generate~ language
- Algorithms ~ use ~ representation

We extract 500 relations. Then we revised them manually and accepted 148 relations to be added to the ontology.

Also, we checked words of all extracted relations and added them to the ontology if they already didn't exist. In this way, 340 new concepts are added to the ontology.

Subsequently, we define more relations and put some of them in a hierarchical structure. Some defined relations are "is_Expert_in", "is_Related_to", "Evaluated_By", "Evaluates", "Use" and "Related". "Use" and "Related" are two relations that are defined hierarchically with specific ends. Some of the "Related" relations are shown in Fig 3.

V. Completion and Integration

In this phase, lots of searches were done to evaluate the completeness of the ontology. By web searching using Google, a number of new concepts were found from different resources and glossaries to be added to the ontology.

As a result, in addition to improving the main categories of the ontology, two new categories are added as "Other_Terms" and "Related_Topic". We place words or terms that do not have any position in the main hierarchy of the ontology (applications, topics, tools, etc.) under "Other_Terms" (such as "Language Constituents").

Also, we add other topics rather than NLP topics and their most important sub-classes under "Related_Topic" node. The added topics are closely related to NLP domain and cooperate with it in researches.

As another assessment, we evaluate the completeness of the ontology with respect to LREC 2016 topics[3]. The topics have compared to ontology classes to calculate the completeness of NLP ontology. Of the 90 topics, 58 (64%) were found in the ontology. Although most of the not found topics are supplementary titles, not exactly related to NLP, like "web service", "policy issues", "metadata", etc., but some of them need to be appended to the main ontology.

To handle all of these new topics for the ontology, we add some more explanation for each concept as annotations, as well as creating new classes. These annotations are as follows:

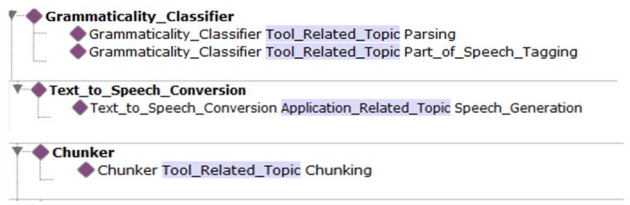

Fig. 3. Examples of Relations between the Ontology Classes.

- "Abbreviation" to show the shortened form of the class label.
- "Gloss" to explain the topic or application purpose shortly.
- "OtherLabel" to define other labels or phrases with the same meaning (at least in the context of NLP).
- "RelatedTerm" to define other terms that almost have same processes or applications. They do not have the exact same meaning, but they can be considered the same in the main application of NLP ontology, expert finding system. For example, "Medicine" and "Medical".
- "ExternalLink" to save the website address or other related links to journals, conferences or maybe organizations.

Adding these annotations can improve precision and/or recall of the future applications that uses the ontology. For example, in an expert finding system, if the query contains "Spam Detection" phrase, the system recognizes the relation with "Spam Filtering" too, because "Spam Detection" is the "otherLabel" of "Spam Filtering" class in the ontology. It should be noticed that by adding these annotations, the number of concepts decreases as some of them are merged to others as their "relatedTerm" or "otherLabel".

Some of these annotations are shown in Fig 4.

VI. Final Review

At last, we ask NLP experts to review and revise the whole ontology. The Review is done to approve the ontology as a suitable ontology for being used in the expert finding applications. Some needed corrections were applied to the ontology after the revision. Some of the concepts were removed, the location of some were changed in the hierarchy and some were merged.

The Final output is the domain-specific NLP ontology with 887 concept classes in the hierarchy, 71 features, and 88 relations. The first level of the NLP ontology with the counts of each class's subclasses is shown in Fig 5, and one of the most important parts of the ontology (Subtree of "Text_Processing" Topic) is shown in Fig 6.

VII. Ontology Population

After completing the ontology construction, we use a classification method to populate the ontology automatically. First, we collect papers available at ACL Anthology[4] and their information (title, abstract, venue, publisher, year and authors) from 2000 to 2017. Then we use Google Scholar to collect more information (citations, h-index, etc.) about the authors of the papers. We also

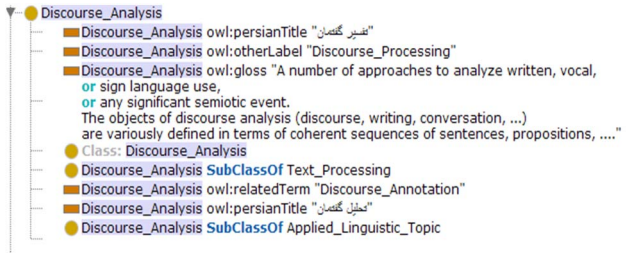

Fig. 4. Different Annotations of "Discourse_Analysis" Class

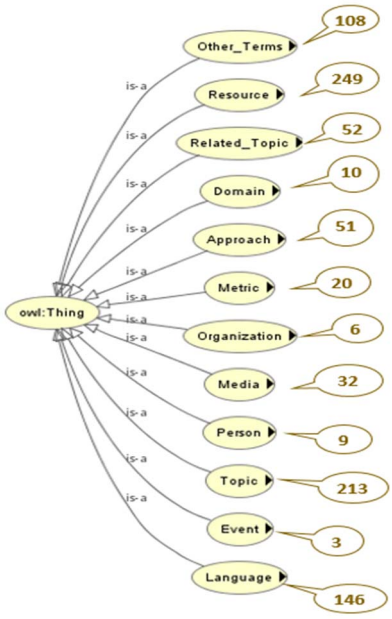

Fig. 5. The First Level of NLP Ontology with the Number of Sub-classes

collect information of the authors that have "Natural_language_proccessing" tag in their Google Scholar profile. Then we design two profile structures for publications and experts, and then we fulfill the profiles with collected information.

To populate the ontology, we add each expert profile to the ontology as an instance of person. Also, we add each publication profile as an instance of paper which is a subclass of "Resource". We also add journals and conferences existing in publication profiles as instances of "Event" and instantiate "Presented_in" relation between them. Also, we instantiate the relation "is_Author_of" between experts and their publications.

Furthermore, we instantiate the "is_Related_to" relation between publication instances and all corresponding concepts in the sub-tree of topic and approach. To do this, we use a machine learning method to find related topics and approaches to each publication.

Therefore, we prepare a training dataset for each concept in the subtree of topic and approach. The training data contains 10 publications related to each concept and is provided manually.

The training data is a collection of information about publications collected from IEEE Xplor[5].

[4] http://aclweb.org/anthology/.
[5] http://ieeexplore.ieee.org

For each paper, we extract the title, abstract and keywords and for all their words, we calculate the tf-idf and build the tf-idf vector. The size of the vector is 8760 words and the value of each is its tf-idf.

We use k- fold cross-validation to evaluate the performance of the classification method. So we partition the training data into 4 sets and for each of them we train the classifier and validate it with remaining partitions. Table ITable I shows that the average precision of this classifier is 81.2%.

Then we calculate the tf-idf and build the vector for all publication instances in the ontology too, and then we use the cosine distance measure to calculate their distances with training tf-idf vectors. We determine a threshold for maximum distance and find most similar topics and approaches for each publication instance.

At the last step, we instantiate the "is_Related_to" relation between publication instances and their related topics and approaches.

Finally the ontology is populated with near 36000 papers and 32000 authors, and about 201000 "is_Related_to", 83500 "is_Author_of", and 29000 "Presented_in" relations.
Evaluation

An ontology can be evaluated through different processes ([10], [11], [12] and [13]):

- Comparison against a gold standard,
- Data-driven evaluation,
- User-based evaluation,
- Application or task-based evaluation.

Due to the best knowledge of authors, there are no other ontologies for NLP domain, so the first choice that is to compare it with a gold standard cannot be done. Moreover, data-driven evaluation is the process of comparing ontology against existing data about the domain that the ontology models it. This process is exactly the procedure that has been followed to construct the ontology. Also, the user-based evaluation has been done under the supervision of an expert of the domain, during various steps of ontology construction. The last one, application-based evaluation, as it is mentioned before, will be done using the expert finding system we will implement as the further work.

Besides other evaluations, we use the two measures recommended by [14] depth and breadth of the ontology. It is concluded that among different measures of depth and breadth, the most important ones are breadth variance and depth variance, and that the best ontologies are generally those that have higher values of depth and breadth variances in their structure.

The calculated metrics for current ontology listed in Table II..

It's worth mentioning that this metrics are appropriate to compare more than one ontologies together, and now that there are no other NLP ontologies, they don't have any gains unless for future alternatives of ontology, to determine that the changes get the ontology to a better situation or not.

Reference [15] recommends another measure to evaluate an ontology, which uses the semantic distance between two classes. This metric relies on comparing concepts according

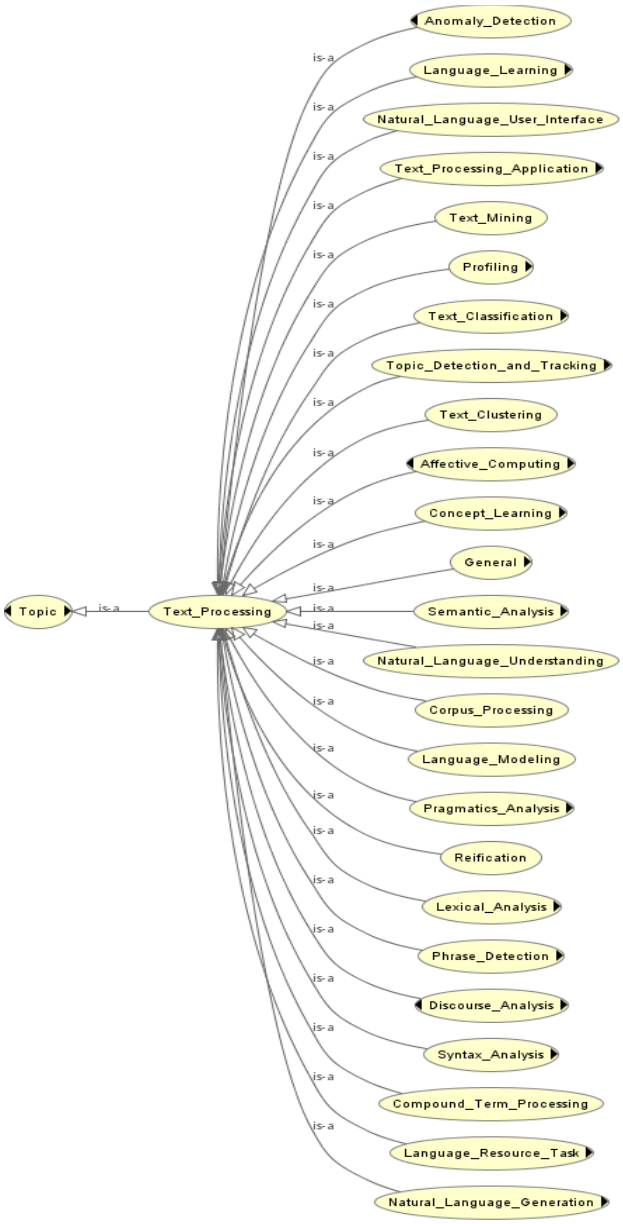

Fig. 6. The First Level Subtree of "Text_Processing" Topic

Table I Precision of Classification in Ontology Population

K	0	1	2	3	Average
Precision	84%	82%	77.5%	81%	81.2%

Table II Depth and Breadth Measures for NLP

	Depth	Breadth
Minimum	2	2
Maximum	14	313
Average	4.91	4.49
Variance	4.46	2.25

to the number of semantic evidence that they have and do not have in common in the ontology.

Based on this principle, a state-of-the-art feature-based measure is proposed ([2], [16]) that measures the semantic distance as a function of the number of non-common

taxonomic ancestors divided (for normalization) by their total number of ancestors, as in (2):

$$d(c_1.c_2) = \log_2(1 + \frac{|T(C_1) \cup T(C_2)| - |T(C_1) \cap T(C_2)|}{|T(C_1) \cup T(C_2)|}) \quad (2)$$

Where T (C_1) is the set of taxonomic ancestors, including itself. [16] proposes a semantic dispersion of an ontology relied on above distance (3):

$$Dispersion(O) = \sqrt{\frac{\sum_{c_1 \in C} d(c_1.root(o))^2}{|C|}} \quad (3)$$

As concluded by [14], the higher values of dispersion show the appropriate distribution of concepts in the ontology. The dispersion of NLP ontology has the value of 0.80 that seems to be a reasonable value.

VIII. Conclusion and Future Work

This paper described our constructed ontology and the methods we used to do it. The evaluation results show that the ontology achieved a good result at the expert's point of view.

Although future work will focus on enhancing the ontology and do more population to cover all resources and experts in NLP. Also, we will focus on developing our Expert finding system according to available information using the ontology. It is expected that using the ontology in an expert finding systems will help the results to be more semantically related to the query than other related works.

References

[1] Kara, S., et al., *An ontology-based retrieval system using semantic indexing.* Information Systems, 2012. **37**(4): p. 294-305.

[2] Batet, M., D. Sánchez, and A. Valls, *An ontology-based measure to compute semantic similarity in biomedicine.* Journal of biomedical informatics, 2011. **44**(1): p. 118-125.

[3] Pisarev, I. and E. Kotova. *Construction of thematic ontologies using the method of automated thesauri development*. in *2016 IEEE NW Russia Young Researchers in Electrical and Electronic Engineering Conference (EIConRusNW)*. 2016. IEEE.

[4] Amini, B., et al., *A reference ontology for profiling scholar's background knowledge in recommender systems.* Expert Systems with Applications, 2015. **42**(2): p. 913-928.

[5] Chiarcos, C. and M. Sukhareva, *Olia–ontologies of linguistic annotation.* Semantic Web, 2015. **6**(4): p. 379-386.

[6] Klimek, B., et al. *OnLiT: An ontology for linguistic terminology*. in *International Conference on Language, Data and Knowledge*. 2017. Springer.

[7] Noy, N.F. and D.L. McGuinness, *Ontology development 101: A guide to creating your first ontology*. 2001, Stanford knowledge systems laboratory technical report KSL-01-05 and ….

[8] Martin, J.H. and D. Jurafsky, *Speech and language processing: An introduction to natural language processing, computational linguistics, and speech recognition*. 2009: Pearson/Prentice Hall Upper Saddle River.

[9] Chen, D. and C. Manning. *A fast and accurate dependency parser using neural networks*. in *Proceedings of the 2014 conference on empirical methods in natural language processing (EMNLP)*. 2014.

[10] Hlomani, H. and D. Stacey, *Approaches, methods, metrics, measures, and subjectivity in ontology evaluation: A survey.* Semantic Web Journal, 2014. **1**(5): p. 1-11.

[11] Hlomani, H. and D.A. Stacey. *Contributing evidence to data-driven ontology evaluation workflow ontologies perspective*. in *5th International Conference on Knowledge Engineering and Ontology Development, KEOD 2013*. 2013.

[12] Ouyang, L., et al. *A method of ontology evaluation based on coverage, cohesion and coupling*. in *2011 Eighth International Conference on Fuzzy Systems and Knowledge Discovery (FSKD)*. 2011. IEEE.

[13] Brewster, C., et al., *Data driven ontology evaluation.* 2004.

[14] Fernández, M., et al. *What makes a good ontology? A case-study in fine-grained knowledge reuse*. in *Asian Semantic Web Conference*. 2009. Springer.

[15] Batet, M. and D. Sánchez. *A semantic approach for ontology evaluation*. in *2014 IEEE 26th International Conference on Tools with Artificial Intelligence*. 2014. IEEE.

[16] Sánchez, D., et al., *Ontology-based semantic similarity: A new feature-based approach.* Expert systems with applications, 2012. **39**(9): p. 7718-7728.