Entity Linking and Lexico-Semantic Patterns for Ontology Learning

Lama Saeeda✉, Michal Med, Martin Ledvinka, Miroslav Blaško, and Petr Křemen

Department of Computer Science, Faculty of Electrical Engineering,
Czech Technical University in Prague, Prague, Czech Republic
{lama.saeeda, michal.med, martin.ledvinka, blaskmir,
petr.kremen}@fel.cvut.cz

Abstract. Ontology learning from a text written in natural language is a well-studied domain. However, the applicability of techniques for ontology learning from natural language texts is strongly dependent on the characteristics of the text corpus and the language used. In this paper, we present our work so far in entity linking and enhancing the ontology with extracted relations between concepts. We discuss the benefits of adequately designed lexico-semantic patterns in ontology learning. We propose a preliminary set of lexico-semantic patterns designed for the Czech language to learn new relations between concepts in the related domain ontology in a semi-supervised approach. We utilize data from the urban planning and development domain to evaluate the introduced technique. As a partial prototypical implementation of the stack, we present Annotace, a text annotation service that provides links between the ontology model and the textual documents in Czech.

Keywords: Entity Linking · Ontology Learning · Lexico-Semantic Patterns.

1 Introduction

Ontology is an essential component for building and understanding the context of any domain of interest. For example, in urban planning and development, the master plan is a legal tool for global planning that aims to support the urban character of the various localities. It addresses the future of the city, including the development of infrastructure and areas for new constructions. Different regulations can apply to different parts of the plan, for example, building regulations. Also, it involves many actors in building and developing the plan, including urban planning experts, inhabitants, experts from the legal and regulation department, and even politicians. Communication between all these parties is not an easy process and involves a broad range of ambiguous technical terms and jargon. For this reason, it is crucial to normalize an efficient way of communication through an urban planning ontology that allows a common understanding of the technical terms that might cause confusion among all participants.

However, using such ontology depends directly on the availability of this ontology in the target domain. Building the ontology manually is tremendously exhaustive in terms of time and effort spent by human experts. Usually, domain experts, besides knowledge engineers, spend a lot of time revising textual resources and documents in order to build a background knowledge that supports the studied domain. This process can be enhanced by utilizing natural language processing side by side with information extraction techniques to help developing the ontology. Ontology learning from a textual corpus is the set of methods and techniques used for building an ontology from scratch, enriching, or adapting an existing ontology in a semi-automatic fashion using several knowledge and information sources [16]. These techniques are divided into two main types, linguistic and statistical approaches. In this paper, we investigate methods that support building the domain ontology based on a seed ontology and a set of domain-related documents in two main tasks:

- Document processing and entity linking task: this step enhances the documents with syntactic and semantic information. It provides links between the textual documents and the concepts that are defined in the seed ontology to add a semantic context to the processed documents. To perform this task, we introduce **Annotace**, a text annotation service that is further discussed in section 4.
- Learning ontological relations task: in this step, a set of rule-based lexico-semantic patterns is used to enhance the process of learning new relations between concepts in a semi-supervised approach.

To further illustrate our approach, consider the following example taken from an urban planning document in Czech.

Cs: "Správní území Prahy členěno na lokality"
En: "Administrative territory of Prague divided into localities"

At first, the entity linking engine enhances the text with semantic information by providing links to the terms in the ontology.

Cs: "**Správní území Prahy** členěno na **lokality**" Where:
Správní území Prahy is linked to mpp[1]:správní-území-prahy and
lokality is linked to mpp:lokalita

Using this information with the following pattern written in HIEL language [14] to extract a part-whole relation from Czech text,

$$(\$subject, hasPart, \$object) :- \$subject : Concept\ COMP\ RB?\ IN?\ \$object : Concept$$

reveals the relation between concepts,

mpp:správní-území-prahy *hasPart* **mpp:lokalita**

[1] mpp: http://onto.fel.cvut.cz/ontologies/slovnik/datovy-mpp-3.5-np/pojem/

where "COMP", "RB?", and "IN?" are specific variables used in the pattern's context. This revealed relation then can be suggested to the user to be added to the ontology.

The rest of the paper is organized as follows. In section 2, we present related works in the domain of entity linking and relation extraction methods. Section 3 explains in detail our approach. Sections 4 and 5 provide an overview of the experiments carried in this research work and the evaluation of the proposed approach, respectively. Finally, we conclude by summarizing the contributions and presenting perspectives in Section 6.

2 Related Work

As discussed in [24], in order to discover new relationships between entities mentioned in the text, the extracted relation requires the process of mapping entities associated with the relation to the knowledge base before it could be populated into the knowledge base. The entity linking task is highly data-dependent, and it is unlikely for a technique to dominate all others across all data sets [24]. The system requirements and the characteristics of the data sets affect the design of the entity linking system.

Any entity linking system is usually based on two steps: 1) candidate entity selection in a knowledge base that may refer to a given entity mention in the text; 2) similarity score definition for each selected candidate entity. Approaches to candidate entity generation are mainly based on string comparison between the textual representation of the entity mention in the text and the textual representation of the entity in the knowledge base. A wide variety of techniques makes use of redirect pages, disambiguation links and hyperlinks in the text to build a "Name Dictionary" that contains information about the named entities and provides a good base for linkage possibilities, as in [11], [9], and [23]. Surface form expansion helps to find other variants for the surface form of the entity mention, for example, abbreviations that are extracted from the context of the processed document as in [28], [17], [10], and [15]. Although some candidate generation and ranking features demonstrate robust and high performance on some data sets, they could perform poorly on others. Hence, when designing features for entity linking systems, the decision needs to be made regarding many aspects, such as the trade-off between accuracy and efficiency, and the characteristics of the applied data set [24]. Using Name Dictionary Based Techniques is not usable in our case since the terms in the domain-specific ontology are similar and some of them share common words, for instance, “lokalita” (en. “locality”), “zastavitelná lokalita” (en. “buildable site”), and “zastavitelná stavební lokalita” (en. “buildable construction site”). Hence, using features like entity pages, redirect pages, hyperlinks, and disambiguation pages as in [11], [9], and [23], bag of words [27] and entity popularity [22], are not useful in our case. Even statistical methods give poor results due to the small corpus and lack of training data. Authors in [16] created a Czech corpus for a simplified entity linking task that focuses on

extracting instances of class "Person". Building such a corpus is a costly task considering the different types of domain-specific entities that exist in our data.

The next task is to calculate a proper score for each candidate entity. In [21] and [3], researchers used a binary classifier to tackle the problem of candidate entity ranking. This method needs many labeled pairs to learn the classifier, and it is not a final-decision method since the final result-set can contain more than one positive class for an entity mention. While researches in [20] and [26] treated the entity ranking problem as an information retrieval task, probabilistic models are also used to link entity mentions in web free text with a knowledge base. The work in [13] proposed a generative probabilistic model that incorporates popularity, name, and context knowledge into the ranking model. Our method is based mainly on three aspects, the string similarity measures of the tokens and the candidate entity name, the number of matched tokens, and the order of these tokens as they appear in the text.

Ontology learning and population methods can be divided into clustering-based approaches that make use of widely known clustering and statistical methods, and pattern-based approaches that mainly employ linguistic patterns. However, the former approaches require large corpora to work well.

Two types of patterns can be applied to natural language corpora. Lexico-syntactic patterns that use lexical representations and syntactical information, and lexico-semantic patterns that combine lexical representations with syntactic and semantic information in the extraction process. Text2Onto [4] combines machine learning approaches with basic linguistic processing to perform relation extraction from text. FRED [8] is a tool for automatically producing RDF/OWL ontologies and linked data from natural language sentences. Both tools do not provide a direct support for documents in Czech language. Java Annotation Patterns Engine (JAPE) [6] is a language to express patterns within the open-source platform General Architecture for Text Engineering (GATE) [5]. Researchers intensively define the patterns using JAPE rules, taking advantages of the linguistic preprocessing components provided by GATE framework as in [19]. However, it is not possible to use these GATE components with our data since GATE does not have models to support resources in the Czech language. Much cleaner rules with considerably less effort and time to create can be written using Hermes Information Extraction Language (HIEL) [14].

In [19], researchers defined a set of lexico-syntactic patterns corresponding to ontology design patterns (ODPs), namely subClassOf, equivalence, and property rules. Lexico-semantic patterns were defined focusing on domain-specific event relation extraction from financial events in [2], and in [12] to spot customer intentions in micro-blogging. To the best of our knowledge, no work has been done on the topic of lexico-semantic patterns for Slavic languages. In this work, we attempt to define a preliminary set of these patterns corresponding to subClassOf, equivalence, part-whole, and property relations.

3 Proposed Approach

Our approach focuses on the Czech language with prospective usage for a bigger class of languages, for example, Slavic ones. The proposed approach is illustrated in figure 3. Following, the main components of the system are discussed in details.

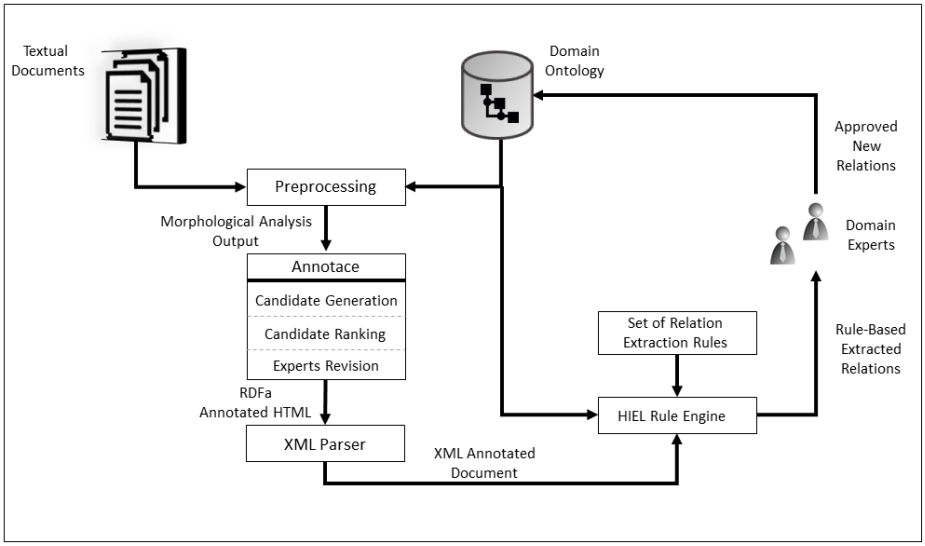

Fig. 1. Entity linking and relation extraction proposed pipeline

3.1 Entity Linking

Preprocessing Any task that deals with textual documents needs to perform a natural language processing step to enhance the parts of the text with further syntactic pragmatic, morphological, and semantic information. Some of the performed steps include tokenization, sentence splitting, and part-of-speech tagging which are dealt with by a morphological analyzer tool called MorphoDiTa, Morphological Dictionary and Tagger [25]. MorphoDiTa[2] uses trained language models for both Czech and English languages.

For the entity linking task, morphological analysis is important because Czech, like many other Slavic languages, is a highly inflective language. Meaning that a word can have different suffixes to determine a linguistic case so that tokens can have many forms belonging to the same lemma and referring to the same semantic entity. For example, "*Metropolitní plán*" (Metropolitan plan in Czech) can appear in several forms like "*Metropolitním plánem*", "*Metropolitního plánu*" and so on. We perform the same processing on the labels of entities in the ontology for the same reason.

[2] `http://ufal.mff.cuni.cz/morphodita` accessed: 2020-03-19

After stop-words removal, it is necessary to match all the remaining tokens since, in the text, most of the tokens might refer to a semantic entity in the ontology. Using regular named entity recognition (NER) tools would not be enough to recognize all the potential mentions. That is because the ontological classes are diverse and not necessarily limited to the standard named entity classes such as geographical location, person, or organization.

Candidate Entity Set Generation and Scoring At this point, we have the clean document enriched with lemmas that should be linked to corresponding semantic classes. First, we find candidate entities in the ontology that may refer to tokens in the text. We apply the famous Jaccard similarity coefficient algorithm on the lemmatized tokens taking into consideration the lexical matching. i.e., the string comparison between the surface form of the entity mention and the name of the entity existing in the knowledge base.

As mentioned earlier, our method is based mainly on three aspects, the string similarity measures of the tokens and the candidate entity name, the number of matched tokens, and the order of these tokens as they appear in the text to ensures a final-decision result.

Given a vocabulary V having a set of entities E, and a processed document D composed of a set of potential entity mentions M_d, we need to find for each entity mention $m \in M_d$ (in our case a sequence of tokens) a mapping to its corresponding entity $e \in E$. In many cases, it can happen that the mapping is not injective since there are more candidate entities in the vocabulary to be linked to a specific mention. Thus, it is needed to rank the entities in the candidate set to choose the most relevant entity and associate it with the sequence of tokens that is considered to be an entity mention of the semantic entity.

For every single token (one word), the annotation service retrieves all possible entities that the surface form of this token might refer to and creates a set of candidate entities for this token E_t. We refer to these annotations as *Words*. A *Word* contains information like the single token's surface form that we are matching the entities against, the lemma, how vital this token is (whether it is extracted as a statistical keyword by Keyword Extractor Tool KER [18]), and a list of Phrases. A *Phrase* contains information like the label and the URI of the retrieved entity in the ontology and whether it is a full match to the token or not.

Even if a phrase indicates a full-match to the token, it does not mean that this token will be annotated with this phrase. The annotation service takes into consideration the neighbors of this token while deciding for the annotation. That means that it looks around the token and it gives a higher score to the phrase if the label of the entity has common sub-strings with the tokens around. In other words, if in the text M_d occurs the sequence $t_1t_2t_3$, t_1 matches the label of the entity e_i in the ontology, but the sequence of tokens, t_1t_2 matches another entity e_j in the ontology, then the service will give a higher score to annotate the multi-word mention t_1t_2 with the entity e_j. In case there is an entity e_k in the ontology with label matching the third token as well, the sequence $t_1t_2t_3$ will be

annotated with the entity e_k. For example, let us assume the document contains the sequence of tokens **"součást otevřené krajiny"** (en. "part of an open landscape"), and in the vocabulary there is e_1 : <mpp:otevřená-krajina>, e_2 : <mpp:krajina>, the mention **"otevřené krajiny"** will be annotated with the entity e_1. The current state of the tool does not support overlapping annotations but it is considered in a newer version.

3.2 Lexico-Semantic Ontology Design Patterns

Even though the domain ontology is rich, it is still far from complete. Updating the ontology manually is an exhaustive process, for that, it is crucial to support the process of developing the ontology with automatic suggestions to the user. Statistical information extraction does not provide satisfactory results when running on a small domain-specific corpus. We define a set of rule-based extraction patterns to help the user in building the ontology. Most of the research on lexico-semantic patterns (LSPs) is done for the English language. Only some attempts have been done on other languages like French and German. To the best of our knowledge, no such work exists on Slavic languages as for Czech. In our case, we define a set of lexico-semantic patterns for Czech language focusing on common ontology relations.

For patterns definition, we use the Hermes Information Extraction Language (HIEL) that enables selecting concepts from the knowledge base and incorporate them into the lexical patterns. HIEL patterns are an ordered collection of tokens that are divided by spaces. They are described by two parts, a left-hand side (LHS) that define the relation to be extracted, and a right-hand side (RHS) that describes the pattern that should be extracted from the text. Once the RHS has been matched in the text to be processed, it is annotated as described by the LHS of the pattern. Usually, the syntax of the pattern is denoted as follows:

$$LHS :\!- \; RHS$$

The language supports lexical features like a limited list of part-of-speech tags, concepts and relations, literals, logical operators (and, or, not), repetition operators (*, +, ?), and wildcards (%, _). We extended the lexico-syntactic pattern restricted symbols and abbreviations used in [7]. The list of the abbreviations and common lexical categories used to formalize our patterns can be found in Table 1.

In our experiments and by the help of domain experts, we performed linguistic analysis and manually defined a preliminary set of lexico-semantic patterns corresponding to ontology design patterns (ODPs) that captures basic ontology relations, such as *subClassOf*, *equivalence*, *part-whole*, and *hasProperty* relations.

In the following patterns, the LHS for the rules is represented as:

$$LHS = (\$subject, relationOfInterest, \$object)$$

In tables 2, 3, 4, and 5, we present only the right-hand side part of the rules due to space presentation limit. We also provide examples extracted from our data.

Table 1. LSPs symbols and lexical categories

Symbols & Abbreviations	Description & Examples
CATV	Phrases of classification. For example, rozlišuje (distinguishes), člení se (is divided into), etc.
COMP	Phrases of composition. For example, zahrnuje (includes), tvořený (formed), skládající se (consisting of), členěno na (divided into).
COMPR	Phrases of reverse composition. For example, vyskytující se v (appearing in), tvoří (creates), je součástí (is part[of]).
CN	Phrases of generic class names. For example, základní typy (base types of).
SYN	Phrases of synonyms. For example, ekvivalent (equivalent).
PROP	Phrases of properties. For example, je přiřazen (is attached).
BE, CD, DT	Verb to be, Cardinal number, Determiner, respectively.
NN, JJ, RB, IN	Noun, Adjective, Adverb, Preposition, respectively.

Table 2. LSPs corresponding to subClassOf rules

P_{id}	**RHS**
P_{11}	$CATV\ CD\ CN\ \$object : Concept\ DT?\ \$subject : Concept$ $CATV\ CD\ CN\ \$object : Concept\ DT?\ Concept\ (\ 'a'|',\ )\ \$subject : Concept$
	example: Metropolitní plán rozlišuje dva základní typy **krajin městskou** a **otevřenou**.
	meaning: Metropolitan plan distinguishes two base types of landscape: municipal landscape and open landscape.
P_{12}	$\$object : Concept\ IN?\ CATV\ IN?\ \$subject : Concept$ $\$object : Concept\ IN?\ CATV\ IN?\ Concept\ (\ 'a'|',\ )\ \$subject : Concept$
	example: **Parkem** [se rozumí] vymezená část území s rozlišením na **městský park** a **krajinný park**.
	meaning: Park [is understood as] delimited part of area, further distinguished into municipal park and landscape park.
P_{13}	$\$subject : Concept\ BE\ \$object : Concept$
	example: **Metropolitní plán** je především **plánem** struktury území.
	meaning: The metropolitan plan is primarily a plan of the area structure.

Table 3. LSPs corresponding to part-whole rules

P_{id}	**RHS**
P_{21}	$\$subject : Concept\ COMP\ RB?\ IN?\ \$object : Concept$
	example: **Správní území Prahy** členěno na **lokality**.
	meaning: Administrative territory of Prague is divided into localities.
P_{22}	$\$subject : Concept\ COMPR\ IN?\ \$object : Concept$
	example: **Veřejná prostranství** tvoří **ulice**.
	meaning: Public areas are created by streets.

Table 4. LSPs corresponding to equivalence rules

P_{id}	**RHS**
P_{31}	$\$subject : Concept\ BE?\ SYN\ NN?\$object : Concept$ $\$subject : Concept\ BE?\ SYN\ NN?\ Concept\ ('a' \mid ',')\ \$object : Concept$
	example: **Metropolitní** je ekvivalentem pojmů **celoměstský** a **nadmístní**.
	meaning: Metropolitan is equivalent of terms citywide and supralocal.
P_{32}	$\$subject : Concept\ DT?\ SYN\ DT?\ \$object : Concept$
	example: **Krajinou za městem**, syn. **krajinným zázemím města**.
	meaning: Landscape outside the city, synonym. city landscape background.

4 Implementation of Annotace - Text Annotation Service

As a part of the processing stack, **Annotace**[3], a text annotation service, was implemented and used in the context of TermIt[4], a terminology management tool based on Semantic Web technologies developed at Czech Technical University in Prague. TermIt allows managing vocabularies and documents that use terms from the vocabularies. The documents can be imported into TermIt document manager and associated with vocabulary. The vocabulary can be empty

[3] Source code is available at `https://github.com/kbss-cvut/annotace` accessed: 2020-03-19

[4] `https://github.com/kbss-cvut/termit` accessed: 2020-03-19

Table 5. LSPs corresponding to hasProperty rules

P_{id}	**RHS**
P_{41}	$\$subject : (Concept \mid (JJ?\ NN?))\ BE\ PROP\ \$object : (Concept \mid (JJ\ NN) \mid NN)$
	example: **Každé lokalitě** je přiřazen **typ struktury**.
	meaning: Every locality has assigned type of structure.
P_{42}	$CD\ CN\ Concept\ IN?\ \$subject : Concept\ DT?\ CD?\ \$object : Concept$ $CD\ CN\ Concept\ IN?\ \$subject : Concept\ DT?\ CD?\ Concept\ ('a' \mid ',')$ $CD?\ \$object : Concept$
	example: Deset typů struktur pro zastavitelné stavební lokality: (01) rostlá struktura, (02) bloková struktura,...
	meaning: Ten types of structures for buildable localitiesa are (01) growing structure, (02) block structure,...

or already augmented with some classes and instances. TermIt allows users to create and manage vocabularies based on related resources, and the annotation service helps to automate this process in two scenarios:

- In the first scenario, a new document is uploaded into the TermIt document manager, and a newly created vocabulary is associated with it. The vocabulary is empty at this point. The task is to help the user to start building the vocabulary based on the text present in the document. Annotace starts analyzing the text based on KER[5] to extract the most significant mentions from the text as a candidate classes in the vocabulary. This step does not involve any semantic technology since there is no semantic information present in the knowledge base yet. The extracted information from the text is then presented to the user as a highlighted text with actions. These actions allow the user to create a new term in the vocabulary. The user can reject the suggested term if it is irrelevant to the associated vocabulary.
- The second scenario has a lot in common with the previous one, but it suggests that the vocabulary has already seed classes and instances. Besides the steps introduced in the first scenario, Annotace starts analyzing the document using the classes in the associated vocabulary to find mentions in the text that refer to specific entities in the vocabulary and provides links between them. These mentions are also presented as highlighted text in the document but differ from the extracted terms in the statistical step by providing a link to the associated term directly. Similar to create and reject

[5] `https://github.com/ufal/ker` accessed: 2020-03-19

actions, the user is allowed to approve the suggested association or change the association to a different term in the vocabulary.

Both scenarios suggest human interaction with the system to approve or reject the output of Annotace. The semi-automatic approach is paramount to keep the high precision of building the ontology and save the user time and efforts needed to be spent with the manual process. Annotace handles data in HTML format and the annotations are created using RDFa [1]. RDFa is an extension to HTML5 that allows to inject linked data annotations in the structure of the HTML document. Whenever a token is recognized as an entity mention for an entity in the vocabulary, a new annotation is injected around this token with properties about this annotation like a unique ID, the resource attribute referring to the URI of the entity in the vocabulary, the type of the annotation in the ontology model, and the accuracy of the prediction represented in the score attribute as depicted in listing 1.

The implementation of the patterns is not part of the stack for the current state of the tool. The patterns are tested separately within Hermes system to evaluate their efficiency. After annotating the document by Annotace with the corresponding ontological classes, Annotace augment the output with their proper tags presented in table 1 and parse the resulted document to XML-based format that serves as an input for the patterns' implementation tool. We consider integrating the patterns in the pipeline of Annotace as part of the ongoing work.

```
<html prefix="ddo:http://onto.fel.cvut.cz/ontologies/application/
    ↪ termit/pojem/">
            <p> Metropolitní plán vymezuje ve <span about="_:4"
                ↪ property="ddo:je-výskytem-termu" resource="http
                ↪ ://onto.fel.cvut.cz/ontologies/slovnik/datovy-
                ↪ mpp-3.5-np/pojem/správní-území-prahy" typeof="
                ↪ ddo:výskyt-termu" score="1.0">správním území
                ↪ Prahy</span> hranici zastavěného území... </p>
```

Listing 1. Annotated HTML with RDFa (output sample)

5 Evaluation

5.1 Description of the Evaluation Corpus

To perform the evaluation, we used a set of documents and vocabularies related to these documents in the urban planning and development field. The documents are on different levels of details regulating spatial and urban planning in Prague. All documents are in Czech. The main document in this set is the *Metropolitan Plan of Prague (MPP)*[6] which is a spatial plan for the Czech capital. It consists

[6] `https://plan.iprpraha.cz/uploads/assets/prohlizeni/zavazna-cast/textova-cast/TZ_00_Textova_cast_Metropolitniho_planu.pdf` accessed: 2020-03-19

of 168 articles divided into ten parts. The current version of MPP vocabulary corresponding to this document contains 59 terms. Other documents including but not limited to the document of the *Law 2006/183 Col., Building Law*[7], the law of urban planning and building regulations in the Czech Republic and the *Prague Building Regulations*[8] in a version from 2016 *(PSP 2016)*. The *Building Law* has 179 paragraphs divided into seven parts, and its corresponding vocabulary has 15 terms currently. On the other hand, *PSP 2016* that regulates the construction of buildings and urban planning in the Czech capital, is conceptualized as a book with 202 pages, describing 87 paragraphs, and the PSP2016 vocabulary consists of 102 terms.

5.2 Evaluation of Annotace

To evaluate the entity linking system, we used the set of documents and vocabularies described in section 5.1. The textual files are loaded into TermIt and automatically annotated using the vocabulary related to the respective documents. The annotations are then revised by a human expert and evaluated based on precision, recall, and F1 measures. The scores are calculated as follows, the True Positives (TP), the number of correct links suggested by Annotace, the False Positives (FP), where the links are suggested by Annotace but they are false, and the False Negatives (FN), the number of mentions in the text that are not suggested by Annotace as a term occurrence but the term is present in the vocabulary. These statistics are then used to calculate the well-known precision, recall, and F1 measures.

Annotace achieved average precision, recall, and F1 measures of 83%, 79%, and 80.9% respectively. It is noticeable that the false negatives occur more often than false positives. There are only a few distinct false positives. In most of the cases, terms are defined in the vocabulary and used in different meaning in the context of the document. As illustrated in figure 5.2, in the vocabulary, it happens that the term **"Lokalita"** (en. "Locality") has intrinsic trope **"Cílový charakter lokality"** (en. "Target character of locality") which in turn, is composed of other intrinsic tropes like **"Struktura"** (en. "Structure"), **"Stabilita"** (en. "Stability"), and **"Využití"** (en. "Usage") and in most of the false positive cases, the word "Struktura" is used in a different context. For example, in the following sentence, *"Metropolitní plán je především plánem struktury území"* (en. "The metropolitan plan is primarily a plan of the area structure"), the word "Struktura" is recognized as the term **"Struktura"** in the vocabulary even though, in this sentence, it means the structure of the area (in Czech, "Území") and is not meant to describe the structure of the locality. The link, in this case, should not be suggested, and hence, it is considered as a false positive. To solve this problem, the specialization classes of the class **"Lokalita"** should be considered in the disambiguation process which we will consider in future work.

[7] `https://www.zakonyprolidi.cz/cs/2006-183` accessed 2020-03-19
[8] Not available online

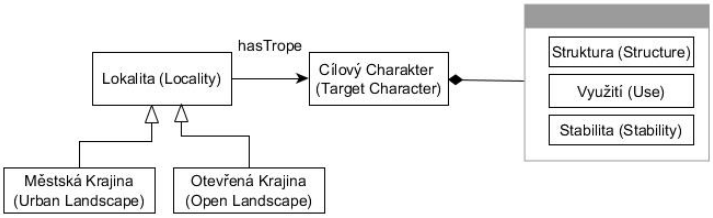

Fig. 2. Example of involving the hierarchy of the ontology in the disambiguation task

On the other hand, false negatives occurred while evaluating the MPP document when some frequently used terms come from other vocabularies and are not present in the vocabulary of MPP and hence, Annotace is not able to retrieve those terms correctly without involving other vocabularies in the process. However, most of the false negative cases happened due to lemma mismatching between the surface form and the term in the ontology, when the morphological tagger erroneously returns different lemmas for the same string.

5.3 Evaluation of Lexico-Semantic Patterns

We evaluated the patterns defined in section 3.2 on the same textual documents that are annotated and parsed by Annotace. Domain experts provided their approval or rejection of the new relations extracted from the annotated documents after applying the patterns. The patterns achieved 65% of precision, 57% of recall, and an average F1 score of 61%. Table 6 allows a closer insight of precision and recall achieved by each pattern.

Table 6. Lexico-semantic patterns evaluation in terms of precision and recall

	Precision	Recall
P11	76%	40%
P12	51%	54%
P13	63%	60%
P21	74%	70%
P22	69%	53%
P31	78%	81%
P32	83%	75%
P41	85%	87%
P42	80%	56%

The false negative cases mostly occurred when the phrase was not recognized in the text as a term occurrence, and hence, the sentence did not match the specified pattern. For this reason, we extended the patterns to extract the subject

or the object as the noun or the combination of adjective-noun. This improved the performance of the patterns and helped to recognize more terms that were not retrieved by Annotace. On the other hand, some patterns suffered from the over-generating problem.

The challenge of the free-word order of Czech language that leads to inverse relation explains many cases where false positives were encountered. For example, pattern P_{12} was able to extract the two sides of the *subClassOf* relation correctly but wrongly reversed the assignment of the super-class and the sub-class in some cases. A possible solution is to consider the case of the words besides their position. Unfortunately, we could not investigate further because the Hermes language allows only the usage of specific tags. However, the free-word order problem of the Czech language is a challenge even after considering syntactic information. The problem is that, for example, the nominative case is similar to the accusative case when the noun is plural in some situations. This would make it hard even for an expert to get the relation correctly based on the ambiguous syntactic information only. Consider the sentence, *"Zastavitelné území tvoří plochy zastavitelné"* (en. "Buildable area creates buildable surfaces") which represents exactly this case where the verb "tvoří" can be used in both directions, and "zastavitelné území", and "plochy zastavitelné" will have the same form in the nominative and accusative linguistic cases.

The type of the recognized relation is another open issue. Pattern P_{21} wrongly retrieved concepts that had a *hyponym-hypernym* relation as a *part-whole* relation. This happens when a word that, according to our experts, intuitively refers to a *part-whole* relation but is used in the text carelessly. Another common issue we found in the data is that the text does not always provide complete information to be extracted. For example, for the sentence *"Metropolitní plán rozlišuje stanici metra, vestibul stanice metra a depo metra."* (meaning Metropolitan plan distinguishes subway station, subway station lobby and subway depot), pattern P_{12} extracted **"Stanice metra"**, **"Vestibul stanice metra"** and **"Depo metra"** to be sub-classes of **"Metropolitní plán"**. However, this is not the case since **"Metropolitní plán"** is the term used to represent the document itself and hence, the extracted terms are sub-classes of a super-class that is not mentioned in the text.

Patterns P_3 and P_4 achieved reasonably high scores. However, there are only a few instances found in the corpus. A larger corpus is necessary to perform a more comprehensive evaluation.

6 Conclusion and Future Work

In this paper, we described a rule-based relation extraction approach to support the process of semi-automatic ontology building, based on a domain-specific seed vocabulary and textual documents. We defined a preliminary set of lexico-semantic rules corresponding to common ontological relations to help to extract relations between concepts based on the analysis of annotated documents written in Czech. As a part of the pipeline, we introduced Annotace, an entity linking

system for the Czech language that enhances textual documents with conceptual context and supports creating the extraction patterns.

We intend to expand the patterns to cover more common relations. As a larger corpus would give a better overview of the proposed pipeline, we will consider evaluating the system on bigger data and a different domain, namely the aviation domain. In the ongoing work, we consider investigating the available rule-based languages and tools that are more flexible, taking into consideration the availability to plug the Czech language models, which is another problem we faced. We plan to configure the preprocessing component in the pipeline to support language models for other Slavic languages that are similar in nature to the Czech language.

Acknowledgments This work was supported by grant No. CK01000204 Improving effectiveness of aircraft maintenance planning and execution of Technology Agency of the Czech Republic and grant No. SGS19/110/OHK3/2T/13 Efficient Vocabularies Management Using Ontologies of the Czech Technical University in Prague.

References

1. Adida, B., Birbeck, M., McCarron, S., Herman, I.: Rdfa core 1.1. W3C technical reports (2010)
2. Borsje, J., Hogenboom, F., Frasincar, F.: Semi-automatic financial events discovery based on lexico-semantic patterns. International Journal of Web Engineering and Technology **6**(2), 115 (2010)
3. Chen, Z., Ji, H.: Collaborative ranking: A case study on entity linking. In: Proceedings of the 2011 Conference on Empirical Methods in NLP (2011)
4. Cimiano, P., Volker, J.: Text2onto - a framework for ontology learning and data-driven change discovery. Lecture Notes in Computer Science (2005)
5. Cunningham, H.: Gate, a general architecture for text engineering. Computers and the Humanities (2002), https://doi.org/10.1023/A:1014348124664
6. Cunningham, H., Maynard, D., Tablan, V.: Jape: a java annotation patterns engine (1999)
7. De Cea, G.A., et al.: Natural language-based approach for helping in the reuse of ontology design patterns. In: International Conference on Knowledge Engineering and Knowledge Management. pp. 32–47. Springer (2008)
8. Gangemi, A., Presutti, V., Recupero, D.R., Nuzzolese, A.G., Draicchio, F., Mongiovi, M.: Semantic Web Machine Reading with FRED. Semantic Web **8**(6), 873–893 (2017)
9. Gattani, A., et al.: Entity extraction, linking, classification, and tagging for social media: a wikipedia-based approach. Proceedings of the VLDB Endowment (2013)
10. Gottipati, S., Jiang, J.: Linking entities to a knowledge base with query expansion. In: Proceedings of the Conference on Empirical Methods in NLP (2011)
11. Guo, S., et al.: To link or not to link? a study on end-to-end tweet entity linking. In: Proceedings of the 2013 Conference of the North American Chapter of the Association for Computational Linguistics (2013)

12. Hamroun, M., et al.: Lexico semantic patterns for customer intentions analysis of microblogging. In: 2015 11th International Conference on Semantics, Knowledge and Grids (SKG) (2015)
13. Han, X., Sun, L.: A generative entity-mention model for linking entities with knowledge base. In: Proceedings of the 49th Annual Meeting of the Association for Computational Linguistics: Human Language Technologies (2011)
14. Ijntema, W., Sangers, J., Hogenboom, F., Frasincar, F.: A lexico-semantic pattern language for learning ontology instances from text. Web Semant. **15**, 37–50 (Sep 2012). https://doi.org/10.1016/j.websem.2012.01.002, http://dx.doi.org/10.1016/j.websem.2012.01.002
15. Jain, A., Cucerzan, S., Azzam, S.: Acronym-expansion recognition and ranking on the web. In: 2007 IEEE International Conference on Information Reuse and Integration (2007)
16. Konkol, M.: First steps in czech entity linking. In: International Conference on Text, Speech, and Dialogue. pp. 489–496. Springer (2015)
17. Lehmann, J., et al.: Lcc approaches to knowledge base population at tac 2010. In: TAC (2010)
18. Libovický, J.: KER - keyword extractor (2016), LINDAT/CLARIN digital library at the Institute of Formal and Applied Linguistics (ÚFAL), Faculty of Mathematics and Physics, Charles University
19. Maynard, D., Funk, A., Peters, W.: Using lexico-syntactic ontology design patterns for ontology creation and population. In: Proceedings of the 2009 International Conference on Ontology Patterns-Volume 516. pp. 39–52. CEUR-WS. org (2009)
20. Nemeskey, D.M., Recski, G., Zséder, A., Kornai, A.: Budapestacad at tac 2010. In: TAC (2010)
21. Pilz, A., Paaß, G.: From names to entities using thematic context distance. In: Proceedings of the 20th ACM International Conference on Information and Knowledge Management (2011)
22. Ratinov, L., Roth, D., Downey, D., Anderson, M.: Local and global algorithms for disambiguation to wikipedia. In: Proceedings of the 49th Annual Meeting of the Association for Computational Linguistics: Human Language Technologies-Volume 1. Association for Computational Linguistics (2011)
23. Shen, W., et al.: Linking named entities in tweets with knowledge base via user interest modeling. In: Proceedings of the 19th ACM SIGKDD (2013)
24. Shen, W., Wang, J., Han, J.: Entity linking with a knowledge base: Issues, techniques, and solutions. IEEE Transactions on Knowledge and Data Engineering (2015)
25. Straková, J., Straka, M., Hajič, J.: Open-source tools for morphology, lemmatization, pos tagging and named entity recognition. In: Proceedings of 52nd Annual Meeting of the Association for Computational Linguistics: System Demonstrations (2014)
26. Varma, V., et al.: Iiit hyderabad in guided summarization and knowledge base population (2019)
27. Zhang, W., et al.: Nus-i2r: Learning a combined system for entity linking. In: TAC (2010)
28. Zhang, W., et al.: Entity linking with effective acronym expansion, instance selection and topic modeling. In: 22 International Joint Conference on AI (2011)