Onto.KOM

Towards a Minimally Supervised Ontology Learning System based on Word Embeddings and Convolutional Neural Networks

Wael Alkhatib, Leon Alexander Herrmann and Christoph Rensing

TU Darmstadt, Multimedia Communications Lab, S3/20,Rundeturmstr. 10, 64283 Darmstadt, Germany

Keywords: Ontology, Neural Language Model, Word Embeddings, Ontology Enrichment, Convolutional Neural Network, Deep Learning.

Abstract: This paper introduces Onto.KOM: a minimally supervised ontology learning system which minimizes the reliance on complicated feature engineering and supervised linguistic modules for constructing the different consecutive components of an ontology, potentially providing domain independent and fully automatic ontology learning system. The focus here is to fill in the gap between automatically identifying the different ontological categories reflecting the domain of interest and the extraction and classification of semantic relations between the concepts under the different categories. In Onto.KOM, we depart from traditional approaches with intensive linguistic analysis and manual feature engineering for relation classification by introducing a convolutional neural network (CNN) that automatically learns features from word-pair offset in the vector space. The experimental results show that our system outperforms the state-of-the-art systems for relation classification in terms of F1-measure.

1 INTRODUCTION

Ontologies form the backbone of the semantic web, which relies on a large population of high quality domain ontologies to enable the increasing need for knowledge integration and interchange for semantic driven modeling. Ontology has been defined as ”a formal specification of a shared conceptualization” (Borst, 1997). Shared conceptualization imposes that ontologies should serve as a shared view of a domain knowledge, whereas formal means it should be represented in a machine understandable format. Manually acquiring knowledge for building domain ontologies is extremely labor-intensive and time-consuming. This fact triggers the need for automatic or semi-automatic ontology learning systems.

Up to now, ontology learning systems have made extensive use of a wide range of shallow linguistic and statistical analysis modules i.e., *Text-to-Onto* (Maedche and Staab, 2000), *OntoLearn* (Velardi et al., 2013) and *INRIASAC* (Grefenstette, 2015). The previously designed systems suffer from many shortcomings concerning ontology coverage, error propagation, reliability and required computation resources. On one hand, linguistic techniques like semantic templates or lexico-syntactic pattern analysis are capable of discovering relatively accurate semantic relations between word-pairs, however, they suffers from deficiency because such patterns cover a small proportion of complex linguistic space. Moreover, all the linguistic pipeline tasks suffer from a performance loss when they are applied to out-of-domain data (McClosky et al., 2010). On the other hand, statistical techniques, i.e., co-occurrence analysis and clustering, can provide higher recall by relying on the implicit relation between words to identify new relations, however, the number of induced incorrect relations is higher which might dramatically effect the quality of the generated ontology. Beside the linguistic and statistical techniques, previously, researchers relied on manually-built lexical databases such as *WordNet* (Miller, 1995) and commonsense knowledge bases like *ConceptNet* (Liu and Singh, 2004) for ontology enrichment with additional concepts and semantic relations. Despite of the high accuracy and good structures of such resources, their coverage is limited to fine-grained concepts.

In recent years, deep learning techniques have proved to substantially outperform traditional machine learning methods across many NLP tasks grounded on neural networks i.e., paraphrase detection, sentiment analysis, knowledge base completion,

Alkhatib W., Alexander Herrmann L. and Rensing C.
Onto.KOM - Towards a Minimally Supervised Ontology Learning System based on Word Embeddings and Convolutional Neural Networks.
DOI: 10.5220/0006483000170026
In *Proceedings of the 9th International Joint Conference on Knowledge Discovery, Knowledge Engineering and Knowledge Management (KEOD 2017)*, pages 17-26
ISBN: 978-989-758-272-1

and question answering. This cutting-edge research field has been inspired by leveraging the distributed word representation in a low dimensional space using word embeddings. Word embeddings represents the words and their context in a reduced linear space, as a vector of numerical values. Word embeddings are proved to be capable of capturing latent semantic and syntactic properties of words (Mikolov et al., 2013b). Word embeddings which are mostly unsupervised, preserve linguistic regularities, such as words similarity i.e., similar words to *frog* are *toad, litoria, ranas* which are different species of frogs. Also they are capable of capturing semantic relationship between words (Mikolov et al., 2013a) i.e., $v(Paris) - v(France) \approx v(Berlin) - v(Germany)$, where $v(w)$ is the embedding of the word w.

This paper describes Onto.KOM: a minimally supervised, fully automatic and domain independent ontology learning system. The main contributions in this framework are the novel algorithms for automatically identifying the different ontological categories based on the word vectors and the reliance on word-pair offset as the only input for relation classification, which can avoid complicated feature engineering.

The rest of the paper is structured as follow: Section 2 introduces potential ontology sources. We provide an overview of related work in Sect. 3. Then, we introduce our methodology and framework in Sect. 4. Section 5 demonstrates the different experiments and comparative analysis of the proposed approaches. Finally, Sect. 6 summarizes the paper and discusses future work.

2 WIKIPEDIA AND WORDNET AS ONTOLOGY SOURCES

Wikipedia is a free crowdsourced encyclopedia with a large volume of high quality, and comprehensive articles. It has been widely used by researchers as a knowledge resource for ontology learning systems (Janik and Kochut, 2008; Kim and Hong, 2015). Wikipedia articles provide a very rich source for ontological entities through the variety of components i.e., infoboxes, templates, categories and internal links between articles. Wikipedia categories build a large network containing links of different types. In many cases there is a subtype relation between two categories and ths can be directly project into taxonomic relationships. DBpedia (Lehmann et al., 2015) and YAGO2 (Hoffart et al., 2013) are two knowledge bases which have been automatically extracted from Wikipedia by exploiting its different constitutive components.

WordNet (Miller, 1995) is a large semantic network of the English language. It organizes words in synonym sets (synsets). All words and phrases in a synset describe a certain context. Furthermore, it differentiates between words in five categories: nouns, verbs, adjectives, adverbs, and function words. Most notably, WordNet is an ontology containing different kinds of semantic relations between nouns, namely synonymy, hyponymy, meronymy, antonymy and morphological relations.

3 RELATED WORK

Many NLP applications has been powered by the revolution of deep learning techniques, including semantic parsing (Yih et al., 2014), search query retrieval (Shen et al., 2014), sentence modeling and classification (Kim, 2014), name tagging and semantic role labeling (Collobert et al., 2011), relation extraction and classification (Liu et al., 2013; Zeng et al., 2014). In the following, we will focus on related work using word embeddings and deep learning for building the different constitutive components of ontologies.

Pembeci (Pembeci, 2016), analyzed the feasibility of using word embeddings for ontology enrichment in an agglutinative language like Turkish. In their work, they showed that words from different ontological categories will be relatively separated from each other in the vector space by using t-SNE (Maaten and Hinton, 2008) to visualize embeddings of certain categories i.e., people, vegetables and animals. Then by looking into the similarity distance distributions of top N similar concepts, where $N \in \{1-50, 50-200, 200+\}$, they found that the first most similar word has a significantly high cosine distribution. The cosine distance of the 20_{th} to 200_{th} most similar words are quite close to each other. In the last experiment, the author developed an algorithm for ontology enrichment that discovers related concepts using word embeddings similarity. For the main concept, an initial set of twelve related concepts was selected. With the use of this set, a relatedness score for every word in the embeddings was calculated and then used to calculate a threshold indicating if a word is related to the main concept or not.

Fu et al. (Fu et al., 2014) approached the task of creating a hierarchy of semantic relations using only word embeddings. They have built a uniform linear projection for the embedding offset of correct hypernym-hyponym relations in order to infer new hypernym-hyponym relations. For some hyponym x and a projection ϕ, the corresponding hypernym y can be found by $y = \phi x$. The hypernym-hyponym offset for words in different domains is quite diverse, thus it cannot be captured with only one projection. As a means of depicting this diversity, they used piecewise linear projections by clustering the offsets and then calculated a projection for each cluster. New hypernym-hyponym relations can be found by analyzing if a given word pair's x, y offset is close to one of the clusters. If this is true, they use the corresponding projection ϕ_k for this cluster.

The SemEval 2016 Task 13 (Bordea et al., 2016), also addressed the task of creating a taxonomy based on extracted hypernym-hyponym relations in a set of domains (environment, food, science, artificial intelligence, plants, and vehicles). In this task, four languages were considered: Dutch, English, French and Italian. It consisted of one monolingual subtask where English was in focus and a multilingual task composed of the other languages. Five teams have submitted results to the monolingual task and two to the multilingual task. (Maitra and Das, 2016) and (Panchenko et al., 2016) contributed to the multilingual task. *JUNLP* is the system developed by Maitra and Das. It used an external open-source multilingual dictionary that is organized in a large network of semantic relations between synsets, called BabelNet to form state-of-the-art ontology, which was used to extract possible hypernym-hyponym relations from Wikipedia articles by applying a number of patterns. The system *TAXI* by Panchenko et al. used a combination of substring matching and Hearst-like lexico-syntactic patterns for the identification of hypernyms. The other two submissions (Tan et al., 2016) and (Cleuziou and Moreno, 2016) considered the monolingual task. The system *USAAR* examined if the property of some hypernyms, that their hyponyms are constructions of the hypernym and some other word, can be utilized for finding new relations. The authors investigated how many hypernym-hyponym relations can be found in the food domain using endocentricity property. In the last submission *QASSIT* (Cleuziou and Moreno, 2016), the authors deployed a genetic algorithm that uses word vectors and pretopological spaces to infer the desired hypernym-hyponym relations. A pretopological space was used to transform terms into a structured space from which the final taxonomy can be extracted.

Another important aspect in ontology learning is relation extraction. The common characteristic of previous research in relation extraction is intensive reliance on complicated feature engineering, linguistic analysis and external knowledge bases to provide a rich representation to feed a classifiers (Boschee et al., 2005; Sun et al., 2011). A very recent work based on convolutional neural networks which automatically learns features from sentences and minimizes the dependence on external toolkits and resources was proposed by (Nguyen and Grishman, 2015). Raw sentences marked with the positions of the two entities of interest are the only input for the system. Finally, deep learning structures have been used also for relation classification. Traditional systems relied on classifiers such as MaxEnt and SVM with series of supervised and manual features (i.g., POS,WordNet, name tagging, dependency parse, patterns) (Hendrickx et al., 2009). While more recent work used lexical and sentence level features based on word embeddings with convolutional neural networks (O-CNN) for sentence classification (Zeng et al., 2014).

This paper is the first step towards a minimally supervised, fully automatic and domain independent ontology leaning system based on word embeddings and convolutional neural networks. The main differences between Onto.KOM and previous automatic and semi-automatic ontology learning systems are: Firstly, the unsupervised approach for identifying the different ontological categories in a text corpus based on clustering the word vectors and using validity indices to select the optimal number of ontological categories. Secondly, we build a robust small ontology using lexico-syntactic patterns and external lexical databases in order to train our CNN classifier with the different semantic relations. Finally, departing from complicated features engineering, our model uses the embedding offset between word pairs as the only feature to identify new semantic relations between concepts.

4 ONTO.KOM METHODOLOGY

In the following we discuss the main constitutive components of the proposed ontology learning system Onto.KOM. In the first phase, we extract all single and multi-word terms representing the domain terminology. Then, we identify the different ontological categories, which are topical categories the terms belongs to, in a specific corpus based on clustering the word vectors and using validity indices to measure the resulting cluster's quality. The output of the first step are the different ontological categories i.e., food, animals and science. Secondly, for each ontological category we build a robust ontology, by adding relations between the terms of a category, using lexico-syntactic patterns and external lexical databases i.e., WordNet. The extracted ontology will be used to train a separate classifier for each category in order to identify and classify new semantic relations. Finally and

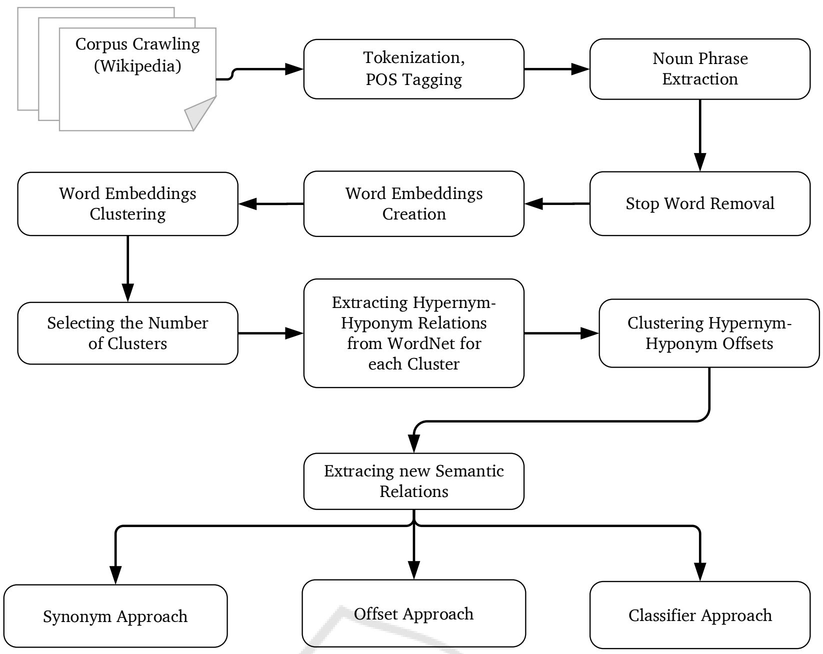


Figure 1: Block diagram of the proposed ontology learning system.

most importantly, rather than using exterior features for relation classification, our model use the embedding offset between word-pair vectors from the extracted ontology to identify new semantic relations.

With minimally supervised, we means that, the linguistic techniques and knowledge bases will be used only on the training phase of the semantic relation classifiers. Having a basic ontology with a coverage of concepts from wide range of domains will make the system capable of implicitly identifying semantic relations between words, without frequent cooccurrence based on capturing their context similarity. For a new textual dataset the system should be capable of identifying the different semantic relations using the word vectors and without any additional feature engineering.

The constitutive components of Onto.KOM, shown in Fig. 1, will be explained in the following:

4.1 Noun Phrase Extraction and Representation

In the first step, we identify the domain terminology by extracting all noun phrases (NPs) in order to form the basis for our semantic relation extraction phase. A linguistic filter will be applied on the corpus to extract all candidate NPs. Afterwards, word vectors for the extracted concepts will be created.

4.1.1 Linguistic Filter

The role of the linguistic filter is to recognize essential concepts and filter out sequence of words that are unlikely to be concepts using linguistic information. The linguistic component pipeline includes tokenization and part of speech tagging (POS) of the text documents for tagging the words as corresponding to a particular part of speech i.e., noun, adjective, verb. A combination of three linguistic filters is used to extract multi-word noun phrases NPs that can reflect essential concepts:

- $Noun\,Noun+$
- $Adj\,Noun+$
- $(Adj|Noun)+Noun$

4.1.2 Word Embeddings Creation

One problem that arises when creating word embeddings directly from text is that multi-word terms, like *machine learning*, are separated, therefore losing critical information about this kind of word constructions. In order to enable the learning of these very common constructions, we concatenate all multi-word terms (e.g., *artificial intelligence* $\rightarrow$ *artificial_intelligence*), then we create a word vector for the concatenated term.

We report experiments with word vectors trained using both Word2vec and GloVe to investigate the

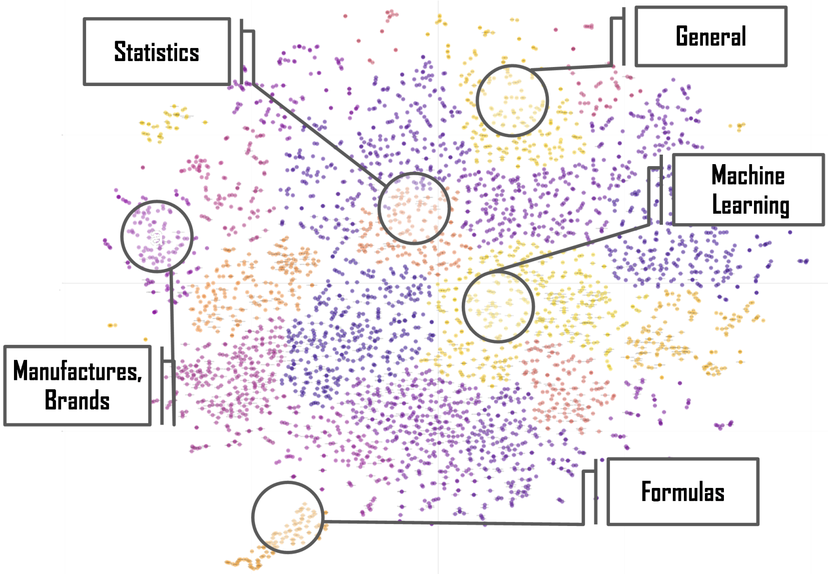


Figure 2: The distribution of word vectors from artificial intelligence articles using t-SNE plot.

effect of different settings on different ontology extraction tasks, namely similarity and relatedness. For GloVe, only one configuration with 300 dimensional vectors, minimum number of occurrences of 5, window size 15 and 30 iterations was used based on the work in (Pennington et al., 2014) which compare GloVe against wide range of word vector models except word2vec. For word2vec, different configurations had been evaluated. The adjusted parameters for each configuration were the size of the context window and the number of dimensions of the word vectors.

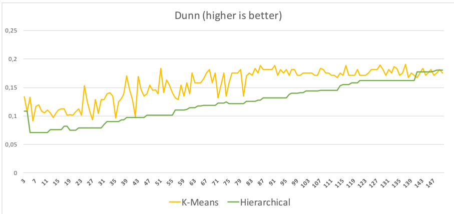
Jastrzebski et al. (Jastrzebski et al., 2017) combine 17 established datasets in the categories of similarity and analogy in order to evaluate word embeddings on all of them. For the final evaluation, six datasets, MEN, MTurk, SimLex999 and WordSimilarity 353, 353R, 353S, were chosen to benchmark the created embeddings on similarity related tasks. Correspondingly, three datasets, BLESS, the Google analogy dataset and SemEval2012, were chosen for the assessment of analogy related tasks. Based on the average performance on similarity and analogy tasks we decided on using GloVe in further steps.

4.2 Identifying Ontological Categories

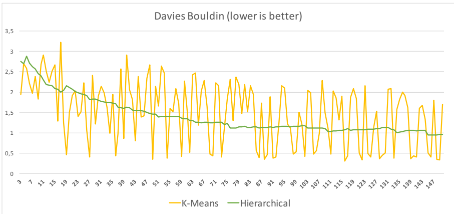
Word embeddings preserve linguistic regularities, such as words similarity and analogy. Figure 2 illustrates the projection of word vectors corresponding to noun phrases from a subset of 6274 Wikipedia articles covering the artificial intelligence category into two-dimensional space using t-SNE. The embeddings created with GloVe conserve semantic similarity so that words with similar context are close in the vector space. Using hierarchical clustering with $K = 20$ to cluster the 300-dimensional word vectors, we can identify relatively separated ontological categories. Concepts belong to *machine learning* and *statistics* are adequately separated in the vector space. These results indicate strong clustering effect, thus a good separation between words belonging to different ontological categories can be achieved.

While t-SNE on its own is a powerful tool for the visualization of word embeddings, in combination with clustering techniques other underlying patterns in the word embeddings can be identified and the different ontological entities can be extracted. A major decision for clustering is which techniques to be used and what is the number of clusters. Clustering Validity Indexes have been widely used in order to specify the optimal number of clusters and the quality of the produced clusters (Desgraupes, 2013). The optimal number of clusters is selected based on the majority vote of three indices, namely Dunn, Davies-bouldin and Silhouette. Lower value of Davies-Bouldin index indicates better clusters quality while higher values for Silhouette and Dunn indices prove better clustering quality.

Figure 3 shows the scores for Dunn and Davies-Bouldin indices over different number of clusters. K-means has higher scores than the hierarchical clustering approach when evaluated using Dunn index as shown in Fig. 3a, however, with number of clusters more that 145 hierarchical clustering outperformed K-means. From Fig. 3b, it is remarkable that the



(a) Dunn index.



(b) Davies-Bouldin index.

Figure 3: Results for two validity indices in relation to the number of clusters.

indices for K-means highly fluctuate due to the random selection of initial centroids. In contrast, the results for hierarchical clustering show that this technique produces more stable results with a low variance in the indices scores over the different number of clusters. We proceeded using hierarchical clustering approach based on the relative comparison of the indices' scores for both algorithms.

4.3 Semantic Relation Extraction using WordNet

Concepts related to different ontological categories i.e., *food* and *animals* occur in different contexts and for that their semantic relations have varied perspectives. Consequently, building a separated model for classifying the semantic relations within the different categories is an essential step to improve the system's overall performance. For each resulting cluster, we build a robust ontology by adding semantic relations between the terms. The extracted ontology will have low coverage of relations in some domain but high precision. This quality of the extracted ontology is essential to minimize the error propagation in ontology enrichment phase. For that, to create this ontology we will rely on lexico-syntactic patterns and external lexical databases. Currently, WordNet is used as a proof of concept to extract taxonomic relations, however, extracting ontological associations using WordNet has short-comings due to the low coverage of concepts for particular domains. Therefore, in future work, lexico-syntactic patterns and other lexical resources i.e., BabelNet will be incorporated in the system.

4.4 Ontology Enrichment

Ontology enrichment methodologies are used for extending an existing ontology with additional instances and relations. Figure 4 illustrates the embedding offset of hypernym-hyponym relations from concepts of two different domains, namely *plants* and *vehicles* using *t-SNE* plot. The different colored markers represent the selected domains. The relation offsets (embedding offsets) are adequately distributed in clusters, which implies indeed that, it can be decomposed into more fine-grained relations. This implies that similar relations and their offsets are near to each other in the vector space and thus have the potential to be used for discovering new relations. In the following, three different methods, namely the synonym, offset and classifier approaches will be introduced.

4.4.1 Synonym Approach

The basic assumption for this approach is that for a given hypernym-hyponym relation (X,Y), one can find new relations with the same hypernym X by searching for "synonyms" for Y. For the relation *coupe* $\rightarrow$ *car*, searching for similar or semantically close words for *coupe* will lead to *compact*, *convertible*, *roadster* or *sedan*. In combination with the corresponding hypernym *car*, new taxonomic relations can be found. The idea in respect to word embeddings is that words similar to Y should be close in the vector space. The procedure for finding an alternative for Y is to find a number of word vectors $v_{Y'}$ that are closest to v_Y the vector representation of Y, based on some threshold δ:

$$distance(v_Y, v_{Y'}) < \delta \quad (1)$$

While identifying many correct relations, this naive approach might also create a high number of false positives. In order to improve on this approach, for a given hypernym X and a set of hyponyms $Y_0, Y_1, \ldots, Y_N$ an alternative Y' has to be a shared alternative between at least n hyponyms in the top K-Nearest results. For example for $n = 2$, the hypernym-hyponyms relations *compact* $\rightarrow$ *car* and *convertible* $\rightarrow$ *car*, the word *roadster* has to be in the closest k-nearest for both *compact* and *convertible* to be considered as a new hyponym of *car*.

4.4.2 Analogy Approach

The offset approach is based on the similarity between the offset of the hypernym-hyponym word pairs in order to find new relations. The offset between two vectors X, Y is the arithmetic difference between them $(Y - X)$. This approach is similar to the work of

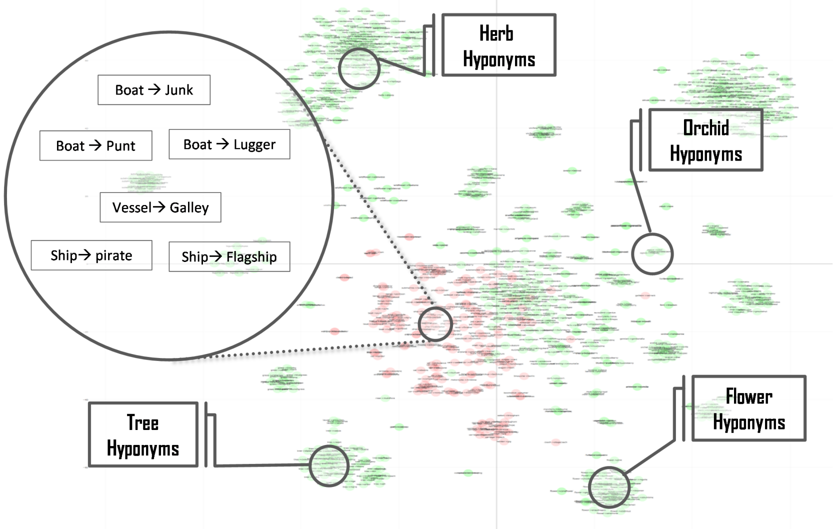


Figure 4: The distribution of the taxonomic relation offset for the plants and vehicle categories using t-SNE plot.

Pocostales (Pocostales, 2016), however, instead of learning offset projection, the idea is to find similar embedding offsets based on the embedding offset of all correct hypernym-hyponym relations. Similar to the synonym approach, this approach utilizes a k-nearest neighbor approach with either euclidean or cosine distance as a threshold for to the corresponding valid relations.

4.4.3 Classifier Approach

Enriching the ontology with additional relations based on the embedding offset is more complex than reliance on similarity scores. Moreover, the taxonomic relations in *vehicles* domain are spatially close, but separate from the relations in the *plants* domain which entails the need for creating separated model for each category. For that, we investigate the feasibility of using the embedding offset between two words as the only input to three different classifiers, namely SVM, Conditional Inference Tree (Ctree) and Convolutional Neural Networks (CNN). Ctree is a non-parametric class of regression trees embedding tree-structured regression models into a well defined theory of conditional inference procedures (Hothorn et al., 2006).

Convolutional neural networks have had a great impact on computer vision community and more recently on a wide range of NLP tasks. We imitate the assumed image structure for CNN by converting the embedding offset into similar structure and feed it to the network. Convolutional neural networks are a type of feed-forward artificial neural networks formed by a sequence of layers. In this work we focus on two types of layers:

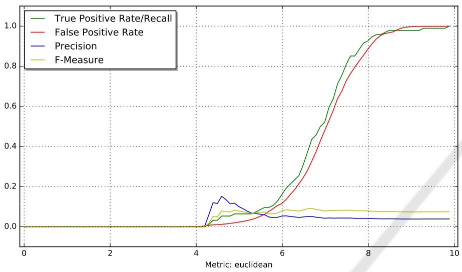
- **Convolution:** A convolutional operator is a weighting matrix (filter) used to extract higher level features. Different feature maps can be generated using various filters with different region sizes or weights.
- **Pooling:** Each convolutional layer is usually followed by a pooling layer. The rationale behind is to further down sampling the features by aggregating the scores for each filter to introduce the invariance to the absolute positions.

The final feature maps generated by the subsequent convolution and pooling operators over the created layers will be connected to a fully-connected layer in order to perform the classification of taxonomic relations.

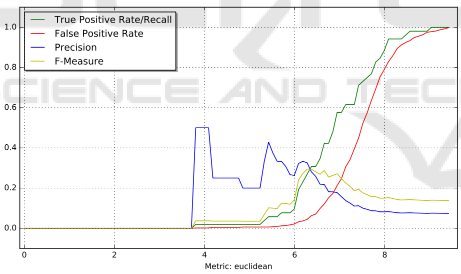
5 EVALUATION

Based on our initial evaluation, we have proceeded with using GloVe to create the word vectors of single and multi-word terms. Hierarchical clustering with Dunn, Davies-bouldin and Silhouette validity indices were used to identify the different ontological categories. The English Wikipedia was used as a corpus for creating the word vectors because of the high quality text. The articles were downloaded directly from the Wikipedia backup dump of 2016. Stanford CoreNLP toolkit (Manning et al., 2014) was used in this work for performing the different NLP tasks (POS, linguistic filter and taxonomic relations extraction). It combines machine learning and probabilistic approaches to NLP with sophisticated, deep linguistic modelling techniques. This toolkit provides state-of-the-art technology for wide range of natural-language processing tasks. Also it is quite widely used, both in the research NLP community, industry, and government.

In the last phase, we investigate the feasibility of using word similarity and relatedness for ontology enrichment. Two ontological categories, namely *vehicles* and *plants* were used for evaluating the three different approaches. The initial semantic relations, forming our basic robust ontology, were extracted from WordNet for both categories. With regard to the generated word embbedings from Wikipedia, the coverage for the *plants* category was 952 relations from 4,699 in WordNet, while 208 relations from a total of 585 for *plants* were found.



(a) Results for synonym approach based on similarity score threshold in the *plants* domain.

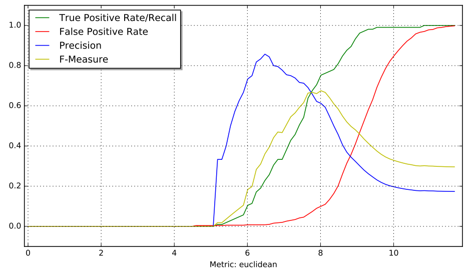


(b) Results for synonym approach based on similarity score threshold in the *vehicles* domain.

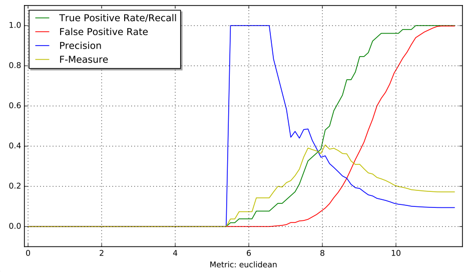
Figure 5: Results of the synonym approach.

Figures 5a and 5b subsequently show the associated graphs of the different performance metrics with regard to the similarity threshold for the two domains using euclidean distance. It is clear that the distance distribution for correct and incorrect synonym relations are similar, which indicates that using only the distance threshold to identify new relations will have poor performance.

With the offset approach, figures 6a and 6b, show a better distinction between false and true relations based on the embeddings offset. However, with small distance threshold many correct relations will be misclassified while with high distance threshold many false relations will be classified as correct taxonomic relations.



(a) Results for the offset approach in the *plants* domain depending on the distance threshold.



(b) Results for the offset approach in the *vehicles* domain depending on the distance threshold.

Figure 6: Results of the offset approach.

Based on the analysis of the first two approaches we can conclude that the embedding offset is more complex than what similarity distance can imply. For that, we tried three different classifiers following different paradigms, namely SVM, Ctree and CNN. In order to train the classifier on negative examples too, a set of 1000 random relations for both domains was extracted from WordNet synsets without taxonomic relations. For the CNN network configurations, initially we used similar structure to the one introduced by *DL4J* for image recognition. We used L2 regularization and initial learning rate of 0.01. Each filter is initialized using *Xavier* initialization (Glorot and Bengio, 2010). We trained our model with a batch size of 200 over 30 iterations, with *Stochastic gradient descent* as optimization algorithm and *Nesterov* (Nesterov, 1983) as an updater function with momentum of 0.9. Table 1 provides the comparative analysis of related work (Zeng et al., 2014) against the proposed CNN classifier as well as SVM and Ctree with the embedding offset as the only input for taxonomic relation classification over a combined dataset of both domains. The results of 5-cross validation folds are quite promising, the CNN model without any additional designated features is capable of providing the best performance equals to O-CNN for taxonomic relations classification and better than other classifiers with exterior features.

Table 1: Classifier, their feature sets and the F1-score for relation classification.

Classifier	***Feature Sets***	***F1-Score***
SVM	POS, stemming, syntactic patterns	60.1
SVM	word pair, words in between	72.5
SVM	POS, stemming, syntactic patterns, WordNet	74.8
MaxEnt	POS, morphological, noun compound, thesauri, Google n-grams, WordNet	77.6
SVM	POS, prefixes, morphological, WordNet, dependency parse, Levin classed, ProBank, FrameNet, NomLex-Plus, Google n-gram, paraphrases, TextRunner	82.2
MVRNN	POS, NER, WordNet	82.4
O-CNN	word pair, words around word pair, WordNet	82.7
SVM	embedding offset	53.2
Ctree	embedding offset	53.0
Proposed CNN	**embedding offset**	**82.7**

6 CONCLUSION AND FUTURE WORK

In this work, we proposed a minimally supervised, fully automatic and domain independent framework for ontology learning. Our experiments showed that word embeddings produced by the GloVe model preserve the linguistic regularities. Also in combination with hierarchical clustering it proved to be quite effective for identifying the different ontological categories in a domain of knowledge. Moreover, the presented work showed that the concept of utilizing word embedding offsets as a basis for relation extraction and identification using CNN networks can provide impressive results equals to best recent work (Zeng et al., 2014) without any manual features engineering. In future work, other external knowledge bases mainly ConceptNet and YAGO2 also linguistic techniques like lexico-syntactic patterns will be integrated to acquire more semantic relations in order to overcome the limitation of using WordNet in particular domains. The current experiments focused on taxonomic relations, however it is quite essential to investigate whether the system is capable of achieving same performance with regard to non-taxonomic relations.

REFERENCES

Bordea, G., Lefever, E., and Buitelaar, P. (2016). Semeval-2016 task 13: Taxonomy extraction evaluation (texeval-2). In *SemEval-2016*, pages 1081–1091. Association for Computational Linguistics.

Borst, W. N. (1997). *Construction of engineering ontologies for knowledge sharing and reuse*. Universiteit Twente.

Boschee, E., Weischedel, R., and Zamanian, A. (2005). Automatic information extraction. In *Proceedings of the International Conference on Intelligence Analysis*, volume 71. Citeseer.

Cleuziou, G. and Moreno, J. G. (2016). Qassit at semeval-2016 task 13: On the integration of semantic vectors in pretopological spaces for lexical taxonomy acquisition. *Proceedings of SemEval*, pages 1315–1319.

Collobert, R., Weston, J., Bottou, L., Karlen, M., Kavukcuoglu, K., and Kuksa, P. (2011). Natural language processing (almost) from scratch. *Journal of Machine Learning Research*, 12(Aug):2493–2537.

Desgraupes, B. (2013). Clustering indices. *University of Paris Ouest-Lab ModalX*, 1:34.

Fu, R., Guo, J., Qin, B., Che, W., Wang, H., and Liu, T. (2014). Learning semantic hierarchies via word embeddings. In *ACL (1)*, pages 1199–1209.

Glorot, X. and Bengio, Y. (2010). Understanding the difficulty of training deep feedforward neural networks. In *Aistats*, volume 9, pages 249–256.

Grefenstette, G. (2015). Inriasac: Simple hypernym extraction methods. *arXiv preprint arXiv:1502.01271*.

Hendrickx, I., Kim, S. N., Kozareva, Z., Nakov, P., Ó Séaghdha, D., Padó, S., Pennacchiotti, M., Romano, L., and Szpakowicz, S. (2009). Semeval-2010 task 8: Multi-way classification of semantic relations between pairs of nominals. In *Proceedings of the Workshop on Semantic Evaluations: Recent Achievements and Future Directions*, pages 94–99. Association for Computational Linguistics.

Hoffart, J., Suchanek, F. M., Berberich, K., and Weikum, G. (2013). Yago2: A spatially and temporally enhanced knowledge base from wikipedia. *Artificial Intelligence*, 194:28–61.

Hothorn, T., Hornik, K., and Zeileis, A. (2006). Unbiased recursive partitioning: A conditional inference framework. *Journal of Computational and Graphical statistics*, 15(3):651–674.

Janik, M. and Kochut, K. J. (2008). Wikipedia in action: Ontological knowledge in text categorization. In *Semantic Computing, 2008 IEEE International Conference on*, pages 268–275. IEEE.

Jastrzebski, S., Leśniak, D., and Czarnecki, W. M. (2017). How to evaluate word embeddings? on importance of data efficiency and simple supervised tasks. *arXiv preprint arXiv:1702.02170*.

Kim, H.-J. and Hong, K.-J. (2015). Building semantic concept networks by wikipedia-based formal concept analysis. *Advanced Science Letters*, 21(3):435–438.

Kim, Y. (2014). Convolutional neural networks for sentence classification. *arXiv preprint arXiv:1408.5882*.

Lehmann, J., Isele, R., Jakob, M., Jentzsch, A., Kontokostas, D., Mendes, P. N., Hellmann, S., Morsey, M., Van Kleef, P., Auer, S., et al. (2015). Dbpedia–a large-scale, multilingual knowledge base extracted from wikipedia. *Semantic Web*, 6(2):167–195.

Liu, C., Sun, W., Chao, W., and Che, W. (2013). Convolution neural network for relation extraction. In *International Conference on Advanced Data Mining and Applications*, pages 231–242. Springer.

Liu, H. and Singh, P. (2004). Conceptneta practical commonsense reasoning tool-kit. *BT technology journal*, 22(4):211–226.

Maaten, L. v. d. and Hinton, G. (2008). Visualizing data using t-sne. *Journal of Machine Learning Research*, 9(Nov):2579–2605.

Maedche, A. and Staab, S. (2000). The text-to-onto ontology learning environment. In *Software Demonstration at ICCS-2000-Eight International Conference on Conceptual Structures*, volume 38. sn.

Maitra, P. and Das, D. (2016). Junlp at semeval-2016 task 13: A language independent approach for hypernym identification. *Proceedings of SemEval*, pages 1310–1314.

Manning, C. D., Surdeanu, M., Bauer, J., Finkel, J. R., Bethard, S., and McClosky, D. (2014). The stanford corenlp natural language processing toolkit. In *ACL (System Demonstrations)*, pages 55–60.

McClosky, D., Charniak, E., and Johnson, M. (2010). Automatic domain adaptation for parsing. In *Human Language Technologies: The 2010 Annual Conference of the North American Chapter of the Association for Computational Linguistics*, pages 28–36. Association for Computational Linguistics.

Mikolov, T., Chen, K., Corrado, G., and Dean, J. (2013a). Efficient estimation of word representations in vector space. *arXiv preprint arXiv:1301.3781*.

Mikolov, T., Sutskever, I., Chen, K., Corrado, G. S., and Dean, J. (2013b). Distributed representations of words and phrases and their compositionality. In *Advances in neural information processing systems*, pages 3111–3119.

Miller, G. A. (1995). Wordnet: a lexical database for english. *Communications of the ACM*, 38(11):39–41.

Nesterov, Y. (1983). A method for unconstrained convex minimization problem with the rate of convergence o (1/k2). In *Doklady an SSSR*, volume 269, pages 543–547.

Nguyen, T. H. and Grishman, R. (2015). Relation extraction: Perspective from convolutional neural networks. In *Proceedings of NAACL-HLT*, pages 39–48.

Panchenko, A., Faralli, S., Ruppert, E., Remus, S., Naets, H., Fairon, C., Ponzetto, S. P., and Biemann, C. (2016). Taxi at semeval-2016 task 13: a taxonomy induction method based on lexico-syntactic patterns, substrings and focused crawling. *Proceedings of SemEval*, pages 1320–1327.

Pembeci, İ. (2016). Using word embeddings for ontology enrichment. *International Journal of Intelligent Systems and Applications in Engineering*, 4(3):49–56.

Pennington, J., Socher, R., and Manning, C. D. (2014). Glove: Global vectors for word representation. In *EMNLP*, volume 14, pages 1532–1543.

Pocostales, J. (2016). Nuig-unlp at semeval-2016 task 13: A simple word embedding-based approach for taxonomy extraction. *Proceedings of SemEval*, pages 1298–1302.

Shen, Y., He, X., Gao, J., Deng, L., and Mesnil, G. (2014). Learning semantic representations using convolutional neural networks for web search. In *Proceedings of the 23rd International Conference on World Wide Web*, pages 373–374. ACM.

Sun, A., Grishman, R., and Sekine, S. (2011). Semi-supervised relation extraction with large-scale word clustering. In *Proceedings of the 49th Annual Meeting of the Association for Computational Linguistics: Human Language Technologies-Volume 1*, pages 521–529. Association for Computational Linguistics.

Tan, L., Bond, F., and van Genabith, J. (2016). Usaar at semeval-2016 task 13: Hyponym endocentricity. *Proceedings of SemEval*, pages 1303–1309.

Velardi, P., Faralli, S., and Navigli, R. (2013). Ontolearn reloaded: A graph-based algorithm for taxonomy induction. *Computational Linguistics*, 39(3):665–707.

Yih, W.-t., He, X., and Meek, C. (2014). Semantic parsing for single-relation question answering. In *ACL (2)*, pages 643–648. Citeseer.

Zeng, D., Liu, K., Lai, S., Zhou, G., Zhao, J., et al. (2014). Relation classification via convolutional deep neural network. In *COLING*, pages 2335–2344.