EoANN: Lexical Semantic Relation Classification Using an Ensemble of Artificial Neural Networks

Rayehe Hosseinipour, Mehrnoosh Shamsfard
Computer Engineering Department, Shahid Beheshti university, Tehran, Iran
Computer Engineering Department, Shahid Beheshti university, Tehran, Iran
r.hosseinipour@mail.sbu.ac.ir
m-shams@sbu.ac.ir

Abstract

Researchers use wordnets as a knowledge base in many natural language processing tasks and applications, such as question answering, textual entailment, discourse classification, and so forth. Lexical semantic relations among words or concepts are important parts of knowledge encoded in wordnets. As the use of wordnets becomes extensively widespread, extending the existing ones gets more attention. Manual construction and extension of lexical semantic relations for WordNets or knowledge graphs are very time consuming. Using automatic relation extraction methods can speedup this process.

In this study, we exploit an ensemble of LSTM and convolutional neural networks in a supervised manner to capture lexical semantic relations which can either be used directly in NLP applications or compose the edges of wordnets. The whole procedure of learning vector space representation of relations is language independent. We used Princeton WordNet 3.1, and FarsNet 3.0 (the Persian wordnet), as gold standards to evaluate the predictive performance of our model and the results are comparable on the two languages. Empirical results demonstrate that our model outperforms the state-of-the-art models.

1 Introduction

Lexical semantic relation classification is the task of identifying s semantic relation**(s)** which holds between word pairs among a set of predefined relation types. Relation classification can be done in a supervised manner, using a dataset, labeled with a certain number of relation classes. In addition to classification with known relations, there are some methods which go even further and learn new semantic relations and suggest new relation categories (Shamsfard and Barforoosh, 2003).

Relation identification plays an essential role in many natural language processing application such as question answering, recognizing textual entailment and discourse understanding.

There are two main approaches for classification of lexical semantic relations; distributional and path-based (Wang et al., 2017).

Path-based approaches try to recognize the type of semantic relation between word pairs according to their co-occurrence information in the corpus. These methods mainly use the dependency path between word pairs as their input feature (Snow et al., 2004; Riedel et al., 2013). As Ziph's law states that most of the words in vocabulary rarely occur in the corpus (Powers, 1998) these methods have some limitation for word pairs who do not co-occur in a context.

On the other hand according to the distributional hypothesis which states "words that occur in similar contexts tend to have similar meanings" (Harris, 1954), distributional approaches try to recognize the relation between words based on their separate occurrence in the corpus which can

Proceedings of Recent Advances in Natural Language Processing, pages 481–486,
Varna, Bulgaria, Sep 2–4, 2019.
https://doi.org/10.26615/978-954-452-056-4_057

be represented for example by their word embedding vectors (Mikolov et al.,2013) and these methods have shown great performance (Baroni et al., 2012; Turney and Pantel, 2010; Roller et al., 2014).

In the last decades, several researches have been conducted on discovering hypernymy as an example of lexical semantic relations, and a key part of taxonomies and state-of-the-art models show significant results (Shwarts et al., 2016; Roller et al., 2016).

However, other types of relations have been less investigated.

Several types of models have been used for the task of semantic relation classification, but the results are not sufficiently admissible (Vu and Shwarts, 2018).

In this paper, we use an ensemble of models to improve prediction performance of relation classification. The idea of ensemble methodology is to combine some weighted classifiers in order to obtain a more accurate one (Rokach et al. 2009).

Main building blocks of this combinational model are some inducers named weak learner which perform slightly better than random. According to Condorcet Jury theorem which states " the ensemble of independent voters each of which performs better than random ($p>0.5$) has a probability of $L>p$ to make the right decision." we used different structured neural networks as the model's weak learners.

The input of our model is a concatenation of word embedding vectors corresponding to target word pairs, and the output is a class label predicted based on learned vector space distributional representation of the semantic relation which holds between them.

Our final model has the best validation F1 score of 0.894 in predicting the relation between FarsNet (Shamsfard, et al., 2010) word pairs and 0.768 to predict Princeton Wordnet (Miller, 1995; Felbaum 1998) relation classes.

We summarize the contribution of this paper as follow:

- We propose EoANN, an ensemble of artificial neural networks for classifying all types of lexical semantic relations in target datasets, without any hand-crafted features.
- Our model addresses the sparseness issue and can classify word pairs which do not necessarily co-occur in the corpus.
- According to human expert reviews, our model goes beyond relation discovery and can be employed to correct the potential error in wordnet edges and suggest new missed relation instances.

The rest of this paper is structured in 6 sections:
Section 2 presents the existing approaches for the classification of lexical semantic relations; the next one presents our model in detail, section 4 describes the data set we used for evaluating our model, section 5 reports experimental results and finally section 6 dedicated to the conclusion and future works.

2 Related Work

There are two main lexical semantic relation extraction models, distributional and path-based (pattern-based) (Wang et al., 2017) and also there are methods that use an integration of these two approaches (Shwartz et al., 2016).

Distributional methods learn the relation between word pairs based on the disjoint occurrence of them. These methods usually use a combination of word embedding vectors (Mikolov et al., 2016) as their input features. Considering v1 and v2 being word embedding vector corresponding to w1 and w2, most common combinations are:

- concatenation of v1 and v2 (**Concat**)
- the offset of v1 and v2 (**Offset**)
- point-wise multiplication of v1 and v2 (**Mult**)
- squared difference between v1 and v2 (**Sqdiff**)

Offset (Roller et al., 2014; Weeds et al., 2014; Fu et al., 2014), **Concat** (Baroni et al., 2012) and **Concat**+**Offset** (Washio and Kato, 2018) is the most common type of feature vector combination which is used in this task. To capture the different notion of interaction information about relation Vu and Shwartz (2018) add **Mult**, studied by Weeds et al. (2014) and **Sqdiff** introduced by themselves as input feature and report **Mult**+**Concat** performs better than other combination.

These methods mostly focus on lexical entailment and relation classes such as hypernym, causality and other instances of relation which

exemplified inference and have a state-of-the-art F1 score of 0.91.

Path-based or pattern-based methods utilize features derived from the context in which word pairs co-occur. For example, the dependency path between a word pair and observed predefined patterns are used as an informative feature to classify the relation. The methods of this category are limited to use only the word pairs that co-occur in corpus (Hearst et al., 1992; Snow et al., 2004; Navigli and Velardi 2010; Shamsfard et al,. 2010; Boella and Di Caro, 2013; Pavlick and Pasca, 2017)

Recently some approaches use an integration of these two methods and combine both distributional and dependency path information to obtain better results. HypNet (shwarts et al., 2016) is an examples of these approaches.

3 Our Model

In this paper, we propose a model to classify lexical semantic relations between a word pair using their word embedding vectors.

The rarity of co-occurring every candidate word pair which possibly involves in a semantic relation leads us to exploit a method which does not necessarily need to see the word pair in a context together.

The output of our model is a class label prediction based on learned vector space distributional representation of the semantic relation which holds between target word pairs.

Although using a single deep neural network (as a distributional method) showed some improvement in capturing semantic relations, in order to get the advantage of the diversity among predictions of separately trained models, we use an ensemble of two artificial neural networks.

The ensemble is a general statistical enhancing technique to improve the representational capacity of the model. This enhancement helps to find a hypothesis which is independent of the space of the model from which it starts to learn.

First, we train two neural networks separately on data our labeled data and evaluate their test results, then put these two models in an ensemble and re-evaluate the result. Comparing two result sets shows 0.1 improvement in F1 score of learned hypothesis.

Models can be assembled in many different ways like boosting, bagging and stacking. We use stacking which involves training a learning algorithm to combine the predictions of several learning algorithms.

The advantage of stacking is to increase the prediction power of the classifier. As the using of another neural network above the weak learners in order to learn the final prediction imposes excess overhead, we use the simplest stacking method which is averaging. Averaging has no parameter, so no training is needed.

We transfer the input embedding vector of word pairs to dense-valued feature vectors, next feed these vectors to both ANN to compose their own distributional representation of them. At the final layer of each, a softmax classifier predicts the label of input sample.

Finally, a weighted averaging mechanism is used to decide the relation class in which input words participate.

3.1 Input of EoANN

Our inputs are raw lexical entries (multi word expressions are excluded) of Wordnets. We first transform every single word to its embedding vectors using word embedding.

Word embedding is a method to map words and phrases from space with one dimension per word, to a continuous low dimensional vector space. There are many word embedding frameworks. We use Fasttext (Piotr et al., 2017) which represents words as the sum of the n-gram vectors. This method is actually an extension of the continuous skip-gram model (Mikolov et al., 2013), which considers sub-word information as well. We denote the word embedding vector of word w by vw ∈ $\mathbb{R}$

Given *R (a, b)* as a sample of semantic relation triple in target Wordnet, *R* is the class of relation which connects a to b and v_a and v_b are the embedding vectors corresponding to them. The input vector and labels of our classifiers is the concatenation of word vectors:

$$h1(a,b) = [va{:}\, vb]$$
$$out(a,b) = R$$

3.2 Weak Learners Structure

We use both convolutional neural network and LSTM network in the simplest structure as our model base inducers. These two learners are chosen because of their power in capturing of

hierarchical patterns and the extraction of the temporal behavior.

The simple CNN inducer is composed of 3 main layers, a convolutional layer with 20 filters of size (1, 2), a pooling layer which is used to reduce the dimensions of feature map and finally a fully connected layer that flattens the results and passes it to a softmax classifier to decide which relation class the input belongs to.

LSTM neural network which we use as another weak learner is composed of a fully connected layer to encode 2-dimentional input feature vector to a dense flat vector, then passes its output to a LSTM layer with 200 memory units and a softmax classifier finally decides about data class label.

Combiner is responsible for getting the final decision by combining individual classifiers predictions. This component holds a majority voting among classifiers and declares the ultimate predicted label.

4 Datasets

In this study, we use four common data set to evaluate the performance of our model, FarsNet (Shamsfard, 2008), Princeton Wordnet (Miller, 1996), Root09 (Santus et al., 2015) and EVALution (Santus et al., 2016) as common semantic relation resources in Persian and English.

Table 1 shows the details about datasets, their relation class and number of instances used for train and test (90% for train and validation and 10% for test).

For embedding model, we used the Wikipedia dump in both English and Persian Languages. Our English word embedding model vocabulary contains 999,994 words and our Persian model has 347,636 words.

Fasttext embedding models had the following parameter set for training:

- Vector dimention:300
- Learning rate:0.04
- Min and max length of char n-gram:[3,6]
- Number of epochs:10

The rest of the parameters are as Fasttext default configuration.

data set	relation classes	# of instances
WordNet	Hypernym, Hyponym, Entailment, Cause, Instance-Hypernym, Instance-Hyponymy, Member, Holonym, Attribute	634,330
Farsnet	Hypernym, Hyponym, Antonym, Instrument, Domain, Instance-Hypernym, Instance-Hyponym Location, Patient	322,554
ROOT09	**Hypernym, co-Hyponym, Random**	**12,762**
EVALution	Hypernym, Antonym, meronym, possession, Attribute, Part Of	7,378

Table 1: data sets we use for evaluating our model, their main relation categories and the number of relation instances of each

5 Experimental Results

We use four wordnet-like data sets as our benchmark to evaluate the performance of our model:

We compared the results on root09 and EVALution with two most recent work, LexNet proposed by Vu and Schwartz (2018) and KSIM previously used and reported to be successful by Levy et al. (2015). We also compared our model performance with the previous effort result in extracting FarsNet relation in Persian which is a semi-automated pattern-based approach (Shamsfard et al., 2010).

Our experimental results which are summarized in table 2, show that our model can classify FarsNet word pairs relations with F1 score of 0.894 which is significant and it has an average F1 of 0.768 for WordNet relation classification.

As shown in table 2 the state-of-the-art models in the best case, has the F1 score of 0.606 on detecting relations in EVALution and 0.81 in ROOT09 and our model with F1 score of 0.655 for first and F1 score of 0.868 for last outperforms these methods.

Model	Data Set	Classifier feature composition	**F1**
EoANN	Root09	LSTM+**CNN Concat**	**0.868**

	EVALution	LSTM+**CNN Concat**	**0.655**
LexNet	Root09	RBF **Sum**+**SqDiff**	0.814
	EVALution	RBF **Concat**+**Mult**	0.6
KSIM	Root09	RBF **Sum**+**SqDiff**	0.723
	EVALution	RBF Concat+Mult	0.505

Table 2: best precision recall and F1 score for root09 and EVALution in 4 compared models

model	dataset	Model and Features	F1 score
EoANN	farsnet	LSTM+CNN **Concat**	**0.894**
Semi-auto	farsnet	Pattern-based +structured_based+ statistical	0.605

Table 3: best F1 score on farsnet

6 Conclusion and Future Works

Our objective in this research was to automatically classify lexical semantic relation employing the power of the simple but effective structured neural networks, which have shown their proficiency in many tasks of natural language processing (Collobert et al., 2011; Yao et al., 2013).

We used both LSTM and convolutional network to benefit the exhibition of temporal behavior by first and the extraction of the hierarchical pattern by last.

We also used the simplest distributional feature as input and entrusted the extraction of the most proper composition of features to the model.

In case of ROOT09 and EVALution our model has an improvement of 0.05 in F1 score from state of the art (LexNet). And for FarsNet dataset we have 0.11 improvement in F1 score.

The next step in extending lexical ontologies is to complete missed relation edges, then to learn new relation classes, which can be added to the target wordnet.

References

George A. Miller. 1995. WordNet: a lexical database for English. Communications of the ACM, 38(11):39-41.

Christiane Fellbaum 1998. WordNet: An Electronic Lexical Database. Cambridge, MA: MIT Press.

Mehrnoush Shamsfard, Akbar Hesabi, Hakimeh Fadaei, Niloofar Mansoory, Ali Famian, Somayeh Bagherbeigi, Elham Fekri, Maliheh Monshizadeh and Mostafa Assi. 2010. Semi-automatic development of farsnet; the Persian wordnet. Proceedings of 5th global WordNet conference, Mumbai, India; 29.

David Austen-Smith, Jeffrey S. Banks. 1996. Information aggregation rationality and the Condorcet jury theorem, American Political Science Review, vol. 90, pages. 34-45.

Mehrnoush Shamsfard and AA Barforoush. 2003. An introduction to HASTI: an ontology learning system. Proceedings of the iasted international conference artificial intelligence and soft computing, Acta Press, Galgary, Canada: 242-247.

Rion Snow, Daniel Jurafsky, and Andrew Y Ng. 2004. Learning syntactic patterns for automatic hypernym discovery. In NIPS.

Ronan Collobert, Jason Weston, Leon Bottou, Michael Karle, Koray Kavukcuoglu, Pavel Kuks. 2011. Natural language processing (almost) from scratch. Journal of machine learning research, pages 2493-537

Sebastian Riedel, Limin Yao, Andrew McCallum, and M. Benjamin Marlin. 2013. Relation extraction with matrix factorization and universal schemas. In Proceedings of NAACL-HLT 2013, pages 74–84.

David M. Powers 1998. Applications and explanations of Zipf's law. Association for Computational Linguistics: 151–160.

Zellig S. Harris. 1954. Distributional Structure. WORD, 10(2-3):146-162.

Tomas Mikolov, Ilya Sutskever, Kai Chen, Gregory S Corrado, and Jeffrey Dean. 2013. Distributed representations of words and phrases and their compositionality. In NIPS, pages 3111–3119.

Marco Baroni, Raffaella Bernardi, Ngoc-Quynh Do, and Chungchieh Shan. 2012. Entailment above the word level in distributional semantics. In Proceedings of EACL 2012, pages 23–32.

Peter D Turney and Patrick Pantel. 2010. From frequency to meaning: Vector space models of semantics. Journal of artificial intelligence research, 37:141–188.

Julie Weeds, Daoud Clarke, Jeremy Reffin, David Weir, and Bill Keller. 2014. Learning to distinguish hypernyms and co-hyponyms. In Proceedings of COL- ING 2014, the 25th International Conference on Computational Linguistics: Technical Papers, pages 2249–2259, Dublin, Ireland. Dublin City University and Association for Computational Linguistics.

Ruiji Fu, Jiang Guo, Bing Qin, Wanxiang Che, Haifeng Wang, and Ting Liu. 2014. Learning semantic hierarchies via word embeddings. In Proceedings of the 52nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers), pages 1199–1209, Baltimore, Maryland. Association for Computational Linguistics.

Koki Washio and Tsuneaki Kato. 2018. Neural Latent Relational Analysis to Capture Lexical Semantic Relations in a Vector Space. arXiv preprint arXiv:1809.03401.

Chengyu Wang, Xiaofeng He, and Aoying Zhou. 2017. A short survey on taxonomy learning from text corpora: Issues, resources and recent advances. In Proceedings of the 2017 Conference on Empirical Methods in Natural Language Processing, pages 1190– 1203.

Stephen Roller, Katrin Erk, and Gemma Boleda. 2014. Inclusive yet selective: Supervised distributional hypernymy detection. In Proceedings of COLING 2014, the 25th International Conference on Computational Linguistics: Technical Papers, pages 1025– 1036, Dublin, Ireland.

Vered Shwartz, Yoav Goldberg, and Ido Dagan. 2016. Improving hypernymy detection with an integrated path-based and distributional method. In Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers), pages 2389–2398, Berlin, Germany. Association for Computational Linguistics.

Stephen Roller and Katrin Erk. 2016. Relations such as hypernymy: Identifying and exploiting Hearst patterns in distributional vectors for lexical entailment. In Proceedings of the 2016 Conference on Empirical Methods in Natural Language Processing, pages 2163–2172, Austin, Texas. Association for Computational Linguistics.

Marti A Hearst. 1992. Automatic acquisition of hyponyms from large text corpora. In Proceedings of the 14th conference on Computational linguistics, pages 539–545.

Roberto Navigli and Paola Velardi. 2010. Learning word-class lattices for definition and hypernym extraction. In ACL, pages 1318–1327.

Guido Boella and Luigi Di Caro. 2013. Supervised learning of syntactic contexts for uncovering definitions and extracting hypernym relations in text databases. In Machine learning and knowledge discovery in databases, pages 64–79. Springer.

Ellie Pavlick and Marius Pasca 2017. Identifying 1950s American jazz musicians: Fine-grained isa extraction via modifier composition. In Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers), pages 2099–2109. Association for Computational Linguistics.

Tu Vu and Vered Shwartz. 2018. Integrating Multiplicative Features into Supervised Distributional Methods for Lexical Entailment. arXiv preprint arXiv:1804.08845.

Lior Rokach. 2009. Ensemble-based classifiers. Artificial Intelligence Review, 33(1-2):1-39.

Bojanowski Piotr, Edouard Grave, Armand Joulin and Tomas Mikolov. 2017. Enriching word vectors with subword information. Transactions of the Association for Computational Linguistics, 5, 135-146.

Enrico Santus, Frances Yung, Alessandro Lenci, and Chu-Ren Huang. 2015. Evalution 1.0: an evolving semantic dataset for training and evaluation of distributional semantic models. In Proceedings of the 4th Workshop on Linked Data in Linguistics (LDL- 2015), pages 64–69.

Enrico Santus, Alessandro Lenci, Qin Lu, and Sabine Schulte im Walde. 2014. Chasing hypernyms in vector spaces with entropy. In Proceedings of the 14th Conference of the European Chapter of the Association for Computational Linguistics, volume 2: Short Papers, pages 38–42, Gothenburg, Sweden. Association for Computational Linguistics.

Omer Levy, Steffen Remus, Chris Biemann, and Ido Dagan. 2015. Do supervised distributional methods really learn lexical inference relations? In Proceedings of the 2015 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, pages 970–976, Denver, Colorado. Association for Computational Linguistics.