Semi-supervised Multi-task Learning using Neural Networks for Semantic Relations Identification

Authors
Affiliation / Address line 1
Affiliation / Address line 2
Affiliation / Address line 3
email@domain

Abstract

Discovering whether words are semantically related and identifying the specific semantic relation that holds between them is of crucial importance for NLP as it is essential for tasks like query expansion in IR. Within this context, different methodologies have been proposed that either exclusively focus on a single lexical relation (e.g. hypernymy vs. random) or learn specific classifiers capable of identifying multiple semantic links (e.g. hypernymy vs. synonymy vs. random). In this paper, we propose another way to look at the problem that relies on the multi-task learning paradigm. In particular, we want to study whether the learning process of a given semantic relation (e.g. hypernymy) can be improved by the concurrent learning of another semantic relation (e.g. co-hyponymy). Within this context, we particularly examine the benefits of semi-supervised learning where just a few labelled examples are available for learning and many unlabelled samples exist. Preliminary results show that improvements can be obtained for specific pairs of semantic relations both in the case of supervised and semi-supervised learning.

1 Introduction

The ability to automatically identify semantic relations is an important issue for Information Retrieval (IR) and Natural Language Processing (NLP) applications such as question answering (Dong et al., 2017), query expansion (Kathuria et al., 2017), or summarization (Gambhir and Gupta, 2017). Semantic relations embody a large number of symmetric and asymmetric linguistic phenomena such as synonymy (bike $\leftrightarrow$ bicycle), co-hyponymy (bike $\leftrightarrow$ scooter), hypernymy (bike $\rightarrow$ tandem) or meronymy (bike $\rightarrow$ chain), but more can be enumerated (Vylomova et al., 2016).

Most approaches focus on modeling a single semantic relation and consist in deciding whether a given relation r holds between a pair of words (x,y) or not. Within this context, the vast majority of efforts (Snow et al., 2004; Roller et al., 2014; Shwartz et al., 2016; Nguyen et al., 2017a) concentrate on hypernymy considered the key organisation principle of semantic memory, but studies exist on antonymy (Nguyen et al., 2017b), meronymy (Glavas and Ponzetto, 2017) and co-hyponymy (Weeds et al., 2014). Another research direction consists in dealing with multiple semantic relations at a time and can be defined as deciding which semantic relation r_i (if any) holds between a pair of words (x,y). This multi-class problem is challenging as it is known that distinguishing between different semantic relations (e.g. synonymy and hypernymy) is difficult (Shwartz et al., 2016). Within the CogALex-V shared task[1] which aims at tackling synonymy, antonymy, hypernymy and meronymy, best performing systems have been proposed by (Shwartz and Dagan, 2016; Attia et al., 2016).

In this paper, we propose another way to look at the problem of semantic relation identification. First, symmetric similarity measures, which capture synonymy (Kiela et al., 2015) play an important role in hypernymy detection (Santus et al., 2017). Second, (Yu et al., 2015) show that learning term embeddings that take into account co-hyponymy similarity improves supervised hypernymy identification. Based on these findings, we propose to study whether the learning process of a given semantic relation can be improved by the concurrent learning of another relation, where semantic relations are either synonymy, co-hyponymy or hypernymy. For that purpose, we propose a simple multi-task learning strategy using a

[1] https://sites.google.com/site/cogalex2016/home/shared-task

hard parameter sharing neural network that takes as input a learning word pair (x,y) encoded as a feature vector representing the concatenation[2] of the respective word embeddings of x and y noted $\overrightarrow{x} \oplus \overrightarrow{y}$. The intuition behind our experience is that if the tasks are correlated, the neural network may learn a model jointly for them while taking into account the shared information which is expected to improve its generalization ability.

Within this context, we also propose to study the generalization capacity of the multi-task learning model based on a limited set of labelled word pairs and a large number of unlabelled samples, i.e. following a semi-supervised paradigm. As far as we know, most related works rely on the existence of a huge number of validated word pairs present in knowledge databases (e.g. WordNet) to perform the supervised learning process. However, such resources may not be available for specific languages or domains. Moreover, it is unlikely that human cognition and its generalization capacity rely on the equivalent number of positive examples. As such, semi-supervised learning proposes a much more interesting framework where unlabelled word pairs can massively[3] be obtained through selected lexico-syntactic patterns (Hearst, 1992) or paraphrase alignments (Dias et al., 2010). To test our hypotheses, we propose a simple self-learning strategy where confidently tagged unlabelled word pairs are iteratively added to the labelled dataset.

Preliminary results based on simple semi-supervised multi-task learning models with state-of-the-art word pairs representations (i.e. concatenation of word embeddings) over the gold standard dataset ROOT9 (Santus et al., 2016) show that improvements can be obtained for specific pairs of semantic relations both in the case of supervised and semi-supervised learning.

2 Related Work

Whether semantic relation identification has been tackled as a one-class or a multi-class problem, two main approaches have been addressed to capture the semantic links between two words (x,y): pattern-based and distributional. Pattern-based (a.k.a. path-based) methods base their decisions on the analysis of the lexico-syntactic patterns (e.g. X *such as* Y) that connect the joint occurrences of x and y. Within this context, earlier works have been proposed by (Hearst, 1992) (unsupervised) and (Snow et al., 2004) (supervised) to detect hypernymy. However, this approach suffers from sparse coverage and benefits precision over recall. To overcome these limitations, recent one-class studies on hypernymy (Shwartz et al., 2016) and antonymy (Nguyen et al., 2017b), as well as multi-class approaches (Shwartz and Dagan, 2016) have been focusing on representing dependency patterns as continuous vectors using long short-term memory (LSTM) networks. Within this context, successful results have been evidenced but (Shwartz et al., 2016; Nguyen et al., 2017b) also show that the combination of pattern-based methods with the distributional approach greatly improves performance.

In distributional methods, the decision whether x is within a semantic relation with y is based on the distributional representation of these words following the distributional hypothesis (Harris, 1954), i.e. on the separate contexts of x and y. Earlier works developed symmetric (Dias et al., 2010) and asymmetric (Kotlerman et al., 2010) similarity measures based on discrete representation vectors, followed by numerous supervised learning strategies for a wide range of semantic relations (Baroni et al., 2012; Roller et al., 2014; Weeds et al., 2014), where word pairs are encoded as the concatenation of the constituent words representations ($\overrightarrow{x} \oplus \overrightarrow{y}$) or their vector difference ($\overrightarrow{x} - \overrightarrow{y}$). More recently, attention has been focusing on identifying semantic relations using neural language embeddings, as such semantic spaces encode linguistic regularities (Mikolov et al., 2013). Within this context, (Vylomova et al., 2016) proposed an exhaustive study for a wide range of semantic relations and showed that under suitable supervised training, high performance can be obtained. However, (Vylomova et al., 2016) also showed that some relations such as hypernymy are more difficult to model than others. As a consequence, new proposals have appeared that tune word embeddings for this specific task, where hypernyms and hyponyms should be closed to each other in the semantic space (Yu et al., 2015; Nguyen et al., 2017a).

In this paper, we propose an attempt to deal with semantic relation identification based on a multi-task

[2] Best configuration reported in (Shwartz et al., 2016) for standard non path-based supervised learning.

[3] Eventhough with some error rate.

strategy. The closest approach to ours is proposed by (Attia et al., 2016), which develops a multi-task convolutional neural network for multi-class semantic relation classification supported by relatedness classification, and can be seen as a domain adaptation problem. Within the scope of our paper, we aim at studying semantic inter-relationships at a much finer grain and understanding the cognitive links that may exist between synonymy, co-hyponymy and hypernymy, that form the backbone of any taxonomy. Moreover, as far as we known, we propose the very first attempt to deal with semantic relation identification based on a semi-supervised approach.

3 Methodology

We propose to combine semi-supervised learning with multi-task learning in order to accurately identify semantic relations between words. The main diffulty of the task is limited training data, which are often specific to a domain due to the fact that they are manually extracted or labelled from resources of the domain. We believe that artificially expanding the training dataset using semi-supervised learning methods can improve the performance on the task. Further, categorizing the different types of relationships between words can provide useful information across the given tasks. For instance, (write an example to motivate multitask learning here). In such a case, we expect multi-task learning to be more efficient than simple one-vs-rest approaches due to sharing the parameter sharing mechanism which has been shown to transfer knowledge across tasks (Caruana, 1998).

3.1 Self-learning for semantic relations identificaiton

- Why use semi-supervised learning?
- Describe self-learning formally
- Do we use the most confident predictions/top-N predictions etc?
- How many examples are we using?

3.2 Multi-task learning

- Why use multi-task learning?
- Feed-forward network (why used nns?)
- Hard parameter sharing of the first layers (how many layers?)
- activation functs, sizes,
- which embeddings used?

4 Set Ups and Experiments

Our goal is to demonstrate the effect of(i) leveraging semi-supervised learning to incorporate noisy examples in our traning set, and (ii) applying multi-task learning when learning the decision function for the different tasks. For semi-supervised learning we experiment with self-training, which iteratively classifies and adds positive and negative examples to the training set. Multi-task learning is achieved while relying on feed-forward neural networks for learning the tasks and using had parameter sharing for the first layers of the network. We now present in more detail the semi-supervised and the multi-task learning setting we use.

4.1 RUMEN Data Set

In order to train and test our classifier, we built a balanced dataset of 18978 WordNet word pairs i.e. 6326 $\times$ $\{$synonyms, hyponyms/hypernyms and unrelated$\}$. All pairs were randomly selected based on WordNet 3.0 noun categories (Miller et al., 1990) such that hyponyms/hypernyms are not necessarily in direct relation and unrelated pairs have as most common parent the root of the hierarchy with a minimum path distance equals to 7.

		Coord Vs Random		Hyper Vs Random	
	System	Accuracy	MaF$_1$	Accuracy	MaF$_1$
ConceptNet	Majority Baseline	0.658	0.397	0.688	0.407
	Logistic Regression	0.890	0.875	0.850	0.804
	NN Baseline	0.904	0.894	0.890	0.831
	Self-learning	0.908	0.897	0.866	0.845
	Multitask learning	0.904	0.894	0.836	0.828
	Multitask learning + Self-learning	0.905	0.895	0.894	0.865

Table 1: ROOT9. ConceptNet Numberbatch embeddings. Accuracy and Macro F$_1$ score. Stratified and lexical split between train/test datasets. Coord Vs Random: Train: 441, Unlabeled: 947, Validation: 189 Test: 666 pairs. Hyper Vs Random: Train: 355, Validation: 153, Unlabeled: 762, Test: 600 pairs. The NN used is a simple feed-forward NN with two layers: Input 600 dimensions, hidden1 50 dimensions and output. Repeated experiments with two types of embeddings. The two parts of the tables are not comparable between them.

		Coord Vs Random		Hyper Vs Random		Mero Vs Random	
	System	Accuracy	MaF$_1$	Accuracy	MaF$_1$	Accuracy	MaF$_1$
ConceptNet	Majority Baseline	0.658	0.397	0.816	0.450	0.654	0.395
	Logistic Regression	0.890	0.875	0.890	0.760	0.780	0.742
	NN Baseline	0.904	0.893	0.913	0.833	0.784	0.756
	Self-learning	0.908	0.897	0.913	0.834	0.783	0.757
	Multitask learning	0.904	0.894	0.905	0.885	0.785	0.760
	Multitask learning + Self-learning	0.905	0.894	0.913	0.830	0.800	0.760

Table 2: BLESS. ConceptNet Numberbatch embeddings. Accuracy and Macro F$_1$ score. Stratified and lexical split between train/test datasets. The NN used is a simple feed-forward NN with two layers: Input 600 dimensions, hidden1 50 dimensions and output. I need to check again the code, because the multitask results are very good.

4.2 Results

In the table below we report preliminary results on applying semi-supervised learning via self learning.

5 Conclusions and Future Work

- If synonymy is not improving other tasks, it may be because similarity and relatedness are usually mixed as shown in (Kiela et al., 2015). So, hypernymy may gain from relatedness but not from synonymy.

- Future works:

- try lexical split of data sets (no word intersection between training, validation and test sets)
- introduce path-based with LSTM.
- semi-supervised learning with lots of related word pairs automatically extracted with paraphrase alignment.
- introduce new embeddings such as ConceptNet or Hypervec.
- deal with hypernymy such that it is always a 3 class problem: given two words x and y, tell if $x \rightarrow y$ (hypernymy), or $y \rightarrow x$ (hyponymy) or nothing. So in the training set, if $x \rightarrow y$ is a hypernym relation, we can also say that $y \rightarrow x$ is a hyponym relation.

- New input 1: vector concatenation + vector difference
- New input 2: vector concatenation + vector difference + similarity measures. We can start with this input: vector concatenation + vector difference + cos(vector1,vector2). Combine unsupervised metrics (e.g. AInfoSimba, Infosimba) with context vectors and path-based vectors to improve overall classification results. Indeed, (Santus et al., 2017) reminds that vector difference or concatenation captures either a specific domain (Vylomova et al., 2016) or a specific pattern (Roller and Erk, 2016) depending on the used word context. However, unsupervised methods are insensitive to these issues. So, we could take a great number of unsupervised metrics as features of a word pair, in the same way we did with Rumen.
- Include OUT word embeddings (Nalisnick et al., 2016) in a similar way as IN word embeddings as input to include contextual information. In OUT embeddings, words that have the same context behavior share a similar semantic space (e.g. “University” is near “Stanford” in that space).

References

Mohammed Attia, Suraj Maharjan, Younes Samih, Laura Kallmeyer, and Thamar Solorio. 2016. Cogalex-v shared task: Ghhh - detecting semantic relations via word embeddings.

Marco Baroni, Raffaella Bernardi, Ngoc-Quynh Do, and Chung chieh Shan. 2012. Entailment above the word level in distributional semantics. In *13th Conference of the European Chapter of the Association for Computational Linguistics (EACL)*, pages 23–32.

Rich Caruana. 1998. Multitask learning. In *Learning to learn*, pages 95–133. Springer.

Gaël Dias, Rumen Moraliyski, Jo ao Paulo Cordeiro, Antoine Doucet, and Helena Ahonen-Myka. 2010. Automatic discovery of word semantic relations using paraphrase alignment and distributional lexical semantics analysis. *Natural Language Engineering*, 16(4):439–467.

Li Dong, Jonathan Mallinson, Siva Reddy, and Mirella Lapata. 2017. Learning to paraphrase for question answering. In *Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 886–897.

Mahak Gambhir and Vishal Gupta. 2017. Recent automatic text summarization techniques: a survey. *Artificial Intelligence Review*, 47(1):1–66.

Goran Glavas and Simone Paolo Ponzetto. 2017. Dual tensor model for detecting asymmetric lexico-semantic relations. In *Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 1758–1768.

Zellig S. Harris. 1954. Distributional structure. *Word*, 10(2-3):146–162.

Marti Hearst. 1992. Automatic acquisition of hyponyms from large text corpora. In *14th Conference on Computational Linguistics (COLING)*, pages 539–545.

Neha Kathuria, Kanika Mittal, and Anusha Chhabra. 2017. A comprehensive survey on query expansion techniques, their issues and challenges. *International Journal of Computer Applications*, 168(12).

Douwe Kiela, Felix Hill, and Stephen Clark. 2015. Specializing word embeddings for similarity or relatedness. In *Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 2044–2048.

Lili Kotlerman, Ido Dagan, Idan Szpektor, and Maayan Zhitomirsky-Geffet. 2010. Directional distributional similarity for lexical inference. *Natural Language Engineering*, 16(4):359–389.

Tomas Mikolov, Kai Chen, Greg Corrado, and Jeffrey Dean. 2013. Efficient estimation of word representations in vector space. *CoRR*, abs/1301.3781.

George A. Miller, Richard Beckwith, Christiane Fellbaum, Derek Gross, and Katherine J. Miller. 1990. Introduction to wordnet: An on-line lexical database. *International Journal of Lexicography*, 3(4):235–244, January.

Eric Nalisnick, Bhaskar Mitra, Nick Craswell, and Rich Caruana. 2016. Improving document ranking with dual word embeddings. In *25th International Conference Companion on World Wide Web (WWW)*, pages 83–84.

Kim Anh Nguyen, Maximilian Köper, Sabine Schulte im Walde, and Ngoc Thang Vu. 2017a. Hierarchical embeddings for hypernymy detection and directionality. In *Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 233–243.

Kim Anh Nguyen, Sabine Schulte im Walde, and Ngoc Thang Vu. 2017b. Distinguishing antonyms and synonyms in a pattern-based neural network. In *15th Conference of the European Chapter of the Association for Computational Linguistics (EACL)*, pages 76–85.

Stephen Roller and Katrin Erk. 2016. Relations such as hypernymy: Identifying and exploiting hearst patterns in distributional vectors for lexical entailment. In *Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 2163–2172.

Stephen Roller, Katrin Erk, and Gemma Boleda. 2014. Inclusive yet selective: Supervised distributional hypernymy detection. In *25th International Conference on Computational Linguistics (COLING)*, pages 1025–1036.

Enrico Santus, Alessandro Lenci, Tin-Shing Chiu, Qin Lu, and Chu-Ren Huang. 2016. Nine features in a random forest to learn taxonomical semantic relations. In *10th International Conference on Language Resources and Evaluation (LREC)*, pages 4557–4564.

Enrico Santus, Vered Shwartz, and Dominik Schlechtweg. 2017. Hypernyms under siege: Linguistically-motivated artillery for hypernymy detection. In *15th Conference of the European Chapter of the Association for Computational Linguistics (EACL)*, pages 65–75.

Vered Shwartz and Ido Dagan. 2016. Cogalex-v shared task: Lexnet - integrated path-based and distributional method for the identification of semantic relations. *CoRR*, abs/1610.08694.

Vered Shwartz, Yoav Goldberg, and Ido Dagan. 2016. Improving hypernymy detection with an integrated path-based and distributional method. In *54th Annual Meeting of the Association for Computational Linguistics (ACL)*.

Rion Snow, Daniel Jurafsky, and Andrew Y. Ng. 2004. Learning syntactic patterns for automatic hypernym discovery. In *17th International Conference on Neural Information Processing Systems (NIPS)*, pages 1297–1304.

Ekaterina Vylomova, Laura Rimell, Trevor Cohn, and Timothy Baldwin. 2016. Take and took, gaggle and goose, book and read: Evaluating the utility of vector differences for lexical relation learning. In *54th Annual Meeting of the Association for Computational Linguistics (ACL)*, pages 1671–1682.

Julie Weeds, Daoud Clarke, Jeremy Reffin, David J. Weir, and Bill Keller. 2014. Learning to distinguish hypernyms and co-hyponyms. In *5th International Conference on Computational Linguistics (COLING)*, pages 2249–2259.

Zheng Yu, Haixun Wang, Xuemin Lin, and Min Wang. 2015. Learning term embeddings for hypernymy identification. In *24th International Joint Conference on Artificial Intelligence (IJCAI)*, pages 1390–1397.