Composite Semantic Relation Classification

Siamak Barzegar[1(✉)], Andre Freitas[2], Siegfried Handschuh[2], and Brian Davis[1]

[1] Insight Centre for Data Analytics, National University of Ireland, Galway, Ireland
{siamak.barzegar,brian.davis}@insight-centre.org
[2] Digital Libraries and Web Information Systems,
University of Passau, Passau, Germany
{andre.freitas,siegfried.handschuh}@uni-passau.de

Abstract. Different semantic interpretation tasks such as text entailment and question answering require the classification of semantic relations between terms or entities within text. However, in most cases it is not possible to assign a direct semantic relation between entities/terms. This paper proposes an approach for composite semantic relation classification, extending the traditional semantic relation classification task. Different from existing approaches, which use machine learning models built over lexical and distributional word vector features, the proposed model uses the combination of a large commonsense knowledge base of binary relations, a distributional navigational algorithm and sequence classification to provide a solution for the composite semantic relation classification problem.

Keywords: Semantic relation · Distributional semantic · Deep learning · Classification

1 Introduction

Capturing the semantic relationship between two concepts is a fundamental operation for many semantic interpretation tasks. This is a task which humans perform rapidly and reliably by using their linguistic and commonsense knowledge about entities and relations. Natural language processing systems which aspire to reach the goal of producing meaningful representations of text must be equipped to identify and learn semantic relations in the documents they process.

The automatic recognition of semantic relations has many applications such as information extraction, document summarization, machine translation, or the construction of thesauri and semantic networks. It can also facilitate auxiliary tasks such as word sense disambiguation, language modeling, paraphrasing, and recognizing textual entailment [5].

However it is not always possible to establish a direct semantic relation given two entity mentions in text. In the Semeval 2010 Task 8 test collection [5] for example 17.39 % of the semantic relations mapped within sentences were assigned with the label *"OTHER"*, meaning that they could not be mapped to the set of

F. Frasincar et al. (Eds.): NLDB 2017, LNCS 10260, pp. 406–417, 2017.
DOI: 10.1007/978-3-319-59569-6_49

9 direct semantic relations[1]. In many cases, the semantic relations between two entities can only be expressed by a composition of two or more operations. This work aims at improving the description and the formalization of the semantic relation classification task by introducing the concept of composite semantic relation classification, in which the relations between entities can be expressed using the composition of one or more relations.

This paper is organized as follows: Sect. 2 describes the semantic relation classification problem and the related work followed by the proposed composite semantic relation classification (Sect. 3), Sect. 4 describes the existing baseline models; while Sect. 5 describes the experimental setup and analyses the results, providing a comparative analysis between the proposed model and the baselines. Finally, Sect. 6 provides the conclusion.

2 Composite Semantic Relation Classification

2.1 Semantic Relation Classification

Semantic relation classification is the task of classifying the underlying abstract semantic relations between target entities (terms) present in texts [10]. The goal of relation classification is defined as follows: given a sentence S with the pairs of annotated target nominals e_1 and e_2, the relation classification system aims to classify the relations between e_1 and e_2 in given texts within the pre-defined relation set [5]. For instance, the relation between the nominal **burst** and **pressure** in the following example sentence is interpreted as **Cause-Effect** (e_2, e_1).

> The $< e_1 >$ *burst* $< /e_1 >$ has been caused by water hammer $< e_2 >$ *pressure* $< /e_2 >$.

2.2 Existing Approaches for Semantic Relation Classification

Different approaches have been explored for relation classification, including unsupervised relation discovery and supervised classification. Existing literature have proposed various features to identify the relations between entities using different methods.

Recently, Neural network-based approaches have achieved significant improvement over traditional methods based on either human-designed features [10]. However, existing neural networks for relation classification are usually based on shallow architectures (e.g., one-layer convolutional neural networks or recurrent networks). In exploring the potential representation space at different abstraction levels, they may fail to perform [15].

The performance of supervised approaches strongly depends on the quality of the designed features [17]. With the recent improvement in Deep Neural

[1] Cause-Effect, Instrument-Agency, Product-Producer, Content-Container, Entity-Origin, Entity-Destination, Component-Whole, Member-Collection, Communication-Topic.

Network (DNN), many researchers are experimenting with unsupervised methods for automatic feature learning. [16] introduce gated recurrent networks, in particular, Long short-term memory (LSTM), to relation classification. [17] use Convolutional Neural Network (CNNs). Additionally, [11] replace the common Softmax loss function with a ranking loss in their CNN model. [14] design a negative sampling method based on CNNs. From the viewpoint of model ensembling, [8] combine CNNs and recursive networks along the Shortest Dependency Path (SDP), while [9] incorporate CNNs with Recurrent Neural Networks (RNNs).

Additionally, much effort has been invested in relational learning methods that can scale to large knowledge bases. The best performing neural-embedding models are Socher(NTN)[12] and Bordes models (TransE and TATEC) [2,4].

3 From Single to Composite Relation Classification

3.1 Introduction

The goal of this work is to propose an approach for semantic relation classification using one or more relations between term mentions/entities.

"The $< e_1 >$ *child* $< /e_1 >$ was carefully wrapped and bound into the $< e_2 >$ *cradle* $< /e_2 >$ by means of a cord."

In this example, the relationship between *Child* and *Cradle* cannot be directly expressed by one of the nine abstract semantic relations from the set described in [5].

However, looking into a commonsense KB (in this case, ConceptNet V5.4) we can see the following set of composite relations between these elements:

$< e_1 >$ *child* $< /e_1 >$ *createdby* $\circ$ *causes* $\circ$ *atlocation* $< e_2 >$ *cradle* $< /e_2 >$

As you increase the number of edges that you can include in the set of semantic relations compositions (the size of the semantic relationship path), there is a dramatic increase in the number of paths which connect the two entities. For example, for the words *Child* and *Cradle* there are 15 paths of size 2, 1079 paths of size 3 and 95380 paths of size 4. Additionally, as the path size grows many non-relevant relationships (less meaningful relations) will be included.

The challenge in *composite semantic relation classification* is to provide a classification method that provides the most meaningful set of relations for the context at hand. This task can be challenging because, as previously mentioned, a simple KB lookup based approach would provide all semantic associations at hand.

To achieve this goal we propose an approach which combines *sequence machine learning models*, *distributional semantic models* and *commonsense relations knowledge bases* to provide an accurate method for composite semantic relation classification.

The proposed model (Fig. 1) relies on the combination of the following approaches:

i Use existing structured commonsense KBs define an initial set of semantic relation compositions.
ii Use a pre-filtering method based on the Distributional Navigational Algorithm (DNA) as proposed by [3]
iii Use sequence-based Neural Network based model to quantify the sequence probabilities of the semantic relation compositions. We call this model Neural Concept/Relation Model, in analogy to a Language Model.

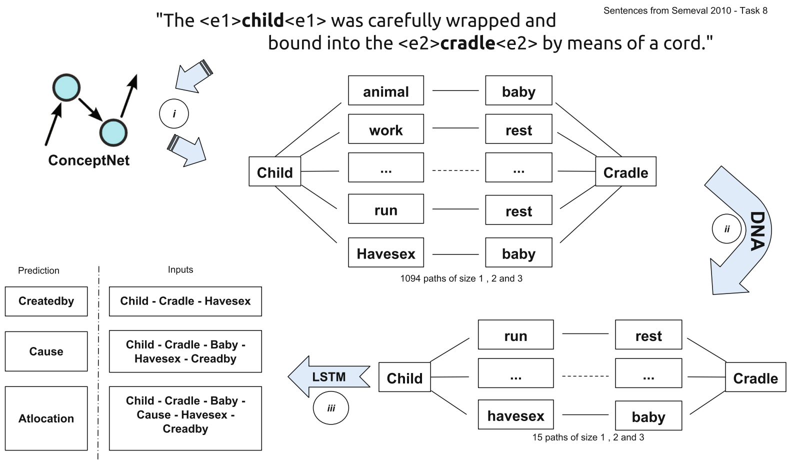

Fig. 1. Depiction of the proposed model relies on the combination of the our three approaches

3.2 Commonsense KB Lookup

The first step consists in the use of a large commonsense knowledge base for providing a reference for a sequence of semantic relations. ConceptNet is a semantic network built from existing linguistic resources and crowd-sourced. It is built from nodes representing words or short phrases of natural language, and labeled abstract relationships between them.

1094 paths were extracted from ConceptNet with two given entities (e.g. *child* and *cradle*) with no corresponding semantic relation from the Semeval 2010 Task 8 test collection (Fig. 1(i)). Examples of paths are:

- **child/canbe/baby/atlocation/cradle**
- child/isa/animal/hasa/baby/atlocation/cradle
- child/hasproperty/work/causesdesire/rest/synonym/cradle
- child/instanceof/person/desires/baby/atlocation/cradle
- child/desireof/run/causesdesire/rest/synonym/cradle
- **child/createdby/havesex/causes/baby/atlocation/cradle**

3.3 Distributional Navigational Algorithm (DNA)

The Distributional Navigational Algorithm (DNA) consists of an approach which uses distributional semantic models as a relevance-based heuristic for selecting relevant facts attached to a contextual query. The approach focuses on addressing the following problems: (i) providing a semantic selection mechanism for facts which are relevant and meaningful in a particular reasoning & querying context and (ii) allowing coping with information incompleteness in a huge KBs.

In [3] DSMs are used as a complementary semantic layer to the relational model, which supports coping with semantic approximation and incompleteness.

For large-scale and open domain commonsense reasoning scenarios, model completeness, and full materialization cannot be assumed. A commonsense KB would contain vast amounts of facts, and a complete inference over the entire KB would not scale to its size. Although several meaningful paths may exist between two entities, there are a large number of paths which are not meaningful in a specific context. For instance, the reasoning path which goes through (1) is not related to the goal of the entity pairs (the relation between *Child* of human and *Cradle*) and should be eliminated by the application of the Distributional Navigation Algorithm (DNA) [3], which computes the distributional semantic relatedness between the entities and the intermediate entities in the KB path as a measure of semantic coherence. In this case the algorithm navigates from $e1$ in the direction of $e2$ in the KB using distributional semantic relatedness between the target node $e2$ and the intermediate nodes en as a heuristic method (Fig. 2).

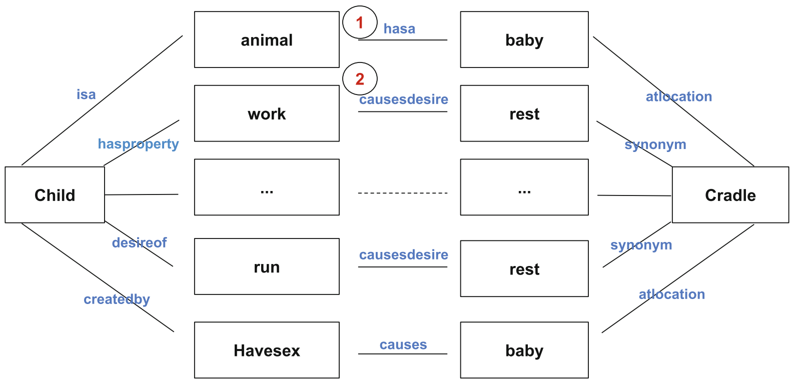

Fig. 2. Selection of meaningful paths

3.4 Neural Entity/Relation Model

The Distributional Navigational Algorithm provides a pre-filtering of the relations maximizing the semantic relatedness coherence. This can be complemented by a predictive model which takes into account the likelihood of a sequence of relations, i.e. the likelihood of a composition sequence. The goal is to systematically compute the sequence of probabilities of a relation composition, in a similar fashion to a language model. For this purpose we use a Long short-term memory (LSTM) recurrent neural network architecture (Fig. 3) [6].

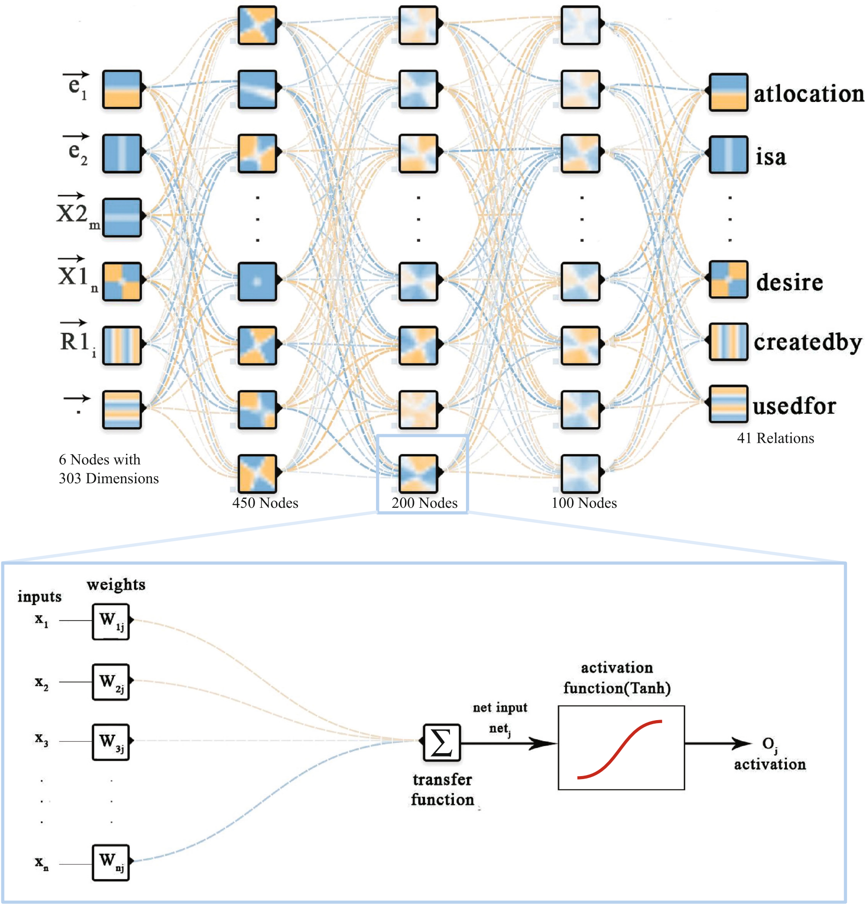

Fig. 3. The LSTM-CSRC architecture

4 Baseline Models

As baselines we use bigram language models which define the conditional probabilities between a sequence of semantic relations r_2 after entities r_1, i.e. $P(r_1 \mid r_2)$.

Algorithm 1. Composite Semantic Relation Classification

I : sentences of semeval 2010-Task 8 dataset
O : predefined entity pairs (e_1, e_2)
W : words in I
R : related relations of w
for all $s \in I$ **do**:
  $S \leftarrow$ *If entities of s are connected in a OTHER relation*
end for
for all $s \in S$ **do**:
  $ep \leftarrow$ *predefined entity pairs of s*
  $p \leftarrow$ *find all path of ep in ConceptNet (with maximum paths of size 3)*
  for all $i \in p$ **do**:
    $sq_i \leftarrow$ *avg similarity score between each word pairs [1]*
  end for
  $msq \leftarrow$ *find max sq*
  for all $i \in p$ **do**:
    filter i If $sq_i < msq - \frac{msq}{2}$
  end for
  $dw \leftarrow$ *convert s into suitable format for deep learning*
end for
$model \leftarrow$ *learning LSTM with dw dataset*

The performance of baselines systems is measured using the *CSRC*[2] *Cloze task*, as defined in Sect. 5.1 where we hold out the last relation and rate a system by its ability to infer this relation.

- **Random Model:** This is the simplest baseline, which outputs randomly selected relation pairs.
- **Unigram Model:** Predicts the next relation based on unigram probability of each relation which was calculated from the training set. In this model, relations are assumed to occur independently.
- **Single Model:**
  The single model is defined by [7]:

$$P(e_1 \mid e_2) = \frac{P(e_1, e_2)}{P(e_1)} \tag{1}$$

  where $P(e_1 \mid e_2)$ is the probability of seeing a_1 and a_2, in order. Let A be an ordered list of relations, $|A|$ is the length of A, For $i = 1, .., |A|$, define a_i to be the *ith* element of A. We rank candidate relations r by maximizing F(r,a), defined as

$$F(r, a) = \sum_{n-1}^{i} logP(r \mid a_i) \tag{2}$$

  With conditional probabilities $P(e_1 \mid e_2)$ calculated using (1).

[2] Composite Semantic Relation Classification.

- **Random Forest:** is an ensemble learning method for classification and other tasks, that operate by constructing a multitude of decision trees at training time and outputting the class that is the mode of the classes. Random decision forests correct for decision trees' habit of overfitting to their training set.

5 Experimental Evaluation

5.1 Training and Test Dataset

The evaluation dataset was generated by collecting all pairs of entity mentions in the Semeval 2010 task 8 [5] which had no attached semantic relation classification (i.e. which contained the relation label *"OTHER"*).

For all entities with unassigned relation labels, we did a *Conceptnet* lookup [13], where we generated all paths from sizes 1, 2 and 3 (number of relations) occurring between both entities(e_1 and e_2) and their relations (R).

For example:

$\mathbf{e1} - R1_i - \mathbf{e2}$
$\mathbf{e1} - R1_i - \mathbf{X1}_n - R2_j - \mathbf{e2}$
$\mathbf{e1} - R1_i - \mathbf{X1}_n - R2_j - \mathbf{X2}_m - R3_k - \mathbf{e2}$

where X contains the intermediate entities between the target entity mentions **e1** and **e2**.

In next step, the Distributional Navigational Algorithm (DNA) is applied over the entity paths [3]. In the final step of generating training & test datasets, the best paths are selected manually out of filtered path sets.

From 602 entity pairs assigned to the *"OTHER"* relation label in Semeval, we found $27,415$ paths between 405 entity pairs in ConceptNet. With the Distributional Navigation Algorithm (DNA), meaningless paths were eliminated, and after filtering, we have $2,514$ paths for 405 entity-pairs.

Overall we have 41 relations and 964 entities. All paths were converted into the following format which will be input into the neural network: $\mathbf{e}_1 - R1_i - \mathbf{X1}_n - R2_j - \mathbf{X2}_m - R3_k - \mathbf{e}_2$ (Table 1).

Table 1. Training data-set for CSRC model

Input	Classification
$\mathbf{e}_1$ $\mathbf{e}_2$ $\mathbf{X1}_n$	$\mathbf{R1}_i$
$\mathbf{e}_1$ $\mathbf{e}_2$ $\mathbf{X2}_m$ $\mathbf{X1}_n$ $\mathbf{R1}_i$	$\mathbf{R2}_i$
$\mathbf{e}_1$ $\mathbf{e}_2$ $\mathbf{X2}_m$ $\mathbf{R2}_i$ $\mathbf{X1}_n$ $\mathbf{R1}_i$	$\mathbf{R3}_i$

We provide statistics for the generated datasets in the Tables 2 and 3. In Table 3 our dataset is divided into a training set and a test set with scale ($75 - 25\,\%$), also we used $25\,\%$ of the training set for cross-validation, 3120 examples for training, 551 for validation and 1124 for testing. Table 2 shows statistics for test dataset of baseline models.

Table 2. Number of different length in the test dataset for baseline models

Test dataset	# Length 2	# Length 4	# Length 6
Baselines	245	391	432

Table 3. Dataset for LSTM model

Dataset	# Train	# Dev	# Test
CSRC	3120	551	1124

5.2 Results

To achieve the classification goal, we generated a LTSM model for the composite relation classification task. In our experiments, a batch size 25, and epoch 50 was generated. An embedding layer using Word2Vec pre-trained vectors was used.

In our experiment, we optimized the hyperparameters of the LSTM model. After several experiments, the best model is generated with:

– Inputs length and dimension are 6 and 303, respectively.
– Three hidden layers with 450, 200 and 100 nodes and $Tanh$ activation,
– Dropout technique (0.5),
– *Adam* optimizer.

We experimented our LSTM model with three different pre-training embedding word vector models:

– Word2Vec (Google News) with 300 dimensions
– Word2Vec (Wikipedia 2016) with 30 dimensions
– No pre-training word embedding

The accuracy for the configuration above after 50 epochs is shown in the table below (Table 4).

Table 4. Validation accuracy

CRSC	W2V Google_News	W2V Wikipedia	No pre training
Accuracy	0.4208	0.3841	0.2196

Table 5 contains the Precision, Recall, F1-Score and Accuracy. Between the evaluated models, the LSTM-CSRC achieved the highest F1 Score and Accuracy. The Single model achieved the second highest accuracy 0.3793 followed by Random forest model 0.3299. The LSTM approach provides an improvement of 9.86 % on accuracy over the baselines, and 11.31 % improvement on the F1-score. Random Forest achieved the highest precision, while LSTM-CSRC achieved the highest recall.

Table 5. Evaluation results on baseline models and our approach, with four metrics

Method	Recall	Precision	F1 score	Accuracy
Random	0.0160	0.0220	0.0144	0.0234
Unigram	0.0270	0.0043	0.0074	0.1606
Single	0.2613	0.2944	0.2502	0.3793
Random forest	0.2476	**0.3663**	0.2766	0.3299
LSTM-CSRC	**0.3073**	0.3281	**0.3119**	**0.4208**

Table 6. The extracted information from Confusion Matrix - Part 1

Relation	# Correct predicted	# Correct predicted rate	Relation	# Correct predicted	# Correct predicted rate
notisa	2	1	memberof	1	0.5
atlocation	172	0.67	hasa	24	0.393
notdesires	6	0.666	hassubevent	12	0.378
similar	5	0.625	partof	16	0.374
desires	36	0.593	haspropertry	12	0.375
hasprerequest	23	0.547	sysnonym	54	0.312
causesdesire	17	0.548	derivedfrom	20	0.307
isa	147	0.492	etymologically derivedfrom	6	0.3
antonym	68	0.492	capableof	13	0.26
instandof	46	0.479	motivationbygoal	3	0.25
usedfor	47	0.475	receivsection	5	0.238
desireof	5	0.5	createdby	4	0.2
hascontext	2	0.5	madeof	3	0.16
haslastsubevent	2	0.5	causes	3	0.15
nothasa	1	0.5	genre	1	0.11

The extracted information from confusion matrix show in Tables 6 and 7.

At table 6 *'Correctly Predicted'* column indicates the proportion of relations are predicted correctly, and *'Correct Prediction Rate'* column indicates the rate of correct predicted. For instance, our model predicts the relation *notisa* 100 % correct.

Table 7 shows the relations which are wrongly predicted (*'Wrongly Predicted'* columns).

Based on the results, the most incorrectly predicted relation is 'isa', which accounts for a large proportion of relations of the dataset (around 150 out of 550). In the second place is '$atlocation'$ relation (172 out of 550). The third place is the '$antonym'$ relation. On the other hand, some relations which are correctly unpredicted, can be treated as semantically equivalent to their prediction, where

Table 7. The extracted information from Confusion Matrix - Part 2

Relation	# Correctly predicted	Rate	Wrong relation 1	# False predicted for relation 1	Wrong relation 2	# False Predicted for relation 2	Wrong relation 3	# False predicted for relation 3
atlocation	172	0.67	antonym	20	Usedfor	17		
desire	36	0.593	isa	6	Capableof	6	Usedfor	5
hasprerequest	23	0.547	sysnonymy	4	antonym	3	atlocation	2
causesdesire	17	0.548	usedfor	7				
isa	147	0.492	atlocation	26	antonym	22	instanceof	22
antonym	68	0.492	isa	17	atlocation	9		
instandof	46	0.479	isa	27	atlocation	8		
usedfor	47	0.475	atlocation	26	isa	18		
hasa	24	0.393	antonym	11	usedfor	6		
hassubevent	12	0.378	causes	5	antonym	4		
partof	16	0.374	synonym	12	antonym	3	hasproperty	3
haspropertry	12	0.375	isa	8				
sysnonym	54	0.312	isa	31	hasproperty	17	atlocation	12
derivedfrom	20	0.307	isa	10	sysnonym	8	etymologically-derivedfrom	8
etymologically-derivedfrom	6	0.3	derivedfrom	6				
capableof	13	0.26	usedfor	13	isa	7		
motivatedbygoal	3	0.25	causes	3	hassubevent	2		
receivsection	5	0.238	atlocation	9	usedfor	3		
createdby	4	0.2	antonym	6	isa	5		
madeof	3	0.16	isa	7	antonym	3	hsaa	2
causes	3	0.15	causesdesire	6	hassubevent	4	derivedfrom	3

the assignment is dependent on a modelling decision. The same situation occurs for '$etymologicallyderivedfrom'$ and '$derivedfrom'$ relations.

Another issue is the low number of certain relations expressed int he dataset.

6 Conclusion

In this paper we introduced the task of composite semantic relation classification. The paper proposes a composite semantic relation classification model which combines *commonsense KB lookup*, a *distributional semantic based filter* and the application of a *sequence machine learning model* to address the task. The proposed LSTM model outperformed existing baselines with regard to f1-score, accuracy and recall. Future work will focus on increasing the volume of the training set for under-represented relations.

References

1. Barzegar, S., Sales, J.E., Freitas, A., Handschuh, S., Davis, B.: Dinfra: a one stop shop for computing multilingual semantic relatedness. In: Proceedings of the 38th International ACM SIGIR Conference on Research and Development in Information Retrieval, pp. 1027–1028. ACM (2015)

2. Bordes, A., Usunier, N., Garcia-Duran, A., Weston, J., Yakhnenko, O.: Translating embeddings for modeling multi-relational data. In: Advances in Neural Information Processing Systems, pp. 2787–2795 (2013)
3. Freitas, A., da Silva, J.C.P., Curry, E., Buitelaar, P.: A distributional semantics approach for selective reasoning on commonsense graph knowledge bases. In: Métais, E., Roche, M., Teisseire, M. (eds.) NLDB 2014. LNCS, vol. 8455, pp. 21–32. Springer, Cham (2004)
4. Garcia-Duran, A., Bordes, A., Usunier, N., Grandvalet, Y.: Combining two and three-way embedding models for link prediction in knowledge bases. J. Artif. Intell. Res. **55**, 715–742 (2016)
5. Hendrickx, I., Kim, S.N., Kozareva, Z., Nakov, P., Ó Séaghdha, D., Padó, S., Pennacchiotti, M., Romano, L., Szpakowicz, S.: 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, pp. 94–99. Association for Computational Linguistics (2009)
6. Hochreiter, S., Schmidhuber, J.: Long short-term memory. Neural Comput. **9**(8), 1735–1780 (1997)
7. Jans, B., Bethard, S., Vulić, I., Moens, M.F.: Skip n-grams and ranking functions for predicting script events. In: Proceedings of the 13th Conference of the European Chapter of the Association for Computational Linguistics, pp. 336–344. Association for Computational Linguistics (2012)
8. Liu, Y., Wei, F., Li, S., Ji, H., Zhou, M., Wang, H.: A dependency-based neural network for relation classification (2015). arXiv:1507.04646
9. Nguyen, T.H., Grishman, R.: Combining neural networks and log-linear models to improve relation extraction (2015). arXiv:1511.05926
10. Qin, P., Xu, W., Guo, J.: An empirical convolutional neural network approach for semantic relation classification. Neurocomputing **190**, 1–9 (2016)
11. dos Santos, C.N., Xiang, B., Zhou, B.: Classifying relations by ranking with convolutional neural networks. In: Proceedings of the 53rd Annual Meeting of the Association for Computational Linguistics and the 7th International Joint Conference on Natural Language Processing, vol. 1, pp. 626–634 (2015)
12. Socher, R., Chen, D., Manning, C.D., Ng, A.: Reasoning with neural tensor networks for knowledge base completion. In: Advances in Neural Information Processing Systems, pp. 926–934 (2013)
13. Speer, R., Havasi, C.: Representing general relational knowledge in conceptnet 5. In: LREC, pp. 3679–3686 (2012)
14. Xu, K., Feng, Y., Huang, S., Zhao, D.: Semantic relation classification via convolutional neural networks with simple negative sampling (2015). arXiv:1506.07650
15. Xu, Y., Jia, R., Mou, L., Li, G., Chen, Y., Lu, Y., Jin, Z.: Improved relation classification by deep recurrent neural networks with data augmentation (2016). arXiv:1601.03651
16. Xu, Y., Mou, L., Li, G., Chen, Y., Peng, H., Jin, Z.: Classifying relations via long short term memory networks along shortest dependency paths. In: Proceedings of Conference on Empirical Methods in Natural Language Processing (to appear) (2015)
17. Zeng, D., Liu, K., Lai, S., Zhou, G., Zhao, J., et al.: Relation classification via convolutional deep neural network. In: COLING, pp. 2335–2344 (2014)