SHIKEBLCU at SemEval-2020 Task 2: An External Knowledge-enhanced Matrix for Multilingual and Cross-Lingual Lexical Entailment

Shike Wang, Yuchen Fan, Xiangying Luo, Dong Yu✉ *
Beijing Language and Culture University, Beijing, China
{shikewang98,savannahfan98}@gmail.com
xiangying_luo@163.com
yudong@blcu.edu.cn

Abstract

Lexical entailment recognition plays an important role in tasks like Question Answering and Machine Translation. As important branches of lexical entailment, predicting multilingual and cross-lingual lexical entailment (LE) are two subtasks of SemEval2020 Task2. In previous monolingual LE studies, researchers leverage external linguistic constraints to transform word embeddings for LE relation. In our system, we expand the number of external constraints in multiple languages to obtain more specialised multilingual word embeddings. For the cross-lingual subtask, we apply a bilingual word embeddings mapping method in the model. The mapping method takes specialised embeddings as inputs and is able to retain the embeddings' LE features after operations. Our results for multilingual subtask are about 20% and 10% higher than the baseline in graded and binary prediction respectively.

1 Introduction

Lexical entailment (LE) refers to the hyponymy-hypernymy relation, also known as TYPE-OF, or IS-A, which is a fundamental asymmetric lexical relation (Vulić et al., 2017). It is a basic requirement for tasks like Question Answering (QA) and Recognizing Textual Entailment (RTE). And more general reasoning over cross-lingual and multilingual LE relationships can improve language understanding in multilingual contexts (Upadhyay et al., 2018). Cross-lingual LE recognition is crucial to tasks such as recognizing cross-lingual textual entailment (Conneau et al., 2018) and machine translation (Padó et al., 2009). Predicting binary and graded scores for multilingual and cross-lingual lexical entailment is the task of SemEval 2020 Task 2 (Glavaš et al., 2020).

There are two subtasks. Subtask A is to predict binary or graded LE on a monolingual pair of words, e.g., (*building, construction*) and the subtask is in six multiple languages (i.e., English, German, Italian, Croatian, Turkish, Albanian). Subtask B, predicting cross-lingual LE, gives a pair of words in two different languages with prefix, e.g., (*en_dinosaur, de_kreatur*[1]). There are 15 cross-lingual pairs as languages above combine with each other, i.e., DE-X, EN-X, IT-X, HR-X and SQ-X sets[2]. Each subtask includes both binary and graded prediction for a given pair of words. Binary prediction is to determine whether there is a LE relation between two concepts, while graded lexical entailment (GR-LE) measures the strength of LE relation on a continuous 0-6 scale (Vulić et al., 2017; Rei et al., 2018). For instance, (*apple, fruit*) gains a score of 6, while the pair (*apple, flower*) gets 1.2. This is because apple is more like a kind of fruit instead of flower. Since LE is an asymmetric relation, the score of (*fruit, apple*) is not the same as (*apple, fruit*).

Researches in monolingual GR-LE mainly focus on training LE-specialiased word embeddings based on external linguistic constraints (Vulić and Mrkšić, 2018; Kamath et al., 2019; Glavaš and Vulić, 2019). The constraints, namely external knowledge, are some synonymy (*pretty, beautiful*), antonymy (*nice, bad*) and lexical entailment word pairs (*sandwich, food*). These are useful for LE relation and are extracted from

* ✉ Corresponding author

[1]The word means *creature* in English

[2]The ISO codes are: German - DE, English - EN, Italian - IT, Croatian - HR, Turkish - TR, Albanian - SQ.

Proceedings of the 14th International Workshop on Semantic Evaluation, pages 255–262
Barcelona, Spain (Online), December 12, 2020.

lexical resources (such as WordNet). The size of external constraints decides the size of word embeddings get trained, thus for each language, the amount of external knowledge needs to be large enough.

For cross-lingual GR-LE prediction, previous works transfer the space from the source language to the target language (Glavaš and Vulić, 2019). The basis of the methods is to obtain unified bilingual word embeddings, which can also be trained by mapping two languages space into a shared one based a bilingual dictionary (Ruder, 2017). After getting bilingual embeddings, operations in cross-lingual models are similar with the monolingual one.

In our work, we apply massive external constraints to train specialised word embeddings for monolingual LE. And we introduce a bilingual word embedding mapping method on cross-lingual subtask. The inputs of the mapping method are the LE-specialised word embeddings which are outputs of subtask A. Several experiments are conducted to prove the effect of external constraints. Our contributions are as follows:

- We expand the number of external constraints in six given languages, and use them in the monolingual LE model, receiving great scores.
- We merge the bilingual word embeddings mapping method with monolingual LE model for the cross-lingual LE prediction.
- We conduct experiments with different number and kinds of external constraints to prove constraints' effectiveness.
- We achieve competitive scores of two subtasks among the competitors, both binary and graded prediction.

2 System Description

The components for our system are shown in Figure 1. We propose a system that focuses on transforming word embeddings for LE relation. For monolingual LE subtask, the inputs are word pairs and external lexical constraints in the same language. We employ LEAR (Vulić and Mrkšić, 2018) to train specialised input vectors based on external constraints. The method was used for English GR-LE. Here we use the model for all six languages with more external constraints added. After transforming the input embeddings, the outputs are monolingual LE-specialised word embeddings (Section 2.1). Next, to solve cross-lingual subtask, we treat the outputs of subtask A as inputs for a bilingual mapping method to map two different vector spaces into a shared one (Section 2.2). The final step of both subtasks is to score the entailment relation using the trained word embeddings (Section 2.3).

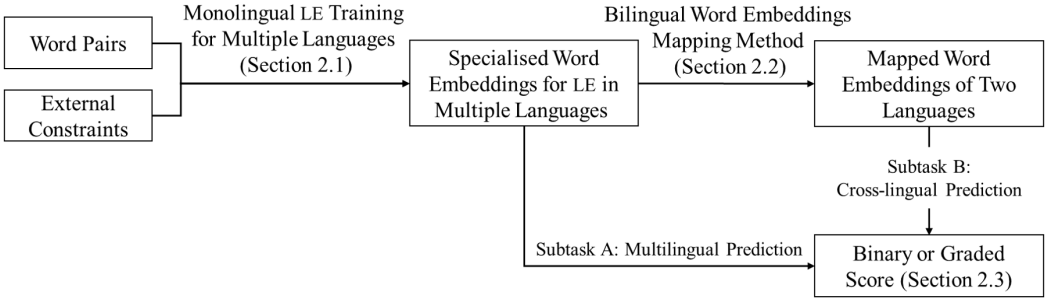
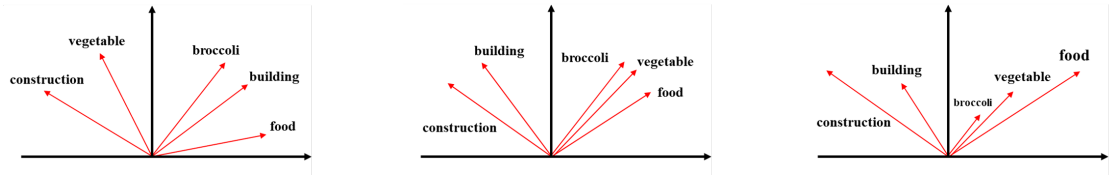


Figure 1: The flowchart for our system

2.1 Subtask A: Monolingual LE Training

The main part of our system for subtask A is the same as LEAR (Lexical Entailment Attract-Repel) (Vulić and Mrkšić, 2018). It is a post-processing method that fine-tunes word embeddings observed in external linguistic constraints. The constraints consists of synonymy pairs *S* such as (*nice, kind*), antonymy pairs *A* such as (*poor, rich*), and lexical entailment pairs *L* such as (*apple, fruit*), i.e., $C = S \cup A \cup L$. The model defines two symmetric objectives: the ATTRACT (*Att*) objective aims to pull synonymy pairs (ATTRACT



(a) The original pre-trained input word embeddings.

(b) Step 1: adjusting word embeddings directions by pulling similar words closer or pushing opposite words away.

(c) Step 2: adjusting norms to reflect concept level according to LE constraints.

Figure 2: The transformations of input word embeddings shown in 2D plane. Figure (a) shows the original input word embeddings. In figure (b), vectors of *(broccoli, food, vegetable)* and *(building, construction)* become closer according to external constraints. Next, as described in lexical entailment pairs *L*, *building* is a kind of *construction* and *broccoli* is a type of *vegetable* as well as a kind of *food*. Thus *food* and *construction* are higher-level concepts and their norms are larger than the others in figure (c).

pairs) closer, while the REPEL (*Rep*) objective pushes antonymy pairs (REPEL pairs) away from each other. Meanwhile, the model adjusts vector norms so that the higher-level concepts have larger norms and lower-level concepts have smaller norms in Euclidean space.

The set of K word pairs for which the *Att* or *Rep* score is to be computed is denoted by $\mathcal{B} = \{(\mathbf{x}_l^{(k)}, \mathbf{x}_r^{(k)}\}_{k=1}^{K}$. These pairs are referred as the *positive* examples. The set of corresponding *negative* examples T is created by coupling each positive ATTRACT example $(\mathbf{x}_l, \mathbf{x}_r)$ with a negative example pair $(\mathbf{t}_l, \mathbf{t}_r)$, where $\mathbf{t}_l$ is the vector closest (within the current batch in terms of cosine similarity) to $\mathbf{x}_l$, and $\mathbf{t}_r$ the vector closest to $\mathbf{x}_r$. The *Att* objective for a batch of ATTRACT constraints $\mathcal{B}_A$ is then given as:

$$Att(\mathcal{B}_A, T_A) = \sum_{k=1}^{K} \left[\tau \left(\delta_{att} + \cos\left(\mathbf{x}_l^{(k)}, \mathbf{t}_l^{(k)}\right) - \cos\left(\mathbf{x}_l^{(k)}, \mathbf{x}_r^{(k)}\right)\right) + \tau \left(\delta_{att} + \cos\left(\mathbf{x}_r^{(k)}, \mathbf{t}_r^{(k)}\right) - \cos\left(\mathbf{x}_l^{(k)}, \mathbf{x}_r^{(k)}\right)\right) \right]. \tag{1}$$

$\tau(x) = max(0, x)$ is the hinge loss and δ_{att} is the similarity margin imposed between the negative and positive vector pairs. Similarly, for each positive REPEL pair $(\mathbf{x}_l, \mathbf{x}_r)$, the negative example pair $(\mathbf{t}_l, \mathbf{t}_r)$ couples the vector $\mathbf{t}_l$ that is most distant from $\mathbf{x}_l$ and $\mathbf{t}_r$, most distant from $\mathbf{x}_r$. The *Rep* objective for a batch of REPEL word pairs $\mathcal{B}_R$ is then defined as:

$$Rep(\mathcal{B}_R, T_R) = \sum_{k=1}^{K} \left[\tau \left(\delta_{rep} + \cos\left(\mathbf{x}_l^{(k)}, \mathbf{x}_r^{(k)}\right) - \cos\left(\mathbf{x}_l^{(k)}, \mathbf{t}_l^{(k)}\right)\right) + \tau \left(\delta_{rep} + \cos\left(\mathbf{x}_l^{(k)}, \mathbf{x}_r^{(k)}\right) - \cos\left(\mathbf{x}_r^{(k)}, \mathbf{t}_r^{(k)}\right)\right) \right]. \tag{2}$$

In addition to these two objectives, LEAR defines a regularization term to preserve the useful semantic content from the original distributional vector space. Let $V(\mathcal{B})$ denote the set of distinct words in a constraint batch $\mathcal{B}$; the regularisation term is then: $Reg(\mathcal{B}) = \lambda_{reg} \sum_{x \in V(\mathcal{B})} \|\mathbf{y} - \mathbf{x}\|_2$, where $\mathbf{y}$ is the transformed vector of any vector $\mathbf{x}$, and λ_{reg} is the regularisation factor.

The most important objective of the method is an asymmetric distance-based objective which aims to rearrange norms of vectors. This is to obtain specialised vectors reflecting the asymmetry of the LE relation. We adopt the best-performing asymmetric objective from Vulić and Mrkšić (2018):

$$LE(\mathcal{B}_L) = \sum_{k=1}^{K} \frac{\|\mathbf{x}_l^{(k)}\| - \|\mathbf{x}_r^{(k)}\|}{\|\mathbf{x}_l^{(k)}\| + \|\mathbf{x}_r^{(k)}\|} \tag{3}$$

$\mathcal{B}_L$ denotes a batch of LE constraints. Finally, the full objective is then defined as:

$$J = Att(\mathcal{B}_S, T_S) + Rep(\mathcal{B}_A, T_A) + Att(\mathcal{B}_L, T_L) + LE(\mathcal{B}_L) + Reg(\mathcal{B}_S, \mathcal{B}_A, \mathcal{B}_L) \tag{4}$$

Figure 2 shows the whole transformation of word embeddings. First, the model adjusts the vectors direction according to *Att* objective or *Rep* objective. This step captures the symmetric similarity of word pairs. And next step is to rearrange vector norms according to $LE(\mathcal{B}_L)$, so that norms reveal the concepts' level. The final transformed word embeddings are saved for the following task.

2.2 Subtask B: Cross-lingual LE Training

The main idea of cross-lingual subtask is to map any two transformed vector spaces from subtask A into one shared space using a dictionary. Because the vector spaces are trained for LE relation, the symmetric and asymmetric features are retained in the mapped spaces. We follow the bilingual word embeddings mapping method proposed by Artetxe et al. (2018).

Let *X* and *Z* be the word embedding matrices in two languages for a given bilingual dictionary so that their *i*th row X_{i*} and Z_{i*} are the embeddings of the *i*th entry. The aim is to learn the transformation matrices W_X and W_Z so the mapped embeddings XW_X and ZW_Z are close to each other. The core step of the method is an orthogonal transformation[3].

The outputs of the method is the mapped vectors of two languages. After the bilingual word embedding training, we use the outputs and following function (Section 2.3) to compute the LE score.

2.3 Scoring Lexical Entailment

After obtaining LE-specialised word embeddings, LE scores are given by a distance function that reflects both the cosine distance between the vectors and the asymmetric difference between their norms (Vulić and Mrkšić, 2018) :

$$I_{LE}(\mathbf{x}, \mathbf{y}) = dcos(\mathbf{x}, \mathbf{y}) + \frac{\|\mathbf{x}\| - \|\mathbf{y}\|}{\|\mathbf{x}\| + \|\mathbf{y}\|} \tag{5}$$

$\mathbf{x}$ and $\mathbf{y}$ represent the vectors of any two words x and y in one subtask. We then normalise the results of the function to a range of (0,6) as a requirement. And for binary detection, we simply transform the graded score into the binary label, using a binarization threshold t. If $I_{LE}(\mathbf{x}, \mathbf{y}) < t$, we predict that the LE relation holds between two given concepts.

3 Experiments

3.1 Experimental Setup

Word Embedding. The English word embedding is the same with Vulić and Mrkšić (2018), which are Skip-Gram with Negative Sampling (SGNS-BOW2) vectors (Mikolov et al., 2013) trained by Levy and Goldberg (2014) on the Polyglot Wikipedia (Al-Rfou' et al., 2013). And for the rest of the required languages, we use word embeddings trained on Common Crawl and Wikipedia using FASTTEXT (Grave et al., 2018). All the vectors are 300-dim.

For all languages except English, we first shrink the input vector spaces according to word frequency lists that contains 50,000 words[4]. This is to make sure our model works smoothly and fast. However, this raises a problem that some words may get word embeddings in the original larger vector space whereas not in the reduced space. Also, we notice that there are some multiword expressions in the datasets, e.g., *macchina_per_scrivere* (*typewritter* in English), and they may not get the corresponding word embeddings either. To address these problems, we conclude in different ways of loading the word embedding.

First, we try to obtain the embeddings from the reduced vector space. If the input word is not in the reduced space, then there are three conditions: whether it is in the priginal larger space, whether it is a multiword expression, or neither. The process is shown in Figure 3. For the words made of multiple words, we separate them by underscores and try to get each part the corresponding word embeddings from the larger vector space. Once one of the parts is not in the space, the embedding of the multiword is randomly initialized. If all parts meet the criteria, the final embeddings of this multiword expression is the average of word embeddings of each part. We distribute words random word embeddings if they do not belong to the above situations.

External Resources. For all the provided languages, one part of external constraints is extracted from ConceptNet[5] (Speer et al., 2017) following the idea of LEAR. For each language, word pairs of synonym and antonym relations are included as symmetric resources and concepts of IsA relation are regarded as

[3]https://github.com/artetxem /vecmap

[4]https://github.com/hermitdave/FrequencyWords/tree/master/content/2016

[5]https://github.com/commonsense/conceptnet5/wiki/Downloads

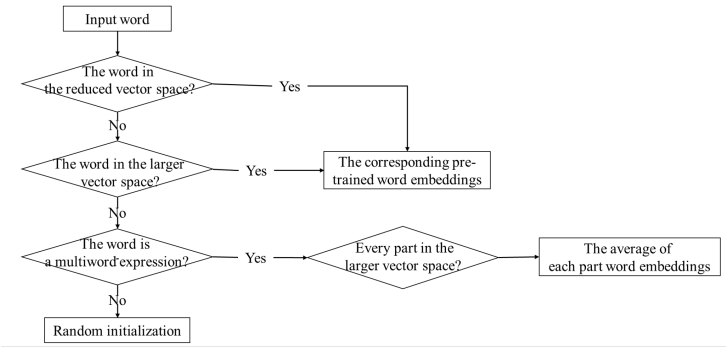


Figure 3: Different situations of loading input word embeddings

asymmetric LE constraints. The number of constraints for each language is displayed in Table 1a. And for English, the other part is the same set as LEAR (Vulić and Mrkšić, 2018): synonymy and antonymy constraints from (Zhang et al., 2014; Ono et al., 2015) are extracted from WordNet (Fellbaum, 1998) and Roget's Thesaurus (Kipfer, 2009), and asymmetric LE constraints are also extracted from WordNet. We add these 1,023,082 pairs of synonyms, 380,873 pairs of antonyms, and 1,545,630 LE pairs into English external lexical constraints.

Language	Antonyms	Synonyms	LE pairs	Total
EN	10108	48443	46068	104619
DE	1247	29699	21807	52753
IT	675	4992	1395	7062
TR	451	1048	107	1606
HR	106	703	57	866
SQ	55	298	6	359

(a) The total number of ConceptNet constraints used in model

Language	Antonyms	Synonyms	LE pairs	Total
EN	187258	593702	1520948	2301908
DE	68801	296700	467482	832983
IT	63121	252227	460798	776146
HR	38665	169151	319863	527679
TR	51110	180748	267724	499582
SQ	21542	138143	280620	440305

(b) The total number of all constraints (both ConceptNet and translated ones) used in model

Table 1: Summary of the number of constraints used in our model.

Expansion of Lexical Constraints. With more lexical constraints, more embeddings will get trained for LE relation. However, for languages like Turkish and Albanian, available external resources are not sufficient (see Table 1a). Therefore, instead of directly searching constraints in such language, we use Google Translator[6] to translate English constraints into other languages to expand their constraints. Furthermore, we apply the translations to construct the bi-dictionaries between every two languages for the cross-lingual subtask. The final number of lexical constraints applied in our model is displayed in Table 1b.

Training Setup. For subtask A, the hyperparameters are: $\delta_{att} = 0.6$, $\delta_{rep} = 0$, $\lambda_{reg} = 10^{-9}$. The threshold t = 3.5. The models are trained for 4 epochs with the AdaGrad algorithm (Duchi et al., 2011). We use the original experimental settings in the bilingual mapping method (Artetxe et al., 2018).

3.2 Results and Discussion

Our final results on the test set and the comparison with the baseline systems for two subtasks are shown in Table 2 and Table 3. The baseline model is from (Glavaš and Vulić, 2019).

Subtask A. Our system surpasses baseline in every monolingual language as a result of external constraints expansion. Our best performing language is the same as baseline: English, but our results are much higher than baseline by 18% and 8% in graded and binary LE scores respectively. Albanian, as the language performs the worst in baseline, we improve the results from 0.32 to 0.56 in graded LE and from 0.57 to 0.72 in binary LE. Even for our worst performing language, Turkish, the score of graded LE is 0.53 and the binary one is 0.70, while the baseline is 0.43 and 0.64.

[6]https://translate.google.com

Type	Model	en	de	it	hr	tr	sq
Graded	**Baseline:** GLEN	51.24	43.31	43.2	38.29	43.06	32.25
(Spearman ρ scores)	Our System	**69.63**	**63.17**	**63.6**	**58.85**	**52.52**	**56.45**
Binary	**Baseline:** GLEN	79.87	59.88	66.27	64.27	64.35	56.86
(F_1 scores)	Our System	**87.9**	**71.43**	**75.94**	**75.37**	**69.85**	**72.12**

Table 2: Results on Monolingual LE prediction of our system and the organizer's baselines.

Type	Model	de-en	de-hr	de-it	de-sq	de-tr	en-hr	en-it
Graded	**Baseline**: GLEN	50.4	40.8	**48.8**	39.6	**46.6**	36.8	**54.1**
(Spearman ρ scores)	Our System	**56.7**	**42.1**	45.7	**52.9**	45.8	**49.4**	53.2
Binary	**Baseline:** GLEN	74.3	62.6	63.7	58.8	**63.2**	65.9	77.2
(F_1 scores)	Our System	**80.6**	**64.0**	**63.8**	**61.4**	62.7	**78.8**	**81.4**

(a) Results on Cross-Lingual LE prediction (Part A).

Type	Model	en-sq	en-tr	hr-it	hr-sq	hr-tr	it-sq	it-tr	sq-tr
Graded	**Baseline**: GLEN	39.3	50.4	35.3	36.6	**40.2**	35.4	**47.4**	37.2
(Spearman ρ scores)	Our System	**55.9**	**51.0**	**43.2**	**49.0**	38.6	**54.4**	45.9	**51.0**
Binary	**Baseline:** GLEN	65.7	74.3	61.6	57.6	63.7	59.8	**67.6**	61.2
(F_1 scores)	Our System	**74.8**	**77.8**	**69.0**	**63.6**	**65.0**	**67.0**	67.5	**62.2**

(b) Results on Cross-Lingual LE prediction (Part B).

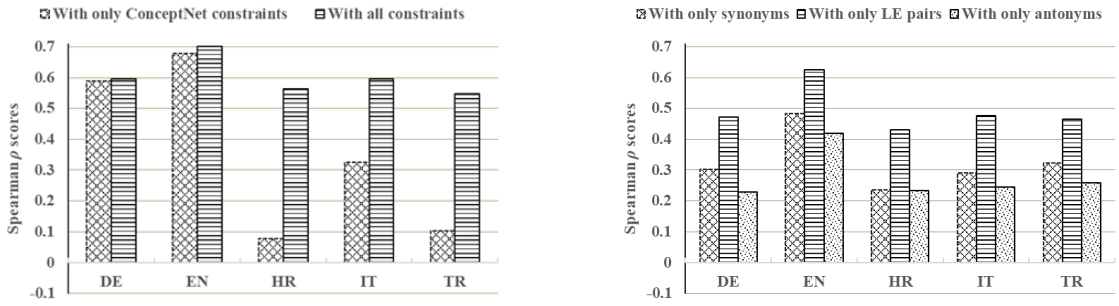
Table 3: Official submission results on Cross-Lingual LE prediction of our system and the organizer's baselines.

We also evaluate the system on the evaluation set to certify that an increase in the number of external lexical knowledge is beneficial to the model. We compare the performance of the model for graded LE prediction in two situations. One is training the model only with lexical constraints extracted from ConceptNet and the other is with all constraints including the translated constraints. Since Albanian is not published in the evaluation set, the comparisons only contain other five languages.

Figure 4a depicts the results for the five languages. The evaluation results reveal the similar pattern as the test results. The scores of English files are the highest, since the number of English constraints is much more than others. As Italian and German hold approximately equal number of constraints, their scores are close to each other. The lowest is Turkish for both evaluation and test sets.

We analyze other factors that may affect the results. We find that among all three kinds of external lexical constraints, LE pairs contribute the most to the model. Figure 4b demonstrates the importance of each kind of constraints. When training with the same number of different constraints, the model performs the best with the help of LE pairs, and the performance of *only with antonyms condition* is the worst. The number of Turkish LE constraints is the smallest among all languages. This explains why the number of Turkish lexical constraints is not the least, but the results are the lowest. And translations also influence the effectiveness of constraints, so we cannot get as useful Turkish LE word embeddings as others.

Subtask B. The circumstances for cross-lingual LE are complex. The bilingual mapping method is not effective for all the bilingual combinations. The baseline surpasses our system in DE-IT, DE-TR, EN-IT, HR-TR, IT-TR sets for graded LE task and DE-TR and IT-TR sets for the binary LE prediction. One of the reasons is that word embeddings from subtask A, as the inputs of mapping method, determine the results of the method. Turkish specialised embeddings are trained not as well as others, so graded results of combinations with Turkish are not outstanding enough.



(a) (b)

Figure 4: Spearman ρ scores of five languages training on evaluation set for graded LE. Figure (a) trains under two situations: 1) Training with constraints only extracted from ConceptNet. 2) Training with all constraints including translated constraints. Figure (b) shows the results of model training with only one kind of external constraints for each time. The number of constraints applied is the same.

4 Conclusion

We use LEAR model to fine-tune the input word vector spaces, and expand the number of external constraints used in the model to obtain more global word embeddings. The results demonstrate that with more knowledge added into the model, especially LE constraints, the relation among vectors will be more significant and beneficial to the monolingual LE relation prediction. And we apply above specialised word embeddings and the mapping method in cross-lingual word embeddings models to predict cross-lingual LE.

For future work, we think bi-dictionary and proper polysemy process should improve the performance. Besides, we will consider applying external constraints in both languages into the cross-lingual model since they work well in monolingual LE prediction.

Acknowledgments

This work is funded by the Humanity and Social Science Youth foundation of Ministry of Education (19YJCZH230) and the BLCU Academic Talents Support Program for the Young and Middle-Aged.

References

Rami Al-Rfou', Bryan Perozzi, and Steven Skiena. 2013. Polyglot: Distributed word representations for multilingual NLP. In *Proceedings of CoNLL*, pages 183–192, August.

Mikel Artetxe, Gorka Labaka, and Eneko Agirre. 2018. Generalizing and improving bilingual word embedding mappings with a multi-step framework of linear transformations. In *Proceedings of the Thirty-Second AAAI Conference on Artificial Intelligence*, pages 5012–5019.

Alexis Conneau, Ruty Rinott, Guillaume Lample, Adina Williams, Samuel Bowman, Holger Schwenk, and Veselin Stoyanov. 2018. XNLI: Evaluating cross-lingual sentence representations. In *Proceedings of EMNLP*, pages 2475–2485. Association for Computational Linguistics, October-November.

John Duchi, Elad Hazan, and Yoram Singer. 2011. Adaptive subgradient methods for online learning and stochastic optimization. *J. Mach. Learn. Res.*, 12(null):2121–2159, July.

Christiane Fellbaum. 1998. *WordNet*. MIT Press.

Goran Glavaš and Ivan Vulić. 2019. Generalized tuning of distributional word vectors for monolingual and cross-lingual lexical entailment. In *Proceedings of ACL*, pages 4824–4830, July.

Goran Glavaš, Ivan Vuli'c, Anna Korhonen, and Simone Ponzetto. 2020. SemEval-2020 task 2: Predicting multilingual and cross-lingual (graded) lexical entailment. In *Proceedings of the 13th International Workshop on Semantic Evaluation*. Association for Computational Linguistics.

Edouard Grave, Piotr Bojanowski, Prakhar Gupta, Armand Joulin, and Tomas Mikolov. 2018. Learning word vectors for 157 languages. In *Proceedings of LREC*.

Aishwarya Kamath, Jonas Pfeiffer, Edoardo Maria Ponti, Goran Glavaš, and Ivan Vulić. 2019. Specializing distributional vectors of all words for lexical entailment. In *Proceedings of the 4th Workshop on Representation Learning for NLP (RepL4NLP-2019)*, pages 72–83, August.

Barbara Ann Kipfer. 2009. *Roget's 21st Century Thesaurus (3rd Edition)*. Philip Lief Group.

Omer Levy and Yoav Goldberg. 2014. Dependency-based word embeddings. In *Proceedings of ACL*, pages 302–308, June.

Tomas Mikolov, Ilya Sutskever, Kai Chen, Greg Corrado, and Jeffrey Dean. 2013. Distributed Representations of Words and Phrases and their Compositionality. *arXiv e-prints*, page arXiv:1310.4546, October.

Masataka Ono, Makoto Miwa, and Yutaka Sasaki. 2015. Word embedding-based antonym detection using thesauri and distributional information. In *Proceedings of NAACL*, pages 984–989, May–June.

Sebastian Padó, Michel Galley, Dan Jurafsky, and Christopher D. Manning. 2009. Robust machine translation evaluation with entailment features. In *Proceedings of ACL*, pages 297–305, August.

Marek Rei, Daniela Gerz, and Ivan Vulić. 2018. Scoring lexical entailment with a supervised directional similarity network. In *Proceedings of ACL*, pages 638–643, July.

Sebastian Ruder. 2017. A survey of cross-lingual embedding models. *CoRR*, abs/1706.04902.

Robyn Speer, Joshua Chin, and Catherine Havasi. 2017. Conceptnet 5.5: An open multilingual graph of general knowledge. In *Proceedings of AAAI Conference on Artificial Intelligence*, pages 4444–4451.

Shyam Upadhyay, Yogarshi Vyas, Marine Carpuat, and Dan Roth. 2018. Robust cross-lingual hypernymy detection using dependency context. In *Proceedings of NAACL*, pages 607–618. Association for Computational Linguistics, June.

Ivan Vulić and Nikola Mrkšić. 2018. Specialising word vectors for lexical entailment. In *Proceedings of NAACL-HLT*, pages 1134–1145, June.

Ivan Vulić, Daniela Gerz, Douwe Kiela, Felix Hill, and Anna Korhonen. 2017. HyperLex: A large-scale evaluation of graded lexical entailment. *Computational Linguistics*, 43(4):781–835, December.

Jingwei Zhang, Jeremy Salwen, Michael Glass, and Alfio Gliozzo. 2014. Word semantic representations using Bayesian probabilistic tensor factorization. In *Proceedings of EMNLP*, pages 1522–1531.