Automated Ontology Matching and Construction using Cross-lingual Embeddings

Anonymous ACL submission

Abstract

There exist numerous structured text representations such as taxonomies, ontologies and lexical databases. Low-resource languages often lack such resources. In this paper we propose a method for matching and constructing ontologies, taxonomies and other forms of graph-trees with textual information using hierarchical information and cross-lingual embeddings.

1 Introduction

There are numerous structured information representations containing texts in forms of titles, descriptions or definitons. Among them we can name ontologies, taxonomies, lexical databases such as WordNet (Miller, 1995). Many of them exist only for English. Many researchers have tried automatically converting such resources from English into their languages. Mostly attempts were focused on using machine translation engines, bilingual dictionaries or parallel corpora (Khodak et al., 2017; Neale, 2018). All this resources are very rare except most popular languages. Some works used word embeddings, which proved to be a powerful tool for dense text representations after papers by Bengio (Bengio et al., 2003) and Mikolov (Mikolov et al., 2013a). However, first word vector representations models were monolingual only. Soon researchers proposed cross-lingual word embedding models (Mikolov et al., 2013b; Lazaridou et al., 2015). Unfortunately, most of the early works in this domain relied on massive parallel corpora. This solution is not really applicable to low-resource languages due to the lack of available data.

Learning mappings between embeddings from different languages or sources has proven to be a rather efficient method for solving this problem to some extent (Ruder et al., 2017).

Alexis Conneau et al. (Conneau et al., 2017) have published a programming library called MUSE to map embeddings from two different sources into a single space. They have reached 81.7% accuracy for English-Spanish and 83.7% for Spanish English pairs for top-1500 source queries in a completely unsupervised mode. For English-Russian and Russian-English their results are not as high and they achieved accuracy of 51.7% and 63.7% respectively. Their FastText embeddings were trained on respective Wikipedia datasets for each corresponding language and do not

Artetxe, Labaka and Agirre have investigated into limitations of MUSE and show its results to be low for some language pairs, e.g. English-Finnish (0.38% accuracy). They also present their own solution called Vecmap (Artetxe et al., 2018) that outperforms MUSE for this task. It gets 37.33% for Spanish-English on average of 10 runs and 37.60% as the best result (they estimate MUSE result to be 21.23% on average of 10 runs and 36.20% at best) and 32.63% on average for the English-Finnish language pair.

In this paper we propose a method for matching ontologies, taxonomies and other forms of graph-trees with textual information using hierarchical information and cross-lingual embeddings. It makes it possible to match ontology graphs containing textual descriptions in different languages. As the example of such taxonomy we use national product classifications: US NIGP-5 and Russian OKPD2. Matching national product classifications is of extreme importance because it facilitates worldwide trade and helps in bridging the gap between inconsistent product standards of the world economies. According to the UN (uns) there are at least 909 (the list seems incomplete - e.g. the currently used OKPD2 for Russia is not listed) classifications from 159 countries and most of them except the most prominent ones are unaligned. OKPD2 is a Russian national classification for goods and services introduced in 2014. It has a four-level hierarchy. Categories consist of a code and its description (e.g. 01.11.11.112 - Seeds of winter durum wheat where code 01.11.1 corresponds to "Wheat"). NIGP-5 is its 2-level US analogue (e.g. 620-80 would be "pens" and 620 – "office supplies").

Taxonomy/Ontology matching is a rather challenging problem because of the vast number of possible variants of matching which is challenging even for specialists and requires a lot of time. This is also made more difficult by the fact that matching is not one-by-one but many-to-many (some categories from one product classification may be broad enough to correspond to several categories from the other classification) and should be made across several hierarchy levels. Moreover, it requires expert and language knowledge which is especially difficult to come by in cases of rare languages and taxonomies. Thus, our product classification matching algorithm should satisfy several criteria.

Due to the lack of resources for rare languages the algorithm should be:

1. be language independent

2. not require parallel texts

   Due to the lack of expert knowledge the algorithm should be:

3. be fully or partially unsupervised

2 Related work

There are numerous papers concerned with taxonomies and ontologies matching. There is even a dedicated taxonomy matching competition Ontology Alignment Evaluation Initiative (Shvaiko et al.). The winner of the last two years is the model called the AgreementMakerLight (Faria et al., 2013). It is monolingual only and relies on hash and word matching. Unsupervised WordNet construction has been demonstrated by Khodak in (2017). They use bilingual dictionaries and word embeddings to build Russian WordNet as in our case. The problem of matching product taxonomies was also studied by Gordeev et al. in (2018).

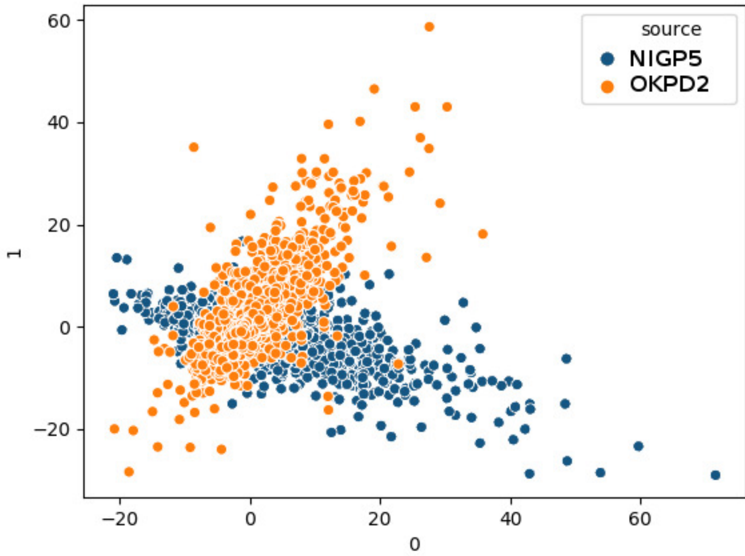

Figure 1: PCA visualisation of averaged FastText MUSE vectors for OKPD2 and NIGP-5 taxonomies

3 Cross-lingual embeddings

MUSE is based on the work by Conneau et al. (Conneau et al., 2017). It consists of two algorithms. The first one which is used only in unsupervised cases is a pair of adversarial neural networks. The first neural network is trained to predict from which distribution $\{X, Y\}$ embeddings come. The second neural networks is trained to modify embeddings X multiplying it by matrix W to prevent the first neural network from making accurate discriminations. Thus, at the end of the training we get a matrix WX which is aligned with matrix Y.

The second method is supervised and the aim is to find a linear mapping W between embedding spaces X and Y which can be solved using Orthogonal Procrustes problem:

$$W^* = argmin_W ||WX - Y||_F = UV^T$$

where UV^T is derived using singular value decomposition SVD$(YX^T) = U\Sigma V^T$ This method is used iteratively with the default number of iterations in MUSE equal to 5. As Søgaard, Ruder and Vulić state Procrustes refinement relies on frequent word pairs to serve as reliable anchors.

Conneau et al. also apply cross-domain similarity local scaling to reduce the hubness problem to which cross-lingual embeddings are prone to

(Dinu et al., 2015). It uses cosine distance between a source embedding and k-target embeddings (the default k in MUSE is 10) instead of the usual cosine distance to generate a dictionary.

$$sim_{source/target} = \frac{1}{k}\sum_{i=1}^{K} cos(x, nn_i)$$

$$CSLS(x, y) = 2cos(x, y) - sim_{source}(x) - sim_{target}(y)$$

Vecmap (Artetxe et al., 2018) is close in its idea to the Procrustes refinement, it computes SVD-factorization SVD$(YX^T) = U\Sigma V^T$ and replaces X and Y with new matrices $X' = U$ and $Y' = V$. The authors also propose normalization and whitening (sphering transformation). After applying whitening new matrices are equal to: $X' = (X^TX)^{-\frac{1}{2}}$ and $Y' = (Y^TY)^{-\frac{1}{2}}$

Jawanpuria et al. (Jawanpuria et al., 2018) propose a method which is also based on SVD-factorization but in smooth Riemannian manifolds instead of Euclidean space.

Ivan Vulić, Wim De Smet and Marie-Francine Moens used BiLDA for cross-language information retrieval which is similar to the task of classification matching. In this LDA variant topic distributions are considered to be equivalent for same articles from different languages (Vulić et al., 2013). However, in our case this method is unlikely to perform because LDA requires longer texts (Yan et al., 2013). Word embeddings methods are preferable for short texts (Maslova and Potapov, 2017).

3.1 Simplistic use of word embeddings for graph representations

In this paper we use a very simple baseline for representing graphs. We represent higher levels of hierarchy as averaged embeddings of lower levels, thus we use a bottom-up approach. We used cross-lingual embeddings in a single vector space provided by the authors of MUSE (i.e. vectors for "cat" and its Russian translation "кот" are close to each other). Using hierarchical information may be beneficial in a range of text classification tasks (Škrlj et al., 2019) as well as in computer vision problems such as ImageNet (Deng et al.). It was also shown by Yang (2018) that using graph and hierarchical is beneficial for many downstream tasks such as SQuAD.

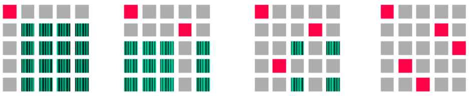

Figure 2: the Hungarian algorithm visualisation

4 Graph Matching

There are many methods for graph matching. Many of them are based on graph edit distance. However, they usually exhibit strong requirements for graph isomorphism. Unfortunately, most taxonomies do not satisfy this requirement (e.g. even in cases of a WordNet one word may have different synsets that may be absent from the other language). Inexact matching methods that overcome this constraint are based on spectral decompositions of graphs using eigenvalues or on weighted graph matchings (Conte et al., 2004). Given that we deal with cross-lingual embeddings it seemed reasonable to adapt a weight matching algorithm for our task. Thus, we used our hierarchical modification of the so-called Hungarian algorithm (Lawler, p. 201) (Riesen and Bunke, 2010, p. 48). We also compared it with the greedy method, where we just select the vector with the highest similarity.

4.1 The Hungarian method

We use a hierarchical version of the Hungarian algorithm. The algorithm takes as the input a similarity matrix of the size $n \times m$. If $m < n$, the matrix is transposed. As given by Brauner et al. in 2004 the algorithm looks the following way:

1. For the initial weight matrix subtract the minimum from each row and then subtract the minimum from each column.

2. Construct a maximum independent set and a minimal cover of same cardinality k. Exit if k = n.

3. Let h be the minimum of all non covered elements. Add $h = 2$ to all elements in each covering line. Then subtract $h = 2$ from each noncovered line.This results in h being subtracted from all noncovered elements, and h being added to all doubly covered elements. All entries that are only covered once remain unchanged.

4. Goto step 2.

For each level of the hierarchy we repeat the algorithm. For nodes that were invalidated at the previous (because they correspond to another branch of the tree) similarities are set to 0. If yet they are chosen by the algorithm we consider that this node is not present in the graph. Thus, we overcome the problems between taxonomies/ontologies.

5 Data

5.1 Taxonomies

Table 1: Taxonomy examples

Category code	Category description (with translation from Russian)	Bid description
325-25	Dog and Cat Food	Dog Food: Blue Buffalo Chicken and Brown Rice Food
43.31.10	Работы штукатурные Plastering Works	Overhaul of the Basement Of The Administration Building

In this study we use two national product classifications as examples of cross-lingual taxonomies. Both NIGP-5 and OKPD2 are used to classify products and services. However, they differ in the way products are described (two-level vs four-level hierarchy) as well as in the amount of described categories (8700 for NIGP-5 (wik, b) vs 17416 for OKPD2 (wik, a)). It means that two graphs that might describe these product classifications are not isomorphic (contain the same number of graph vertices connected in the same way and may be transformed into one another) by itself. It does not imply that they may not be made isomorphic by disregarding some vertices (e.g. using some threshold or similarity measure) and then aligned using graph matching methods but it complicates their alignment. It should be also noted that some notions from one classification may not exist in the other (e.g. popular in Russia curd snacks and traditional Russian felt footwear 'valenki' do not appear in NIGP-5).

The data for the Russian taxonomy OKPD2 was collected from them Russian state website Goszakupki [1], which contains purchases made by state entities. The data for the US national classification was collected from the US state website data.gov [2]. We have used only marketplace bids by the state of Maryland because they were the only found entries that contained bids descriptions not matching code-descriptions that are required for training Doc2Vec. Extracts from taxonomies can be seen in Table 1.

5.2 WordNet

We use English WordNet provided by the NLTK package (Miller, 1995; Bird, 2006). It contains 117'659 synsets. Among them the vast majority does not have hyponyms (97'651) and 30'062 lacks hypernyms (29'716 lacks any hierarchical relations). In this study we examined only nouns (82'115). Because of the small size of the MUSE model we had to use only 22'566 synsets that are included both in the model and in the WordNet.

6 Experiments

In this paper we have conducted two experiments. In the first experiment we compared different methods for products taxonomy matching. In the second we used the same methods for unsupervised WordNet construction.

6.1 NIGP and OKPD2 matching

Several methods and their combinations were used for mapping taxonomy embeddings. As the first method as described by (Gordeev et al., 2018) we tried to train doc2vec embeddings and map them directly using Vecmap or MUSE. However, due to the differences between graphs we failed to reach any meaningful result.

We also made a simple baseline with category descriptions transformed into English using Google Translate. We considered category descriptions from each taxonomy as bags of words, then for a set of words from the first taxonomy we calculated averaged similarity to all category descriptions from the second taxonomy and chose the category from the second taxonomy with the largest similarity. Thus, our method resembles Monge-Elkan similarity (Christen, 2012, p. 111):

$$mapping\{A_i, B\}_{i=1}^{|A|} = max_{j=1}^{|B|}\{sim(A_i, B_j)\}$$

where

$$sim = \frac{|A_i \cap B_j|}{2} + \frac{|B_j \cap A_i|}{2}$$

We used our custom similarity function to fine the function in the cases when the first set of strings is short in comparison with the second set (or the opposite).

[1] www.zakupki.gov.ru

[2] https://www.data.gov/

We also used hierarchical information to modify mappings from direct string comparison and averaged Word2vec descriptions.

For taxonomy matching using cross-lingual embeddings we tried two approaches: top-down and bottom-up. For the top-down approach we took upper-level category descriptions and transformed them into embeddings using pre-trained MUSE embeddings (Conneau et al., 2017). We used the averaging scheme (we checked various embedding weighting schemes (e.g. TF-IDF) but did not observe any considerable difference.) For the bottom-up approach the lowest level of hierarchy was also attained with averaging description embeddings. Upper-level vectors were gained via averaging their constituent node embeddings. As in the top-down approach. After getting a vector for each category we built layer-wise similarity matrices between taxonomies. Then we applied either Hungarian or greedy method for matching the taxonomies. After getting closest categories for the first layer, we looked only for categories that corresponded to the category chosen at the upper level (e.g. if the chosen category code is 64.12 at the next level we look only for categories 64.12.1, 64.12.2).

6.2 Russian WordNet construction using cross-lingual word embeddings

We use an extension approach where we take the existing English WordNet and transform its synsets into Russian (Neale, 2018). As with taxonomies we built a Wordnet tree using hypernym-hyponym relations between synsets. Also we build a list of Russian nouns using the rnnmorph library [3]. Then for the bottom WordNet layer we looked for the closest Russian nouns using most-similar function (cosine similarity) provided by the library Gensim (Řehůřek and Sojka, 2010). After that, just as in the case with taxonomy matching we constructed embeddings for the upper level averaging bottom-level vectors. We used the same hyponym-hyperonym relations as for English.

7 Annotation procedure

Mappings made by all methods were manually annotated on random 5% of examples according to the corresponding similarity metric (cosine distance for vectors and our string similarity function for strings similarity). The annotation included two classes: True, False. If examples were too specific or too broad (if the annotator was aware of this fact), the match was considered incorrect.

8 Results

8.1 Taxonomy matching

Table 2: Taxonomy matching results

Method	Accuracy %
bottom-up hungarian	47.5
bottom-up greedy	45
top-down hungarian	33
top-down greedy	25.5
translated strings matching	41

As can be seen from Table 2 the hierarchical Hungarian method and bottom-up graph embedding representations both help in boosting the results of taxonomy matching. Still translated strings matching is quite a strong baseline which may be the preferred solution when there is a translation engine for the studied language. In the other case the procedure would be: first, to train word-embeddings using some corpora from a common domain (e.g. Wikipedia), then align them using MUSE or Vecmap. After that, those embeddings may be used to map category descriptions. It should be also noted that mappings annotated as wrong were not completely incorrect and were usually on topic (e.g. acids -> oils; engine maintenance -> auto body repair; sewage treatment equipment -> sewage treatment services). We get slightly worse results for string matching in comparison with (Gordeev et al., 2018) because we use a more rigid annotation procedure and rely on expert knowledge of the taxonomies. Thus, hierarchical information carries more importance in our experiment.

8.2 Wordnet construction

As can be seen from Table 5 our results are slightly worse than the ones provided by Khronak. However, our algorithm does not require parallel dictionaries and thus is more applicable to low-resource languages. Also our method reveals hierarchical relations which are absent from English. For example, correctly matches experimenter and physicist because in Russian there is a special synset with this meaning (and their are not always research workers). Also sometimes hypernyms and

[3] https://github.com/IlyaGusev/rnnmorph

Table 3: WordNet examples

Russian child	Russian parent	English child	English parrent
экспериментатор (experimenter)	физик (physicist)	experimenter	research worker.n.01
астрофизик astrophysicist	физик	astrophysicist	astronomer.n.01
биофизик (biophysicist)	физик	biophysicist	physicist.n.01
рыба (fish)	лосось (salmon)	panfish	fish.n.02
краснопёрка (redeye)	лосось (salmon)	halibut	flatfish.n.01
буксир (tow)	катер (motorboat)	tow	draw.n.09
катамаран (catamaran)	катер (motorboat)	catamaran	sailboat.n.01

Table 4: Illustration of category allignment

Source Category Code	Source Category description	Target Category Code	Target Category Description (translated from Russian)	Result
800-16	Shoes and Boots: Boots, Rubber	43.31.10	Сапоги резиновые Rubber boots	True
958-78	Management Services Property Management Services	84.11.19.110	Услуги государственного управления имуществом State property management services	Partially True (state)
936-70	Roofing Equipment and Machinery Maintenance and Repair	33.12.23.000	Услуги по ремонту и техническому обслуживанию оборудования для металлургии Services in repair and maintenance service of the equipment for metallurgy	False

Table 5: WordNet results

Method	F1-Score
Universal Wordnet	50.2
Extended OpenMultilingual Wordnet	53
Khrobak	67.6
Cross-lingual embeddings	66.1

homonyms get messed up and we get that ’salmon’ is ’hypernym’ for ’fish’.

9 Conclusion

In this work we have demonstrated several successful methods for unsupervised cross-lingual matching of national product classification systems, we have also shown that the same methods can be used for unsupervised WordNet construction.

The first method taxonomy categories descriptions using a custom string similarity function. Russian category descriptions were converted to English using Google Translate. It performed reasonably well and achieved 41% accuracy. However, this method is unlikely to be suitable for languages that have less resources and are more distant from English than Russian because of the worse search engine quality for such languages (Wu et al., 2016).

We demonstrate that using translation information from a pre-trained translation engine or using embeddings pre-aligned in a common space may help in solving this task.

We also use a similar method for automatic WordNet construction for the Russian language. We do not get a state-of-the-art result, yet our method does not require bilingual corpora and is, thus, more applicable to a wider range of languages.

References

a. All-Russian classifier of products - Wikipedia [Obshcherossijskij klassifikator produkcii — Wikipedia].

b. NIGP Code - Wikipedia.

UNSD — National Classifications.

Mikel Artetxe, Gorka Labaka, and Eneko Agirre. 2018. A robust self-learning method for fully unsupervised cross-lingual mappings of word embeddings.

Y Bengio, R Ducharme, and P Vincent. 2003. A neural probabilistic language model. *Journal of Machine Learning Research*, 3:1137–1155.

Steven Bird. 2006. NLTK: the natural language toolkit. In *Proceedings of the COLING/ACL on Interactive . . .* , pages 69–72. Association for Computational Linguistics.

N Brauner, R Echahed, G Finke, H Gregor, and F Prost. 2004. A complete assignment algorithm and its application in constraint declarative languages. *Les cahiers du laboratoire Leibniz*, 111.

Peter Christen. 2012. *Data Matching: Concepts and Techniques for Record Linkage, Entity Resolution, and Duplicate Detection*. Springer-Verlag, Berlin Heidelberg.

Alexis Conneau, Guillaume Lample, Marc'Aurelio Ranzato, Ludovic Denoyer, and Hervé Jégou. 2017. Word Translation Without Parallel Data.

D. Conte, P. Foggia, C. Sansone, and M. Vento. 2004. Thirty Years Of Graph Matching In Pattern Recognition. *International Journal of Pattern Recognition and Artificial Intelligence*, 18(03):265–298.

J Deng, J Krause, AC Berg 2012 IEEE Conference on . . . , and Undefined 2012. Hedging your bets: Optimizing accuracy-specificity trade-offs in large scale visual recognition. *ieeexplore.ieee.org*.

Georgiana Dinu, Angeliki Lazaridou, and Marco Baroni. 2015. Improving zero-shot learning by mitigating the hubness problem. In *In Proceedings of the 3rd In- ternational Conference on Learning Representations (ICLR2015), workshop track*.

Daniel Faria, Catia Pesquita, Emanuel Santos, Matteo Palmonari, Isabel F. Cruz, and Francisco M. Couto. 2013. The AgreementMakerLight Ontology Matching System. pages 527–541.

D Gordeev, A Rey, and D Shagarov. 2018. Unsupervised Cross-lingual Matching of Product. In *Proceedings of the FRUCT'23*, pages 459–464, Bologna. FRUCT Oy, Finland.

Pratik Jawanpuria, Arjun Balgovind, Anoop Kunchukuttan, and Bamdev Mishra. 2018. Learning Multilingual Word Embeddings in Latent Metric Space: A Geometric Approach.

Mikhail Khodak, Andrej Risteski, Christiane Fellbaum, and Sanjeev Arora. 2017. Automated WordNet Construction Using Word Embeddings. In *Proceedings of the 1st Workshop on Sense, Concept and Entity Representations and their Applications*, pages 12–23, Stroudsburg, PA, USA. Association for Computational Linguistics.

Eugene L. Lawler. *Combinatorial Optimization : Networks And Matroids*.

Angeliki Lazaridou, Georgiana Dinu, and Marco Baroni. 2015. Hubness and Pollution: Delving into Cross-Space Mapping for Zero-Shot Learning. In *Proceedings of the 53rd Annual Meeting of the Association for Computational Linguistics and the 7th International Joint Conference on Natural Language Processing (Volume 1: Long Papers)*, volume 1, pages 270–280, Stroudsburg, PA, USA. Association for Computational Linguistics.

Natalia Maslova and Vsevolod Potapov. 2017. Neural network doc2vec in automated sentiment analysis for short informal texts. In *Lecture Notes in Computer Science*, volume 10458 LNAI, pages 546–554.

Tomas Mikolov, Kai Chen, Greg Corrado, and Jeffrey Dean. 2013a. Distributed Representations of Words and Phrases and their Compositionality. *Nips*, pages 3111–3119.

Tomas Mikolov, Quoc V. Le, and Ilya Sutskever. 2013b. Exploiting Similarities among Languages for Machine Translation.

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

Steven Neale. 2018. A Survey on Automatically-Constructed WordNets and their Evaluation: Lexical and Word Embedding-based Approaches. In *Proceedings of the Eleventh International Conference on Language Resources and Evaluation (LREC 2018)*, Miyazaki, Japan. European Language Resources Association (ELRA).

Kaspar Riesen and Horst Bunke. 2010. *Graph Classification and Clustering Based on Vector Space Embedding*, volume 77 of *Series in Machine Perception and Artificial Intelligence*. World Scientific.

Sebastian Ruder, Ivan Vulić, and Anders Søgaard. 2017. A Survey Of Cross-lingual Word Embedding Models.

P Shvaiko, J Euzenat IEEE Transactions on knowledge And, and Undefined 2013. Ontology matching: state of the art and future challenges. *ieeexplore.ieee.org*.

Blaž Škrlj, Matej Martinc, Jan Kralj, Nada Lavrač, and Senja Pollak. 2019. tax2vec: Constructing Interpretable Features from Taxonomies for Short Text Classification.

Anders Søgaard, Sebastian Ruder, and Ivan Vulić. 2018. On the Limitations of Unsupervised Bilingual Dictionary Induction.

Ivan Vulić, Wim de Smet, and Marie Francine Moens. 2013. Cross-language information retrieval models based on latent topic models trained with document-aligned comparable corpora. *Information Retrieval*, 16(3):331–368.

Yonghui Wu, Mike Schuster, Zhifeng Chen, and Quoc V. Le. 2016. Google's Neural Machine Translation System: Bridging the Gap between Human and Machine Translation.

Xiaohui Yan, Jiafeng Guo, Shenghua Liu, Xueqi Cheng, and Yanfeng Wang. 2013. Learning Topics in Short Texts by Non-negative Matrix Factorization on Term Correlation Matrix. In *Proceedings of the 2013 SIAM International Conference on Data Mining*, pages 749–757, Philadelphia, PA. Society for Industrial and Applied Mathematics.

Zhilin Yang, Jake Zhao, Bhuwan Dhingra, Kaiming He, William W. Cohen, Ruslan Salakhutdinov, and Yann LeCun. 2018. GLoMo: Unsupervisedly Learned Relational Graphs as Transferable Representations.

Radim Řehůřek and Petr Sojka. 2010. Software Framework for Topic Modelling with Large Corpora. In *Proceedings of the LREC 2010 Workshop on New Challenges for NLP Frameworks*, pages 45–50, Valletta, Malta. ELRA.