Translation Inference through Multi-lingual Word Embedding Similarity

Kathrin Donandt, Christian Chiarcos

Applied Computational Linguistics (ACoLi)
Goethe-Universität Frankfurt, Germany
{donandt|chiarcos}@cs.uni-frankfurt.de

Abstract. This paper describes our contribution to the Shared Task on Translation Inference across Dictionaries (TIAD-2019). In our approach, we construct a multi-lingual word embedding space by projecting new languages in the feature space of a language for which a pretrained embedding model exists. We use the similarity of the word embeddings to predict candidate translations. Even if our projection methodology is rather simplistic, our system outperforms the other participating systems with respect to the F1 measure for the language pairs which we predicted.

1 Background

The Second Shared Task on Translation Inference across Dictionaries (TIAD-2019) has been conducted in conjunction with the Conference on Language, Data and Knowledge (LDK, Leipzig, Germany, May 2019). As in the first edition, the objective is to automatically obtain new bilingual dictionaries based on existing ones.

Our contribution is based on the application of a technology originally developed for a related, but broader problem, the identification of cognates in dictionaries of languages that are either diachronically or culturally related with each other. We consider two words from languages A and B to be cognates (in a broad sense) if they share the same etymological origin, either because they have been inherited from a language C which is ancestral to both A and B or if they are loanwords from the same source.[1] Cognates are characterized by a systematic phonological relationship with each other, but also, a systematic semantic relation. A typical example is German *Bank* and English *bench* which

[1] Note that this technical definition is broader than the definition typically applied in linguistics: In a strict sense, cognates are only those of the first case. However, for languages with a long and intense history of contact, the differentiation between cognates in a strict sense and mutual loans is not always clear-cut. A typical case is the large number of common roots in Turkic and Mongolian which can be variously attributed to either a loan between (proto-) languages or to their common (but hypothetical) ancestral language. In both situations, however, the linguistic characteristics (systematic phonological similarity and semantic relatedness) are comparable, thus motivating a generalized, technical definition of cognates.

(in one sense of the German word) are semantically identical, and they reflect the same sound correspondence of *k* and *ch* that we also find in word pairs such as *Kinn*/*chin*. An example for the second case is English *bank* (as in river bank) and German -*bank* in compounds such as *Sandbank* 'sandbank'. Again, senses are similar (an accumulation of soil in or at a water stream), but here, the English word is a loan (either from Low German or Scandinavian). For *Bank* and *bank* as 'financial institute', the situation is more complex: In both German and English, this is a loan from Italian *banca*, but the origin of this word is a Germanic (Langobardian) word with the meaning 'bench', i.e., the place where clients had to wait in order to receive financial support. Cognate candidates can be identified by means of phonological and semantic similarity metrics, and the latter are the basis for the implementation that we describe with this paper. Our research has been conducted in the context of the Independent Research Group 'Linked Open Dictionaries' (LiODi), funded by the German Federal Ministry of Education and Science (BMBF), and applied to language contact studies in the Caucasus area.

Out of the context of this project, one aspect that we appreciate in the TIAD shared task is that it does not depend nor require the use of parallel corpus data. This is in fact the situation for most languages studied in our project. For many low-resource languages, the only language data that is available is provided in the form of dictionaries (or plain word lists) on the one hand and in the form of monolingual, non-translated text on the other, often in the form of field recordings. Parallel (translated) data, however, is scarce – if available at all (and if so, often limited to religious texts, e.g., translations of the Bible with limited representativeness for the language as a whole and not necessarily written by a native speaker). The technological challenges of the TIAD task thus closely mirror the technical (data) challenges in the LiODi project, except that it is limited for translation equivalence whereas we aim to include less direct semantic relations, as well.

In our approach, we make use of the similarity of cross-lingual word embeddings – genuine and projected ones – in order to build new bilingual dictionaries. The projection of words of one language into the embedding space of another and thus the generation of a multi-lingual word embedding space is a widely discussed topic. For an overview of different methodologies of cross-lingual word embedding generation, we refer to [3]. Different from more sophisticated projection methods, we use the provided bilingual resources and obtain word embeddings for a new language by summing up the vector representation(s) of the translation(s) into a language for which we already have either pretrained or projected word embeddings. It should be noted that our simple embedding-based approach allows us to generalize beyond translation information provided by dictionaries, as it naturally includes similarity assessments in accordance with the underlying embedding space. However, as such more remotely related terms receive scores much lower than those of words between which a translation path over different dictionaries exists, the threshold-based implementation we apply for the TIAD shared task systematically restricts our results to the latter.

Despite its simplicity, our implementation achieves good results in comparison to the other systems submitted to TIAD. When similarity beyond translation equivalence is excluded (as by applying a selection threshold), our system is basically a neural reconstruction of [4], which has also been used as a baseline in this task (see Sect. 4.2).

2 Data & Preprocessing

We use the TSV edition of the dictionaries provided by the task organizers. Whereas we only use the languages and language pairs provided in these dictionaries, it would be possible to add more language pairs to be processed by our approach, as long as they are provided in the TIAD-TSV format. In the context of the H2020 project 'Pret-a-LLOD. Ready-to-use Multilingual Linked Language Data for Knowledge Services across Sectors', we are currently in the process of creating such translation data, with OntoLex-Lemon and TIAD-TSV data for 130 language varieties and 373 language pairs already being available from http://github.com/acoli-repo/acoli-dicts/. For the TIAD-2019 task, selected parts of this data have been taken into consideration, however, only for estimating prediction thresholds (Sect. 4.1), not as a source of external lexicographical information.

For initializing the embedding space, we employ readily available monolingual embeddings from a particular 'reference language'. Different reference languages would be feasible in this context, and many different types of embeddings are available for each. Here, we build on the 50 dimensional GloVe v.1.2 embeddings for English, trained on Wikipedia and Gigaword [2].[2] Our implementation can be applied to any other sort of embeddings as long as they are provided in the same format (one word plus its embedding per line, the first column holding the word [without whitespaces] followed by whitespace-separated columns of doubles). Only for reasons of time, we chose the lowest-dimensional embeddings provided, and no other reference languages nor embeddings have been experimented with. We would expect that higher-dimensional embeddings, and embeddings from other languages with many dictionary pairs (e.g., Spanish) would outperform our implementation, and this should be addressed in subsequent research.

In order to make our approach computationally more efficient, we prune the embedding space before running our experiments from words not occurring in the (English) dictionary data we are working with. We add multi-word expressions consisting of words present in the model by taking the sum of their representations. As we want to project other languages into the semantic space of our reference language, we need to store the information to which language a word of the model belongs to. We solve this by adding a language tag to the word:

Example entry in the original embedding model:

[2] Download link: http://nlp.stanford.edu/data/glove.6B.zip

house 0.60137 0.28521 -0.032038 -0.43026 ... -1.0278 0.039922 0.20018

Same entry in the updated embedding model:
"house"@en 0.60137 0.28521 -0.032038 -0.43026 ... -1.0278 0.039922 0.20018

3 Approach

We developed a baseline approach, and submitted the results of this approach to the Shared Task organizers. Later on, we elaborated our baseline and compared its performance to our baseline's performance by applying an interal evaluation procedure (4.3). The results of our elaboration were not submitted to the Shared Task organizers due to time constraints.

3.1 Baseline

Using the pretrained embedding model for our reference language, we generate word embeddings for all the other languages of the provided dictionaries. We project the words of these languages into the embedding space of the reference language, e.g. for the Basque word *abentura* and its English translations *adventure* and *venture* (according to the Basque-English Apertium dictionary), we calculate the new embedding by summing up the embeddings of the possible translations into English:

$$\overrightarrow{\text{"}abentura\text{"@}eu} = \overrightarrow{\text{"}adventure\text{"@}en} + \overrightarrow{\text{"}venture\text{"@}en}$$

In a first iteration, we thus project all those languages into the embedding space of our reference language English for which a dictionary with English translations exist (here: EU → EN, EO → EN).[3] In order to improve coverage, we also included the inverse direction (EN → CA, EN → ES, EN → GL). We refer to these languages (CA, EU, ES, EO, GL) as 'first-level' languages. In the second (and subsequent) iterations, we then project all the languages into the enriched embedding space for which at least one dictionary exists that connects it with another language in the embedding space. If more than one dictionary does exist (e.g., for French EO → FR, FR → ES, FR → CAT), we sum over all translations in the embedding space. AN, AST, FR and IT are the 'second-level languages' for which embeddings are created in this way. Note that once embeddings for a particular language are created, they are not updated nor extended any more, even if in a further iteration additional dictionaries also containing this language are encountered. For example, we obtain EO embeddings from EO → EN and nothing else, even if after adding CA to the embedding space, EO could be reached via EN → CA → EO, as well. This also means that lexical material for Esparanto provided by the CA → EO dictionary which is not covered in the EO

[3] In the following, we refer to the languages solely by using the language abbreviations used in the Shared Task description (https://tiad2019.unizar.es/task.html)

$\rightarrow$ EN dictionary will not be represented in the embedding space. Again, this is an implementation decision taken in the interest of time and to be reconsidered in subsequent research. With a larger set of dictionaries and language pairs, we would continue these iterations until no more languages can be reached in the language graph represented by the dictionaries.

In order to generate a new bilingual dictionary $A \rightarrow B$, we use this enriched, multilingual word embeddings and predict candidate translations of a word a in language A by choosing the words in language B whose embeddings are most similar to the embedding of a, and which have the same part-of-speech as a. As similarity measure we chose cosine similarity, which is ignorant of vector length, and faster in lookup than Euclidian distance. This also allows us to add rather than average in the process of projecting new language into the embedding space. It should be noted that speed is a decisive criterion in the intended application of the technique, as we aim to provide on-the-fly cognate detection.

3.2 Experiments with sense clustering

We elaborated our baseline approach by taking into consideration the sense(s) of a word to be translated. We follow the definition of senses in the TIAD-OntoLex edition of the Apertium dictionaries by assuming that in a bilingual dictionary, each translation $b_1, b_2, ...$ provided for a source language word a entails a specific sense $a_{b_1}, a_{b_2}, ...$ The number of OntoLex senses in the TIAD data is thus identical to the number of translations provided. Unfortunately, the Apertium data does not provide sense-level linking beyond different bilingual dictionaries, so that sense-level linking cannot be used as a feature for translation inference. We do not have information on whether sense a_{b_1} induced from the $A \rightarrow B$ dictionary has any relation with the sense $a_{c_{15}}$ induced from a $A \rightarrow C$ dictionary.

However, if the basic assumption is correct, then we can approximate the factual number of senses of a source language word a as the maximum number n of translations in any bilingual dictionary. Assuming that translations referring to the same word sense tend to be closer in the embedding space, we can use this information for disambiguation and try to identify the most representative translation for every sense. We do so by performing a lookup in the dictionaries to get all translations, and then cluster (the embeddings of) these translations into n clusters. For every cluster, we then return the target language word(s) closest to the cluster center.

We generate "sense embeddings" for a as follows:

1. Look up all possible translations of a in the dictionaries, e.g.

   $(a, b_{11}), ..., (a, b_{1n}), (a, b_{21}), ..., (a, b_{2m})$, etc.,

   where
   b_{ij}: the j'th translation of a in a dictionary $A \rightarrow B_i$ (or $B_i \rightarrow A$)
   N: number of dictionaries where A appears as source or target language;
   $i = 1, ..., N$.

2. Apply k-means clustering[4] over all possible translations b_{ij}, where

   $k = max_i J_i$,

   where J_i: number of entries containing a in $A \rightarrow B_i$.

3. For each of the k cluster centers, take the closest embedding (irrespective of its language) as a sense embedding of word a. This is the basis for finding the closest target language embedding, resp., word, per sense cluster.

We then predict possible translations of a given word in a source language into a specified target language by looking up the most similar word embeddings of the target language to each of the sense embeddings of the word in the source language.

While the baseline predicted n words in the target language as candidate translations, we now predict at least n/k words for each of the k sense embeddings, with distance relative to the 'sense cluster' rather than the source language embedding.

4 Results

We first ran an internal evaluation in order to get an idea of the quality of our approach. After that, we determine a cosine similarity threshold for our predicted translations in order to select the best translation candidates. Later, we use the internal evaluation strategy again to compare our baseline with the elaboration. Unlike most other systems in TIAD-2019, we provide only the data points exceeding our thresholds, whereas a lower threshold (and parameter estimation by the task organizers) may have produced better results on the blind test set.

4.1 Threshold Estimation

Our baseline approach returns the n most similar word embeddings of the target language to the word embedding to be translated, but obviously, not all of these might be good translation candidates. In order to filter out good candidates, we determine a threshold for the cosine similarity value. Therefore, we take the EN $\rightarrow$ PT, PT $\rightarrow$ FR, FR $\rightarrow$ EN dictionaries available from FreeDict,[5] using the TIAD-TSV version we provide under https://github.com/acoli-repo/acoli-dicts. If both translation directions are available for a language pair, we join the

[4] We use the scikit-learn implementation of k-means clustering (https://scikit-learn.org/stable/modules/generated/sklearn.cluster.KMeans.html) with default parameters.

[5] http://www.freedict.de/

two dictionaries to obtain a higher coverage. As expected, the results in table 1 show decreasing cosine similarity decrease is correlated with lower precision and higher recall. We chose the cosine similarity threshold 0.97, a rather high threshold in order to achieve sufficient precision, without suffering too much from the drop in recall in terms of F1. Yet, considering recall here is debatable: It is not as expressive in this scenario as it requires a full coverage of the languages by the dictionary, which is not the case for dictionaries in general (see also the discussion in the TIAD-2017 overview report [1]).

Table 1. Comparison of Cosine Similarity Thresholds, using FreeDict dictionaries as evaluation

Translation	Threshold	Precision	Recall	F1
FR → PT	0.98	37.00%	12.00%	18.12%
	0.97	36.00%	12.00%	18.00%
	0.96	35.00%	13.00%	18.96%
	0.95	34.00%	13.00%	18.81%
	0.9	27.00%	14.00%	18.44%
PT → EN	0.98	47.00%	10.00%	16.49%
	0.97	47.00%	11.00%	17.83%
	0.96	46.00%	11.00%	17.75%
	0.95	46.00%	12.00%	19.03%
	0.9	39.00%	13.00%	19.50%
EN → FR	0.98	19.00%	12.00%	14.71%
	0.97	19.00%	13.00%	15.44%
	0.96	19.00%	13.00%	15.44%
	0.95	19.00%	14.00%	16.12%
	0.9	17.00%	17.00%	17.00%

4.2 Shared Task Evaluation

We generated translations in one direction per language pair: EN → FR, FR → PT and PT → EN. We submitted the baseline implementation with a threshold of 0.97 to the TIAD Shared Task, where we achieved overall precision of 0.64, recall of 0.22, F1 of 0.32 and coverage of 0.43. Unsurprisingly, all these scores are substantially higher than the numbers we obtained for evaluation against FreeDict data. In comparison with the performance of the other participants' systems, our system shows one of the highest overall performance regarding the F1-measure, see Tab. 2. It should be noted that the OTIC and W2VEC implementations were provided by the task organizers as baseline implementations, but have not been outperformed by any participant system. For our implementation, a similar performance as OTIC [4] would be expected as we consider our approach a neural reconstruction of the OTIC approach.

The "One Time Inverse Consultation" (OTIC) method [4] is a graph-based approach, where lexical overlap in the pivot language is used for scoring trans-

lation candidates from source and target language. With some degree of simplification, the score is for a source language word a and a target language word b is based on the relative number of pivot language elements provided both as translations of a and b relative to the number of pivot language elements provided as translations of a or b. In combination with a prediction threshold, this lexical overlap score is then used for predicting translation pairs.

In our approach, this resembles the way how multilingual embeddings are constructed: A non-English word is represented as the sum (of the embeddings) of the (directly or indirectly associated) English words. If two words are identical in their English translations, they thus have cosine similarity 1, every English translation they do not share leads (when added to the embedding) to a slight decrease in this score. In our approach, this decrease will be smaller for semantically related words and larger for non-related words, but if we assume that, on average, the deviation from a perfect match per non-shared English translation is equal, the decrease in cosine similarity will directly reflect the number of non-shared English translations relative to the number of shared English translations. We suspect that the observed drop in F1 in comparison to OTIC is due to the construction method of the multilingual embedding space, where dictionaries connecting two first-, resp., two second-level languages are being ignored, as well as due basing this on English as the only pivot language, whereas OTIC can adopt different pivots depending on the structure of the dictionary graph.

In the light of this explanation, it is somewhat surprising to see that the evaluation results of the Shared Task organizers seem to confirm the trend that 2nd level languages predict better translation candidates, as previously noticed in Sect. 4.3: For PT → FR, we get the highest F1 value, even if for both of these languages, the embeddings are produced in the second iteration of our algorithm.

Table 2. TIAD-2019 Shared Task results, top 5 systems in terms of F1, according to https://tiad2019.unizar.es/results.html

System	Precision	Recall	F1	Coverage
OTIC	0.64	0.26	0.37	0.45
W2VEC	0.66	0.24	0.35	0.51
FRANKFURT	0.64	0.22	0.32	0.43
LyS-DT	0.36	0.31	0.32	0.64
LyS-ES	0.33	0.30	0.31	0.64

Overall, the scores achieved by all systems (as well as the baselines) in the TIAD-2019 shared task were considerably low, possibly indicating conceptual differences in the composition of Apertium data and the blind test set. We thus performed an additional evaluation on the Apertium data itself. The sense-clustering extension was evaluated only in this internal evaluation.

4.3 Internal Evaluation

To assess the quality of a newly generated bilingual dictionary by our approach, we leave out one of the provided dictionaries when calculating the embeddings for the languages based on the reference language model and try to reconstruct it using a similarity threshold of 0.97 for predicting translation pairs. Only in this evaluation, we compare our base implementation with the sense-clustering extensions, see Tab. 3.

For more than two thirds of the language pairs (72%, 13/18), the sense clustering yields a slight improvement in F1, but only at a marginal scale. In general, precision and recall vary substantially. We obtain the highest precision for EU $\rightarrow$ EN in both the baseline and extension. The fact that high-precision language pairs show a (small) drop in precision in the sense clustering extension is possibly related to the fact that most of them involve the reference language. Dictionaries including OC have comparably high recall, PT $\rightarrow$ CA and FR $\rightarrow$ CA also obtain high recall. The highest precision values (over 69%) are obtained by dictionaries including EN for the baseline, precision drops for these dictionaries in the sense extension, while recall increases in most of them.

The fact that embeddings for a language are not trained on their own data as it is the case for the reference language might have a small effect on precision, but the F1 values of language pairs that do include the reference language English are not in general higher than of pairs that do not. Our "2nd level languages" AN, AST, FR, IT, OC, PT and RO are not added in the first iteration like CA, EO, ES, EU and GL ("1st level languages") and have to be generated using already projected word embeddings instead of those of the reference language. The number of pivots used in the projection procedure does not seem to be negatively correlated with the quality of the generated dictionaries though, at least for those languages we could evaluate in our internal evaluation (we could not include AN, AST, IT, and RO as they are connected to only one other language, resp., and thus, as there is no alternative way of getting their embeddings with our projection procedure if their dictionaries are excluded from the translation graph): The highest F1 value is obtained by predicting CA and ES, both 1st level languages, from OC, a 2nd level language, whereas the lowest F1 value is obtained by predicting FR (2nd level language) from EO (1st level language). In general, predicting dictionaries with EO as input yields low F1 values, and all except one (FR $\rightarrow$ ES) predicted dictionary from 2nd level languages yield higher recall values than most of the other predicted dictionaries.

5 Discussion & Conclusion

As the results of our internal evaluation (and later on also of the submission) show, the fact that a word embedding was projected and the number of pivot languages necessary for the projection do not seem to worsen the quality of the generation of translation pairs. We have therefore concluded that our approach is a viable way for generating new translation pairs. We might, however, expect

Table 3. Internal Evaluation for reference language (EN), first-level languages (CA, EO, ES, EU, GL), and second-level languages (FR, OC, PT)

	Baseline			With sense clustering		
Dictionary	**Precision**	**Recall**	**F1**	**Precision**	**Recall**	**F1**
EU, EN	**84.25%**	**31.67%**	**46.03%**	82.83%	30.10%	44.16%
EN, GL	**81.05%**	24.93%	38.13%	80.00%	**28.47%**	**41.99%**
ES, PT	68.29%	**37.50%**	48.41%	**75.85%**	35.97%	**48.79%**
EN, CA	**72.09%**	30.16%	42.52%	68.18%	**31.04%**	**42.66%**
EN, ES	**71.19%**	40.08%	51.29%	66.48%	**43.01%**	**52.23%**
EO, EN	**69.15%**	**19.58%**	**30.52%**	66.16%	18.27%	28.64%
ES, CA	62.32%	**39.92%**	**48.67%**	**67.83%**	37.00%	47.88%
ES, GL	58.17%	**44.36%**	**50.33%**	**67.59%**	38.45%	49.02%
PT, GL	59.17%	**39.14%**	47.11%	**66.27%**	37.06%	**47.53%**
EO, FR	61.07%	12.30%	20.48%	**66.15%**	**17.19%**	**27.29%**
OC, CA	59.03%	**59.05%**	**59.04%**	**65.94%**	52.26%	58.31%
OC, ES	55.03%	54.72%	54.87%	**65.35%**	**55.81%**	**60.21%**
PT, CA	56.26%	**57.85%**	57.04%	**63.15%**	53.20%	**57.75%**
EO, CA	53.40%	25.40%	34.42%	**56.22%**	**31.63%**	**40.48%**
EO, ES	51.47%	30.34%	38.18%	**53.13%**	**35.00%**	**42.20%**
FR, ES	49.88%	23.40%	31.86%	**52.96%**	**25.89%**	**34.78%**
EU, ES	**52.34%**	33.38%	40.76%	47.97%	**36.65%**	**41.55%**
FR, CA	**51.34%**	56.36%	53.73%	50.26%	**61.10%**	**55.15%**

better results when using more sophisticated projection methodologies, for example by learning projection matrices jointly for the languages to be projected into the English embedding space - this remains to be tested. A critical point of our algorithm is to generate a multi-lingual embedding space in a greedy fashion, other languages are projected into the space of the reference language: In each iteration, we only consider the dictionaries whose target language is in the language set of already added languages plus the reference language, e.g. we obtain EO from EO → EN and nothing else. We did not exploit yet any data provided by the dictionaries EO → CA, EO → ES and EO → FR; the integration of these dictionaries and therefore, the usage of all available resources would be a more comprehensive approach and probably improve the quality of our generation of candidate translation pairs. Choosing an alternative pretrained model for the reference language with a higher-dimensional embedding space would be another way to improve – GloVe embeddings are available in higher dimensions as well, and likewise, alternative embeddings could be explored. Furthermore, one could use more than just one reference language, and create a joint embedding space by means of either concatenation (e.g., of English-based and Spanish-based embeddings for each word) or by training a joint embedding space as mentioned above.

The sense clustering we used in the elaboration of our baseline approach fails to yield significant improvements. One possible explanation for this is that we have too little input for clustering. Another cause might be that senses are

represented by written representation pairs as described in Sect. 3.2. This results in more emphasis on certain "senses" (pairs of written representations) when predicting translations. In general, the remarkably poor performance of all submitted systems (and the provided baseline implementations) on the blind test set may reflect the possibly quite distinct nature of training and blind test data: The blind test data originates from proprietary learner dictionaries provided by KDictionaries, whereas the Apertium dictionaries that constitute the training data have been designed for machine translation. Both are similar on a superficial level, but it is not unlikely that learner's dictionaries put a stronger emphasis on covering the different semantic senses in their translations in a near-exhaustive way (regardless of their real-world frequency), whereas MT dictionaries do have a preference for covering prototypical senses (being ignorant against infrequent/domain-specific uses may actually improve the system for most of its applications). In order to quantify the impact of sense granularity independently from proprietary data, we would encourage the task organizers to provide an provide open test set along with the blind test set.

As mentioned above, we developed our system originally in the context of our research project for the purpose of facilitating the search for semantically similar words. Vector representations of words in a semantic space are particularly promising when searching for remote semantic links, e.g., as required for cognate detection, but maybe less so for finding literal translations. Yet, as the evaluation indicates, our system also produces acceptable results for the task of translation inference across dictionaries in the sense that we rank among the best-performing implementations for this task.

Our contribution and the key benefit of our approach and is to be seen in the fact that it is slim, fast, and trivially extensible beyond literal translations: Translation inference (like cognate detection) requires a simple lookup in the embedding space. We see that as an important component of on-the-fly cognate detection, as it allows to identify semantically related form forms over a multilingual graph of dictionaries without actually traversing this graph at query time.

Acknowledgments

The research described in this paper was primarily conducted in the project 'Linked Open Dictionaries' (LiODi, 2015-2020), funded by the German Ministry for Education and Research (BMBF) as an Independent Research Group on eHumanities. The conversion of FreeDict dictionaries into TIAD-TSV data that we used for estimating the prediction threshold was performed in the context of the Research and Innovation Action "Pret-a-LLOD. Ready-to-use Multilingual Linked Language Data for Knowledge Services across Sectors" funded in the European Union's Horizon 2020 research and innovation programme under grant agreement No 825182.

References

1. Alper, M.: Auto-generating bilingual dictionaries: Results of the TIAD-2017 shared task baseline algorithm. In: Proceedings of the LDK 2017 Workshops: 1st Workshop on the OntoLex Model (OntoLex-2017), Shared Task on Translation Inference Across Dictionaries & Challenges for Wordnets co-located with 1st Conference on Language, Data and Knowledge (LDK 2017), Galway, Ireland, June 18, 2017. pp. 85–93 (2017), http://ceur-ws.org/Vol-1899/baseline_report.pdf
2. Pennington, J., Socher, R., Manning, C.D.: Glove: Global vectors for word representation. In: Empirical Methods in Natural Language Processing (EMNLP). pp. 1532–1543 (2014), http://www.aclweb.org/anthology/D14-1162
3. Ruder, S.: A survey of cross-lingual embedding models. CoRR **abs/1706.04902** (2017), http://arxiv.org/abs/1706.04902
4. Tanaka, K., Umemura, K.: Construction of a bilingual dictionary intermediated by a third language. In: Proceedings of the 15th Conference on Computational Linguistics - Volume 1. pp. 297–303. COLING '94, Association for Computational Linguistics, Stroudsburg, PA, USA (1994)