Word-embedding based bilingual terminology alignment

Andraž Repar[1], Matej Martinc[2], Matej Ulčar[3], Senja Pollak[4]

[1]International Postgraduate School, Institut Jozef Stefan, Jamova 39, Ljubljana, Slovenia
[2] Institut Jozef Stefan, Jamova 39, Ljubljana, Slovenia
[3] Faculty of Computer and Information Science, University of Ljubljana, Večna pot 113, Ljubljana, Slovenia
[4] Institut Jozef Stefan, Jamova 39, Ljubljana, Slovenia
E-mail: andraz.repar@ijs.si, matej.martinc@ijs.si, matej.ulcar@fri.uni-lj.si, senja.pollak@ijs.si

Abstract

The ability to accurately align concepts between languages can provide significant benefits in many practical applications. In this paper, we extend a machine learning approach using dictionary and cognate-based features with novel cross-lingual embedding features using pretrained fastText embeddings. We use the tool VecMap to align the embeddings between Slovenian and English and then for every word calculate the top 3 closest word embeddings in the opposite language based on cosine distance. These alignments are then used as features for the machine learning algorithm. With one configuration of the input parameters, we managed to improve the overall F-score compared to previous work, while another configuration yielded improved precision (96%) at a cost of lower recall. Using embedding-based features as a replacement for dictionary-based features provides a significant benefit: while a large bilingual parallel corpus is required to generate the Giza++ word alignment lists, no such data is required for embedding-based features where the only required inputs are two unrelated monolingual corpora and a small bilingual dictionary from which the embedding alignments are calculated.

Keywords: terminology alignment; word embeddings; embeddings alignment; machine learning

1. Introduction

The ability to accurately align concepts between languages can provide significant benefits in many practical applications. For example, in terminology, terms can be aligned between languages to provide bilingual terminological resources, while in the news industry, keywords can be aligned to provide better news clustering or search in another language. Accurate bilingual resources can also serve as seed data for various other NLP tasks, such as multilingual vector space alignment.

Bilingual terminology alignment[1] is the process of aligning terms between two candidate term lists in two languages. The primary purpose of bilingual terminology extraction is to build a term bank - i.e. a list of terms in one language along with their equivalents in the other language. With regard to the input text, we can distinguish between alignment on the basis of a parallel corpus and alignment on the basis of a comparable corpus. For the translation industry, bilingual terminology extraction from parallel corpora is extremely relevant due to the large amounts of sentence-aligned parallel corpora available in the form of translation memories. Consequently, initial attempts at bilingual terminology extraction involved parallel input data (Kupiec, 1993; Daille et al., 1994; Gaussier, 1998), and the interest of the community has continued until today. However, most parallel corpora are owned by private companies[2], such as language service providers, who consider them to be their intellectual property and are reluctant to share them publicly. For this reason (and in particular for language pairs not involving English) considerable efforts have also been invested into researching bilingual terminology extraction from comparable corpora (Fung & Yee, 1998; Rapp, 1999; Chiao & Zweigenbaum, 2002; Cao & Li, 2002; Daille &

[1] Note that bilingual terminology alignment has a narrower focus than *bilingual terminology extraction*, but the two terms are often used interchangeably in various papers. The latter covers extraction and alignment of terms between languages.

[2] However, some publicly available parallel corpora do exist. A good overview can be found at the OPUS web portal (Tiedemann, 2012).

Morin, 2005; Morin et al., 2008; Vintar, 2010; Bouamor et al., 2013; Hazem & Morin, 2016, 2017).

The approach designed by Aker et al. (2013) and replicated and adapted in Repar et al. (2019) served as the basis of our work. It was developed to align terminology between languages with the help of parallel corpora using machine-learning techniques. They use terms from the Eurovoc (Steinberger et al., 2002) thesaurus and train an SVM binary classifier (Joachims, 2002) (with a linear kernel and the trade-off between training error and margin parameter c = 10). The task of bilingual alignment is treated as a binary classification task – each term from the source language S is paired with each term from the target language T and the classifier then decides whether the aligned pair is correct or incorrect. Aker et al. (2013) run their experiments on the 21 official EU languages covered by Eurovoc with English always being the source language (20 language pairs altogether). They evaluate the performance on a held-out term pair list from Eurovoc using recall, precision and F-measure for all 21 languages. Next, they propose an experimental setting for a simulation of a real-world scenario where they collect English-German comparable corpora of two domains (IT, automotive) from Wikipedia, perform monolingual term extraction using the system by Pinnis et al. (2012) followed by the bilingual alignment procedure described above and manually evaluate the results (using two evaluators). They report excellent performance on the held-out term list with many language pairs reaching 100% precision and the lowest recall being 65%. For Slovenian, which is our main interest, the reported results were excellent with perfect or nearly perfect precision and good recall. The reported results of the manual evaluation phase were also good, with two evaluators agreeing that at least 81% of the extracted term pairs in the IT domain and at least 60% of the extracted term pairs in the automotive domain can be considered exact translations. Repar et al. (2019) tried to reproduce their approach and after initially having little success they were at the end able to achieve comparable results with precision exceeding 90% and recall over 50%.

Despite the problem of bilingual term alignment lending itself well to the binary classification task, there have been relatively few approaches utilising machine learning. Similar to Aker et al. (2013), Baldwin & Tanaka (2004) generate corpus-based, dictionary-based and translation-based features and train an SVM classifier to rank the translation candidates. Note that they only focus on multi-word noun phrases (noun + noun). A similar approach, again focusing on noun phrases, is also described by Cao & Li (2002). Finally, Nassirudin & Purwarianti (2015) also reimplement Aker et al. (2013) for the Indonesian-Japanese language pair and further expand it with additional statistical features.

This paper is organised as follows: the present section introduces the problem and related work, Section 2 describes the datasets used for the experiments, Section 3 lists the features used in the machine learning process, Section 4 contains a description of the experiments and lists their results and Section 5 provides the conclusion.

2. Resources

The approach described in this paper requires four types of resources. The first two are the same as in Aker et al. (2013) and Repar et al. (2019), whereas the third and fourth resources are required for the additional experiments conducted for this paper:

- aligned term pairs in two languages that serve as training data
- a parallel corpus to generate a Giza++ word alignment list
- pretrained embeddings in two languages
- a (small) bilingual dictionary

We create term pairs from the Eurovoc (Steinberger et al., 2002) thesaurus, which at the time of Repar et al. (2019) consisted of 7,083[3] terms, by pairing Slovenian terms with English ones. The test set consisted of 600 positive (correct) term pairs — taken randomly out of the total 7,083 Eurovoc term pairs — and around 1.3 million negative pairs which were created by pairing each source term with 200 distinct incorrect random target terms. Aker et al. (2013) argue that this was done to simulate real-world conditions where the classifier would be faced with a larger number of negative pairs and a comparably small number of positive ones. The 600 positive term pairs were further divided into 200 pairs where both (i.e. source and target) terms were single words, 200 pairs with a single word only on one side and 200 pairs with multiple-word terms on both sides. The remaining positive term pairs (approximately 6,200) were used as training data along with additional 6,200 negative pairs. These were constructed by taking the source side terms and pairing each source term with one target term (other than the correct one). Using Giza++, we created source-to-target and target-to-source word alignment dictionaries based on the DGT translation memory (Steinberger et al., 2013). The resulting dictionary entries consist of the source word s, its translation t and the number indicating the probability that t is an actual translation of s. To improve the performance of the dictionary-based features, the following entries were removed from the dictionaries:

- entries where probability is lower then 0.05
- entries where the source word was less than 4 characters and the target word more than 5 characters long and vice versa in order to avoid translations of stop word to content words)

In addition to the resources described above, we used fastText (Bojanowski et al., 2016) pre-trained word embedding vectors to calculate distances (or similarities) between terms. We aligned monolingual fastText embeddings using the VecMap (Artetxe et al., 2018) tool which can align embeddings with the help of a small bilingual dictionary. We used a bilingual dictionary compiled from two sources: single word terms from Eurovoc and Wiktionary entries extracted using the wikt2dict tool (Acs, 2014). Using the aligned embedding vectors, we then calculated cosine distances between all words present in Eurovoc terms in one language and all words present in Eurovoc terms in the other language.

Using the fastText-based lists of aligned words, we created 3-tuples[4] of most similar — based on cosine similarity — source-to-target and target-to-source words, such as:

- ksenofobija [‘xenophobia’, ‘0.744’], [‘racism’, ‘0.6797’], [‘anti-semitism’, ‘0.654’]
- ženska [‘woman’, ‘0.7896’], [‘women’, ‘0.73’], [‘female’, ‘0.722’]

[3] While new terms are constantly added to Eurovoc, we decided not to use them to allow for better comparison between the approaches

[4] This number was determined experimentally.

where the tuple contains the source language word along with their three most likely corresponding words in the target language and their cosine similarities. The 3-tuples of most similar words were used to construct additional features for the machine learning algorithm.

3. Feature construction

The updated approach in this paper uses three types of features that express correspondences between the words (composing a term) in the target and source language. The dictionary and cognate-based features are same as in Repar et al. (2019), while embedding-based features are newly developed. The three feature types are as follows (for a detailed description see Table 1):

- 7 dictionary-based (using Giza++) features which take advantage of dictionaries created from large parallel corpora of which 6 are direction-dependent (source-to-target or target-to-source) and 1 direction-independent — resulting in altogether 13 features
- 7 cognate-based features (on the basis of Gaizauskas et al. (2012)) which utilize string-based word similarity between languages
- 5 cognate-based features using specific transliteration rules which take into account the differences in writing systems between two languages: e.g. Slovenian and English. Transliteration rules were created for both directions (source-to-target and target-to-source) separately and cognate-based features were constructed for both directions — resulting in an additional 10 cognate-based features with transliteration rules. The following transliteration rules were used: *x:ks, y:j, w:v, q:k* for English to Slovenian and *č:ch, š:sh, ž:zh* for Slovenian to English
- 5 direction-dependent combined[5] features where the term pair alignment is correct if either the dictionary or the cognate-based method returns a positive result — resulting in a total of 10 combined features
- 12 novel direction-dependent embedding-based features utilising fastText embeddings — resulting in a total of 24 features
- 5 novel combined features constructed in the same manner as the existing combined features but replacing Giza++ word lists with fastText-based lists of top 3 aligned words - resulting in a total of 10 novel combined features
- 3 term length features: sourceTargetLengthMatch, sourceTermLength, targetTermLength

To match words with morphological differences, we do not perform direct string matching but utilise Levenshtein Distance. Two words were considered equal if the Levenshtein Distance Levenshtein (1966) was equal or higher than 0.95.

[5] For combined features, a word is considered as covered if it can be found in the corresponding set of Giza++ translations or if one of the cognate-based measures (Longest Common Subsequence, Longest Common Substring, Levensthein Distance, Needleman-Wunsch Distance, Dice) is 0.70 or higher (set experimentally by Aker et al. (2013))

Feature	Cat	Description
isFirstWordTranslated	Dict	Checks whether the first word of the source term is a translation of the first word in the target term (based on the Giza++ dictionary)
isLastWordTranslated	Dict	Checks whether the last word of the source term is a translation of the last word in the target term
percentageOfTranslatedWords	Dict	Ratio of source words that have a translation in the target term
percentageOfNotTranslatedWords	Dict	Ratio of source words that do not have a translation in the target term
longestTranslatedUnitInPercentage	Dict	Ratio of the longest contiguous sequence of source words which has a translation in the target term (compared to the source term length)
longestNotTranslatedUnitInPercentage	Dict	Ratio of the longest contiguous sequence of source words which do not have a translation in the target term (compared to the source term length)
Longest Common Subsequence Ratio	Cogn	Measures the longest common non-consecutive sequence of characters between two strings
Longest Common Substring Ratio	Cogn	Measures the longest common consecutive string (LCST) of characters that two strings have in common
Dice similarity	Cogn	2*LCST / (len(source) + len(target))
Needlemann-Wunsch distance	Cogn	LCST / min(len(source), len(target))
isFirstWordCognate	Cogn	A binary feature which returns True if the longest common consecutive string (LCST) of the first words in the source and target terms divided by the length of the longest of the two words is greater than or equal to a threshold value of 0.7 and both words are longer than 3 characters
isLastWordCognate	Cogn	A binary feature which returns True if the longest common consecutive string (LCST) of the last words in the source and target terms divided by the length of longest of the two words is greater than or equal to a threshold value of 0.7 and both words are longer than 3 characters
Normalized Levensthein distance (LD)	Cogn	1 - LD / max(len(source), len(target))
isFirstWordCovered	Comb	A binary feature indicating whether the first word in the source term has a translation or transliteration in the target term
isLastWordCovered	Comb	A binary feature indicating whether the last word in the source term has a translation or transliteration in the target term
percentageOfCoverage	Comb	Returns the percentage of source term words which have a translation or transliteration in the target term
percentageOfNonCoverage	Comb	Returns the percentage of source term words which have neither a translation nor transliteration in the target term
difBetweenCoverageAndNonCoverage	Comb	Returns the difference between the last two features
isFirstWordMatch	Emd	Checks whether the first word of the source term is the most likely translation of the first word in the target term (based on the aligned embeddings)
isLastWordMatch	Emd	Checks whether the last word of the source term is the most likely translation of the last word in the target term (based on the aligned embeddings)
percentageOfFirstMatchWords	Emb	Ratio of source words that have a first match (i.e. first position in the 3-tuple) in the target term
percentageOfNotFirstMatchWords	Emb	Ratio of source words that do not have a first match (i.e. first position in the 3-tuple) in the target term
longestFirstMatchUnitInPercentage	Emb	Ratio of the longest contiguous sequence of source words which has a first match (first position in the 3-tuple) in the target term (compared to the source term length)
longestNotFirstMatchUnitInPercentage	Emb	Ratio of the longest contiguous sequence of source words which do not have a first match (first position in the 3-tuple) in the target term (compared to the source term length)
isFirstWordTopnMatch	Emd	Checks whether the first word of the source term is in the 3-tuple of most likely translations of the first word in the target term (based on the aligned embeddings)

isLastWordTopnMatch	Emd	Checks whether the first word of the source term is not in the 3-tuple of most likely translations of the first word in the target term (based on the aligned embeddings)
percentageOfTopnMatchWords	Emb	Ratio of source words that have a match (i.e. any position in the 3-tuple) in the target term
percentageOfNotTopnMatchWords	Emb	Ratio of source words that do not have a match (i.e. any position in the 3-tuple) in the target term
longestTopnMatchUnitInPercentage	Emb	Ratio of the longest contiguous sequence of source words which has a match (any position in the 3-tuple) in the target term (compared to the source term length)
longestNotTopnMatchUnitInPercentage	Emb	Ratio of the longest contiguous sequence of source words which do not have a match (any position in the 3-tuple) in the target term (compared to the source term length)
isFirstWordCoveredEmbeddings	Comb	A binary feature indicating whether the first word in the source term has a match (any position in the 3-tuple) or transliteration in the target term
isLastWordCoveredEmbeddings	Comb	A binary feature indicating whether the last word in the source term has a match (any position in the 3-tuple) or transliteration in the target term
percentageOfCoverageEmbeddings	Comb	Returns the percentage of source term words which have a match (any position in the 3-tuple) or transliteration in the target term
percentageOfNonCoverageEmbeddings	Comb	Returns the percentage of source term words which do not have a match (any position in the 3-tuple) or transliteration in the target term
diffBetweenCoverageAnd-NonCoverageEmbeddings	Comb	Returns the difference between the last two features

Figure 1: Features used in the experiments. Note that some features are used more than once because they are direction-dependent.

4. Experimental setup and results

The constructed features were then used to train an SVM binary classifier (Joachims, 2002) (with a linear kernel and the trade-off between training error and margin parameter c = 10). We selected three configurations from Repar et al. (2019) for comparison:

- Training set 1:200: a very unbalanced training set (ratio of 1:200 between positive and negative examples [6]) greatly improves the precision of the classifier at a cost of somewhat lower recall, when compared to a balanced train set or a less unbalanced train set (e.g., ratio of 1:10 between positive and negative examples).
- Training set filtering 3: In Repar et al. (2019), we have performed an error analysis and found that many incorrectly classified term pairs are cases of partial translation where one unit in a multi-word term has a correct Giza++ dictionary translation in the corresponding term in the other language. Based on the problem of partial translations, leading to false positive examples, we focused on the features that would eliminate such partial translations from the training set. After a systematic experimentation, we noticed that we can drastically improve precision if we only keep positive term pairs with the following feature values: isFirstWordTranslated = True, isLastWordTranslated = True, percentageOfCoverage > 0.66, isFirstWordTranslated-reversed = True, isLastWordTranslated-reversed = True, percentageOfCoverage-reversed > 0.66.

[6] 1:200 imbalance ratio was the largest imbalance we tried, since the testing results indicated that no further gains could be achieved by further increasing the imbalance.

- Cognates: the dataset is additionally filtered according to the following criteria: isFirstWordCognate = True and isLastWordCognate = True, isFirstWordTranslated = True and isLastWordCognate = True, isFirstWordCognate = True and isLastWordTranslated = True and we also use a Gaussian kernel instead of the linear one, since this new dataset structure represents a classic "exclusive or" (XOR) problem which a linear classifier is unable to solve.

The selection was made based on our experience and previous work with this approach. The three selected configurations were among the best performing in previous experiments and we believed they had the highest potential for improvement. For a complete description of the decisions that led to these configurations, please refer to Repar et al. (2019).

No.	Config EN-SL	Training set size	Pos/Neg ratio	Precision	Recall	F-score
	Dictionary-based and cognate-based features					
1	Training set 1:200	1,303,083	1:200	0.4299	**0.7617**	0.5496
2	Training set filtering 3	645,813	1:200	0.9342	0.4966	0.6485
3	Cognates approach	672,345	1:200	0.8732	0.5167	0.6492
	Dictionary-based, embedding-based and cognate-based features					
1	Training set 1:200	1,303,083	1:200	0.5375	0.680	0.6004
2	Training set filtering 3	695,058	1:200	0.8170	0.5133	0.6305
3	Cognates approach	706,113	1:200	0.8991	0.5200	**0.6589**
	Embedding-based and cognate-based features only					
1	Training set 1:200	1,303,083	1:200	0.3232	0.4967	0.3916
2	Training set filtering 3	322,605	1:200	0.9545	0.2450	0.3899
3	Cognates approach	394,362	1:200	**0.9618**	0.3617	0.5242

Table 2: Results on the English-Slovenian term pair.

First, we simply added the new embedding-based features to the dataset to see if they improved the overall performance. Later, we removed the dictionary-based features from the dataset to see whether the novel embedding-based features could replace them without a major impact on the performance. As can be observed from Table 2, the results are a mixed bag when using all available features. Without any training set filtering, the new features improve precision at the expense of recall, but are less effective when filtering is applied. Nevertheless, when we use additional trainset filters for the Cognates approach, we can observe a slight increase in both precision and recall resulting in the overall highest F-score. When we use only embedding-based and cognate-based features, which would be beneficial for language pairs without access to large parallel corpora needed to create Giza++ word alignments, there is a significant drop in recall in all cases, but precision actually increases when trainset filtering is applied and the Cognates approach achieves the overall best precision.

5. Conclusion

In this paper, we continued our experiments on bilingual terminology alignment using a machine learning approach by adding new features based on fastText word embedding

vectors. We took advantage of the availability of large pre-trained datasets by Bojanowski et al. (2016), and a cross-lingual word embedding mapping tool Vecmap by Artetxe et al. (2018) to create word alignment dictionaries similar to the output of traditional word alignment tools, such as Giza++ (Och & Ney, 2003). The single most important advantage of this approach is that while Giza++ requires a large parallel corpus, fastText vectors are trained on monolingual data and Vecmap needs only a (much smaller) bilingual dictionary. Bilingual dictionaries are readily available for many language pairs via Wiktionary (Acs, 2014).

The experiments showed that the new features can have a positive impact on the F-score (depending on the configuration), but precision was somewhat lower compared to when we were using only Giza++ features. When we removed Giza++ features and using only the new embedding-based features (alongside cognate features which are based on word similarity and require no pre-existing bilingual data), we observed somewhat lower recall and slightly higher precision. This means that the embedding-based features can be used instead of Giza++ features for language pairs where no large parallel bilingual corpora are available.

In terms of future work, we plan on creating additional features using contextual embeddings, such as ELMo (Peters et al., 2018) and BERT (Devlin et al., 2018), which could potentially help us improve recall, and explore more granular and detailed training set filtering techniques. We also plan to expand the experiments and test other configurations in a more systematic way.

6. Acknowledgements

The work was supported by European Union's Horizon 2020 research and innovation programme under grant agreement No 825153, project EMBEDDIA (Cross-Lingual Embeddings for Less-Represented Languages in European News Media). We also acknowledge the project the financial support from the Slovenian Research Agency for research core funding for the programme Knowledge Technologies (No. P2-0103) and the project TermFrame - Terminology and Knowledge Frames across Languages (No. J6-9372).

7. References

Acs, J. (2014). Pivot-based multilingual dictionary building using Wiktionary. In *Proceedings of the Ninth International Conference on Language Resources and Evaluation LREC*.

Aker, A., Paramita, M. & Gaizauskas, R. (2013). Extracting bilingual terminologies from comparable corpora. In *Proceedings of the 51st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, volume 1. pp. 402–411.

Artetxe, M., Labaka, G. & Agirre, E. (2018). Generalizing and improving bilingual word embedding mappings with a multi-step framework of linear transformations. In *Thirty-Second AAAI Conference on Artificial Intelligence*.

Baldwin, T. & Tanaka, T. (2004). Translation by machine of complex nominals: Getting it right. In *Proceedings of the Workshop on Multiword Expressions: Integrating Processing*. pp. 24–31.

Bojanowski, P., Grave, E., Joulin, A. & Mikolov, T. (2016). Enriching Word Vectors with Subword Information. *arXiv preprint arXiv:1607.04606*.

Bouamor, D., Semmar, N. & Zweigenbaum, P. (2013). Context Vector Disambiguation for Bilingual Lexicon Extraction from Comparable Corpora. In *Proceedings of the 51st Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers)*. pp. 759–764.

Cao, Y. & Li, H. (2002). Base Noun Phrase Translation Using Web Data and the EM Algorithm. In *Proceedings of the 19th International Conference on Computational Linguistics - Volume 1*. pp. 1–7.

Chiao, Y.C. & Zweigenbaum, P. (2002). Looking for Candidate Translational Equivalents in Specialized, Comparable Corpora. In *Proceedings of the 19th International Conference on Computational Linguistics - Volume 2*. pp. 1–5.

Daille, B., Gaussier, E. & Langé, J.M. (1994). Towards Automatic Extraction of Monolingual and Bilingual Terminology. In *Proceedings of the 15th Conference on Computational Linguistics - Volume 1*. pp. 515–521.

Daille, B. & Morin, E. (2005). French-English Terminology Extraction from Comparable Corpora. In *Natural Language Processing – IJCNLP 2005*. pp. 707–718.

Devlin, J., Chang, M.W., Lee, K. & Toutanova, K. (2018). BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding. *arXiv preprint arXiv:1810.04805*.

Fung, P. & Yee, L.Y. (1998). An IR Approach for Translating New Words from Nonparallel, Comparable Texts. In *Proceedings of the 17th International Conference on Computational Linguistics - Volume 1*. pp. 414–420.

Gaizauskas, R., Aker, A. & Yang Feng, R. (2012). Automatic bilingual phrase extraction from comparable corpora. In *24th International Conference on Computational Linguistics*. pp. 23–32.

Gaussier, E. (1998). Flow Network Models for Word Alignment and Terminology Extraction from Bilingual Corpora. In *Proceedings of the 17th International Conference on Computational Linguistics - Volume 1*. pp. 444–450.

Hazem, A. & Morin, E. (2016). Efficient Data Selection for Bilingual Terminology Extraction from Comparable Corpora. In *Proceedings of COLING 2016, the 26th International Conference on Computational Linguistics: Technical Papers*. pp. 3401–3411.

Hazem, A. & Morin, E. (2017). Bilingual Word Embeddings for Bilingual Terminology Extraction from Specialized Comparable Corpora. In *Proceedings of the 8th International Joint Conference on Natural Language Processing (Volume 1: Long Papers)*. pp. 685–693.

Joachims, T. (2002). *Learning to classify text using support vector machines: Methods, theory and algorithms*. Kluwer Academic Publishers.

Kupiec, J. (1993). An algorithm for finding noun phrase correspondences in bilingual corpora. In *Proceedings of the 31st annual meeting on Association for Computational Linguistics*. pp. 17–22.

Levenshtein, V.I. (1966). Binary Codes Capable of Correcting Deletions, Insertions and Reversals. *Soviet Physics Doklady*, 10, p. 707.

Morin, E., Daille, B., Takeuchi, K. & Kageura, K. (2008). Brains, Not Brawn: The Use of Smart Comparable Corpora in Bilingual Terminology Mining. *ACM Trans. Speech Lang. Process.*, 7(1), pp. 1:1–1:23.

Nassirudin, M. & Purwarianti, A. (2015). Indonesian-Japanese term extraction from bilingual corpora using machine learning. In *Advanced Computer Science and Information Systems (ICACSIS), 2015 International Conference on*. pp. 111–116.

Och, F.J. & Ney, H. (2003). A systematic comparison of various statistical alignment models. *Computational linguistics*, 29(1), pp. 19–51.

Peters, M.E., Neumann, M., Iyyer, M., Gardner, M., Clark, C., Lee, K. & Zettlemoyer, L. (2018). Deep contextualized word representations. In *Proc. of NAACL*.

Pinnis, M., Ljubešić, N., Stefanescu, D., Skadina, I., Tadić, M. & Gornostaya, T. (2012). Term extraction, tagging, and mapping tools for under-resourced languages. In *Proceedings of the 10th Conference on Terminology and Knowledge Engineering (TKE 2012), June*. pp. 20–21.

Rapp, R. (1999). Automatic Identification of Word Translations from Unrelated English and German Corpora. In *Proceedings of the 37th Annual Meeting of the Association for Computational Linguistics on Computational Linguistics*. pp. 519–526.

Repar, A., Martinc, M. & Pollak, S. (2019). Reproduction, replication, analysis and adaptation of a term alignment approach. *Language Resources and Evaluation*, pp. 1–34.

Steinberger, R., Eisele, A., Klocek, S., Pilos, S. & Schlüter, P. (2013). DGT-TM: A freely available translation memory in 22 languages. In *Proceedings of the 8th international conference on Language Resources and Evaluation (LREC'2012)*.

Steinberger, R., Pouliquen, B. & Hagman, J. (2002). Cross-lingual document similarity calculation using the multilingual thesaurus eurovoc. *Computational Linguistics and Intelligent Text Processing*, pp. 101–121.

Tiedemann, J. (2012). Parallel Data, Tools and Interfaces in OPUS. In N.C.C. Chair), K. Choukri, T. Declerck, M.U. Dogan, B. Maegaard, J. Mariani, J. Odijk & S. Piperidis (eds.) *Proceedings of the Eight International Conference on Language Resources and Evaluation (LREC'12)*. Istanbul, Turkey: European Language Resources Association (ELRA).

Vintar, Š. (2010). Bilingual term recognition revisited: The bag-of-equivalents term alignment approach and its evaluation. *Terminology. International Journal of Theoretical and Applied Issues in Specialized Communication*, 16(2), pp. 141–158.