Lang Resources & Evaluation
https://doi.org/10.1007/s10579-019-09477-1

ORIGINAL PAPER - REPLICABILITY & REPRODUCIBILITY

# Reproduction, replication, analysis and adaptation of a term alignment approach

**Andraž Repar**[1,2] · **Matej Martinc**[1,2] · **Senja Pollak**[2,3]



**Abstract** In this paper, we look at the issue of reproducibility and replicability in bilingual terminology alignment (BTA). We propose a set of best practices for reproducibility and replicability of NLP papers and analyze several influential BTA papers from this perspective. Next, we present our attempts at replication and reproduction, where we focus on a bilingual terminology alignment approach described by Aker et al. (Extracting bilingual terminologies from comparable corpora. In: Proceedings of the 51st annual meeting of the association for computational linguistics, vol. 1 402–411, 2013) who treat bilingual term alignment as a binary classification problem and train an SVM classifier on various dictionary and cognate-based features. Despite closely following the original paper with only minor deviations—in areas where the original description is not clear enough—we obtained significantly worse results than the authors of the original paper. We then analyze the reasons for the discrepancy and describe our attempts at adaptation of the approach to improve the results. Only after several adaptations, we achieve results which are close to the results published in the original paper. Finally, we perform the experiments to verify the replicability and reproducibility of our own code. We publish our code and datasets online to assure the reproducibility of the results of our experiments and implement the selected BTA models in an online

✉ Andraž Repar
repar.andraz@gmail.com

Matej Martinc
matej.martinc@ijs.si

Senja Pollak
senja.pollak@ijs.si

[1] Jožef Stefan Postgraduate School, Ljubljana, Slovenia

[2] Jožef Stefan Institute, Ljubljana, Slovenia

[3] Usher institute, Medical school, University of Edinburgh, Edinburgh, UK

Published online: 18 November 2019

platform making them easily reusable even by the technically less-skilled researchers.

**Keywords** Bilingual term alignment · Reproducibility · Machine learning · Cognates

## 1 Introduction

The issue of reproducibility has been on the radar of researchers at least for the past 25 years, particularly in the life science research (e.g. Yentis et al. 1993; Prinz et al. 2011; Camerer et al. 2016). More recently, many other disciplines have started to acknowledge the crisis of reproducibility, among them also human language technology research (Pedersen 2008; Kano et al. 2009; Fokkens et al. 2013; Branco et al. 2017; Wieling et al. 2018). However, the basic terminology has remained confusing with different authors using different terms for the same concepts which is why Cohen et al. (2018) describe the three dimensions of reproducibility in natural language processing (NLP) and provide a set of definitions for the various concepts used when discussing reproducibility in NLP. They first differentiate between the concepts of **replicability** (or repeatability), which they define as *the ability to repeat the experiment described in a study*, and **reproducibility**, which describes the outcome—whether *the replicability efforts lead to the same conclusions.* Then they further break down reproducibility into reproducibility of a **conclusion** (defined as an explicit statement in the paper arrived at on the basis of the results of the experiments), reproducibility of a **finding** (a relationship between the values for some reported figure of merit) and reproducibility of a **value** (actual measured or calculated numbers).

In this paper we extend our reproducibility study (Repar et al. 2018), presented at the Workshop on Research Results Reproducibility and Resources Citation (4REAL Workshop, Branco et al. (2018)) organized within the scope of the 11th Language Resources and Evaluation Conference (LREC 2018). Our original motivation came from our interest and need for a terminology alignment tool, and the paper by Aker et al. (2013) titled "Extracting Bilingual Terminology from Parallel Corpora" seemed a perfect candidate for reproduction with nearly perfect results, coverage of the Slovenian-English pair (which were the languages of our interest) and what seemed like a well described and simple to replicate method. The authors treat aligning terms in two languages as a binary classification problem. They use an SVM binary classifier (Joachims 2002) and training data terms taken from the Eurovoc thesaurus (Steinberger et al. 2002) and construct two types of features: dictionary-based (using word alignment dictionaries created with Giza++ (Och and Ney 2003)) and cognate-based (effectively utilizing the similarity of terms across languages). Given that the results looked very promising—precision on the held-out set was 1 or close to 1 for many language pairs, we thought we could use the approach in our work and we set out to replicate it. We expected a straightforward process, but it turned out to be anything but: the results of our experiments were very vastly different from the original paper. For example, while the original paper

reports an extremely high precision (1 or close to 1) for the language pairs we have focused on, our experiments showed a precision below 0.05. Based on the reproducibility dimensions mentioned above, in our original reproducibility experiment from Repar et al. (2018) we were not able to reproduce any of the three dimensions: the values and findings in our experiments were vastly different, and—had we stopped at this point—we would have concluded that the proposed machine learning approach is not suitable for bilingual terminology alignment. Only after a great deal of tweaking and optimization have we managed to get to a respectable precision level (similar to the results in the original paper).

In the present paper, we aim to explore the issue of reproducibility and replicability in the field of terminology alignment further. To do so, we extend the work in Repar et al. (2018) with the following:

- an overview of bilingual terminology extraction and alignment approaches in terms of replicability and reproducibility.
- extending the original reproducibility experiment to two additional languages, resulting in Slovenian, French and Dutch as target languages from three different language families.
- providing very detailed description of feature construction.
- additional filtering and refinement of the cognate-based features.
- a reproducibility experiment with source code from Repar et al. (2018).
- implementation of our code into an online data mining platform ClowdFlows.
- a discussion on good practices for reproducibility and replicability in NLP.

This paper is organized as follows: After the introduction in Sect. 1, we present the related work and the analysis of bilingual terminology alignment papers from the point of view of replicability and reproducibility (Sect. 2). Section 3 contains the main replicability and reproducibility experiments, and is followed by Sect. 4, which describes our attempts at improving the results of the replicated approach, while Sect. 5 contains the results of manual evaluation. Section 6 describes the reproducibility experiment using our code from Repar et al. (2018) and Sect. 7 the implementation of the system in the ClowdFlows platform, for making it accessible to a wider community. Section 8 contains the conclusions and presents ideas for future work. The code and datasets of our experiments are published online, to enable future reproducibility and replicability.[1]

## 2 Overview of bilingual terminology extraction and alignment approaches

In this section we first look at the related work on bilingual terminology extraction and alignment and then analyze several related papers from the viewpoint of replicability and reproducibility.

[1] http://source.ijs.si/mmartinc/4real2018.

### 2.1 Related work

We start by providing a clarification regarding the terminology used in this paper. Following the distinction between two basic approaches made by Foo (2012):

- *extract-align* where we first extract monolingual candidate terms from both sides of the corpus and then align the terms, and
- *align-extract* where we first align single and multi-word units in parallel sentences and then extract the relevant terminology from a list of candidate term pairs.

we propose the following two definitions:

- *Bilingual terminology extraction* is the process which, given the input of related specialized monolingual corpora, results in the output of terms aligned between two languages. The process can either start with extracting monolingual candidate terms and aligning them between two languages (i.e. extract-align) or with aligning phrases and then extracting terms (i.e. align-extract) or any other sequence of actions.
- *Bilingual terminology alignment* is the process of aligning terms between two candidate term lists in two languages.

Bilingual terminology alignment has a narrower focus than bilingual terminology extraction, but the two terms are often used interchangeably in various papers. For example, the title of the paper we were trying to replicate "Extracting bilingual terminologies from comparable corpora" is somewhat misleading in this regard, since the paper primarily deals with bilingual terminology alignment, while they utilize monolingual terminology extraction (specifically the approach by Pinnis et al. (2012) without any modifications) only in the manual evaluation experiments.

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 (in the TMX file format). 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 continued until today (Ha et al. 2008; Ideue et al. 2011; Macken et al. 2013; Haque et al. 2014; Arčan et al. 2014; Baisa et al. 2015). 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

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

involving English) considerable efforts have also been invested into researching bilingual terminology extraction from comparable corpora (Fung and Yee 1998; Rapp 1999; Chiao and Zweigenbaum 2002; Cao and Li 2002; Daille and Morin 2005; Morin et al. 2008; Vintar 2010; Bouamor et al. 2013; Hazem and Morin 2016, 2017).

Despite the problem of bilingual term alignment lending itself well to the binary classification task, there have been relatively few approaches utilizing machine learning. For example, similar to Aker et al. (2013), Baldwin and 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 and Li (2002). Finally, Nassirudin and Purwarianti (2015) also reimplement Aker et al. (2013) for the Indonesian-Japanese language pair and further expand it with additional statistical features. In the best scenario, their accuracy, precision and recall all exceed 90% but the results are not directly comparable since Nassirudin and Purwarianti (2015) use tenfold cross-validation while Aker et al. (2013) use a held-out test set. In addition, Nassirudin and Purwarianti (2015) have a balanced test set while Aker et al. (2013) use a very unbalanced one (ratio of positive vs. negative examples 1:2000).

## 2.2 Analysis of past papers on bilingual terminology extraction from the viewpoint of reproducibility and replicability

In an ideal reproducibility and replicability scenario, a scientific paper would contain an accurate and clear description of the datasets used and experiments conducted and the authors would provide a single link containing all the datasets (versions, subsets etc.) used for the experiments along with the experiment source code (or alternatively, an online tool to run the experiments). These could then be used to replicate the experiments and reproduce the results using the descriptions provided in the paper.

We have analyzed several[3] bilingual terminology extraction papers from the past 25 years from the point of view of dataset, code and tool availability. The summary of results is available in Table 1.

### *2.2.1 Dataset availability*

In terms of dataset availability, we looked at whether the paper contains some description of how the datasets were constructed and which could (theoretically) be used to reconstruct the datasets. Note that under "dataset", we include corpora, gold

[3] The selection process was as follows: the starting point were selected seminal papers on the field, as well as two queries in the ACL Anthology database: "term alignment" and "bilingual terminology extraction". We analyzed the papers found by these two queries as well as additional papers mentioned in the related works sections of these papers and the main criterion for including a paper in our analysis was that it primarily deals with bilingual terminology extraction (and not for example latent semantic analysis, such as Bader and Chew (2008)). However, no strict systematic review with inclusion and exclusion criteria was made, as such a survey would be beyond the needs of this paper.

**Table 1** An analysis of bilingual terminology extraction papers from the point of view of reproducibility and replicability

| Paper | Dataset | Code | Tool | Google Scholar citations as of September 2019 |
|---|---|---|---|---|
| Kupiec (1993) | Links | No | No | 333 |
| Daille et al. (1994) | No | No | No | 268 |
| Fung and Yee (1998) | Description | No | No | 427 |
| Gaussier (1998) | No | No | No | 84 |
| Rapp (1999) | Description | No | No | 552 |
| Chiao and Zweigenbaum (2002) | Description | No | No | 135 |
| Cao and Li (2002) | Description | No | No | 141 |
| Morin et al. (2007) | No | No | No | 113 |
| Daille and Morin (2005) | Obsolete | No | Obsolete | 56 |
| Morin et al. (2008) | Links | No | Obsolete | 22 |
| Ha et al. (2008) | Description | No | No | 4 |
| Lee et al. (2010) | Description | No | No | 22 |
| Vintar (2010) | No | No | Obsolete | 53 |
| Ideue et al. (2011) | No | No | Yes[a] | 9 |
| Macken et al. (2013) | No | No | No | 48 |
| Bouamor et al. (2013) | Description | No | No | 24 |
| Aker et al. (2013) | Links | No | No | 36 |
| Arčan et al. (2014) | Links | No | No | 18 |
| Haque et al. (2014) | Links | No | No | 11 |
| Kontonatsios et al. (2014) | Description | No | No | 14 |
| Baisa et al. (2015) | No | No | Yes | 5 |
| Hazem and Morin (2016) | Links | No | No | 12 |
| Hazem and Morin (2017) | Links | No | No | 2 |

[a]A Perl module (Term Extract) was used, however the link leads to a Japanese website

standard termlists, seed dictionaries and all other linguistic resources needed to conduct the experiments in the paper. For example, we consider the following paragraph from Rapp (1999) to be a valid description of a dataset: *As the German corpus, we used 135 million words of the newspaper Frankfurter Allgemeine Zeitung (1993 to 1996), and as the English corpus 163 million words of the Guardian (1990 to 1994).* On the other hand, this paragraph from Ideue et al. (2011) is not considered a valid description: *We extracted bilingual term candidates from a Japanese-English parallel corpus consisting of documents related to apparel products.* In the former example, dataset reconstruction would be difficult but not impossible, while in the latter it is impossible. An even better option is to link to actual datasets or refer to papers where datasets are described and linked, which is why we also looked for dataset links and/or references in the analyzed papers. Note that there are several examples where links are provided only for a selection of the datasets used in the experiments (e.g., Morin et al. (2008)).

As evident from Table 1, dataset availability is the least problematic aspect of reproducibility and replicability in terminology (extraction and) alignment papers with approximately two thirds of the analyzed papers (15 out of 23) either containing a description of the resources used for the experiments, providing links to them or refering to papers where they are described.

We expected the earlier papers to have less information on datasets than latter ones, but this turned out not to be the case. In fact, the earliest paper analyzed—Kupiec (1993)—provides a reference to a publicly available corpus (Canadian Hansards (Gale and Church 1993)). The first paper to have a separate section with data/resource description is Rapp (1999) and from this point on, almost all papers have such a section—usually titled "Data and Resources", "Resources and Experimental Setup", "Linguistic resources" or similar.

However, it is rarely documented what version of the dataset was used and whether an entire dataset was used or only a part of it (as in random selection, train-test split, etc.). In most cases, little information is provided on the actual subsets used for the experiments. Another aspect of dataset use is the languages: when one of the languages involved is English, it is much easier to find datasets than for other language combinations. Finally, there is also the issue of keeping the links active. For example, many of the links in Daille and Morin (2005) and Morin et al. (2008) are not active anymore while Bouamor et al. (2013) state that the corpora and terminology gold standard lists created for the paper will be shared publicly, but no links are provided.

The most significant problem encountered during our analysis was the fact that terminology alignment is most often not the sole focus of a paper, such as in Haque et al. (2014), where the experiments start with monolingual terminology extraction from two languages and the extracted terms are then aligned. As terminology extraction and alignment go hand-in-hand, it may often be impossible to make a clear distinction between the terminology extraction and terminology alignment datasets. This means that the dataset results in Table 1 are not a true apple-to-apple comparison: one paper might link to the parallel corpus used to extract terms from, while another to a gold standard termlist. Our main criterion was whether the dataset description (or link) could be used to replicate the experiments described in the paper.

An ideal terminology (extraction and) alignment dataset would therefore consist of a bilingual or multilingual (parallel or comparable) corpus along with reference (gold standard) term lists containing terms that can be found in the corpus. Such corpora are TTC wind energy and TC mobile technology[4], which contain data for six languages (English, French, German, Spanish, Russian, Latvian, Chinese), or the Bitter corpus[5], which contains data for the EN-IT language pair. The first was used in Hazem and Morin (2016), while the second one by Arčan et al. (2014). Since such datasets are scarce, researchers employ various methodologies for constructing their own datasets. One method, used by Aker et al. (2013), is to take one of the available multilingual translation memories containing EU documentation (such as

[4] http://www.lina.univ-nantes.fr/?Reference-Term-Lists-of-TTC.html.

[5] https://hlt-mt.fbk.eu/technologies/bittercorpus.

Europarl (Koehn 2005) or DGT (Steinberger et al. 2013)) as the corpus and a glossary (e.g., IATE (Johnson and Macphail 2000)) or thesaurus (e.g., Eurovoc (Steinberger et al. 2002)) as the terminology gold standard list. Another strategy, used by Hazem and Morin (2017), is to collect a comparable corpus manually (i.e. scientific articles in French and English from the Elsevier[6] website) and a domain specific terminological resource (i.e. UMLS[7]) as a reference termlist. Hazem and Morin (2017) also filter out those terms from the termlist that do not appear often enough in their corpus. In other cases (e.g., Haque et al. (2014)), the datasets are not available because the papers were written as part of industrial projects and the datasets are private.

#### *2.2.2 Code and tool availability*

We have discovered that no paper has made experiment code available and only a few provide access or links to tools where the experiments were conducted. But even when links to tools are provided, reproducibility and replicability may be hindered: for example, the link provided in Ideue et al. (2011) leads to a Japanese website. Another issue is the long-term availability of resources. For example, Daille and Morin (2005) conducted their experiments in *ACABIT*, an open source terminology extraction software. However, the link given in the paper does not work anymore. From the analyzed papers, the only example of bilingual term extraction and alignment tool, which is publicly available, is the Sketch Engine term extraction module, described by Baisa et al. (2015).

None of the papers analyzed in this section fulfill the ideal scenario described at the start of this section (i.e. a single link with code and all datasets) which severly hinders any replicability attempts as will be evident from our own experiments described in this paper.

## 3 Replicating a machine learning approach to bilingual term alignment and reproducing its results

This section describes our efforts in replicating a machine learning approach to bilingual term alignment described in Aker et al. (2013),by which we extend our initial experiments and analysis (Repar et al. 2018). Section 3.1 describes the original approach and Sect. 3.2 contains an overview of our attempts to replicate it.

### 3.1 Description of the original approach

The original approach designed by Aker et al. (2013) was developed to align terminology from comparable (or 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

[6] https://www.elsevier.com/.

[7] https://www.nlm.nih.gov/research/umls/.

trade-off between training error and margin parameter c = 10). The task of bilingual alignment is treated as a binary classification—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. They then extract features (dictionary and cognate-based) to be used by the classifier. They 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 20 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 of our main interest, as well as for the additional target languages that we selected, namely French and Dutch, the reported results were excellent with perfect or nearly perfect precision and good recall for all three language pairs. 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.

#### *3.1.1 Features*

Aker et al. (2013) use two types of features that express correspondences between the words (composing a term) in the target and source language (for a detailed description see Table 2:

- 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, and

- 5 cognate-based (on the basis of Gaizauskas et al. (2012)) which utilize string-based word similarity between languages.

To match words with morphological differences, they do not perform direct string matching but utilize Levenshtein Distance. Two words were considered equal if the Levenshtein Distance (Levenshtein 1966) was equal or higher than 0.95. For closed-compounding languages, they check whether the compound source term has an initial prefix that matches the translation of the first target word, provided that translation is at least 5 characters long.

**Table 2** Features used in the experiments

| Feature | Cat | Description | Type |
|---|---|---|---|
| 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) | Bin |
| isLastWordTranslated | Dict | Checks whether the last word of the source term is a translation of the last word in the target term | Bin |
| percentageOfTranslatedWords | Dict | Ratio of source words that have a translation in the target term | Num |
| percentageOfNotTranslatedWords | Dict | Ratio of source words that do not have a translation in the target term | Num |
| 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) | Num |
| 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) | Num |
| Longest Common Subsequence Ratio (LCSSR) | Cogn | Measures the longest common non-consecutive sequence of characters between two strings (divided by the length of the longest string) | Num |
| Longest Common Substring Ratio (LCSTR) | Cogn | Measures the longest common consecutive string (LCST) of characters that two strings have in common (divided by the length of the longest string) | Num |
| Dice similarity | Cogn | 2*LCST / (len(source) + len(target)) | Num |
| Needlemann-Wunsch distance | Cogn | LCST / min(len(source), len(target)) | Num |
| Normalized Levenshtein distance (nLD) | Cogn | 1 − LD / max(len(source), len(target)) | Num |
| isFirstWordCovered | Comb | A binary feature indicating whether the first word in the source term has a translation or transliteration in the target term | Bin |
| isLastWordCovered | Comb | A binary feature indicating whether the last word in the source term has a translation or transliteration in the target term | Bin |
| percentageOfCoverage | Comb | Returns the percentage of source term words which have a translation or transliteration in the target term | Num |
| percentageOfNonCoverage | Comb | Returns the percentage of source term words which have neither a translation nor transliteration in the target term | Num |
| difBetweenCoverageAndNonCoverage | Comb | Returns the difference between the last two features | Num |

Note that some features are used more than once because they are direction-dependent

Additional features are also constructed by:

- Using language pair specific transliteration rules to create additional cognate-based features. The purpose of this task was to try to match the cognate terms while taking into account the differences in writing systems between two languages: e.g. Greek 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 additional 10 cognate-based features with transliteration rules.
- Combining the dictionary and cognate-based features in a set of combined features where the term pair alignment is correct if either the dictionary or the cognate-based method returns a positive result. This process resulted in additional 10 combined features.[8]

At the end of the feature construction phase, there were 38 features: 13 dictionary-based, 5 cognate-based, 10 cognate-based features with transliteration rules and 10 combined features.

### *3.1.2 Data source and experiments*

Using Giza++, Aker et al. (2013) create 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.)

The next step is the creation of term pairs from the Eurovoc (Steinberger et al. 2002) thesaurus, which at the time consisted of 6797 terms. Each non-English language was paired with English. The test set consisted of 600 positive (correct) term pairs—taken randomly out of the total 6797 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

[8] 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, Levenshtein Distance, Needleman-Wunsch Distance, Dice) is 0.70 or higher (set experimentally by Aker et al. (2013)).

200 pairs with multiple-word terms on both sides. The remaining positive term pairs (approximately 6200) were used as training data along with additional 6200 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 this approach, Aker et al. (2013) achieve excellent results with 100% precision and 66% recall for Slovenian and French and 98% precision and 82% recall for Dutch.

### 3.2 Replication of the approach

The first step in our approach was to replicate the algorithm described by Aker et al. (2013). The initial premise is the same: given two lists of terms from the same domain in two different languages, we would like to align the terms in the two lists to get one bilingual glossary to be used in a variety of settings (computer-assisted translation, machine translation, ontology creation etc.). We followed the approach described above faithfully except in the following aspects[9]:

- Instead of the entire set of Eurovoc languages, we have initially focused only on the English-Slovenian language pair (Repar et al. 2018). In the current paper, we add two additional language pairs (English-French, English-Dutch) to see whether our findings can be generalised across different languages. We selected languages from different language families, as the importance of cognates is dependent on the similarity between languages (for example, Dutch and English (being both Germanic languages) presumably have a higher number of cognates).
- We use newer datasets. The Eurovoc thesaurus version that we used contained 7,083 terms for Slovenian[10] and 7,181 terms for French[11] and Dutch.[12] Similarly, the DGT translation memory contains additional content not yet present in 2013.[13] For English-Slovenian, we at first used the entire DGT corpus up to and including the *DGT-TM-release 2017* for deriving GIZA alignments. Later we also experimented with precomputed dictionaries by Aker et al. (2014). When performing the experiments on the other languages pairs, we did not create our own GIZA alignment, but only used the precomputed ones by Aker et al. (2014).
- Since no particular cleaning of training data (e.g., manual removal of specific entries) is described in the paper for the languages of our interest, we do not perform any.

We think that regardless of these differences, the experiments should yield similar results.

[9] Note that our original replication paper Repar et al. (2018) wrongly states that we did not utilize the compounding solution implemented by Aker et al. (2013) for addressing compouding issues in languages such as German. In fact, we did implement it and used it in all experiments.

[10] http://source.ijs.si/mmartinc/4real2018/blob/master/term_list_sl.csv.

[11] http://source.ijs.si/mmartinc/4real2018/blob/master/term_list_fr.csv.

[12] http://source.ijs.si/mmartinc/4real2018/blob/master/term_list_nl.csv.

[13] The versions of the resources used in Aker et al. (2013) were not documented or made available.

### 3.2.1 Problems with replicating the approach

While the general approach is clearly laid out in the article, there are several spots where further clarification would be welcome:

- There is no sufficient information about the Giza++ settings or whether the input corpora have been lemmatized. In order to improve term matching, we experimented with and without lemmatization of the Giza++ input corpora.
- There is no information about the specific character mappings rules other than a general principle of one character in the source being mapped to one or more character in the target. Since the authors cover 20 languages, it is understandable that they cannot include the actual mapping rules in the article. Therefore, we have created our own mapping rules for English-Slovenian and English-French according to the instructions in the original paper:
  - Mapping the English term to the Slovenian writing system (the character before the colon is replaced by the sequence of characters after the colon): *x:ks, y:j, w:v, q:k.*
  - Mapping the Slovenian term to the English writing system: *č:ch, š:sh, ž:zh.*
  - Mapping the French term to the English writing system: we deleted all accents e.g., é:e, ê:e.
  - Mapping the Dutch term to the English writing system: we deleted all accents and replace the digraph ij with two separate letters ij.

- Instead of the unclear Needleman–Wunsch distance formula from Aker et al. (2013) $\frac{LCST}{min[len(source)+len(target)]}$ (which implies that we should take the minimum value of the sum of the length of the target and source term) we opted for $\frac{LCST}{min[len(source),len(target)]}$ as in Nassirudin and Purwarianti (2015).
- We were not completely certain how to treat examples such as "passport—potni list", where a single-word source term is translated by a multi-word target term and both combinations (passport—potni and passport—list) can be found in the Giza++ dictionary. In this case, our implementation returns values of 1 for both *isFirstWordTranslated* and *isLastWordTranslated* features despite the fact that the source term only has one word.
- There was a slight ambiguity on how to calculate cognate-based features: on the level of words or on the level of entire terms. We opted for the second, since the names of the cognate-based features did not imply that cognates are calculated on the word level (as was the case with the dictionary-based features) and since there was no mention in the original paper on how to combine cognate-based scores for specific word pairs in the multi-word term pairs in order to get a final cognate score for the whole term pair.
- In the original article, the *isFirstWordCovered* feature is described as "a binary feature indicating whether the first word in the source term has a translation (i.e. has a translation entry in the dictionary regardless of the score) or transliteration (i.e. if one of the cognate metric scores is above 0.7) in the target term." While

the dictionary-based part is clear, for calculating the cognate-based feature values (e.g., of the first word in the source term), the values of the cognate metric scores concern the entire target term. As we did not find this fully intuitive, and we believe other interpretations are possible, we experimented with these settings in the adaptation of the approach (see Sect. 4.8).
To avoid ambiguities, we provide a separate document with examples of constructed features, together with the code (http://source.ijs.si/mmartinc/4real2018/blob/master/feature_examples.docx).

### *3.2.2 Results*

The evaluation on the test set created as described in the original paper by Aker et al. (2013) shows that compared to the results reported by the authors (see line 1 in Tables 3, 4 and 5), our results are significantly worse. Despite all our efforts to follow the original approach, we were unable to match the results achieved in the original paper when running the algorithm without any changes to the original approach. When trying to follow the original paper's methodology, precision is only 3.59% and recall is 88% for the English-Slovenian language pair. The results for the other two language pairs are comparable (see line 2 in Tables 3, 4 Table 5 for details).

In Sect. 4, we provide the results of detailed analysis and additional experiments that we performed in order to reach results comparable to the original approach.

### *3.2.3 Attempts at establishing contact with the authors*

When replicating an existing paper, especially when the code is not made available, contacting the authors for clarification (or for providing/running the code) is the most obvious step when encountering the problems or ambiguities. However, due to busy schedules of researchers, change of professional paths or other similar reasons, getting detailed help might be impossible.

This is true for our case as well. Initially, we were hopeful of getting useful feedback, as the authors already provided the software to other researchers in the past (see Arčan et al. (2014)). However, despite a friendly response, we have been able to get only a limited number of answers and many questions remained unanswered, and the auhors have not been able to share their code. We have first contacted the original authors of the paper when we were running the experiments reported in Repar et al. (2018) and did receive some answers confirming our assumptions (e.g. regarding mapping terms to the different writing systems and that the test set data was selected individually for each language pair), but several other issues remained unaddressed (in particular, what was the exact train and test data selection strategy for the EN-SL language pair). Further inquiries proved unsuccessful due to time constraints on the part of the original authors. As we expanded the paper with additional languages and experiments, we again contacted the main author, provided him the code and the paper and asked for help in

**Table 3** Results on the English–Slovenian term pair

| No. | Config EN-SL | Training set size | Pos/neg ratio | Precision | Recall | F-score |
|---|---|---|---|---|---|---|
| 1 | Reported by Aker et al. (2013) | 12,400 | 1:1 | 1 | 0.6600 | 0.7900 |
| 2 | Replicated approach | 12,966 | 1:1 | 0.0359 | 0.8800 | 0.0689 |
| 3 | Giza++ terms only | 8306 | 1:1 | 0.0645 | **0.9150** | 0.1205 |
| 4 | Giza++ cleaning | 12,966 | 1:1 | 0.0384 | 0.7789 | 0.0731 |
| 4a | Lemmatization | 12,966 | 1:1 | 0.0373 | 0.8150 | 0.0713 |
| 5 | Training set 1:200 | 1,303,083 | 1:200 | 0.4299 | 0.7617 | 0.5496 |
| 6 | Training set filtering 1 | 6426 | 1:1 | 0.5969 | 0.64167 | 0.6185 |
| 7 | Training set filtering 2 | 35,343 | 1:10 | 0.9042 | 0.5350 | **0.6723** |
| 8 | Training set filtering 3 | 645,813 | 1:200 | **0.9342** | 0.4966 | 0.6485 |
| 9 | Term length filtering | 6426 | 1:1 | 0.8144 | 0.4900 | 0.6119 |
| 10 | Cognates approach | 672,345 | 1:200 | 0.8732 | 0.5167 | 0.6492 |

No. 1 presents the results reported by the authors, No. 2 our replication of the approach and No. 3–10 our modifications of the first replicated approach with the aim of improving the results

**Table 4** Results on the English–French language pair

| No. | Config EN-FR | Training set size | Pos/neg ratio | Precision | Recall | F-score |
|---|---|---|---|---|---|---|
| 1 | Reported by Aker et al. (2013) | 12,400 | 1:1 | 1 | 0.6600 | 0.7900 |
| 2 | Replicated approach | 13,160 | 1:1 | 0.0323 | **0.8483** | 0.0622 |
| 3 | Giza++ terms only | 8892 | 1:1 | 0.0437 | 0.8433 | 0.0830 |
| 4 | Giza++ cleaning | 13,160 | 1:1 | 0.0317 | 0.7917 | 0.0610 |
| 5 | Training set 1:200 | 1,322,580 | 1:200 | 0.5273 | 0.6767 | **0.5927** |
| 6 | Training set filtering 1 | 2650 | 1:1 | 0.4623 | 0.5517 | 0.5030 |
| 7 | Training set filtering 2 | 14,575 | 1:10 | 0.9422 | 0.3533 | 0.5139 |
| 8 | Training set filtering 3 | 266,325 | 1:200 | **0.9791** | 0.3117 | 0.4728 |
| 9 | Term length filtering | 2650 | 1:1 | 0.6791 | 0.3950 | 0.4995 |
| 10 | Cognates approach | 311,952 | 1:200 | 0.8603 | 0.3900 | 0.5367 |

No. 1 presents the results reported by the authors, No. 2 our replication of the approach and No. 3–10 our modifications of the first replicated approach with the aim of improving the results

identification of any possible mistakes leading to the results, however, we were ultimately not able to get any information which would explain the differences.

We think the original paper is generally well-written and that the main reason for occasional lack of clarity is its scope: as the authors deal with more than 20 language pairs, it would be impossible to provide specific information regarding all of them. Providing more examples would be useful, but still the code and the exact dataset are in our opinion the only way to be able to fully replicate the experiments.

**Table 5** Results on the English–Dutch language pair

| No. | Config EN-NL | Training set size | Pos/neg ratio | Precision | Recall | F-score |
|---|---|---|---|---|---|---|
| 1 | Reported by Aker et al. (2013) | 12,400 | 1:1 | 0.9800 | 0.8200 | 0.8000 |
| 2 | Replicated approach | 13,160 | 1:1 | 0.0227 | 0.8850 | 0.0442 |
| 3 | Giza++ terms only | 7310 | 1:1 | 0.0636 | **0.9317** | 0.1191 |
| 4 | Giza++ cleaning | 13,160 | 1:1 | 0.0340 | 0.8500 | 0.0654 |
| 5 | Training set 1:200 | 1,322,580 | 1:200 | 0.5053 | 0.6300 | 0.5608 |
| 6 | Training set filtering 1 | 4250 | 1:1 | 0.5122 | 0.4917 | 0.5017 |
| 7 | Training set filtering 2 | 23,375 | 1:10 | 0.6842 | 0.4333 | 0.5306 |
| 8 | Training set filtering 3 | 427,125 | 1:200 | **0.9356** | 0.3633 | 0.5234 |
| 9 | Term length filtering | 4250 | 1:1 | 0.7621 | 0.3683 | 0.4966 |
| 10 | Cognates approach | 468,933 | 1:200 | 0.9101 | 0.5233 | **0.6646** |

No. 1 presents the results reported by the authors, No. 2 our replication of the approach and No. 3–10 our modifications of the first replicated approach with the aim of improving the results

## 4 Analysis and adaptation: experiments for improving the replicated approach

The results in our replicated experiments differ dramatically from the results obtained by Aker et al. (2013). Their approach yields excellent results with perfect or almost perfect precision and respectable recall for all three languages under our consideration.

For the EN-SL language pair, the reported results have the precision of 100% and the recall of 66%, meaning that with 600 positive term pairs in the test set, their classifier returns only around 400 positive term pairs. In contrast, in our replication attempts the classifier returned a lot of falsely classified positive term pairs. In addition to 526 true positive examples (out of a total of 600), the classifier also returns 14,194 misclassified examples—incorrect term pairs wrongly classified as correct. Similar statistics can be observed for the other two language pairs.

These results are clearly not useful for our goals which is to use the methods to continuously populate a termbase with as little manual intervention as possible. In this section we present the analysis of ambiguities in the description of the approach and the issues spotted when inspecting the results of the replicated approach, and propose several methods aiming at improving the results. To do so, we have performed experiments with regard to the following aspects:

- Giza++ terms only: using only those terms that can be found in the Giza++ training corpora (i.e. DGT).
- Giza++ cleaning.
- Lemmatization.
- Changing the ratio of positive/negative examples in the training set.
- Training set filtering.

The experiments have been initially presented for Slovenian in our short paper in the 4REAL workshop (Repar et al. 2018). Here, we provide additional analysis and extend the experiments to the other two languages under consideration. The results are reported in Sect. 4.1 to 4.5.

In the 4REAL paper, precision was already relatively high (see for example line 8 in Table 3), which is why our additional experiments focused on improving recall. We implemented several additional approaches as reported in Sect. 4.6 to 4.8:

- Removing the Needleman–Wunsch Distance feature.
- Term length filtering.
- Adding new cognate-based features.

### 4.1 Giza++ terms only

We thought that one of the reasons for low results can be that not all EUROVOC terms actually appear in the Giza++ training data (i.e. DGT translation memory). The terms that do not appear in the Giza++ training data could have dictionary-based features similar to the generated negative examples, which could affect the precision of a classifier that was trained on those terms. We found that only 4,153 out of 7,083 Slovenian terms of the entire EUROVOC thesaurus do in fact appear in a DGT translation memory. Using only these terms in the classifier training set did provide modest improvements of precision, recall and F-score across all three languages. For details, see line 3 in Tables 3, 4 and 5.

### 4.2 Giza++ cleaning

The output of the Giza++ tool contained a lot of noise and we thought it could perhaps have a detrimental effect on the results. There is no mention of any sophisticated Giza++ dictionary cleaning in the original paper beyond removing all entries where probability is lower then 0.05 and entries where the source word is less than 4 characters and the target word more than 5 characters in length and vice versa (introduced to avoid stopword-content word pairs). For clean Giza++ dictionaries, we used the resources described in Aker et al. (2014), available via the META-SHARE repository[14] (Piperidis et al. 2014), specifically, the transliteration-based approach which yielded the best results according to the cited paper.

For Slovenian and Dutch, precision and F-score improved marginally at a cost of a lower recall, while for French, precision, recall and F-score all decreased. For details, see line 4 in Tables 3, 4 and 5.

### 4.3 Lemmatization

The original paper does not mention lemmatization which is why we assumed that all input data (Giza++ dictionaries, EUROVOC thesaurus) is not lemmatized. They

[14] http://metashare.tilde.com, last accessed: February 14, 2019.

state that to capture words with morphological differences, they don't perform direct string matching but utilize Levenshtein Distance and two words are considered equal if the Levenshtein Distance (Levenshtein 1966) is equal or higher than 0.95. This led us to believe that no lemmatization was used. Nevertheless, we thought lemmatizing the input data could potentially improve the results which is why we adapted the algorithm to perform lemmatization (using Lemmagen (Juršič et al. 2010)) of the Giza++ input data and the EUROVOC terms. We have also removed the Levenshtein distance string matching and replaced it with direct string matching (i.e. word A is equal to word B, if word A is exactly the same as B), which drastically improved the execution time of the software.

We considered lemmatization as a factor that could explain the difference in results obtained by us and Aker et al. (2013), but our experiments on lemmatized and unlemmatized clean Giza++ dictionaries show that lemmatization does not have a significant impact on the results. Compared to the configuration with unlemmatized clean Giza++ dictionaries, in the configuration with lemmatized Giza++ dictionaries precision was slightly lower (by 0.1%), recall was a bit higher (by around 4%) and F-score was lower by 0.2%. For details, see Table 3, line 4a. As lemmatization significantly slows down the experimentation, we tested the results first on Slovenian, where the influence of the lemmatization should be the largest as it is a morphologically-rich language. As lemmatization did not improve the results, we did not repeat the experiments for French and Dutch.

### 4.4 Changing the ratio of positive/negative examples in the training set

In the original paper, the training set is balanced (i.e. the ratio of positive vs. negative examples is 1) but the test set is not (the ratio is around 1:2000). Since our classifier had low precision and relatively high recall, we figured that an unbalanced training set with much more negative than positive examples could improve the former. To test this, we experimented with training the classifier on unbalanced train sets with different ratios between positive and negative examples. The general tendency we noticed during experimentation is that a very unbalanced train set (ratio of 1:200 between positive and negative examples[15]) greatly improves the precision of the classifier at a cost of somewhat lower recall, when compared to balanced train set or less unbalanced train set (e.g., ratio of 1:10 between positive and negative examples). For details, see line 5 in Tables 3, 4 and 5.

### 4.5 Training set filtering

The original paper mentions that their classifier initially achieved low precision on Lithuanian language training set, which they were able to improve by manually removing 467 positive term pairs that had the same characteristics as negative examples from the training set. No manual removal is mentioned for Slovenian, French and Dutch.

[15] 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.

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. Some EN-SL examples can be seen in Table 6, and similar errors were observed for for the other two language pairs.

Based on this problem of partial translations, leading to false positive examples, we focused on the features that would eliminate these 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 in the training set:

- isfirstwordTranslated = True.
- islasttwordTranslated = True.
- percentageOfCoverage $> 0.66$.
- isfirstwordTranslated-reversed = True.
- islasttwordTranslated-reversed = True.
- percentageOfCoverage-reversed $> 0.66$.

Using this approach, we managed to greatly increase precision at a cost of significant drop in recall values for all three languages. For details see line 6 (*Training set filtering 1*) in Tables 3, 4 and 5. When combining this approach with an unbalanced dataset described in the previous section, we managed to improve precision even further, but again at a cost of lower recall. For details, see lines 7 and 8 (*Training set filtering 2 and 3*) in Tables 3, 4 and 5.

### 4.6 Cognate feature analysis and removing the Needleman–Wunsch Distance feature

We performed an analysis of the results on the English–Slovenian language pair achieved with the best configuration for precision (line 8—*Training set filtering 3* in Table 3) in our experiments (Repar et al. 2018) and discovered that cognate term pairs were not being considered by the classifier. In a way, this was expected since in the previous step we have filtered the training set based on mostly dictionary-based features.

When analyzing the performance of the cognate-based features, we found that four (Longest Common Subsequence Ratio (LCSSR) Longest Common Substring Ratio (LCSTR), Dice Similarity (Dice), Normalized Levenshtein Distance (nLD)) out of five perform as expected with cognate term pairs having high values, but Needleman-Wunsch Distance (NWD) did not. As already mentioned in the beginning, the formula provided by the authors for computing NWD feature possibly contained an error, therefore we opted for the implementation as mentioned in Nassirudin and Purwarianti (2015). Table 7 shows the behaviour of the five cognate-based features. When we are dealing with actual cognates, all five features have high values, but when the two terms in questions are not cognates, only NWD stays high.

**Table 6** Examples of negative term pairs misclassified as positive

| EN | SL | Giza++ |
|---|---|---|
| Agrarian reform | Kmetijski odpadki | Agrarian, kmetijske, 0.29737 |
| Brussels region | Območje proste trgovine | Region, območje, 0.0970153 |
| Energy transport | Nacionalni prevoz | Transport, prevoz, 0.442456 |
| Fishery product | Tekstilni izdelek | Product, izdelek, 0.306948 |

Column 1 contains the English term, column 2 contains the Slovenian term and column 3 contains the Giza++ dictionary entry (from the non-clean version, see Sect. 4.2) responsible for positive dictionary-based features

**Table 7** Cognate-based features values (showing issues with NWD)

| EN | SL | LCSSR | LCSTR | Dice | nLD | NWD |
|---|---|---|---|---|---|---|
| hospitalisation | hospitalizacija | 0.73 | 0.60 | 0.60 | 0.73 | 0.6 |
| monopsony | monopson | 0.89 | 0.89 | 0.94 | 0.89 | 1.00 |
| fish | predstavniška demokracija | 0.12 | 0.12 | 0.20 | 0.12 | 0.75 |
| Yemen | osna obremenitev | 0.25 | 0.25 | 0.38 | 0.25 | 0.80 |

The first two term pairs are actual cognates with all five cognate-based features having high values. The last two pairs are not cognates and show the issues with the Needleman-Wunsch Distance (NWD), which is the only measure that keeps a high value. Note that due to character mapping rules (see Section 3.2.1.), the word "predstavniška" was transformed into "predstavnishka"

For this reason, we ran our experiments without the NWD feature, but the results did not improve since the SVM classifier is known to be capable of handling noisy features.

### 4.7 Term length filtering

Based on error analysis, one of the major issues confusing the classifier were training examples with differing word lengths. E.g., the source term in the example would have one word, but the target term would have two. An analysis of the terms in Eurovoc for the three language pairs in question showed that 26% of the EN-SL term pairs, 34% of the EN-FR term pairs and 48% of the EN-NL term pairs have different word lengths of the source and target terms (the reason for the high ratio in EN-NL is the use of compounds in Dutch). This turned out to be one of the characteristics leading to low classification performance: for Slovenian with the replicated configuration (line 2 in Table 3) the classifier returned a total of 14,721 positively classified examples. 14,193 out of these were false positives—incorrectly aligned term pairs. A further 13376 out of these had different lengths of the source and target terms. A visual inspection of feature values indicated that there is often no clear difference between positive and negative term pairs (see Table 8).

Since this was an issue, we experimented with additional term length filtering. We took the positively classified examples from the *training set filtering 1* approach as described in Sect. 4.5 (see line 6 in the tables) and added an additional filter: if the two terms do not have the same number of words, we change the prediction from positive to negative. Using this additional filter, we achieved good precision for Slovenian (81%), and respectable for French (68%) and Dutch (76%). On the other hand, recall values were badly affected, since one third of positive term pairs in the constructed test set are terms of different word length (meaning that highest possible theoretical recall with this approach is 66%). Recall was again best for Slovenian with a value close to 50% and a bit worse for French and Dutch with a value at around 40% and 37% respectively. Consequently, F-scores were the highest for Slovenian and lower for Dutch and French. For details, see line 9 in Tables 3, 4 and 5.

From the original paper it is clear, that authors were aware of the possible complexity of terms of unequal length, as they consider terms of different lengths in the test set construction. So, we exclude the possibility that authors did not have such examples in the test set.

### 4.8 Cognate-based feature approach

The analysis showed that all *Training set filtering* approaches tend to overestimate the importance of Giza++ features and underestimate cognate-based features. This results in a low recall for correct cognate term pairs, which are rarely classified as positive, if their Giza++ based feature values do not show similarity with Giza++ based feature values for non-cognate correct term pairs. For example, Giza++ dictionary does not contain a Slovenian translation *pacifizem* for the English term *pacifism*, which means that the values of features *isFirstWordTranslated*, *isLastWordTranslated*, *isFirstWordTranslated-reversed* and *isLastWordTranslated-reversed* are False and the values for features *percentageOfCoverage* and *percentageOfCoverage-reversed* are zero, therefore the classifier would have a strong inclination to classify this correct term pair as incorrect, even though cognate based feature values clearly indicate that these two terms are cognates.

In order to improve the detection of cognate terms, we first propose two new cognate based features:

- isFirstWordCognate: 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. For example, the value of the feature for the English-Slovenian term pair *Klaipeda county* - *Klaipedsko okrožje* would be True because the LCST for the first words in both terms is *Klaiped*, which has a length of 7. The length of the longest of the two first words in the terms (*Klaipedsko*) is 10 and 7 divided by 10 is 0.7, which is equal to the threshold value.
- isLastWordCognate: a binary feature which returns True if the longest common consecutive string (LCST) of the last words in the source and target terms

**Table 8** A comparison of dictionary feature values

| Source term | raw material | provision | additional resources | provision |
|---|---|---|---|---|
| Target term | surovine | računovodska rezervacija | surovine | urbanistični predpisi |
| Correctly aligned | True | True | False | False |
| isFirstWordTranslated | 1 | 0 | 0 | 0 |
| isLastWordTranslated | 1 | 1 | 1 | 1 |
| pctOfTransWords | 0.5 | 1 | 0.5 | 1 |
| pctOfNotTransWords | 0.5 | 0 | 0.5 | 0 |
| longestTransUnitInPct | 0.5 | 1 | 0.5 | 1 |
| longestNotTransUnitInPct | 0.5 | 0 | 0.5 | 0 |
| isFirstWordTranslated_R | 0 | 0 | 0 | 0 |
| isLastWordTranslated_R | 1 | 1 | 1 | 1 |
| pctOfTransWords_R | 1 | 0.5 | 1 | 0.5 |
| pctOfNotTransWords_R | 0 | 0.5 | 0 | 0.5 |
| longestTransUnitInPct_R | 1 | 0.5 | 1 | 0.5 |
| longestNotTransUnitInPct_R | 0 | 0.5 | 0 | 0.5 |

The first two term pairs are correctly alligned term pairs from the training set (line 2 in Table 3), the second two are not correctly alligned term pairs. We can observe that the dictionary feature values are very similar—compare for example *raw material/surovine* and *additional resources/surovine*

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. For example, the value of the feature for the English-Slovenian term pair *Latin America - Latinska Amerika* would be True because the LCST for the last words in both terms is *Ameri*, which has a length of 5. The length of the longest of the two last words in the terms is 7 and 7 divided by 5 is 0.714, which is greater than the threshold value.

As having the same number of words in the source and target term could play a role in classification, we also add three new features responsible for encoding term length information:

- sourceTargetLengthMatch: a binary feature that returns True if the number of words in source and target terms match.
- sourceTermLength: returns the number of words in the source term.
- targetTermLength: returns the number of words in the target term.

Analysis of the filtered training set showed that it contained a small number of positive cognate based term pair examples, therefore the first step was to include more of them into the dataset. We build three separate datasets, each of them filtered according to the following feature values:

- isFirstWordCognate = True and isLastWordCognate = True.
- isFirstWordTranslated = True and isLastWordCognate = True.
- isFirstWordCognate = True and isLastWordTranslated = True.

The terms from these three datasets are added to the original filtered train set (we make sure that each positive term pair is represented in the new dataset only once by removing all the duplicates). The new dataset contains two distinct groups of terms, one with favorable Giza++ based features (and unfavorable cognate based features) and one with favorable cognate based features (and in some cases unfavorable Giza++ based features). Since this new dataset structure represents a classic "exclusive or" (XOR) problem which a linear classifier is unable to solve, we also replace the linear kernel of our SVM classifier with the Gaussian one.

Using this approach, precision was close to 90% (Slovenian, French) or just over 90% (Dutch), recall was just over 50% for Slovenian, around 52% for Dutch and close to 40% for French. For details, see line 10 in Tables 3, 4 and 5.

### 4.9 Best results

Overall, the setting with the best precision is Train set filtering 3. Compared to the replicated approach (line 2 in Tables 3, 4 and 5), it has an unbalanced dataset of 1:200 (see Section 4.4) and employs the term filtering strategy described in Sect. 4.5. However, for a small gain in recall at the price of a slight decrease in precision, a good alternative is the Cognates approach (line 10 in Tables 3, 4 and 5), which is

based on the Train set filtering 3 approach and additionally includes the cognate detection strategies described in Sect. 4.8.

## 5 Manual evaluation

The first part of this section contains the manual evaluation replicated from Aker et al. (2013), already reported in Repar et al. (2018), while the second part is novel and contains an evaluation using a new dataset and has a specific focus on cognate term pairs.

### 5.1 Replicating the manual evaluation experiments from the original paper

Similar to the original paper, we also performed manual evaluation. We selected a random subset of term pairs classified as positive by the classifier (using the *Training set filtering 3* configuration (line 8 in Table 3) that yielded the best precision). While the authors of the original approach extract monolingual terms using the term extraction and tagging tool TWSC (Pinnis et al. 2012), we use a workflow for monolingual term extraction by Pollak et al. (2012). Both use a similar approach - terms are first extracted using morphosyntactic patterns and then filtered using statistical measures: TWSC uses pointwise mutual information and TF*IDF, while Pollak et al. (2012) is based on an approach by Vintar (2010) and compares the relative frequencies of words composing a term in the domain-specific (i.e. the one we are extracting terminology from) corpus and a general language corpus.

In contrast to the original paper where they extracted terms from domain-specific Wikipedia articles (for the English-German language pair), we are using two translation memories—one containing finance-related content, the other containing IT content. Another difference is that extraction in the original paper was done on comparable corpora, but we extracted terms from parallel corpora - which is why we expected our results to be better. Each source term is paired with each target term (just as in the original paper - if both term lists contained 100 terms, we would have 10,000 term pairs) and extract the features for each term pair. The term pairs were then presented to the classifier that labeled them as correct or incorrect term translations. Afterwards, we took a random subset of 200 term pairs classified as correct and showed them to an experienced translator[16] fluent in both languages who evaluated them according to the criteria set out in the original paper:

- **1—Equivalence**: The terms are exact translations/transliterations of each other (e.g., *type—tip*).
- **2—Inclusion**: Not an exact translation/transliteration, but an exact translation/transliteration of one term is entirely contained within the term in the other language (e.g., *end date—datum*).
- **3—Overlap**: Not category 1 or 2, but the terms share at least one translated/transliterated word (e.g., *user id—uporabniško ime*).

[16] The original paper used two annotators, hence two lines for each domain in Table 4.

– **4—Unrelated**: No word in either term is a translation/transliteration of a word in the other (e.g., *level—uporabnik*[17]).

The results of the manual evaluation can be found in Table 9. Manual evaluation showed that 72% of positive term pairs in the Finance domain, and 79% of positive term pairs in the IT domain were correctly classified by the classifier. The differences between the *Finance* and *IT* datasets can be partially explained by the *Finance* dataset containing more MWE terms than the *IT* dataset (84 vs. 51 for SL and 78 vs. 49 for EN). On the one hand, this means that the chances of aligning a single word term in one language with a multi word term in another language is greater, hence the greater number of partial translations in *Finance* (category 2 - Inclusion), while on the other, single word terms means less characters for the algorithm to work with, hence the greater number of outright mistakes in *IT* (category 4 - Unrelated). Compared to the original paper, we believe these results are comparable when taking into account the different monolingual extraction procedures , the different language pairs and the human factor related to different annotators.

### 5.2 Evaluation on a Karst terminology gold-standard

As mentioned in Sect. 4, the best configuration in terms of precision used in Repar et al. (2018) (line 8 in Tables 3, 4 and 5) overestimates dictionary-based and underestimates cognate-based features. To alleviate this, we added additional features and filtering strategies to our approach to try to improve cognate term pair alignment (see lines 9 and 10 in the results tables). However, evaluating its performance on EUROVOC is difficult as many terms have favorable dictionary-based features due to the fact that both the Giza++ dictionary and EUROVOC are made from the same content (i.e. EU documentation). For the evaluation in this section, we therefore selected a domain, with a content type which is unlikely to be found in DGT (Steinberger et al. 2013), i.e. karstology, which is the science in the field of geomorphology, specializing in the study of karst formations.

To evaluate our bilingual term alignment approach, we used a gold standard of EN-SL aligned karst terminology,[18] which was manually created by the authors of the karstology corpus (Vintar and Grčić-Simeunović 2016). The gold standard consists of 52 English-Slovenian term pairs. For the evaluation experiment, we aligned all Slovenian term with all English terms, resulting in a dataset of 52 positive examples and 2652 negative examples. With the best configuration for precision (line 8 in Table 3), selected also as the best configuration in Repar et al. (2018), precision was 100%, but recall was only 40.4%. Many term pairs containing cognates such as “eogenetic cave—eogenetska jama”, “epigenic aquifer—epigeni vodonosnik” or “karst polje—kraško polje”, were not aligned. With the final cognate approach (line 10 in Table 3), we managed to retain 100% precision and raise the recall to 50% by finding 7 additional cognate term pairs (*aggressive*

[17] “uporabnik” means “user”.

[18] http://source.ijs.si/mmartinc/4real2018/tree/master/datasets/karst_corpus.

*water—agresivna voda*, *eogenetic cave—eogenetska jama*, *precipitation—precipitacija*, *ponor cave—ponorna jama*, *epigenic aquifier—epigeni vodonosnik*, *karst polje—kraško polje*, *linear stream cave—linearna epifreatična jama*). However one half of correct term pairs remain undiscovered. We believe this is due to 1) domain-specific words which are not cognates and are missing from the Giza++ dictionary (e.g., *porous aquifer—medzrnski vodonosnik* and *denuded cave—brezstropa jama*), and 2) valid cognate words which do not meet the threshold described in Sect. 4.8 (e.g. *oxidization—oksidacija*, *percolation—perkolacija* and *liquefication—likvifakcija*).[19]

## 6 Replicability and reproducibility of our own terminology alignment results

As mentioned before, availability of the source code can drastically improve the reproducibility of experiments, since very detailed descriptions of procedures used in the experiments are beyond the scope of most papers because of length limitations and negative effects on the readability of the paper. Since we wanted to ensure the full reproducibility of our approach, we decided to publish the source code for all the conducted experiments and results that are published in the paper. As we were aware that just the presence of source code itself does not guarantee complete reproducibility, we decided that the published code should comply to the following three criteria:

- Instructions on how to use the code should be as unambiguous, simple and clear as possible.
- Code should be bug free and running it according to the instructions should yield the exact same results as published in the paper.
- Running the code should require as little time and technical skills as possible.

In order to validate that the published code complies to these criteria, we asked three students[20] to try to reproduce the results published in the paper (Repar et al. 2018) and after that answer the following questions related to the proposed criteria:

- Did you manage to reproduce the results?
- If not, what do you think was the main problem?
- If yes, how much time did you need for replicating the experiment?
- Were the instructions clear?
- Did you run into any specific problems during any part of the replicability attempt? If yes, please describe it.
- Do you have any suggestions on how to further improve the reproducibility of the results?

[19] It might also make sense to include morphological information as a feature of the machine learning algorithm, since all these word have endings typical of cognates in their respective languages.

[20] 2 Master students (one in Economy and one in Computer science) and 1 first year PhD student in ICT.

**Table 9** Manual evaluation results

| Domain | 1 | 2 | 3 | 4 |
|---|---|---|---|---|
| Reported in Aker et al. (2013) | | | | |
| IT, Ann. 1 | 0.81 | 0.06 | 0.06 | 0.07 |
| IT, Ann. 2 | 0.83 | 0.07 | 0.07 | 0.03 |
| Auto, Ann. 1 | 0.66 | 0.12 | 0.16 | 0.06 |
| Auto, Ann. 2 | 0.60 | 0.15 | 0.16 | 0.09 |
| Replication | | | | |
| Finance | 0.72 | 0.09 | 0.12 | 0.07 |
| IT | 0.79 | 0.01 | 0.09 | 0.12 |

Ann. stands for "Annotator" since the original paper uses two annotators

We also imposed a time limit of 8 hours (one working day) for the entire replicability attempt. If that limit was reached, the replicability attempt would count as unsuccessful.

The feedback we got was interesting and made us reconsider the initial source code criteria. Two out of three students managed to reproduce all the published results in less than an hour without any major problems. They did however point out some mistakes and ambiguities in the instructions on how to run the code. These were mostly connected with the programming environment used by the students, one of them using PyPI Python package manager for acquiring dependencies while the other one used the Conda environment, for which the usage instructions were not published.

The third student managed to reproduce the results in about two hours and reported some major problems with dependencies installation. He was the only person trying to reproduce the experiments in the Windows environment while the other two students used a Linux operating system, and he reported problems with the Python implementation of the Lemmagen lemmatizer (Juršič et al. 2010), which he was unable to install properly on the Windows platform. He managed to overcome the problem by manually removing the dependency from the code, by which he limited the flexibility of the published source code (he could only use it for the classification on the pre-generated train and test sets) but did not make the reproduction impossible.

While he was successful at reproducing the results for eight out of nine experiments published in the paper, he also reported a slight deviation (by less than 0.05 percentage point) from the reported recall and F-score in one of the experiments. Although we are not sure what is the exact reason for this deviation, we suspect it could be connected to the difference in operating systems.

These experiments show that programming environment and the choice of the operating system can have an unexpected negative impact on the reproducibility. While attaching code usage instructions for every possible programming environment and operating system is practically impossible, we do believe that the results of this experiment show that a published source code should comply to one additional criteria:

– Instructions should clearly specify on which operating system and in which programming environment the reported results were produced.

We have updated the usage instructions for our source code to comply with these criteria.

## 7 Reusability of our code in the ClowdFlows online platform

Because we want to make sure that our terminology alignment system is also available to a wider audience of users with lower level of technical skills (e.g., translators or linguists) and because we want to encourage a very simple reusability of our system, we have implemented the system into a cloud-based visual programming platform ClowdFlows (Kranjc et al. 2012). The ClowdFlows platform employs a visual programming paradigm in order to simplify the representation of complex data mining procedures into visual arrangements of their building blocks. Its graphical user interface is designed to enable the users to connect processing components (i.e. widgets) into executable pipelines (i.e. workflows) on a design canvas by a drag and drop technique, reducing the complexity of composition and execution of these workflows. The platform also enables online sharing of the composed workflows.

We took pretrained models of our terminology alignment system for English-Slovenian, English-French and English-Dutch alignment and packed them in a widget *Terminology alignment*, so it can be used out-of-the-box. The widget takes two columns of the Pandas dataframe (McKinney 2011) containing the source and target terms as inputs and returns a dataframe containing aligned term pairs. The user needs to define the names of the columns in the dataframe containing source and target language termlists, and the language of alignment as parameters. The user can also switch between configurations *Training set filtering 3* with the best precision and *Cognates approach* with the on average best F-score for all three languages while still having good precision by either enabling or disabling the *Maximize recall* widget parameter. Such an end to end system for bilingual terminology alignment in ClowdFlows is implemented at: http://clowdflows.org/workflow/13789/.[21] Another widget called *Terminology alignment evaluation* is used for determining the performance of the system (if we have a gold standard available), taking as input the dataframe produced by the *Terminology alignment* widget and a dataframe containing true alignments, and outputting the performance score in terms of precision, recall and F-score.

Workflow in Fig. 1 (available at http://clowdflows.org/workflow/13753/) is a ClowdFlows implementation for terminology alignment and evaluation. The source and target terminologies are both loaded from a CSV file with the help of the *Load Corpus From CSV* widget and fed as input to the *Terminology alignment* widget,

[21] Note that the execution time of term alignment increases rapidly with the increase in number of terms, e.g., alignment of hundred terms takes around five minutes, while it takes about one hour for alignment of thousand terms.

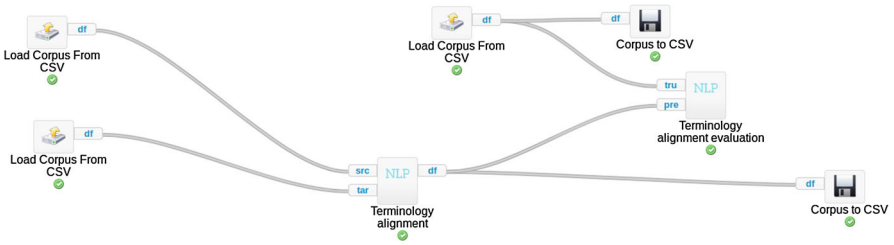


**Fig. 1** ClowdFlows implementation of the system for terminology alignment and evaluation available at http://clowdflows.org/workflow/13753/

which returns a dataframe with alignments. These are written to a CSV file with the *Corpus to CSV* widget and also fed to the *Terminology alignment evaluation* widget together with the dataframe containing true alignments (which was also loaded from a CSV file with the *Load Corpus From CSV* widget) in order to estimate the performance of the system. In addition, term alignment widget can also be incorporated into a bilingual terminology extraction workflow (Pollak et al. 2012). The workflow with the newly added term alignment widget, is available at http://clowdflows.org/workflow/13723/), where a user can now input text from a specific domain in Slovenian and English and get aligned terminology as output.

## 8 Conclusions and future work

Based on our research and attempts at replicating a bilingual terminology alignment paper reproducing its results, we propose a set of best practices any bilingual terminology extraction paper (and more generally every NLP paper) should fulfill to facilitate reproducibility and replicability of the experiments:

- *Dataset availability.* Availability of datasets (i.e. gold standard term lists, corpora) is an essential prerequisite for successful replication.
- *Experiment code availability.* The main task of reproducibility and replicability experiments is often to reconstruct the experiments in computer code. It is a cumbersome process which inevitably requires that the reproducer/replicator makes educated guesses at some point since a detailed description of the code is beyond the scope of most papers. Having the original code available greatly increases the ease of reproducibility and replicability experiments.
- *Tool availability.* Availability of a tool or application (online or offline) where experiments can be conducted eases reproducibility and replicability, but also enables the reusability of results by a larger community.
- Finally, releasing intermediate results, configuration settings and the actual outcomes of individual experiments, while not essential, would provide future researchers with an even greater possibility of successful reproduction of the paper's results.

A prerequisite for successful reproduction and replication is a clearly written research paper. However as is evident from our example, it is often difficult to include all necessary implementation notes given the length restrictions of the paper. For this reason, another best practice would be to provide relevant implementation examples alongside the code (which is what we did for feature construction.[22]) Finally, as the experiment in Sect. 6 showed, even code itself is sometimes not enough without additional implementation notes and information on the operating systems and software used. In addition, testing the code by non-authors is strongly recommended.

Our attempts focused on the approach to bilingual term alignment using machine learning by Aker et al. (2013). They approach term alignment as a bilingual classification task—for each term pair, they create various features based on word dictionaries (i.e. created with Giza++ from the DGT translation memory) and word similarities across languages. They evaluated their classifier on a held-out set of term pairs and additionally by manual evaluation. Their results on the held-out set were excellent, with 100% precision and 66% recall for the English-Slovenian and English-French language pair and 98% precision and 82% recall for English-Dutch.

Our reproduction attempt focused on three language pairs: English-Slovenian, English-Dutch and English-French (in contrast with the original article where they had altogether 20 language pairs) and we were unable to reproduce the results following the procedures described in the paper. In fact, our results have been dramatically different from the original paper with precision being less than 4% and recall close to 90% for all three language pairs under consideration. We then tested several different strategies for improving the results ranging from Giza++ dictionary cleaning, lemmatization, different ratios of positive and negative examples in the training and test sets, training set filtering based on feature values and term length, and adding new cognate-based features. The most effective strategies employed unbalanced training set and training set filtering based on certain feature values which resulted in precision exceeding 90% for all three language combinations (*Training set filtering 3* configuration, line 8 in Tables 3, 4 and 5). It is possible that in the original experiments authors performed a similar training set filtering strategy, because the original paper mentions that their classifier initially achieved low precision on Lithuanian language training set, which they were able to improve by manually removing positive term pairs that had the same characteristics as negative examples from the training set. However, no manual removal is mentioned for Slovenian, Dutch or French. Further attempts were directed at boosting recall and the performance of cognate-based features. By adding additional cognate-based features, we were able to improve recall by around 16% for Dutch, 8% for French and by around 2% for Slovenian (over the *Training set filtering 3* configuration) at a cost of a moderate drop in precision.

For evaluation we focused only on Slovenian, which is our native language and of primarily interest for our applied tasks. We performed manual evaluation similar to the original paper and reached roughly the same results with our adapted approach. In addition, because we discovered that Eurovoc data is of limited use for

[22] http://source.ijs.si/mmartinc/4real2018/blob/master/feature_examples.docx.

evaluating the performance of cognate-based features, we ran experiments on an English-Slovenian karstology gold standard term list. With the *Cognates approach* configuration (line 10 in Tables 3, 4 and 5), we improved recall by 11% (compared to the *Training set filtering 3* configuration) and a qualititive analysis of the results showed that the new strategies for boosting the performance of cognate-based features do indeed result in more cognate term pairs being properly aligned.

This paper demonstrates some of the obstacles for research reproducibility and replicability, with the prime one being code unavailability. Had we had access to the code of the original experiments, it is highly likely that replicating the original paper would be a trivial matter. Also in this particular case, the discrepancy in the results could be attributed to the scope of the original paper - with more than 20 languages—which is also a demonstration of very impressive approach—it would be impossible to describe procedures for all of them. We weren't able to reproduce the results of the original paper, but after developing the optimization approaches described above over the course of several months, we were able to reach a useful outcome at the end. We believe that providing supplementary material online, i.e. the code and datasets, is the only way of assuring complete reproducibility of results. For this reason, in order to help with any future reproducibility/replicability attempts of our paper, we are publishing the code at: http://source.ijs.si/mmartinc/4real2018.

In terms of future work, we plan to expand the feature set by introducing the features derived from the distributions in parallel corpora (e.g. co-frequency, logDice and other measures, see Baisa et al. (2015)), as well as investigate novel methods using cross-lingual embeddings. In terms of reproducibility, we plan to extend the study to a systematic comparison of different term alignment and term extraction methods.

**Acknowledgements** This paper is 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). The authors acknowledge also the financial support from the Slovenian Research Agency for research core funding for the programme Knowledge Technologies (No. P2-0103). The authors also acknowledge the project TermFrame—Terminology and Knowledge Frames across Languages (No. J6-9372), which was financially supported by the Slovenian Research Agency. We would also like to thank the company Iolar, for allowing us to use the data from the translation memories in one of the experiments.



## References

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: Vol. 1*. Long Papers (pp 402–411).

Aker, A., Paramita, M. L., Pinnis, M., & Gaizauskas, R. (2014). Bilingual dictionaries for all EU languages. In *Proceedings of 9th International Conference on Language Resources and Evaluation*. (pp 2839–2845).

Arčan, M., Turchi, M., Tonelli, S., & Buitelaar, P. (2014). Enhancing statistical machine translation with bilingual terminology in a CAT environment. https://doi.org/10.13140/2.1.1019.8404.

Bader, B. W., & Chew. P. A. (2008). Enhancing multilingual latent semantic analysis with term alignment information. In *Proceedings of the 22nd International Conference on Computational Linguistics: Vol. 1. Association for Computational Linguistics* (pp 49–56).

Baisa, V., Ulipová, B., & Cukr, M. (2015). Bilingual terminology extraction in Sketch Engine. In *9th Workshop on Recent Advances in Slavonic Natural Language Processing*. (pp 61–67).

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).

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: Vol. 2: Short Papers*. (pp 759–764).

Branco, A., Calzolari, N., & Choukri, K. (eds) (2018). 4REAL 2018—Workshop on Replicability and Reproducibility of Research Results in Science and Technology of Language, ELRA.

Branco, A., Cohen, K. B., Vossen, P., Ide, N., & Calzolari, N. (2017). Replicability and reproducibility of research results for human language technology: introducing an LRE special section.

Camerer, C. F., Dreber, A., Forsell, E., Ho, T. H., Huber, J., Johannesson, M., et al. (2016). Evaluating replicability of laboratory experiments in economics. *Science*, *351*(6280), 1433–1436.

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: Vol. 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: Vol. 2*. (pp 1–5).

Cohen, K. B., Xia, J., Zweigenbaum, P., Callahan, T., Goss, F., Ide, N., Névéol, A., Grouin, C., & Hunter, L. E. (2018). Three dimensions of reproducibility in natural language processing. In *Proceedings of the Eleventh International Conference on Language Resources and Evaluation (LREC 2018)*. (pp 156–165).

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: Vol. 1*. (pp 515–521).

Daille, B., & Morin, E. (2005). French-English terminology extraction from comparable corpora. *Natural Language Processing - IJCNLP*, *2005*, 707–718.

Fokkens, A., Van Erp, M., Postma, M., Pedersen, T., Vossen, P., & Freire, N. (2013). Offspring from reproduction problems: What replication failure teaches us. In *Proceedings of the 51st Annual Meeting of the Association for Computational Linguistics: Vol. 1: Long Papers*. (pp 1691–1701).

Foo, J. (2012). Computational terminology: Exploring bilingual and monolingual term extraction. PhD thesis, Linköping University Electronic Press.

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: Vol. 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).

Gale, W., & Church, K. (1993). A program for aligning sentences in bilingual corpora. *Computational Linguistics*, *19*(1), 75–102.

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: Vol. 1*. (pp 444–450).

Ha, L. A., Fern, G., Mitkov, R., Corpas, G. (2008). Mutual bilingual terminology extraction. In *Proceedings of the Sixth International Conference on Language Resources and Evaluation (LREC'08)*. (pp 1818–1824).

Haque, R., Penkale, S., & Way, A. (2014). Bilingual termbank creation via log-likelihood comparison and phrase-based statistical machine translation. In *Proceedings of the 4th International Workshop on Computational Terminology (Computerm)*. (pp 42–51).

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: Vol. 1: Long Papers*. (pp 685–693).

Ideue, M., Yamamoto, K., Utiyama, M., & Sumita., E. (2011). A comparison of unsupervised bilingual term extraction methods using phrase tables. In *Proceedings of the 13th Machine Translation Summit*. (pp 346–351).

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

Johnson, I., & Macphail, A. (2000). IATE–Inter-Agency Terminology Exchange: Development of a single central terminology database for the institutions and agencies of the European Union. In *Proceedings of the Workshop on Terminology resources and computation, LREC 2000 Conference*.

Juršič, M., Mozetič, I., Erjavec, T., & Lavrač, N. (2010). Lemmagen: Multilingual lemmatisation with induced ripple-down rules. *Journal of Universal Computer Science*, *16*(9), 1190–1214.

Kano, Y., Baumgartner, W. A, Jr., McCrohon, L., Ananiadou, S., Cohen, K. B., Hunter, L., et al. (2009). U-compare: Share and compare text mining tools with uima. *Bioinformatics*, *25*(15), 1997–1998.

Koehn, P. (2005). Europarl: A parallel corpus for statistical machine translation. In *Proceedings of the 10th Machine Translation Summit: Vol. 5*. (pp 79–86).

Kontonatsios, G., Korkontzelos, I., Tsujii, J., & Ananiadou, S. (2014). Combining string and context similarity for bilingual term alignment from comparable corpora. In *Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing (EMNLP)*. (pp 1701–1712).

Kranjc, J., Podpečan, V., & Lavrač, N. (2012). Clowdflows: A cloud based scientific workflow platform. In *Proceedings of Machine Learning and Knowledge Discovery in Databases, ECML/PKDD (2)*. Springer. (pp 816–819).

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).

Lee, L., Aw, A., Zhang, M., & Li, H. (2010). Em-based hybrid model for bilingual terminology extraction from comparable corpora. In *Proceedings of the 23rd International Conference on Computational Linguistics: Posters, Association for Computational Linguistics*. (pp 639–646).

Levenshtein, V. I. (1966). Binary codes capable of correcting deletions, insertions and reversals. *Soviet Physics Doklady*, *10*, 707.

Macken, L., Lefever, E., & Hoste, V. (2013). Texsis: Bilingual terminology extraction from parallel corpora using chunk-based alignment. *Terminology International Journal of Theoretical and Applied Issues in Specialized Communication*, *19*(1), 1–30.

McKinney, W. (2011). Pandas: A foundational Python library for data analysis and statistics. Python for High Performance and Scientific Computing. (pp 1–9).

Morin, E., Daille, B., Takeuchi, K., Kageura, K. (2007). Bilingual terminology mining-using brain, not brawn comparable corpora. In *Proceedings of the 45th Annual Meeting of the Association of Computational Linguistics*. (pp 664–671).

Morin, E., Daille, B., Takeuchi, K., & Kageura, K. (2008). Brains, not brawn: The use of smart comparable corpora in bilingual terminology mining. *ACM Transactions on Speech and Language Processing*, *7*(1), 1.

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

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

Pedersen, T. (2008). Empiricism is not a matter of faith. *Computational Linguistics*, *34*(3), 465–470.

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)*. (pp 20–21).

Piperidis, S., Papageorgiou, H., Spurk, C., Rehm, G., Choukri, K., Hamon, O., Calzolari, N., Del Gratta, R., Magnini, B., & Girardi, C. (2014). Meta-share: One year after. In *Proceedings of the 9th International Conference on Language Resources and Evaluation*. (pp 1532–1538).

Pollak, S., Vavpetič, A., Kranjc, J., Lavrač, N., & Vintar V,. (2012). NLP workflow for on-line definition extraction from English and Slovene text corpora. *11th Conference on Natural Language Processing, KONVENS 2012 - Empirical Methods in Natural Language Processing* (pp. 53–60). Vienna: Austria.

Prinz, F., Schlange, T., & Asadullah, K. (2011). Believe it or not: How much can we rely on published data on potential drug targets? *Nature Reviews Drug Discovery*, *10*(9), 712.

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. (2018). Machine learning approach to bilingual terminology alignment: Reimplementation and adaptation. In *Proceedings of the Eleventh International Conference on Language Resources and Evaluation (LREC 2018)*.

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: Chair) NCC, Choukri K, Declerck T, Dogan MU, Maegaard B, Mariani J, Odijk J, Piperidis S (eds) Proceedings of the Eight International Conference on Language Resources and Evaluation (LREC'12), European Language Resources Association (ELRA), Istanbul, Turkey.

Vintar, Š., & Grčić-Simeunović (2016). Definition frames as language-dependent models of knowledge transfer. Fachsprache : internationale Zeitschrift für Fachsprachenforschung, - didaktik und Terminologie. (pp 43–58).

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), 141–158.

Wieling, M., Rawee, J., & van Noord, G. (2018). Reproducibility in computational linguistics: Are we willing to share? *Computational Linguistics*, *44*(4), 641–649.

Yentis, S., Campbell, F., & Lerman, J. (1993). Publication of abstracts presented at anaesthesia meetings. *Canadian Journal of Anaesthesia*, *40*(7), 632–634.