TermEnsembler

An ensemble learning approach to bilingual term extraction and alignment

Andraž Repar, Vid Podpečan, Anže Vavpetič, Nada Lavrač & Senja Pollak
Jožef Stefan Institute

This paper describes TermEnsembler, a bilingual term extraction and alignment system utilizing a novel ensemble learning approach to bilingual term alignment. In the proposed system, the processing starts with monolingual term extraction from a language industry standard file type containing aligned English and Slovenian texts. The two separate term lists are then automatically aligned using an ensemble of seven bilingual alignment methods, which are first executed separately and then merged using the weights learned with an evolutionary algorithm. In the experiments, the weights were learned on one domain and tested on two other domains. When evaluated on the top 400 aligned term pairs, the precision of term alignment is over 96%, while the number of correctly aligned multi-word unit terms exceeds 30% when evaluated on the top 400 term pairs.

Keywords: bilingual terminology alignment, terminology extraction, ensemble learning, evolutionary algorithm

1. Introduction

With the onset of globalized markets, the need for effective multilingual communication has never been greater. Language industry, a term used to describe collectively the companies that offer translation and other related language services, has been steadily growing for several years and the increase in the volume of translated words brought along the need to streamline the translation process with automated solutions. In the 1990s, translation companies embraced computer-assisted translation (CAT) tools that allow them to store translations in a database and recycle them in future translation tasks.

https://doi.org/10.1075/term.00029.rep
Terminology 25:1 (2019), pp. 93–120. issn 0929-9971 | e-issn 1569-9994

Parallel to this process, another distinct (but related) development took place which revolved around terminology in the translation process. While several solutions and tools have been proposed, terminology remains one of the main problem areas for the translation industry. For example, a 2014 report[1] by SDL, a market leader in translation and terminology management software solutions, showed that among 140 companies, 51 percent of the respondents did not have a terminology management process in place, while a survey by Schmitz and Straub (2016) showed that among 800 respondents, 89.5 percent often or constantly experience that different organizational areas or employees use different terms for the same concept and that 51.9 percent of employees often or constantly cannot understand terms immediately. SDL Translation Technology Insights Series survey,[2] which focused on translation quality conducted among a mix of translation buyers, language service providers and freelance translators, found that "inconsistencies in the use of terminology" is the number one reason of translation rework (i.e. when translation is deemed not good enough and the source text has to be translated again) and recommended that, in order to improve translation quality, terminology management be prioritized.

Due to the early adoption of CAT tool technology in the translation industry, most translation companies have large repositories of translation memories. To illustrate, Gouadec (2007) reported that among more than 430 translation job advertisements surveyed, 95 percent contain a requirement for a "translation memory skill." In the period since that study, translation memories have remained a central component of any translation company business model.

This paper addresses the above-mentioned needs of the translation industry by proposing a system for semi-automated terminology extraction and alignment, currently focusing on English and Slovenian. The system, developed for one of the largest language service providers in Southeast Europe, consists of:

- A concept-oriented terminology database, where all the data is stored, allowing import from and export into industry-standard terminology management formats.
- A terminology extraction workflow, including automated extraction or import of manually defined monolingual terminology, followed by a novel approach to term alignment utilizing an evolutionary algorithm to combine the results of several individual bilingual term alignment methods.

---

1. SDL Research – Terminology: An End-to-End Perspective (http://www.sdl.com/download/terminology-an-endtoend-perspective/71114/). Accessed 3 March 2017.

2. Research Study 2016: Translation Technology Insights – Productivity (https://www.sdl.com/download/tti16-productivity/109572/). Accessed 3 March 2017.

– A web interface for managing the database and controlling the extraction and alignment algorithms.
– Additional functionalities for extraction of good example sentences and identification of the domain in which the term is used.

The novel approach to bilingual term alignment is the main contribution of this work. We systematically compare several existing term alignment methods, propose a novel Phrase-Table-Based Alignment (PTBA) method based on Pialign (Neubig et al. 2011), as well as a novel methodology using an evolutionary algorithm to combine solutions of an ensemble of elementary term alignment algorithms. We evaluate the performance of the system on three different domains, where one domain was used for training and two domains were used for testing the proposed approach.

This paper is structured as follows: Section 2 describes the related work, Section 3 describes the system and its methodology, Section 4 contains the experiments and results, while Section 5 contains the conclusions and plans for future work.

2. Related work

Terminology extraction refers to structuring terminological knowledge from unstructured text. Parallel translation databases (i.e. translation memories), which are omnipresent in the translation industry, lend themselves nicely to automated terminology extraction. In addition to terminology, various other types of information can be extracted, such as named entities, collocations or good examples.

In terms of input text, we can distinguish between monolingual terminology extraction, where terms are extracted from text in one language, and bilingual or multilingual terminology extraction, where the goal is to extract and align terms from text in two or more languages. A brief survey of related work is presented in Sections 2.1 and 2.2, respectively.

2.1 Monolingual term extraction

In the broadest sense, there are two different approaches to monolingual term extraction: linguistic and statistical. The linguistic approach utilizes the distinctive linguistic aspects of terms – most often their syntactic patterns, while the statistical approach takes advantage of term frequencies in the corpus. However, most state-of-the-art systems are hybrid, using a combination of the two approaches; e.g., Justeson and Katz (1995) first define part-of-speech patterns of terms and then use simple frequencies to filter the term candidates.

Many terminology extraction algorithms are based on the concepts of termhood and unithood defined by Kageura and Umino (1996). Termhood is "the degree to which a stable lexical unit is related to some domain-specific concepts" and unithood is "the degree of strength or stability of syntagmatic combinations and collocations." Termhood-based statistical measures function on a presumption that a term's relative frequency will be higher in domain-specific corpora than in the general language. Several approaches utilizing termhood have been developed, including those by Ahmad et al. (2000) and Vintar (2010). Common statistical measures are used to measure unithood, such as mutual information (Daille et al. 1994) or t-test (Wermter and Hahn 2005).

In the last few years, word embeddings – vectors of real numbers representing words on a corpus – have become a very popular natural language processing technique. The turning point was the paper by Mikolov et al. (2013) describing word2vec, a word embedding toolkit that can create vector space models much faster than previous attempts. Several attempts have already been made to utilize word embeddings for terminology extraction (e.g. Amjadian et al. (2016), Wang et al. (2016), Khan et al. (2016) and Zhang et al. (2018)).

2.2 Bilingual term extraction and alignment

At the highest level, bilingual terminology extraction can be divided into extraction from comparable and extraction from parallel corpora, where parallel corpora are composed of source texts and their translations in one or more different languages, while comparable corpora are composed of monolingual texts collected from different languages using similar sampling techniques (McEnery et al. 2006). For alignment of terms between the two languages, the methods typically utilize the idea that a term and its translation tend to occur in similar lexical contexts (Daille and Morin 2005).

In the language-industry context, taking into account parallel bilingual sentence pairs, stored in the translation memory, brings significant advantages to the task of terminology extraction. Broadly speaking, there are two distinct approaches to bilingual terminology extraction from parallel corpora according to Foo (2012):

- 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, and
- Extract-align, where we first extract monolingual candidate terms from both sides of the corpus and then align the terms.

A state-of-the-art align-extract approach is proposed by Macken et al. (2013) utilizing a chunk-based alignment method to produce a list of candidate term pairs, which are then filtered using statistical methods.

The extract-align approach is the more common of the two. Kupiec (1993) describes an algorithm for noun phrase extraction followed by alignment with a statistical estimation algorithm, achieving precision of 90 percent on the highest ranking candidate pairs. Vintar (2010) describes an extract-align approach named "bag-of-equivalents", where after monolingual extraction, the term pairs are aligned with the help of word alignment probabilities. Baisa et al. (2015) describe a frequency-based term alignment algorithm utilizing a variation of logDice to score the strength of the candidate term pair alignment. Haque et al. (2014) first generate candidate terms monolingually and then build a phrase table using the Moses toolkit (Koehn et al. 2007) and compare the extracted terms with the phrases in the table. Precision among the top 100 candidate term pairs often exceeds 90 percent. Aker et al. (2013) treat bilingual term alignment as a binary classification task, achieving good results. More recently, Hazem and Morin (2017) experiment with word embeddings used to augment bilingual terminology extraction from specialized comparable corpora (achieving precision of 70.9 percent).

The approach proposed in this paper is based on the idea of utilizing evolutionary algorithms which mimic biological evolution (i.e. reproduction, mutation, selection) to optimize the stated objective. Specifically, we use the genetic algorithm implementation in DEAP (*Distributed Evolutionary Algorithms in Python*) by Fortin et al. (2012) to build a term alignment ensemble.

3. TermEnsembler system and methodology

In this section, we describe the functionality of the developed TermEnsembler system, starting with the system overview and the background technologies used, and then focusing on bilingual term alignment as the main contribution of this paper.

3.1 System overview

The TermEnsembler system extracts bilingual terminology from English and Slovenian texts, and stores it into a concept-based terminology database, meaning that the entries are organized to correspond to a concept (cf. the general theory of terminology proposed by Wüster (1979)), but a concept might have more than one corresponding designator. It is a semi-automated system, meaning that the user can select several extraction parameters and manually curate the monolingual

extraction results for better bilingual alignment. While the system currently supports two languages (English and Slovenian), additional languages can be added by implementing appropriate language-specific background technologies similar to the ones described in this paper. In addition to the extraction of individual terms in each of the two languages (extracted using the approach described in Section 3.2), it also stores aligned term pairs (aligned using the approach described in Section 3.3). We have also developed a method for extracting good examples and domains, but as these are additional functionalities, we refer the reader to the previous papers by Repar and Pollak (2017a, 2017b).

The system relies on several background resources and technologies, used in different components of the system:

- *Preprocessing:* Texts are extracted from the translation memory (TMX) and preprocessed using the part-of-speech tagger and Wordnet lemmatizer from NLTK (Bird et al. 2009) for English and using the ReLDI tagger and lemmatizer (Ljubešić and Erjavec 2016) for Slovenian.
- *Monolingual term extraction:* Monolingual term extraction method LUIZ-CF by Pollak et al. (2012), extending LUIZ (Vintar 2010), is used as a basis for our upgraded LUIZ-CF++ term extraction approach.
- *Bilingual term alignment:* We use the Pialign phrase table extraction functionality (Neubig et al. 2011) as a basis for implementing three different bilingual term alignment approaches PTBA-1, PTBA-2 and PTBA-3 used in our experiments. In the reimplementation of bilingual LUIZ, we use Giza++ for word alignment (Och and Ney 2003). For weight assignment in our ensemble approach, we use the evolutionary computation framework DEAP (Distributed Evolutionary Algorithms in Python) by Fortin et al. (2012).

The overall structure of the system is shown in Figure 1. The starting point is a bilingual corpus in the standard translation memory format TMX, from which also available metadata, such as term domain or language variety can be extracted. The text is extracted and preprocessed resulting in a list of aligned lemmatized and POS-tagged sentence pairs. These pairs are sent into the additional metadata extraction (e.g., when domain information is not available in the TMX) and the monolingual extraction process, which results in two separate monolingual term lists (for TL1 and TL2). At this point, these two term lists can be curated by the user of the system. The (raw or curated) term lists are then taken as input to the bilingual alignment process (described in detail in Figure 2), which produces the final list of aligned term pairs. Finally, these term pairs are entered in the termbase alongside the metadata extracted in the step described above.

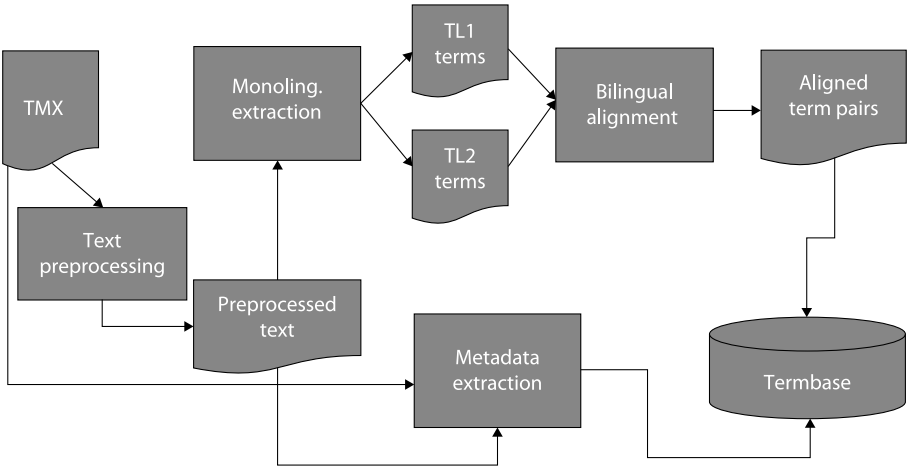

Figure 1. TermEnsembler: Methodology and components of the TermEnsembler system. Note that at several points human curation is possible (after monolingual extraction, after bilingual alignment or when accepting terms and metadata in the termbase. The monolingual step can also be skipped if the monolingual term lists are manually provided

3.2 Monolingual term extraction: LUIZ-CF++ upgrade of LUIZ-CF

The implemented monolingual term extraction approach LUIZ CF++ is based on the LUIZ hybrid approach by Vintar (2010) and refined with scoring and ranking functions implemented in LUIZ-CF by Pollak et al. (2012). The LUIZ approach is based on a list of part-of-speech patterns and a formula for comparison of term frequency between a domain corpus and a general language corpus (we used frequency lists from corpus Kres (Logar et al. 2012) for Slovenian and the British National Corpus (2007) for English).

In LUIZ-CF++, used in our experiments, we upgraded the LUIZ-CF monolingual term extraction approach by implementing the following additional functionalities:

- *Near-duplicates detection*: When importing the terms, the near duplicates (e.g. the orthography with or without spaces or hyphens, British and American English spellings) are detected and not created as new entries, but can be added as term variants of existing entries.
- *Nested term filtering*: According to Frantzi et al. (2000), nested terms are the terms that appear within other longer terms, and may or may not appear by themselves in the corpus. If the difference between a term and its nested term is below a certain threshold (which, in our case, can be defined by the user), only the longer term is returned. If not, both terms are included in the final output.

3.3 Bilingual term alignment: A novel ensemble learning approach

In this section, we describe the core part of TermEnsembler, i.e. the bilingual term alignment methodology implementing the *extract-align* approach explained in Section 2.2. Having implemented seven elementary term alignment approaches (3 existing, one modified, and 3 novel variants based on Pialign), this section introduces a novel ensemble-based approach combining the selected elementary term alignment approaches using an evolutionary algorithm.

We start by a brief outline of the proposed term alignment approach, illustrated in Figure 2. The input to the proposed TermEnsembler's bilingual term alignment methodology are two term lists (TL1 and TL2), which are automatically extracted using the monolingual extraction component (described in Section 3.2) or are human-defined. These two term lists are fed into seven individual bilingual term alignment algorithms that produce a total of 7 separate lists of aligned term pairs (*aligned term lists* or ATL), ranked by their alignment probability score as described in Section 3.3.1. The outputs of each alignment method are first normalized (separately) to the [0,1] interval, then fed into the evolutionary weights optimization algorithm described in Section 3.3.3 (which uses an external *ground truth list* (GTL) of manually annotated term pairs) to produce an optimal set of weights. These weights are then used to merge the seven ATLs into the final merged ATL using the procedure from Section 3.3.2.

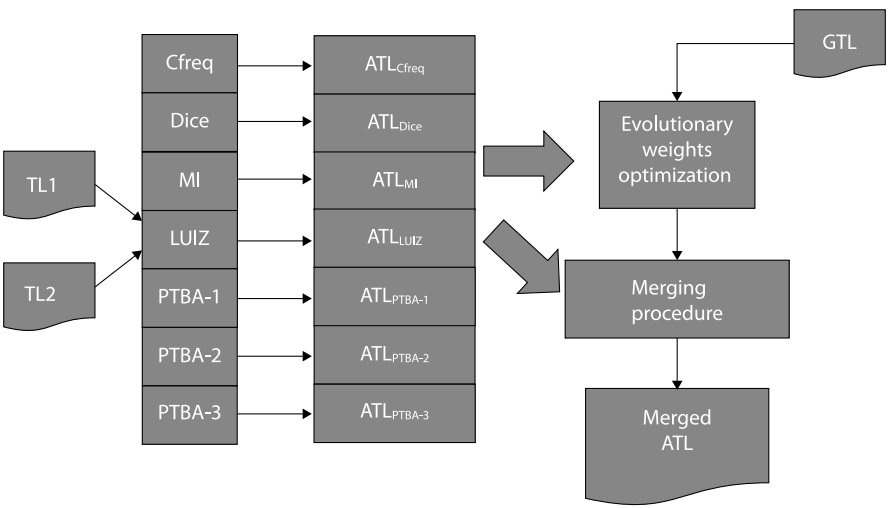

Figure 2. TermEnsembler's bilingual term alignment methodology

3.3.1 *Individual bilingual term alignment algorithms*

Each term alignment component described in this section produces a list of aligned term pairs ranked by their alignment scores, which are normalized between 0 and 1. The calculation of the scores is described below. The first four reimplemented approaches produce each one output (one aligned term list), while the last, novel approach, has three variants, leading to a total of seven output lists of aligned term pairs.

Co-frequency

Co-frequency *cofreq*(t_S, t_T) simply counts the number of sentences in which a term (t_S) from a source language S and a term (t_T) from target language T co-occur in the same sentence pair. The higher the co-occurrence count, the higher the probability that the terms are a correct term pair. This is the simplest of the used approaches and is completely language independent, but it does not take into account any language specifics. Because of that, it also requires a larger input corpus to produce sensible results.

Dice

This approach to bilingual terminology extraction is based on the Dice algorithm (Dice 1945). The co-frequency score from the previous component is used in the calculation of the Dice score, defined as follows:

$$dice(t_S,t_T) = 2\,\frac{cofreq(t_S,t_T)}{freq(t_S) + freq(t_T)}$$

where (t_S) and (t_T) are source and target terms, respectively. The *freq(t)* function stands for the frequency of term t in the entire corpus. A score based on Dice is used also in Sketch Engine (Baisa et al. 2015).

Mutual information

Similar as Dice, MI (Church and Hanks 1990) calculates term alignment by taking into account the co-frequency of source and target terms and the individual frequency of each term. It is defined as follows:

$$MI(t_S,t_T) = \log_2 \frac{cofreq(t_S,t_T)}{freq(t_S)\,freq(t_T)}$$

It usually contains the multiplication with N (in our case the number of candidate terms), but since in our case N is constant across terms, we can omit it if we just want to rank the terms.

BI-LUIZ+

We used a modified version of the bilingual component of the LUIZ approach, described by Vintar (2010). This approach takes as input two lists of term candidates (one for the source language and one for the target language) and word alignment pairs (with probabilities). The original paper uses the Twente aligner (Hiemstra 1998), while we used the GIZA++ (Och and Ney 2003).[3]

Using the alignments, the best matches (1 or more) are computed for each source term as follows: given a source term, we iterate through all target terms. For each target term we compute a score by summing the probabilities that a target token is a translation of a source token. Note that in the original paper by Vintar (2010) the equivalence score takes all single-word probabilities and divides them by the number of words, but dividing is not performed in our re-implementation as in the testing phase it produced worse results.[4] If the score is non-zero, we add the target term to the list of candidates.

Novel Phrase-Table-Based Alignment (PTBA) approaches PTBA-1, PTBA-2 and PTBA-3

The proposed PTBA approaches are novel bilingual term alignment approaches that we have developed based on Pialign (Neubig et al. 2011), an unsupervised model for joint phrase alignment and extraction using nonparametric Bayesian methods and inversion transduction grammars. Pialign follows a similar approach to phrase table generation in statistical machine translation (SMT) (Koehn et al. 2007), however, instead of first generating word alignments and then extracting a phrase table consistent with these alignments, it joins the phases of alignment and extraction by constructing a generative model that includes phrases at many levels of granularity, from single words to full sentences. Similar to Haque et al. (2014), the PTBA approach uses machine translation phrase tables for term alignment, but differs from it in several aspects described below.

The proposed PTBA approach takes as input a corpus and produces the list of aligned terms as output. Specifically, the Pialign alignments are read and used for mapping that stores for each English word all the computed Slovenian alignments along with the frequency of each alignment. As illustration, take the following example:

manager → upravitelj (20%), upravljavec (30%), upravljavec premoženja (50%)

The same mapping is also created for the reverse direction (Slovenian to English). For each aligned sentence pair found to contain some English and Slovenian terms, we compute the matching of all English terms from this sentence against

3. We had to use a different alignment method since the Twente aligner does not work anymore.

4. In communication with Vintar it has been confirmed that division has been later excluded from the formula.

all phrases from this sentence, and the best matching is retained. The matching is computed as the ratio of the most similar substring (i.e. if the phrase contains the entire term, the result is 100%). As a result, for each English phrase found in a sentence we record which terms found in this sentence are a part of this phrase. The matching procedure is repeated also for Slovenian. Finally, for each sentence we retain only the term-to-phrase mappings that exist in both directions. That is, we store a mapping if an English term from some sentence matches an English phrase from the same sentence and a Slovenian term from the aligned Slovenian sentence matches with the aligned Slovenian phrase.

As a side result of this term-to-phrase matching procedure, we propose the following procedure to obtain a list of direct candidates for aligned terms (i.e. we identify the phrase alignments consisting of a single term). The conditions are that the best term-to-phrase matching score is at least 95% for English and 90% (as the language is morphologically more varied) for Slovene and the difference in length of term string and phrase string is not greater than 4. An example, where a term matches the phrase with nearly no differences is a term *upravitelj* and the phrase *upravitelji*. As this is the only element of the phrase, we assume that the aligned phrase is the term's equivalent in English (e.g. *manager*).

The matching problem is addressed as follows: For each sentence, we have a list of phrases in English, their aligned counterparts in Slovenian, a list of terms for each English phrase and a list of terms for each Slovenian phrase. When computing the matching between English and Slovenian terms we also take into account the possibility that the terms can consist of several words.

We define the matching score of a multi-word English term to a multi-word Slovenian term as the sum of best single word alignment scores among all word combinations between the terms. Consider the following example:

English sentence	*The name of the share class Allianz….*
Slovenian sentence	*Ime razreda delnic Allianz…*
English phrase	*The name of the share class*
Slovenian phrase	*Ime razreda delnic*
English terms	*share, share class*
Slovenian terms	*delnica, razred delnic*

The matching algorithm computes the sum of all best word alignment scores. For example *score(share, delnica) + score(class, razred)* is the alignment score for terms *share class* and *razred delnic* (the word (mis)alignments *share-razred* and *class-delnica* have very low or possibly zero scores and are not added to the sum).

The matching scores are accumulated for all phrases and all sentences. In the end, we obtain the probability distributions for the translation of English terms into Slovenian and Slovenian terms into English. Using this information, we can produce three translation tables: *symmetric, English to Slovenian, and Slovenian to English*, respectively. The symmetric table consists of only those aligned terms

where the greedy probabilistic translation is the same in both directions. That is, a pair of English and Slovenian terms have each other listed as the most probable translation. The other two translation tables simply list the most likely translation in each direction. In this way, we have defined three different PTBA term alignment methods, resulting in three separate outputs of the PTBA term alignment method:

- *PTBA-1 Aligned Term list*, containing the results of the symmetric translation table.
- *PTBA-2 Aligned Term list,* containing the results of the English to Slovenian and Slovenian to English translation tables.
- *PTBA-3 Aligned Term list,* containing the list of direct alignment candidates produced as a side result of the term-to-phrase matching procedure.

3.3.2 *Final term pair ranking by ensemble-based weighting of separate lists of term pairs*

This section presents the key part of the developed methodology for ranking of aligned term pairs, i.e. the mechanism for assigning weights to separate lists of term pairs obtained by individual term alignment algorithms, and the merging mechanism using an ensemble weighting approach.

The ensemble score (Escore) is computed from two separate weighting scores:

- the algorithm weight (w), and
- the term pair score (score), normalized to [0,1].

A merging procedure for computing the final ensemble score Escore takes the individual term pair scores (score) from each of the seven elementary algorithms, together with weights for each approach provided by the user or assigned by automated means (i.e. the evolutionary algorithm approach explained below) and returns the final aligned term list, re-normalized on the [0,1] interval.

Merging procedure

1. For all **term pairs** (t_S, t_T) **compute** $Escore(t_S, t_T)$:

$$
\begin{aligned}
Escore(t_S, t_T) = {} & w_{cofreq}\, score_{cofreq}(t_S, t_T) + \\
& w_{dice} \cdot score_{dice}(t_S, t_T) + \\
& w_{mi} \cdot score_{mi}(t_S, t_T) + \\
& w_{mi} \cdot score_{mi}(t_S, t_T) + \\
& w_{luiz} \cdot score_{luiz}(t_S, t_T) + \\
& w_{PBA1} \cdot score_{PTBA\text{-}1}(t_S, t_T) + \\
& w_{PBA2} \cdot score_{PTBA\text{-}2}(t_S, t_T) + \\
& w_{PBA3} \cdot score_{PTBA\text{-}3}(t_S, t_T)
\end{aligned}
$$

2. Compute *Normalized Escore*(t_S, t_T)$\in$[0,1]
3. Rank *term pairs* (t_S, t_T) in decreasing order of their *Normalized Escore*(t_S, t_T)

3.3.3 *Evolutionary weighting of term alignment algorithms*

To be able to effectively search the large space of various weight values, we decided to use an evolutionary algorithm to find an optimal configuration. Specifically, we utilized the genetic algorithm (GA) implementation in DEAP (*Distributed Evolutionary Algorithms in Python*) by Fortin et al. (2012), an evolutionary computation framework, which can be used for rapid prototyping and testing of ideas and is designed to make algorithms explicit and data structures transparent. The GA algorithm starts with a random population and then applies crossover (producing new (children) members of the population from existing (parent) members) and mutation (randomly changing individual members – similar to biological mutation) operations for a successive number of generations. In each generation, the children are evaluated using a custom evaluation function and those that perform better than the parents are retained, while those that perform worse are discarded which eventually leads to an optimal result.

We start by generating a population of random sets of seven real numbers in the form of 7-tuples of weights of the 7 individual bilingual term alignment outputs:

$$(w_{cofreq}, w_{dice}, w_{mi}, w_{luiz}, w_{PBA1}, w_{PBA2}, w_{PBA3})$$

Each 7-tuple is used to generate a final bilingual term list (see Section 3.3.2) and is evaluated against a database of manually annotated term pairs provided in the training dataset. We used the parameters suggested in the DEAP documentation: number of generations: 100; population: 100; crossover probability: 0.5; mutation probability: 0.2.

We repeated the GA algorithm execution 20 times, and then calculated the average precision and standard deviation of the best performing 7-tuple of weights in each GA repetition. We selected the overall best performing 7-tuple learned on the training domain (training dataset) and tested its performance on two separate domains (test datasets). DEAP can be set up to optimize a single objective (i.e. precision among the Top 400 term pairs as in Section 4.4.1) or multiple objectives (i.e. precision among the Top 400 term pairs and number of correct *multi-word unit* (MWU) term pairs as in Section 4.4.2) at the same time.

4. Experiments and results

This section describes the experiments conducted to evaluate the TermEnsembler bilingual term alignment methodology and the datasets used in the experiments, followed by the results of the experiments and a qualitative analysis of errors.

4.1 Experimental setting

In these experiments, our goal was to find the best weight configuration for the 7 outputs produced by the individual term alignment components. To do so, we first evaluated the outputs individually in terms of overall precision and precision of MWU (*multi-word unit*) terms and then tried to find the best weight configuration using the evolutionary algorithm. We learned the best weight configuration on one domain (*Financial*) and then tested it on two others, non-related domains (*IT* and *Automotive*), by which we show that it is applicable to different domains.

The experimental setting was as follows. In creating the monolingual term lists as described in Section 3.2, we included only the terms that appear more than 10 times in the dataset.

– The evaluation criterion was the precision of term alignment, where the criterion for annotation was proper alignment, and not whether the individual English and Slovenian units are actually terms or not.

The latter requires further clarification.

– As bilingual term alignment is the main focus of this paper, we were primarily concerned with whether the terms are aligned properly (whether the terms are translation equivalents) and not whether the terms are true terms in each language.[5] For illustration, consider the following two examples:

  exchange rate – menjalni tečaj
  end of march – konec marca

  In the first example, both terms (English and Slovenian) are true terms according to the definition of a term from ISO 1087 ("verbal designation of a general concept in a specific subject field"), while the terms in the second example are much less likely to be considered terms in the sense of ISO 1087. However, for the purposes of evaluating the bilingual alignment algorithm both examples were considered correct.

5. An evaluation by a subject-matter expert reviewing the top 200 term pairs produced by the system showed that 74.5% of them are true terms.

– The evaluation was performed by a single annotator, which is the only realistic setting in a language-industry environment. Nevertheless, for inter-annotator evaluation, we acquired a second annotator to annotate a subset of the final output produced (and previously annotated by the main annotator) with the final weight configuration (see Section 4.4) on the Financial domain. The inter-annotator agreement was high, with both annotators agreeing in more than 95% of term pairs and Cohen's kappa (Cohen 1968) reaching 0.900. This denotes almost perfect agreement according to Landis and Koch (1977), and we can safely assume that annotations performed by a single annotator are highly accurate.

Note that in addition to measuring the precision of term alignment, we initially also considered measuring the recall, for which we would need a dataset containing manually annotated term pairs. However, measuring recall proved to be practically less relevant. The client arrived at the conclusion that in a production environment of a language service provider, the recall is not of particular importance, while it is much more important that term extraction output be precise, requiring no or minimal further processing or manual selection. As will be shown in Section 4.4, TermEnsembler produces a large number of correct term pairs, which satisfies the needs of the client. However, for the purpose of this article, we did evaluate the recall on a small gold standard term list in Section 4.4.3.

4.2 Data

In our experiments we used three distinct datasets, all coming from a production environment of a language service provider.

– Financial. This translation memory contains segments from a long-term translation project in the financial domain, specifically annual reports of investment funds and various related documentation. It has 18,197 segments (i.e. bilingual segment pairs) with 396,295 words in English and 354,862 words in Slovenian. The default configuration of the monolingual extractor returned 1,723 English and 1,953 Slovenian terms. This dataset was used to find the best weight configuration with the evolutionary algorithm.
– IT. This translation memory was used in a long-term software localization project. Most segments contain user interface strings and a smaller portion also contains user assistance (i.e. help articles) content. It has 40,599 segments (i.e. bilingual segment pairs) with 523,819 words in English and 473,430 words in Slovenian. The default configuration of the monolingual extractor returned 2,234 English and 2,477 Slovenian terms. This dataset was used to test the best weight configuration found with the evolutionary algorithm on the Financial dataset.

– Automotive. This translation memory was used in a long-term project for a customer from the automotive industry and contains segments from user manuals, internal service documentation and customer-facing promotional materials. It has 65,516 segments (i.e. bilingual segment pairs) with 861,665 words in English and 779,145 words in Slovenian. The default configuration of the monolingual extractor returned 3,122 English and 3,879 Slovenian terms. This dataset was used to test the best weight configuration found with the evolutionary algorithm on the Financial dataset.

Detailed statistics for each dataset, including the number of terms obtained by monolingual terminology extraction, are presented in Table 1.

Table 1. Detailed statistics of the three datasets used in the experiments

	Financial	IT	Automotive
Total segments	18,197	40,599	65,516
Total English words	396,295	523,819	861,665
Total Slovenian words	354,862	473,430	779,145
Unique English words	11,365	21,711	25,591
Unique Slovenian words	20,093	31,973	43,406
English terms	1,723	2,234	3,122
Slovenian terms	1,953	2,477	3,879

4.3 Experimental comparison of individual bilingual term alignment components

In this section, we systematically compare the performance of individual bilingual term alignment components from two aspects. First, we focus on the overall precision of the Top N term pairs produced by each component, and then we turn our attention to MWU (*multi-word unit*) term pairs found in the top N term pairs produced by the individual components.

4.3.1 *Precision of individual term alignment components*

Table 2 provides the results for precision for each method on the Financial dataset. We can observe that two PTBA methods have the highest precision, followed by another PTBA method and the three frequency-based components (Co-frequency, Dice and Mutual information), while BI-LUIZ+ has the lowest precision.

Table 2. Precision of individual bilingual alignment components on the Financial dataset on the Top 100, Top 200, Top 400 and Top 800 term pairs according to their (normalized) alignment score

	Total term pairs	Top 100		Top 200		Top 400		Top 800/ Total	
		Corr.	Prec.	Corr.	Prec.	Corr.	Prec.	Corr.	Prec.
Co-freq	1,492	60	0.600	111	0.555	175	0.438	292	0.366
Dice	1,492	57	0.570	128	0.640	272	0.680	511	0.693
MI	1,492	59	0.590	120	0.600	229	0.573	398	0.498
BI-LUIZ+	1,561	43	0.430	82	0.410	136	0.340	228	0.285
PTBA-1	591	93	0.930	183	0.915	350	0.875	**486**	**0.822**
PTBA-2	1,341	74	0.740	148	0.740	246	0.616	436	0.546
PTBA-3	674	**98**	**0.980**	**193**	**0.965**	**360**	**0.900**	523	0.777

Note that since the total number of term pairs of PTBA-3 and PTBA-1 is lower than 800, the last column denotes precision on the total number of pairs (i.e. Top 674 and Top 591, respectively).

4.3.2 *Single vs. multi-word unit terms*

While precision is the most important performance indicator of a bilingual term alignment algorithm, we also wanted to have more details on the ratio between single and multi-word terms in the outputs, because the client communicated that having translations of multi-words terms is much more useful than just simple one-word units. Since we are looking at bilingual term pairs, we consider a pair to be a single-word unit if both terms (English and Slovenian) are single-word units, and multi-word if at least one of the terms is a multi-word unit (MWU). For illustration, see the three examples below:

issuance – izdaja *SINGLE-WORD UNIT*
registrar – agent za registracijo *MULTI-WORD UNIT*
stock market – borzni trg *MULTI-WORD UNIT*

Specifically, we looked at how many of the top N terms produced by individual components are correct MWU term pairs. This decision was again reached in communication with the client who wanted to have the ability to request a specific number (N) of term pairs to be returned by TermEnsembler and our goal was to make the returned term pairs as good as possible, both in terms of overall precision and in the number of correct MWU terms.

In Table 3, we can observe that the Dice algorithm produces the most correct term pairs in all 4 scenarios, closely followed by MI. BI-LUIZ+ produces a lot of multi-word terms but its precision (calculated as correct MWU terms divided by all MWU terms in the top N term pairs) is relatively low, while the PTBA methods

produce few MWU term pairs in the Top 100 pairs, but improve in this respect in Top 200, Top 400 and Top 800 scenarios.

Table 3. Total number of MWU term pairs (and their precision) in top N terms, correct MWU term pairs on the Financial dataset

	Top 100		Top 200		Top 400		Top 800/Total	
	Cor/tot	**Prec**	**Cor/tot**	**Prec**	**Cor/tot**	**Prec**	**Cor/tot**	**Prec**
Co-freq	2/21	0.420	7/49	0.143	17/128	0.133	49/383	0.128
Dice	**52**/94	0.553	**106**/175	0.606	**198**/320	0.619	**358**/589	0.608
MI	50/87	0.575	102/178	0.573	187/351	0.533	295/678	0.435
BI-LUIZ+	43/100	0.430	82/200	0.410	103/363	0.284	136/680	0.200
PTBA-1	20/24	0.833	51/61	0.836	133/170	0.782	199/273	0.729
PTBA-2	15/38	0.395	39/85	0.459	90/234	0.385	194/527	0.368
PTBA-3	14/14	**1.000**	54/57	**0.947**	130/146	**0.890**	218/278	**0.784**

Note that since the total number of term pairs of PTBA-3 and PTBA-1 is lower than 800, the last column denotes precision on the total number of pairs (i.e. Top 674 and Top 591, respectively).

4.4 Results of the TermEnsembler's bilingual term alignment approach

The key question in our system is how to determine the optimal configuration of weights for the merging script described in Section 3.3. Table 2 and Table 3 above clearly show that some of the methods are much more effective than the others. Similar to the reasoning in Section 4.3, we want to test two distinct scenarios:

- In the first one, we want to find the best overall precision.
- In the second one, we want to find the best compromise between the overall precision and the number of correct multi-word units.

We decided to focus the evaluation of the weight configuration on the top 400 term pairs, because the client believes that 400 terms are enough to produce a useful terminological resource in a standard translation project. In other words, we try to optimize the configuration to return the best results on the top 400 term pairs. Also, the starting point for comparison is the result of the PTBA-3 component that has an overall precision of 0.900 and returns 130 correct multi-word unit term pairs (see Table 2). This means that any weight configuration would need to improve on these results.

As evident from Table 4, assigning the same weight to all components does not yield results superior to the PTBA-3 component. The same is true if we assign weights according to their individual precision (calculated in Table 2) relative to the lowest value (i.e. the weight of BI-LUIZ+ is 1.0 and the rest are calculated

proportionally). This is why we decided to use the DEAP evolutionary algorithm described in Section 3.3 for weight configuration.

4.4.1 *Optimizing for optimal precision*

In the first experiment, we wanted to construct a weight configuration that would result in the highest possible precision, which means that we minimize the number of incorrect pairs. We performed 20 repetitions of the evolutionary algorithm execution. The average precision of the best performing 7-tuples of weights in each of the 20 repetitions was 0.949 with a standard deviation of 0.009. The overall best precision of 0.960 was achieved by three different weight configurations (see Table 5),[6] showing that the evolutionary algorithm exceeds the results of PTBA-3 by 6% (see Table 4).

Table 4. Results of the various weight configurations on the Financial domain

	Top 400
PTBA-3	0.900
Equal weights	0.725
Precision weights	0.732
Evolutionary algorithm	**0.960**

To test whether this configuration can be applied universally, we used it to evaluate precision on two additional domains: *Automotive* and *IT*. To do so, we tested all three configurations from Table 5 and calculated the average overall precision. As can be observed from Table 6, the weight configuration produced by the evo-

6. The calculated weights show that the PTBA-3 component is always the most significant one, followed by PTBA-1, and next Cofreq followed by all other methods (which can in some cases even have negative weights). Several factors that contribute to the actual magnitude of weights have to be taken into account when interpreting the results. First, the weights are computed using different heuristics. Second, the components produce results of different lengths and those returning a small number of mostly correct results are likely to obtain a higher weight. Next, the evolutionary algorithm will try to adjust the weights in such way that segments of high ranked correct results will make it to the final list. If the same or similar segment of correct results appears at the bottom of the list of another component, its promotion to the final list is likely to be too costly as this would also promote several incorrect results. For example, the reason for the negative weights in some of the repetitions in Table 5 is that the scores assigned by a particular component (i.e. PTBA-2) are too high compared to other components. This is confirmed by the results of the manual evaluation of individual components in Table 2 where we can observe that PTBA-2 has a significantly lower precision than PTBA-1 or PTBA-3. The weights of the remaining 4 components are significantly lower, close to 0, with the highest one of them being Cofreq.

Table 5. The best performing weight configurations when optimizing overall precision using an evolutionary algorithm

Rep #	Cofreq	Dice	MI	Luiz	PTBA-1	PTBA-2	PTBA-3
3	0.619	0.196	0.010	0.053	4.481	−2.867	11.046
8	0.327	0.086	0.008	0.022	1.564	0.137	5.494
10	0.561	0.106	−0.017	0.104	2.177	−0.758	10.268

lutionary algorithm returns good results on unseen data (*IT* and *Automotive*) as well, with precision on unseen data actually exceeding the precision on the training data (i.e. *Financial* domain).

Table 6. Precision of the weight configuration produced by the evolutionary algorithm on the Financial domain and applied to the Automotive and IT domain. The results were obtained as an average precision of the three weight configurations shown in Table 5

	Top 400
Financial	0.960±0.000
Automotive	0.984±0.001
IT	0.984±0.001

4.4.2 *Optimizing for a compromise between optimal precision and number of correct multi-word unit term pairs*

In the next step, we modified the evolutionary algorithm to optimize the configuration for the highest precision and the largest number of multi-word units at the same time. While the equal weight configuration and the weight configuration based on individual precision values produce a higher number of MWUs, they also introduce a fair amount of noise resulting in lower precision. As is evident from Table 7, the configuration produced by the evolutionary algorithm has the highest precision while maintaining a decent amount of MWUs (a high number of which are also correct – MWU precision of 0.919). The results closest to this configuration are returned by the PTBA-3 component, but the number of MWUs is significantly lower.

These results were achieved by running 20 repetitions of the evolutionary algorithm and selecting the best weight configuration based on the following criterion: the best configuration has the highest number of correct MWUs and must have an overall precision greater than the best individual component (in our case, PTBA-3). The best weight configuration was thus produced in repetition 19 and had the weights shown in Table 8.

Table 7. Overall precision, total number of MWUs, number of correct MWUs and precision of MWUs of the configuration produced by the evolutionary algorithm compared to various other configurations, measured on the Financial domain

	Precision	Total MWUs	Correct MWUs	MWU precision
PTBA-3	0.900	146	130	**0.890**
Equal weights	0.725	311	205	0.659
Precision weights	0.733	**312**	**208**	0.667
Evolutionary algorithm	**0.955**	185	170	0.919

Table 8. The best performing weight configuration when optimizing for a compromise between optimal precision and number of correct multi-word unit term pairs

Rep #	Cofreq	Dice	MI	Luiz	PTBA-1	PTBA-2	PTBA-3
19	0.219	0.229	0.009	0.116	2.855	−4.739	11.470

Once again, we tested whether the configuration produced by the evolutionary algorithm can be used universally by applying it to two additional domains: Automotive and IT. The results can be found in Table 9.

Table 9. Top 400 results of the weight configuration produced by the evolutionary algorithm on the Financial domain and applied to the Automotive and IT domain

	Precision	Total MWUs	Correct MWUs	MWU precision
Financial	0.955	**185**	**170**	0.919
Automotive	**0.990**	153	151	**0.987**
IT	0.985	130	126	0.969

In both domains, the results are similar to what we observed in the *Financial* domain. In fact, the results are even better in the two new domains with overall precision in the Top 400 term pair candidates exceeding 98%, and the MWU precision above 96%. The actual ratio of correct MWU terms among the Top 400 terms is 38% on the *Automotive* domain and 32% on the *IT* domain. We decided to use this configuration as the final configuration in the client's production environment.

4.4.3 *Recall of the TermEnsembler system*

Due to the client's preference, the majority of our experiments were focused on precision, but we did also evaluate recall on a corpus subsample, where a gold standard termlist of 88 financial terms was produced by manual expert annotation. With the final weight configuration (used in the production environment) the recall of the TermEnsembler system was 60%.

4.5 Qualitative analysis of errors

To better understand the types of errors that the system makes, for each of the three domains we have performed a qualitative analysis of the first 50 incorrect term pairs[7] among the list of 800 top ranked term pairs suggested by the system, using the final weight configuration suggested by the evolutionary algorithm. We observed that most of the errors are due to discrepancies between the English and Slovenian monolingual extraction process, rather than due to the incorrect alignment procedure, and that many incorrect term pairs can be considered "partially correct". We illustrate several examples of incorrect alignments below, starting with minor errors followed by some more severe cases of misaligned terms.

In some of the highly ranked term pairs, one part of a term in one language is missing because the term was incorrectly extracted, which results in partially correct term pair, such as (the word in brackets was not extracted):

Financial: *interest (rate) – obrestna mera*
Automotive: *(quick) repair kit – komplet za hitro popravilo*
IT: *missing (value) – manjkajoča vrednost*

A particularly difficult issue for the system are product names. Because they may not follow standard language rules regarding the construction of terms, they are difficult to detect without a pre-defined product name list or a well performing named entity recognition system. Consequently, many of the incorrectly extracted named entities contain parts of product names. The Financial dataset in particular has a high number of named entities, which is a reason for lower results compared to the other two corpora. Such examples include:

Equity – delnica[8]
BNP Paribas – Paribas
Flexible Bond Strategy – Bond Strategy

In a limited number of cases, the monolingual terms and the alignment itself are correct, but the resulting term pair is not correct. In the two examples from the Automotive dataset, the source text uses *miles per gallon* to denote gas mileage, but the Slovenian translation (due to the preferences of the customer) uses *kilometers per 100 liters*. A similar case can be observed with units denoting weight.

Mile – km
Lb – kg

---

7. The positions of the 50th incorrect term pair for all three domains: 518 for Financial, 756 for Automotive, and 661 for IT.

8. Note that "equity" can appear either as a common noun (i.e. equity = assets) or as a part of a proper noun (e.g., Global Equity Climate Change).

In a smaller number of cases close to the bottom of the list of extracted term pairs, the alignment is completely off and the meaning of the source term is not the same as the meaning of the target term (which can be explained by the frequent co-occurrence of the terms in the text), for example:

Financial: *gross national income – svetovna banka*
Automotive: *similar heavy object – pritrjen nosilec koles*
IT: *folder number – znesek kredita*

Finally, we compared the ratio between the two major error types in the three domains (see Table 10). In the *Financial* and *Automotive* domains, the majority of the incorrect terms can be ascribed to the category "Partially correct", which are predominantly errors arising from incorrect monolingual extraction (but could also be related to incorrect translation or wrong alignment of the two terms). Because the monolingual term is missing a word or several words or contains redundant words, the resulting term pair was not classified as correct. However, the alignment is not completely wrong nor completely useless, because the term can be quickly corrected in a semi-automated terminology setting.

Table 10. A comparison of the two major error type among the 50 analysed incorrect term pairs

	Financial	**Automotive**	**IT**
Different meaning	38%	12%	**56%**
Partially correct	**62%**	**88%**	44%

5. Conclusions and future work

This paper describes TermEnsembler, a terminology extraction and alignment system, created from the point of view of language service providers in the language and translation industry. It consists of a concept-oriented terminology database with industry-standard file format support for easy sharing with other terminological applications, an online user interface for database management and semi-automatic term extraction, a monolingual terminology extraction algorithm (currently supporting English and Slovenian) and a novel bilingual alignment methodology with several components.

The first step is monolingual extraction based on the work of Vintar (2010) and Pollak et al. (2012) with some additional modifications, such as a filter for nested terms and near-duplicate recognition. The final result of this step are two lists of terms (one for each language) with the terms ordered by their termhood score. The next step, which is the central part of the paper, involves bilingual

alignment of the terms in the two lists. We have implemented and evaluated a total of seven methods – implementing approaches from the related work and the newly proposed approaches – which all return a list of aligned English-Slovenian term pairs. The evaluation of each approach separately shows that the highest precision was obtained by the newly developed phrase-table-based term alignment approach PTBA-3 which directly matches the extracted terms with phrases from the phrase table.

For final implementation, we experimented with different merging methods for the 7 outputs by assigning weights to produce a final list of term pairs. After initial experiments with equal weight and precision-based weights, we opted for an ensemble optimization approach using the genetic algorithm implementation from the evolutionary algorithm framework DEAP by Fortin et al. (2012), which takes random weight configurations and tries to optimize them towards a certain goal over a successive number of generations.

We have trained the bilingual alignment approach in TermEnsembler on one domain and tested it on two different domains achieving excellent results, with more than 96% of the top 400 term pair alignments produced by the system evaluated as correct by a human evaluator. In addition, we have also tried to optimize the system for producing a greater number of multi-word terms because they are particularly complicated for translation. When optimizing the evolutionary algorithm for overall precision and number of correct multi-word terms, at least a third of the top 400 term pair alignments returned by our system were correct multi-word terms, with precision computed on the MWUs reaching 0.919. All in all, we believe the high precision of our system among the top 400 terms would require only minor manual human curation to produce a viable term list for day-to-day work in the language industry.

We also briefly looked into whether bilingual term alignment improves the quality of monolingual terms. An experienced translator compared the top 200 terms returned by the initial algorithm (the LUIZ-CF variant described in Pollak et al. (2012)) for each of the two languages in all three domains and compared them with the top 200 terms produced by TermEnsembler after bilingual term alignment. The results show that TermEnsembler does improve the monolingual quality of terms (precision) by around 10%.

In terms of future work, we have identified several lines of research. We will continue adding new languages, implementing and systematically evaluating different monolingual term-extraction approaches. For bilingual alignment, we will initially focus on a systematic optimization of the evolutionary algorithm parameters and then look into implementing user-friendly parameters that would allow the users to tweak the weights towards greater overall precision or larger number of MWU terms. We will also test other, potentially faster optimization methods such

as differential evolution and Newton-like methods as well as develop machine-learning solutions for term alignment, combining the proposed statistical scores and cognate-based features, as in Aker et al. (2013). Finally, given a recent trend of well performing word-embeddings methods leading to excellent results in various natural-language processing tasks, we aim to address bilingual term-extraction as a well-suited task for developing cross-lingual embedding based term alignment methods, stimulated by the work of Conneau et al. (2018).

Acknowledgements

The system's interface and the elementary term extraction approaches were designed and developed in the scope of the TermIolar project by the Jožef Stefan Institute and Iolar d.o.o. The authors acknowledge the contribution of Simon Bratina and Davorin Sečnik (of Iolar d.o.o.) to functional specifications, additional requirements, evaluation of the interim results and providing important feedback and suggestions. The authors thank also Špela Vintar for her clarifications in the reimplementation of bilingual LUIZ term alignment.

The authors acknowledge the financial support of Slovenian Research agency for funding part of this research in the scope of basic research program Knowledge Technologies (Grant No. P2-0103) and the project Terminology and Knowledge Frames across Languages (Grant No. J6-9372). 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 results of this paper reflect only the author's view and the Commission is not responsible for any use that may be made of the information it contains.

References

Ahmad, Khurshid, Lee Gillam, and Lena Tostevin. 2000. "Weirdness Indexing for Logical Document Extrapolation and Retrieval (WILDER)." In *Proceedings of the 8th Text Retrieval Conference (TREC-8)*, 717–724. Washington, USA.

Aker, Ahmet, Monica Paramita, and Rob Gaizauskas. 2013. "Extracting Bilingual Terminologies from Comparable Corpora." In *Proceedings of the 51st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, 402–411. Sofia, Bulgaria.

Amjadian, Ehsan, Diana Inkpen, Tahereh Paribakht, and Farahnaz Faez. 2016. "Local-Global Vectors to Improve Unigram Terminology Extraction." In *Proceedings of the 5th International Workshop on Computational Terminology*, 2–11. Osaka, Japan.

Baisa, Vít, Barbora Ulipová, and Michal Cukr. 2015. "Bilingual Terminology Extraction in Sketch Engine." In *9th Workshop on Recent Advances in Slavonic Natural Language Processing, RASLAN 2015 – Proceedings*, 61–67. Karlova Studánka, Czech Republic.

Bird, Steven, Ewan Klein, and Edward Loper. 2009. *Natural Language Processing with Python: Analyzing Text with the Natural Language Toolkit*. Sebastopol: O'Reilly Media Inc.

Church, Kenneth Ward, and Patrick Hanks. 1990. "Word Association Norms, Mutual Information, and Lexicography." *Computational Linguistics* 16 (1): 22–29.

Cohen, Jacob. 1968. "Weighted Kappa: Nominal Scale Agreement Provision for Scaled Disagreement or Partial Credit." *Psychological Bulletin* 70 (4): 213. https://doi.org/10.1037/h0026256

Conneau, Alexis, Guillaume Lample, Marc'Aurelio Ranzato, Ludovic Denoyer, and Hervé Jégou. 2018. "Word Translation Without Parallel Data." (https://arxiv.org/abs/1710.04087) Accessed 2 February 2019.

Daille, Béatrice, and Emmanuel Morin. 2005. "French-English Terminology Extraction from Comparable Corpora." In *Proceedings of the 2nd International Joint Conference on Natural Language Processing*, 707–718. Jeju Island, South Korea.

Daille, Béatrice, Éric Gaussier, and Jean-Marc Langé. 1994. "Towards Automatic Extraction of Monolingual and Bilingual Terminology." In *Proceedings of the 15th Conference on Computational linguistics*, 515–521. Kyoto, Japan. https://doi.org/10.3115/991886.991975

Dice, LR. 1945. "Measures of the Amount of Ecologic Association between Species." *Ecology* 26 (3): 297–302. https://doi.org/10.2307/1932409

Foo, Jody. 2012. *Computational Terminology: Exploring Bilingual and Monolingual Term Extraction*. Linköping: Linköping University Electronic Press.

Fortin, Félix-Antoine, François-Michel De Rainville, Marc-André Gardner, Marc Parizeau, and Christian Gagné. 2012. "DEAP: Evolutionary Algorithms Made Easy." *Journal of Machine Learning Research* 13 (no. Jul): 2171–2175.

Frantzi, Katerina, Sophia Ananiadou, and Hideki Mirna. 2000. "Automatic Recognition of Multi-Word Terms:. the C-Value/NC-Value Method." *International Journal on Digital Libraries* 3(2): 115–130. https://doi.org/10.1007/s007999900023

Gouadec, Daniel. 2007. *Translation as a Profession*. Amsterdam/Philadephia: John Benjamins. https://doi.org/10.1075/btl.73

Haque, Rejwanul, Sergio Penkale, and Andy Way. 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)*, 42–51. Dublin, Ireland. https://doi.org/10.3115/v1/W14-4806

Hazem, Amir, and Emmanuel Morin. 2017. "Bilingual Word Embeddings for Bilingual Terminology Extraction from Specialized Comparable Corpora." In *Proceedings of the 8th International Joint Conference on Natural Language Processing*, 685–693. Taipei, Taiwan.

Hiemstra, Djoerd. 1998. "Multilingual Domain Modeling in Twenty-One: Automatic Creation of a Bi-Directional Translation Lexicon from a Parallel Corpus." In *Proceedings of the 8th CLIN Meeting*, 41–58. Amsterdam, The Netherlands.

Justeson, John, and Slava Katz. 1995. "Technical Terminology: some Linguistic Properties and an Algorithm for Identification in Text." *Natural Language Engineering* 1 (1): 9–27. https://doi.org/10.1017/S1351324900000048

Kageura, Kyo, and Bin Umino. 1996. "Methods of Automatic Term Recognition: A Review." *Terminology* 3 (2): 259–289. https://doi.org/10.1075/term.3.2.03kag

Khan, Muhammad Tahir, Yukun Ma, and Jung-jae Kim. 2016. "Term Ranker: A Graph-Based Re-Ranking Approach." In *Proceedings of the 29th International Florida Artificial Intelligence Research Society Conference*, 310–315. Key Largo, USA.

Koehn, Philipp, Hieu Hoang, Alexandra Birch, Chris Callison-Burch, Marcello Federico, Nicola Bertoldi, Brooke Cowan et al. 2007. "Moses: Open Source Toolkit for Statistical Machine Translation." In *Proceedings of the 45th Annual Meeting of the ACL on Interactive Poster and Demonstration Sessions*, 177–180. Prague, Czech Republic. https://doi.org/10.3115/1557769.1557821

Kupiec, Julian. 1993. "An Algorithm for Finding Noun Phrase Correspondences in Bilingual Corpora." In *Proceedings of the 31st Annual Meeting on Association for Computational Linguistics*, 17–22. Columbus, USA. https://doi.org/10.3115/981574.981577

Landis, Richard, and Gary Koch. 1977. "The Measurement of Observer Agreement for Categorical Data." *Biometrics* 33 (1): 159–174. https://doi.org/10.2307/2529310

Ljubešić, Nikola, and Tomaž Erjavec. 2016. "Corpus vs. Lexicon Supervision in Morphosyntactic Tagging: the Case of Slovene." In *Proceedings of the 10th International Conference on Language Resources and Evaluation (LREC 2016)*, 23–28. Portorož, Slovenia.

Logar, Nataša, Miha Grčar, Marko Brakus, Tomaž Erjavec, Špela Arhar Holdt, and Simon Krek. 2012. *Korpusi slovenskega jezika Gigafida, KRES, ccGigafida in ccKRES: gradnja, vsebina, uporaba [Slovenian language corpora Gigafida, KRES, ccGigafida, ccKRES: creation, content, use]*. Ljubljana: Trojina, zavod za uporabno slovenistiko; Fakulteta za družbene vede.

Macken, Lieve, Els Lefever, and Veronique Hoste. 2013. "Texsis: Bilingual Terminology Extraction from Parallel Corpora using Chunk-Based Alignment." *Terminology* 19 (1): 1–30. https://doi.org/10.1075/term.19.1.01mac

McEnery, Tony, Richard Xiao, and Yukio Tono. 2006. *Corpus-Based Language Studies: An Advanced Resource Book*. London: Taylor & Francis.

Mikolov, Tomas, Kai Chen, Greg Corrado, and Jeffrey Dean. 2013. "Efficient Estimation of Word Representations in Vector Space." (https://arxiv.org/abs/1301.3781) Accessed 10 July 2018.

Neubig, Graham, Taro Watanabe, Eiichiro Sumita, Shinsuke Mori, and Tatsuya Kawahara. 2011. "An Unsupervised Model for Joint Phrase Alignment and Extraction." In *Proceedings of the 49th Annual Meeting of the Association for Computational Linguistics*, 632–641. Portland, USA.

Och, Franz Josef, and Hermann Ney. 2003. "A Systematic Comparison of Various Statistical Alignment Models." *Computational Linguistics* 29 (1): 19–51. https://doi.org/10.1162/089120103321337421

Pollak, Senja, Anže Vavpetič, Janez Kranjc, Nada Lavrač, and Špela Vintar. 2012. "NLP Workflow for On-Line Definition Extraction from English and Slovene Text Corpora." In *Proceedings of KONVENS 2012*, 53–60. Vienna, Austria.

Repar, Andraž, and Senja Pollak. 2017a. "Good Examples for Terminology Databases in Translation." In *Electronic Lexicography in the 21st century. Proceedings of eLex 2017 Conference*, 651–661. Leiden, Netherlands.

Repar, Andraž, and Senja Pollak. 2017b. "Ontology-Based Translation Memory Maintenance." In *Proceedings of the 20th International Multiconference Information Society 2017*, 19–22. Ljubljana, Slovenia.

Schmitz, Klaus Dirk, and Daniela Straub. 2016. "Tight Budgets and a Growing Number of Languages Impede Terminology Work." *tcworld magazine for international information management* (http://www.tcworld.info/e-magazine/technical-communication/article/tight-budgets-and-a-growing-number-of-languages-impede-terminology-work/). Accessed 24 August 2018.

The British National Corpus, version 3 (BNC XML Edition). 2007. *Distributed by Bodleian Libraries*, University of Oxford, on behalf of the BNC Consortium. (URL: http://www.natcorp.ox.ac.uk/). Accessed 10 March 2017.

Vintar, Špela. 2010. "Bilingual Term Recognition Revisited. The Bag-of-Equivalents Term Alignment Approach." *Terminology* 16 (2): 141–158. https://doi.org/10.1075/term.16.2.01vin

Wang, Rui, Wei Liu, and Chris McDonald. 2016. "Featureless Domain-Specific Term Extraction with Minimal Labelled Data." In *Proceedings of the Australasian Language Technology Association Workshop*, 103–112. Melbourne, Australia.

Wermter, Joachim, and Udo Hahn. 2005. "Paradigmatic Modifiability Statistics for the Extraction of Complex Multi-Word Terms." In *Proceedings of the Conference on Human Language Technology and Empirical Methods in Natural Language Processing*, 843–850. Vancouver, Canada.

Wüster, Eugene. 1979. *Introduction to the General Theory of Terminology and Terminological Lexicography*. Vienna: Springer.

Zhang, Zigi, Jie Gao, and Fabio Ciravegna. 2018. "SemRe-Rank: Incorporating Semantic Relatedness to Improve Automatic Term Extraction Using Personalized PageRank." (https://arxiv.org/abs/1711.03373) Accessed 7 January 2019.

Address for correspondence

Andraž Repar
International Postgraduate School Jožef Stefan
Jožef Stefan Institute
Jamova 39, Ljubljana
Slovenia
repar.andraz@gmail.com

Co-author information

Vid Podpečan
Jožef Stefan Institute
vid.podpecan@ijs.si

Anže Vavpetič
Jožef Stefan Institute
hi@anzevavpetic.com

Nada Lavrač
Jožef Stefan Institute
nada.lavrac@ijs.si

Senja Pollak
Jožef Stefan Institute
senja.pollak@ijs.si