Identification of Bilingual Terms from Monolingual Documents for Statistical Machine Translation

Mihael Arcan[1] **Claudio Giuliano**[2] **Marco Turchi**[2] **Paul Buitelaar**[1]

[1] Unit for Natural Language Processing, Insight @ NUI Galway, Ireland
{mihael.arcan , paul.buitelaar}@insight-centre.org
[2] FBK - Fondazione Bruno Kessler, Via Sommarive 18, 38123 Trento, Italy
{giuliano, turchi}@fbk.eu

Abstract

The automatic translation of domain-specific documents is often a hard task for generic Statistical Machine Translation (SMT) systems, which are not able to correctly translate the large number of terms encountered in the text. In this paper, we address the problems of automatic identification of bilingual terminology using Wikipedia as a lexical resource, and its integration into an SMT system. The correct translation equivalent of the disambiguated term identified in the monolingual text is obtained by taking advantage of the multilingual versions of Wikipedia. This approach is compared to the bilingual terminology provided by the Terminology as a Service (TaaS) platform. The small amount of high quality domain-specific terms is passed to the SMT system using the XML markup and the Fill-Up model methods, which produced a relative translation improvement up to 13% BLEU score points

1 Introduction

Translation tasks often need to deal with domain-specific terms in technical documents, which require specific lexical knowledge of the domain. Nowadays, SMT systems are suitable to translate very frequent expressions but fail in translating domain-specific terms. This mostly depends on a lack of domain-specific parallel data from which the SMT systems can learn. Translation tools such as Google Translate or open source phrase-based SMT systems, trained on generic data, are the most common solutions and they are often used to translate manuals or very specific texts, resulting in unsatisfactory translations.

This problem is particular relevant for professional translators that work with documents coming from different domains and are supported by generic SMT systems. A valuable solution to help them in handling domain-specific terms is represented by online terminology resources, e.g. IATE - Inter-Active Terminology for Europe,[1] which are continuously updated and can be easily queried. However, the manual use of these services can be very time demanding. For this reason, the identification and embedding of domain-specific terms in an SMT system is a crucial step towards increasing translator productivity and translation quality in highly specific domains.

In this paper, we propose an approach to automatically detect monolingual domain-specific terms from a source language document and identify their equivalents using Wikipedia cross-lingual links. For this purpose we extend The Wiki Machine API,[2] a tool for linking terms in text to Wikipedia pages, adding two more components able to first identify domain-specific terms, and to find their translations in a target language. The identified bilingual terms are then compared with those obtained by TaaS (Skadinš et al., 2013). The embedding of the domain-specific terms into an SMT system is performed by use of the XML markup approach, which uses the terms as preferred translation candidates at run time, and the Fill-Up model (Bisazza et al., 2011), which emphasizes phrase pairs extracted from the bilingual terms.

Our results show that the performance of our technique and TaaS are comparable in the identification of monolingual and bilingual domain-specific terms. From the machine translation point of view, our experiments highlight the benefit of integrating bilingual terms into the SMT system, and the relative improvement in BLEU score of the Fill-Up model over the baseline and the XML markup approach.

[1] http://iate.europa.eu/ [2] https://bitbucket.org/fbk/thewikimachine/

2 Methodology

Given a source document, it is processed by our pipeline that: (*i*) with the help of The Wiki Machine, it identifies, disambiguates and links all terms in the document to the Wikipedia pages; (*ii*) the terms and their links are used to identify the domain of the document and filter out the terms that are not domain-specific; (*iii*) the translation of such terms is obtained following the Wikipedia cross-lingual links; (*iv*) the bilingual domain-specific terms are embedded into the SMT system using different strategies. In the rest of this section, each step is described in detail.

2.1 Bilingual Term Identification

Term Detection and Linking The Wiki Machine is a tool for linking terms in text to Wikipedia pages and enriching them with information extracted from Wikipedia and Linked Open Data (LOD) resources such as DBPedia or Freebase. The Wiki Machine has been preferred among other approaches because it achieves the best performance in term disambiguation and linking (Mendes et al., 2011), and facilitates the extraction of structured information from Wikipedia.

The annotation process consists of a three-step pipeline based on statistical and machine learning methods that exclusively uses Wikipedia to train the models. No linguistic processing, such as stemming, morphology analysis, POS tagging, or parsing, is performed. This choice facilitates the portability of the system as the only requirement is the existence of a Wikipedia version with a sufficient coverage for the specific language and domain. The first step identifies and ranks the terms by relevance using a simple statistical approach based on *tf-idf* weighting, where all the n-grams, for n from 1 to 10, are generated and the *idf* is directly calculated on Wikipedia pages. The second step links the terms to Wikipedia pages. The linking problem is cast as a supervised word sense disambiguation problem, in which the terms must be disambiguated using Wikipedia to provide the sense inventory and the training data (for each sense, a list of phrases where the term appears) as first introduced in (Mihalcea, 2007). The application uses an ensemble of word-expert classifiers that are implemented using the kernel-based approach (Giuliano et al., 2009). Specifically, domain and syntagmatic aspects of sense distinction are modelled by means of a combination of the latent semantic and string kernels (Shawe-Taylor and Cristianini, 2004). The third step enriches the linked terms using information extracted from Wikipedia and LOD resources. The additional information relative to the pair term/Wikipedia page consists of alternative terms (i.e., orthographical and morphological variants, synonyms, and related terms), images, topic, type, cross language links, etc. For example, in the text "click right mouse key to pop up menu and Gnome panel", The Wiki Machine identifies the terms *mouse*, *key*, *pop up menu* and *Gnome panel*. For the ambiguous term *mouse*, the linking algorithm returns the Wikipedia page 'Mouse_(computing)', and the other terms used to link that page in Wikipedia with their frequency, i.e., *computer mouse*, *mice*, and *Mouse*.

In the context of the experiments reported here, we were specifically interested in the identification of domain-specific bilingual terminology to be embedded into the SMT system. For this reason, we extend The Wiki Machine adding the functionality of filtering out terms that do not belong to the document domain, and of automatically retrieving term translations.

Domain Detection To identify specific terms, we assign a domain to each linked term in a text, after that we obtain the most frequent domain and filter out the terms that are out of scope. In the example above, the term *mouse* is accepted because it belongs to the domain *computer_science*, as the majority of terms (*mouse*, *pop up menu* and *Gnome panel*), while the term *key* in the domain *music* is rejected.

The large number of languages and domains to cover prevents us from using standard text classification techniques to categorize the document. For this reason, we implemented an approach based on the mapping of the Wikipedia categories into the WordNet domains (Bentivogli et al., 2004). The Wikipedia categories are created and assigned by different human editors, and are therefore less rigorous, coherent and consistent than usual ontologies. In addition, the Wikipedia's category hierarchy forms a cyclic graph (Zesch and Gurevych, 2007) that limits its usability. Instead, the WordNet domains are organized in a hierarchy that contains only 164 items with a degree of granularity that makes them suitable for Natural Language Processing tasks. The approach we are proposing overcomes the Wikipedia category sparsity, allows us reducing the number of domains to few tens instead of some hundred thousands (800,000 categories in the English Wikipedia) and does not require any language-specific training data. Wikipedia categories that contain more pages (∼1,000) have been manually mapped to WordNet domains. The domain for a term is obtained as follows. First, for each term, we extract its set of categories, C, from the Wikipedia page linked to it. Second, by means of a recursive procedure, all possible outgoing paths (usually in a large number) from each category in C are followed in the graph of Wikipedia categories. When one of the mapped categories to a WordNet domain is found, the approach stops and associates the relative WordNet domain to the term. In this way, more and more domains are assigned to a single term. Third, to isolate the most relevant one, these domains are ranked according the number of times they have been found following all the paths. The most frequent domain is assigned to the terms. Although this process needs the human intervention for the manual mapping, it is done once and it is less demanding than annotating large amounts of training documents for text classification, because it does not require the reading of the document for topic identification.

Bilingual Term Extraction The last phase consists in finding the translation of the domain terminology. We exploit the Wikipedia cross-language links, which, however, provide an alignment at page level not at term level. To deal with this issue we introduced the following procedure. If the term is equal to the source page title (ignoring case) we return the target page; otherwise, we return the most frequent alternative form of the term in the target language. From the previous example, the system is able to return the Italian page *Mouse* and all terms used in the Italian Wikipedia to express this concept of *Mouse* in *computer_science*. Using this information, the term *mouse* is paired with its translation into Italian.

2.2 Integration of Bilingual Terms into SMT

A straightforward approach for adding bilingual terms to the SMT system consists of concatenating the training data and the terms. Although it has been shown to perform better than more complex techniques (Bouamor et al., 2012), it is still affected by major disadvantages that limits its use in real applications. In particular, when small amounts of bilingual terms are concatenated with a large training dataset, terms with ambiguous translations are penalised, because the most frequent and general translations often receive the highest probability, which drives the SMT system to ignore specific translations.

In this paper, we focus on two techniques that give more priority to specific translations than generic ones: the Fill-Up model and the XML markup approach. The Fill-Up model has been developed to address a common scenario where a large generic background model exists, and only a small quantity of in-domain data can be used to build an in-domain model. Its goal is to leverage the large coverage of the background model, while preserving the domain-specific knowledge coming from the in-domain data. Given the generic and the in-domain phrase tables, they are merged. For those phrase pairs that appear in both tables, only one instance is reported in the Fill-Up model with the largest probabilities according to the tables. To keep track of a phrase pair's provenance, a binary feature that penalises if the phrase pair comes from the background table is added. The same strategy is used for reordering tables. In our experiments, we use the bilingual terms identified from the source data as in-domain data. Word alignments are computed on the concatenation of the data. Phrase extraction and scoring are carried out separately on each corpus. The XML markup approach makes it possible to directly pass external knowledge to the decoder, specifying translations for particular spans of the source sentence. In our scenario, the source term is used to identify a span in the source sentence, while the target term is directly passed to the decoder. With the setting *exclusive*, the decoder uses only the specified translations ignoring other possible translations in the translation model.

3 Experimental Setting

In our experiments, we used different English-Italian and Italian-English test sets from two domains: (*i*) a small subset of the GNOME project data[3] (4,3K tokens) and KDE4 Data[4] (9,5K) for the IT domain and (*ii*) a subset of the EMEA corpus (11K) for the medical domain.

In order to assess the quality of the monolingual and bilingual terms, we create a terminological gold standard. Two annotators with a linguistic background and English and Italian proficiency were asked

[3] https://l10n.gnome.org/ [4] http://i18n.kde.org/

to mark all domain-specific terms in a set of 66 English and Italian documents of the GNOME corpus, and a set of 100 paragraphs (4,3K tokens) from the KDE4 corpus.[5] Domain-specificity was defined as all (multi-)words that are typically used in the IT domain and that may have different Italian translations in other domains. The average Cohen's Kappa of GNOME and KDE_anno computed at token level was 0.66 for English and 0.53 for Italian. Following Landis and Koch (1977), this corresponds to a substantial and moderate agreement between the annotators.

Finally the gold standard dataset was generated by the intersection of the annotations of the two annotators. In detail, for the GNOME dataset the annotators marked 93 single-word and 134 multi-word expressions (MWEs), resulting 227 terms in overall. For the KDE_anno dataset, 321 monolingual terms for the GNOME dataset were annotated, whereby 192 of them were multi-word expressions. This results in 190 unique bilingual terms for the GNOME corpus and 355 for the KDE_anno dataset.

We compare the monolingual and bilingual terms identified by our approach to the terms obtained by the online service TaaS,[6] which is a cloud-based platform for terminology services based on the state-of-the-art terminology extraction and bilingual terminology alignment methods. TaaS provides several options in term identification, of which we selected TWSC, Tilde wrapper system for CollTerm, (Pinnis et al., 2012). TWSC is based on linguistic analysis, i.e. part of speech tagging and morpho-syntactic patterns, enriched with statistical features. TaaS allows for lookup in several manually and automatically built monolingual and bilingual terminological resources and for our experiment we use EuroTermBank (ETB), Taus Data and Web Data. Accessing several resources, TaaS may provide several translations for a unique source term, but not an indicator of their translation quality. To avoid assigning the same probability to all the translations of the same source term, we prioritise a translation by the resource it was provided. In our case, we favour first the translation provided by ETB. If no translation is available, we use the translation provided by Taus Data or eventually from Web Data. Before starting the term extraction approach, TaaS requires manual specification of the source and target languages, the domain, and the source document. Since we focused on the IT and medical domains we set the options to 'Information and communication technology' and 'Medicine and pharmacy', respectively.

For each translation task, we use the statistical translation toolkit Moses (Koehn et al., 2007), where the word alignments were built with the GIZA++ toolkit (Och and Ney, 2003). The IRSTLM toolkit (Federico et al., 2008) was used to build the 5-gram language model. For a broader domain coverage, we merged parts of the following parallel resources: JRC-Acquis (Steinberger et al., 2006), Europarl (Koehn, 2005) and OpenSubtitles2013 (Tiedemann, 2012), this results in a generic training corpus of $\sim$37M tokens and a development set of $\sim$10K tokens.

In our experiments, an instance of Moses trained on the generic parallel dataset was used in three different scenarios: (*i*) as baseline SMT system without embedded terminology; (*ii*) in the XML markup approach for translating remaining parts that were not covered by the embedded terminology; (*iii*) in the Fill-Up method as background translation model.

4 Evaluation

In this Section, we report the performance of the different term identification tools and term embedding methods for the two domains: IT and the medical domain. For evaluating the extracted monolingual and bilingual terms, we calculate precision, recall and f-measure using the manually labelled KDE_anno and GNOME datasets. In addition, we perform a manual inspection of a subset of the bilingual identified terms. The BLEU metric (Papineni et al., 2002) was used to automatically evaluate the quality of the translations. The metric calculates the overlap of n-grams between the SMT system output and a reference translation, provided by a professional translator.

4.1 Monolingual Term Identification

In Table 1, the column 'Ident.' represents the number of identified terms for each tool, whereby we observed TaaS always extracts more terms than The Wiki Machine. While extracting Italian terms, TaaS extracts twice as more terms as The Wiki Machine, which can be explained by the overall lower

[5] In the rest of the paper, we refer to the annotated part of KDE4 as KDE_anno

[6] https://demo.taas-project.eu/

	English						Italian					
KDE_anno	Ident.	unigram	MWE	Precision	Recall	F1	Ident.	unigram	MWE	Precision	Recall	F1
TaaS	431	144	287	0.442	0.594	0.507	518	147	371	0.326	0.511	0.398
The Wiki Machine	327	247	80	0.400	0.406	0.403	207	184	23	0.429	0.268	0.330
GNOME	Ident.	unigram	MWE	Precision	Recall	F1	Ident.	unigram	MWE	Precision	Recall	F1
TaaS	311	119	192	0.260	0.355	0.301	359	110	249	0.272	0.415	0.329
The Wiki Machine	275	199	76	0.303	0.364	0.330	196	167	29	0.331	0.275	0.301

Table 1: Evaluation of monolingual term identification for the KDE_anno and GNOME dataset.

amount of Italian pages in Wikipedia compared to the English version. Focusing on the amount of identified single-word and multi-word expressions, it is interesting to notice that TaaS, independently of the language, extracts around twice as more MWEs than single words. Differently, The Wiki Machine identifies mostly single-word terms, whereby they represent around three-fourth of all identified terms for English and around 12% for Italian.

For the KDE_anno dataset, TaaS in most cases (except in precision for the Italian KDE_anno dataset) outperforms The Wiki Machine approach in all metrics. Especially we observed a higher recall produced by the TaaS approach, which can be deduced from the higher number of extracted MWEs compared to The Wiki Machine approach. On the English GNOME dataset, The Wiki Machine performs comparable results to TaaS, with a slightly higher recall and F1. On the Italian side, The Wiki Machine identifies less MWEs than TaaS, which results in a low recall and F1.

In summary, we observe that TaaS performs best on the KDE_anno dataset, whereas The Wiki Machine and TaaS perform comparable results on the GNOME dataset. Analysing the overall results, we notice that precision, recall and F1 are generally better in English than in Italian. This is due to the fact that Italian tends to use more words to express the same concept compared to English.

4.2 Bilingual Term Identification

Table 2 reports the performance of The Wiki Machine and TaaS in the identification of bilingual terms evaluated against the manually produced list of terms. In both language pairs and datasets, TaaS and The Wiki Machine mostly identify similar amounts of bilingual terms (column 'Ident.') and match with the gold standard (column 'Mat.'). Only for KDE_anno, It→En, TaaS identifies almost 50% more bilingual terms than The Wiki Machine.

It is worth noticing that, although TaaS is accessing high quality manually-produced termbases, e.g. ETB in our results, there is no evidence that it works significantly better than The Wiki Machine accessing Wikipedia. In fact, in terms of F1, The Wiki Machine performs best on the GNOME annotated test set, while it is outperformed by TaaS on KDE_anno. In both cases, differences in performance are minimal. According to the precision measure, The Wiki Machine seems to be able to produce more accurate bilingual terms.

The automatic evaluation shows difficulties (low F1 scores) for The Wiki Machine and TaaS in identifying bilingual terms that perfectly match the gold standard. To better understand the quality of term translations, we asked one of the annotators involved in the creation of the gold standard to perform a manual evaluation of a subset of fifty bilingual terms randomly selected from each list. We used the four error categories proposed in (Aker et al., 2013): 1) The terms are exact translations of each other in the domain; 2) Inclusion: Not an exact translation, but an exact translation of one term is entirely contained within the term in the other language; 3) Overlap: Not category 1 or 2, but the terms share at least one translated word; 4) Unrelated: No word in either term is a translation of a word in the other. The percentages of bilingual terms assigned to each class are shown in Table 3.

In terms of comparison between the two tools, the manual evaluation confirms that there is no evidence that a tool produces better term translations than the other in all the test sets. In fact, except for KDE_anno En→It where TaaS outperforms The Wiki Machine, the percentage of bilingual terms assigned to class 1 for both the tools is almost similar. In terms of absolute scores, the manual evaluation shows that the quality of the identified bilingual terms is relatively high (merging the terms assigned to classes 1

GNOME En→It	Ident.	Mat.	Precision	Recall	F1
TaaS	145	20	0.138	0.105	0.119
The Wiki Machine	156	25	0.160	0.130	0.144
GNOME It→En	Ident.	Mat.	Precision	Recall	F1
TaaS	139	21	0.151	0.110	0.127
The Wiki Machine	140	23	0.164	0.121	0.139
KDE_anno En→It	Ident.	Mat.	Precision	Recall	F1
TaaS	249	65	0.261	0.183	0.215
The Wiki Machine	229	49	0.202	0.138	0.164
KDE_anno It→En	Ident.	Mat.	Precision	Recall	F1
TaaS	228	58	0.254	0.163	0.199
The Wiki Machine	155	48	0.292	0.135	0.185

Table 2: Automatic evaluation of bilingual terms extracted from GNOME and KDE_anno.

GNOME En→It	1	2	3	4
TaaS	0.66	0.08	0.00	0.26
The Wiki Machine	0.70	0.08	0.06	0.16
GNOME It→En	1	2	3	4
TaaS	0.78	0.08	0.02	0.12
The Wiki Machine	0.68	0.12	0.04	0.16
KDE_anno En→It	1	2	3	4
TaaS	0.90	0.00	0.06	0.04
The Wiki Machine	0.70	0.10	0.06	0.14
KDE_anno It→En	1	2	3	4
TaaS	0.70	0.10	0.10	0.10
The Wiki Machine	0.64	0.22	0.08	0.06

Table 3: Manual evaluation of bilingual terms based on four error categories (1-4).

and 2, we reach a score, in most of the cases, larger than 80%). This is in contrast with the automatic evaluation, which reports limited performances (F1 $\sim$ 0.2) for both methods. The main reason is that the automatic evaluation requires a perfect match between the identified and the gold standard bilingual terms to measure an improvement in F1, while the manual evaluation can reward bilingual terms that do not perfectly match any gold standard terms but are correct translations of each other. An example is the multi-word bilingual term "settings of the network connection → impostazioni della connessione di rete" that is present in the gold standard as a single multi-word term, while it is identified by The Wiki Machine as two distinct bilingual terms, i.e. "network connection → connessione di rete" and "settings → impostazioni". From the translation point of view, both the distinct terms are correct and they are assigned to class 1 during the manual evaluation, but they are ignored by the automatic evaluation.

The analysis of terms assigned to error class four shows that both methods are affected by similar problems. The main source of error is the correct detection of the source term domain, which results in a translated term that does not belong to the correct domain. For instance, in the bilingual term "stringhe → shoe and boot laces", the term "stringhe" ("strings" in the IT domain) is translated into "laces". Similarly, the English term "launchers" ("lanciatori" in Italian in the IT domain) is translated into "lanciarazzi multiplo" ("multiple rocket launchers" in English), which is clearly not an IT term. Furthermore, The Wiki Machine seems to have more problems in identifying the right morphological variation, e.g. "indirizzi ip → ip address", where "indirizzi" is a plural noun and needs to be translated into "addresses". This is expected because page titles in Wikipedia are not always inflected. An interesting example highlighted by the annotator in the TaaS translations is: "percorso di ricerca" → "how do i access refresh grid texture?", where the Italian term ("search path" in English) is translated with a completely wrong translation. In the next Section we evaluate whether the automatic identified bilingual terms can improve the performance of an SMT system and if it is robust to the aforementioned errors.

4.3 Embedding Terminology into SMT

Our further experiments focused on the automatic evaluation of the translation quality of the EMEA, GNOME and KDE test sets (Table 4). The obtained bilingual terminology from TaaS and The Wiki Machine was embedded through the Fill-Up and XML markup approaches. The approximate randomization approach in MultEval (Clark et al., 2011) is used to test whether differences among system performances are statistically significant with a p-value < 0.05. The parameters of the baseline method and the Fill-Up models were optimized on the development set.

Injecting the obtained TaaS bilingual terms improves the BLEU score in several cases. XML markup outperforms the general baseline approach in three (out of eight) datasets, whereby three of them are statistically significant (GNOME En→It, KDE_anno En↔It). Embedding the same bilingual terminology into the Fill-Up model helped to outperform the baseline approach for all test sets, whereby only the result for EMEA En→It is not statistically significant.

	GNOME		KDE_anno		EMEA		KDE4	
	En→It	It→En	En→It	It→En	En→It	It→En	En→It	It→En
general baseline	15.39	21.62	15.58	22.64	25.88	25.75	19.22	23.54
XML Mark-up (TaaS)	15.87	22.45*	**17.62***	**23.88***	25.84	25.74	18.97	24.27*
Fill-Up Model (TaaS)	**16.22***	**22.73***	17.61*	23.45*	25.95	26.02*	**19.69***	**24.56***
XML Mark-up (The Wiki Machine)	15.49	20.57	17.19*	23.44*	25.59	24.97	17.74	22.16
Fill-Up Model (The Wiki Machine)	15.82	21.70	16.48*	23.28*	**26.35***	**26.44***	19.61*	24.14*

Table 4: Automatic BLEU Evaluation on GNOME, KDE and EMEA datasets with different term embedding strategies (bold results = best performance ; * statistically significant compared to baseline).

Finally, we investigate the impact of embedding the identified terms provided by The Wiki Machine. When we suggest translation candidates with the XML markup, it only slightly outperforms the baseline approach for GNOME En→It, but statistically significant improves the translations for the KDE_anno test set for both language directions. Similarly to previous observations, the Fill-Up model improves further the translations, i.e. the translations are statistically significant better than the baseline for both language pairs of both KDE test sets as well as for EMEA.

To better understand our translation results, we manually inspected the EMEA En→It sentences, which have the best translation performance. For each of the source sentence and the translation method, we analyse the translated sentences and the bilingual terms that match at least one word in the source sentence. Both translation strategies tried to encapsulate the bilingual terms, but there is clear evidence that the Fill-Up model better embeds the target terms in the context of the translation. For instance in the following example, the target sentence produced by the XML markup (XML trg) does not contain the article "la", uses a wrong conjunction ("di" instead of "per") and wrongly orders the adjective with the noun ("adulti pazienti" instead of "pazienti adulti"). All these issues are correctly addressed by the Fill-Up model (Fill-Up trg) which produces a smoother translation.

source sentence: adult patients receive therapy for tumours
reference sentence: pazienti adulti ricevono la terapia per i tumori
bilingual terms: therapy → terapia, patients → pazienti, adult → adulti
XML trg: adulti pazienti ricevono terapia di tumori
Fill-Up trg: pazienti adulti ricevono la terapia per i tumori

Analysing the number of suggested bilingual terms per sentence, we notice that The Wiki Machine tends to propose more terms than TaaS (on average, The Wiki Machine 3.1, TaaS 2.5 per sentence). Of these terms, TaaS provides on average more translations for each unique source term than The Wiki Machine (on average, TaaS 1.51, The Wiki Machine 1).

In addition to evaluating the performance of TaaS and The Wiki Machine separately, for the EMEA dataset we concatenate the terminological lists provided by the tools and supply it to the XML markup and the Fill-Up approach. Embedding the combined terminology with the XML markup produces a BLEU score of 25.59 for En→It and 24.92 for It→En. This performance is similar to the scores obtained using the terminology provided by The Wiki Machine, but worse compared to TaaS. Passing the whole terminology to the Fill-Up model, the BLEU score increases up to 26.57 for En→It and 27.02 for It→En, which are the best BLEU scores for the EMEA test set. This experiment shows the complementarity of the two term identification methods and suggests a novel research direction.

5 Related Work

The main focus of our research is on bilingual term identification and the embedding of this knowledge into an SMT system. Since previous research (Wu et al. (2008); Haddow and Koehn (2012)) showed that an SMT system built by using a large general resource cannot be used to translate domain-specific terms, we have to provide the system domain-specific lexical knowledge.

Wikipedia with its rich lexical and semantic knowledge was used as a resource for bilingual term identification in the context of SMT. Tyers and Pieanaar (2008) describe method for extracting bilingual dictionary entries from Wikipedia to support the machine translation system. Based on exact string matching they query Wikipedia with a list of around 10,000 noun lemmas to generate the bilingual dictionary. Besides the interwiki link system, Erdmann et al. (2009) enhances their bilingual dictionary by using redirection page titles and anchor text within Wikipedia. To filter out incorrect term translation pairs, the authors use the backward link information to prove if a redirect page title or an anchor text represents a synonymous expression. Niehues and Waibel (2011) analyse different methods to integrate the extracted Wikipedia titles into their system, whereby they explore methods to disambiguate between different translations by using the text in the articles. In addition, the authors use morphological forms of terms to enhance the extracted bilingual dictionary. The results show that the number of out-of-vocabulary words could be reduced by 50% on computer science lectures, which improved the translation quality by more than 1 BLEU point. Arcan et al. (2013) restrict term identification to the observed domain by using the frequency information of Wikipedia categories. Different from these approaches we focus on domain-specific dictionary generation, ignoring identified terms which do not belong to the domain to be observed. Furthermore, we take advantage of the Wikipedia category graph representation and its linking to WordNet domain, which allowed us to identify the domain we were interested in.

Furthermore, research has been done on the integration of domain-specific parallel data into SMT, either by retraining small domain-specific and large general resources as one concatenated parallel data (Koehn and Schroeder, 2007), adding new phrase pairs directly into the phrase table (Langlais, 2002; Ren et al., 2009; Haddow and Koehn, 2012) or assigning adequate weights to the in- and out-of-domain translation models (Foster and Kuhn (2007); Läubli et al. (2013)). Bouamor et al. (2012) address the problem of finding the best approach to integrate new obtained knowledge in an SMT system, and show that they should be used as additional parallel sentences to train the translation model. In our approach, we use the XML markup and the Fill-Up approach, which handles the in-domain parallel data equally to the out-domain data. Furthermore, Okita and Way (2010) investigate the effect of integrating bilingual terminology in the training step of an SMT system, and analyse in particular the performance and sensitivity of the word aligner. As opposed to their approach, we do not have prior knowledge about the bilingual terminology, since we extract it from the document to be translated.

6 Conclusion

In this paper we presented an approach to identify bilingual domain-specific terms starting from a monolingual text and to integrate these into an SMT system. With the help of terminological and lexical resources, we are able to discover a small amount (∼200) of high-quality domain-specific terms and enhanced the performance of an SMT system trained on large amounts (1.8M) of parallel sentences. Monolingual and bilingual term evaluation showed no evidence that one of the tested tools (The Wiki Machine or TaaS) produces better terms than the other in all the test sets. Depending on the manual mapping between the Wikipedia categories and WordNet domains and the existence of a Wikipedia version, our approach is language and domain independent, does not need training data and is able to overcome the sparseness and coherence problems of the Wikipedia categories. Evaluation of the two systems on different language directions and domains shows significant improvements over the baseline in terms of two BLEU scores (up to 13%) and confirms the applicability of such techniques in a real scenario. It is interesting to notice that the Fill-Up technique regularly outperforms the XML markup approach, taking advantage of all terms and not only the overlapping terms in the text to be translated. Our contribution shows a different context of using Fill-Up and extends the usability of it in terms of embedding terminological knowledge into SMT. In future work, we plan to focus on exploiting morphological term variations taking advantage of the alternative terms (i.e., orthographical and morphological variants, synonyms, and related terms) provided by The Wiki Machine. This will make it possible to increase the coverage adding new terms and the accuracy of the proposed method for bilingual term identification.

Acknowledgments

This publication has emanated from research supported in part by a research grant from Science Foundation Ireland (SFI) under Grant Number SFI/12/RC/2289 and by the European Union supported projects EuroSentiment (Grant No. 296277), LIDER (Grant No. 610782) and MateCat (ICT-2011.4.2-287688).

References

Ahmet Aker, Monica Paramita, and Robert Gaizauskas. 2013. Extracting bilingual terminologies from comparable corpora. In *Proceedings of ACL*, Sofia, Bulgaria.

Mihael Arcan, Susan Marie Thomas, Derek De Brandt, and Paul Buitelaar. 2013. Translating the FINREP taxonomy using a domain-specific corpus. In *Machine Translation Summit XIV*, pages 199–206.

Luisa Bentivogli, Pamela Forner, Bernardo Magnini, and Emanuele Pianta. 2004. Revising the wordnet domains hierarchy: semantics, coverage and balancing. In *Proceedings of the Workshop on Multilingual Linguistic Ressources*, pages 101–108. Association for Computational Linguistics.

Arianna Bisazza, Nick Ruiz, and Marcello Federico. 2011. Fill-up versus Interpolation Methods for Phrase-based SMT Adaptation. In *Proceedings of IWSLT*.

Dhouha Bouamor, Nasredine Semmar, and Pierre Zweigenbaum. 2012. Identifying bilingual multi-word expressions for statistical machine translation. In *Proceedings of the Eight International Conference on Language Resources and Evaluation (LREC'12)*, Istanbul, Turkey, may. European Language Resources Association (ELRA).

Jonathan Clark, Chris Dyer, Alon Lavie, and Noah Smith. 2011. Better Hypothesis Testing for Statistical Machine Translation: Controlling for Optimizer Instability . In *Proceedings of the Association for Computational Lingustics*.

Maike Erdmann, Kotaro Nakayama, Takahiro Hara, and Shojiro Nishio. 2009. Improving the extraction of bilingual terminology from wikipedia. *ACM Trans. Multimedia Comput. Commun. Appl.*, 5(4):31:1–31:17, November.

Marcello Federico, Nicola Bertoldi, and Mauro Cettolo. 2008. Irstlm: an open source toolkit for handling large scale language models. In *INTERSPEECH*, pages 1618–1621. ISCA.

George Foster and Roland Kuhn. 2007. Mixture-model adaptation for smt. In *Proceedings of the Second Workshop on Statistical Machine Translation*, StatMT '07, pages 128–135, Stroudsburg, PA, USA. Association for Computational Linguistics.

Claudio Giuliano, Alfio Massimiliano Gliozzo, and Carlo Strapparava. 2009. Kernel methods for minimally supervised wsd. *Computational Linguistics*, 35(4):513–528.

Barry Haddow and Philipp Koehn. 2012. Analysing the Effect of Out-of-Domain Data on SMT Systems. In *Proceedings of the Seventh Workshop on Statistical Machine Translation*, Montréal, Canada. Association for Computational Linguistics.

Philipp Koehn and Josh Schroeder. 2007. Experiments in domain adaptation for statistical machine translation. In *Proceedings of the Second Workshop on Statistical Machine Translation*, StatMT '07, pages 224–227, Stroudsburg, PA, USA. Association for Computational Linguistics.

Philipp Koehn, Hieu Hoang, Alexandra Birch, Chris Callison-Burch, Marcello Federico, Nicola Bertoldi, Brooke Cowan, Wade Shen, Christine Moran, Richard Zens, Chris Dyer, Ondřej Bojar, Alexandra Constantin, and Evan Herbst. 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*, ACL '07, pages 177–180, Stroudsburg, PA, USA. Association for Computational Linguistics.

Philipp Koehn. 2005. Europarl: A Parallel Corpus for Statistical Machine Translation. In *Conference Proceedings: the tenth Machine Translation Summit*, pages 79–86. AAMT.

J. Richard Landis and Gary G. Koch. 1977. Measurement of Observer Agreement for Categorical Data. In *Biometrics*, volume 33, pages 159–174.

Philippe Langlais. 2002. Improving a general-purpose statistical translation engine by terminological lexicons. In *Proceedings of the 2nd International Workshop on Computational Terminology (COMPUTERM) '2002, Taipei, Taiwan*, pages 1–7.

Samuel Läubli, Mark Fishel, Martin Volk, and Manuela Weibel. 2013. Combining statistical machine translation and translation memories with domain adaptation. In Stephan Oepen, Kristin Hagen, and Janne Bondi Johannesse, editors, *Proceedings of the 19th Nordic Conference of Computational Linguistics (NODALIDA 2013), May 22?24, 2013, Oslo University, Norway*, Linköping Electronic Conference Proceedings, pages 331–341, Oslo, May. Linköpings universitet Electronic Press.

Pablo N Mendes, Max Jakob, Andrés García-Silva, and Christian Bizer. 2011. Dbpedia spotlight: shedding light on the web of documents. In *Proceedings of the 7th International Conference on Semantic Systems*, pages 1–8. ACM.

Rada Mihalcea. 2007. Using Wikipedia for Automatic Word Sense Disambiguation. In *Proceedings of NAACL-HLT*, pages 196–203.

Jan Niehues and Alex Waibel. 2011. Using Wikipedia to Translate Domain-specific Terms in SMT. In *nternational Workshop on Spoken Language Translation*, San Francisco, CA, USA.

Franz Josef Och and Hermann Ney. 2003. A systematic comparison of various statistical alignment models. *Computational Linguistics*, 29.

Tsuyoshi Okita and Andy Way. 2010. Statistical Machine Translation with Terminology. In *Proceedings of the First Symposium on Patent Information Processing (SPIP)*, Tokyo, Japan.

Kishore Papineni, Salim Roukos, Todd Ward, and Wei-Jing Zhu. 2002. Bleu: a method for automatic evaluation of machine translation. In *Proceedings of the 40th Annual Meeting on Association for Computational Linguistics*, ACL '02, pages 311–318.

Mārcis Pinnis, Nikola Ljubešić, Dan Ştefănescu, Inguna Skadiņa, Marko Tadić, and Tatiana Gornostay. 2012. Term extraction, tagging, and mapping tools for under-resourced languages. In *Proceedings of the Terminology and Knowledge Engineering (TKE2012) Conference*.

Zhixiang Ren, Yajuan Lü, Jie Cao, Qun Liu, and Yun Huang. 2009. Improving statistical machine translation using domain bilingual multiword expressions. In *Proceedings of the Workshop on Multiword Expressions: Identification, Interpretation, Disambiguation and Applications*, MWE '09, pages 47–54, Stroudsburg, PA, USA. Association for Computational Linguistics.

John Shawe-Taylor and Nello Cristianini. 2004. *Kernel Methods for Pattern Analysis*. Cambridge University Press, New York, NY, USA.

Raivis Skadinš, Marcis Pinnis, Tatiana Gornostay, and Andrejs Vasiljevs. 2013. Application of online terminology services in statistical machine translation. In *Proceedings of the XIV Machine Translation Summit*, Nice, France.

Ralf Steinberger, Bruno Pouliquen, Anna Widiger, Camelia Ignat, Tomaz Erjavec, Dan Tufis, and Dániel Varga. 2006. The jrc-acquis: A multilingual aligned parallel corpus with 20+ languages. In *Proceedings of the 5th International Conference on Language Resources and Evaluation (LREC'2006)*.

Jörg Tiedemann. 2012. Parallel data, tools and interfaces in opus. In Nicoletta Calzolari (Conference Chair), Khalid Choukri, Thierry Declerck, Mehmet Uğur Doğan, Bente Maegaard, Joseph Mariani, Jan Odijk, and Stelios Piperidis, editors, *Proceedings of the Eight International Conference on Language Resources and Evaluation (LREC'12)*, Istanbul, Turkey, may. European Language Resources Association (ELRA).

Francis M. Tyers and Jacques A. Pieanaar. 2008. Extracting bilingual word pairs from wikipedia. In *Collaboration: interoperability between people in the creation of language resources for less-resourced languages (A SALTMIL workshop)*.

Hua Wu, Haifeng Wang, and Chengqing Zong. 2008. Domain adaptation for statistical machine translation with domain dictionary and monolingual corpora. In *Proceedings of the 22nd International Conference on Computational Linguistics - Volume 1*, COLING '08, pages 993–1000.

Torsten Zesch and Iryna Gurevych. 2007. Analysis of the wikipedia category graph for nlp applications. In *Proceedings of the TextGraphs-2 Workshop (NAACL-HLT)*, pages 1–8, Rochester, April. Association for Computational Linguistics.