Translation for the rest of us:
Usable word-level translation for automated text analysis

A. Maurits van der Veen (maurits@wm.edu)
William & Mary

Draft, May 2022

Abstract

Automated machine translation has been improving in quality rapidly for some years, and now approaches that of professional human translation. However, for large volume translations, these methods remain costly, in terms of money, computational resources, or time. In contrast, word-level translation is both free and fast, simply mapping each word in a source language deterministically to a target language. While the resulting translations are ungainly, for many text analysis methods this is not a problem. In this paper, I first outline a way to produce high-quality translation dictionaries using aligned word embeddings, and then show that applying text analysis methods including sentiment analysis, emotion analysis, and topic modeling to the translations generated using these dictionaries produce results that are very similar to those produced by human translation or state-of-the-art neural machine translation. The translation dictionaries as well as the code used to generate them are available on Github.

Note for CompText 2022: the comparison to nmt (using OPUS-MT) is not yet done (awaiting completion of the translation, Colab willing :-).

Translation for the rest of us:
Usable word-level translation for automated text analysis

Introduction

Much of social science relies on texts. Speeches, debates, media coverage, memoirs, tweets, and many other texts are key to our understanding of the world and the people around us. Often, the sheer volume of relevant texts makes it infeasible for individual scholars to read everything of interest. Accordingly, automated text analysis has rapidly become a popular approach (Grimmer & Stewart, 2013). For comparative analyses across linguistic borders, translation becomes essential. Machine translation offers a way to do this automatically, and recent advancements in neural machine translation produce translations of impressive quality, approaching those produced by professional human translators (Graham et al., 2019).

Google Translate is perhaps the best known automated translation engine available to the general public. Previous research has shown that applying automated text analysis techniques to Google Translate output produces results that are highly similar to those from performing the same techniques on human translations of the same texts (de Vries et al., 2018). DeepL is another, more recent machine translation service. While such services are invaluable for collections of a few hundred or even a few thousand texts, their costs rise rapidly for applications featuring tens of thousands of texts or more, as is increasingly common for text analysis applications.

Fortunately, many of the most widely used text analysis methods do not require high-quality, grammatically correct translations. In fact, many techniques take a "bag of words" approach, which ignores grammar and word order altogether. For such applications, the most important factor is whether key terms are translated correctly. As I show here, word-level translation can produce translations whose suitability for common text analysis techniques approaches that of high quality neural machine translation or even human translation. Moreover, it produces such translations both rapidly and at essentially no cost, making it feasible to compare translations into different target languages. The great advantage of word-level translation is that any word only needs to be translated once, before being added to the translation dictionary.

The dictionary creation approach outlined here takes advantage of word embeddings, which approximate high-dimensional semantic spaces and can be mapped across languages, thus providing a vehicle for automated translation. The core insight is that, if semantic spaces for two different languages are accurately mapped onto one another, translating a word simply means identifying the target language word whose location in the shared space is closest to that of the source language word.

In reality, the challenge is not as straightforward, as embeddings are imperfect, can contain words from other languages, and have other idiosyncratic characteristics. Accordingly, blindly choosing the nearest target language neighbor produces poor quality translations. The approach described here addresses such shortcomings, among others by taking into account word frequencies, diacritical marks, capitalization, etc.

The paper proceeds as follows. I begin with a brief overview of competing approaches to machine translation. Next, I outline my method of producing high-quality, word-level translation dictionaries and provide summary information about the translation dictionaries produced so far. The third section introduces the European Parliament speech corpus used to assess the quality of these dictionaries. These speeches are translated by professional human translators into all official languages of the European Union, providing an ideal corpus for comparing translation quality across multiple language pairs. Finally, I assess the translation quality of the word-level translations by investigating how closely the results on three standard text analysis tasks — sentiment analysis, measuring emotions in text, and topic modeling — compare between the professional translation into English, state-of-the-art neural machine translation, and the word-level translations produced here. The translation dictionaries, along with replication code for producing these dictionaries as well as for producing additional dictionaries for other language pairs, are available on Github.

The appeal of word-level machine translation

Recent years have brought rapid advancements in machine translation, both at the word and document level. For document-level translation, neural machine translation produces the best results. Already in 2018, analyses showed that texts translated using Google Translate produced term-document matrices that were comparable to those produced from texts translated by professional human translators, and that applying a technique such as topic modeling to the automatically translated texts generated highly similar results (de Vries et al.). The quality of Google Translate's output has only increased since then, and other commercial services have appeared offering similarly high-quality translation, most notably DeepL (www.deepl.com). However, commercial translation services come at a cost. Even though both Google Translate and DeepL are very reasonably priced, the cost of translating the volumes of texts now regularly used in automated text analysis studies can quickly mount to thousands or even tens of thousands of dollars.

Open-source technologies for neural machine translation (Junczys-Dowmunt et al., 2018; Klein et al., 2018, 2020) eliminate these costs, and with sufficient training can produce translations that approach the quality of Google Translate or DeepL. Moreover, pre-trained translation engines for

a growing number of language pairs can be found online, obviating the need for users to collect their own training data for and engage in the time-consuming training of the translation engines. Even the computational resources such technologies require do not need to acquired directly, as Google's Colab makes it possible to run them online. However, the time demands of producing translations remain considerable.

The corpora used in the validation tests in this paper total about one million short parliamentary speeches, adding up to 190 million words. This makes it a medium-size dataset for automated text analysis work. Translating them using a commercial neural machine translation engine such as DeepL would cost around $30,000. In contrast, using OPUS-MT, a comparatively fast, pre-trained open machine translation model (cite), is free, but would take approximately 240 hours on Google Colab, or 10 days running non-stop.[1] In contrast, translating the same corpora using the dictionaries created here costs nothing and takes a matter of minutes. Moreover, as I show in this paper, the resulting translations are of sufficient quality for many real-world applications.

Method

All leading automated machine translation approaches build on the insight that one can "know a word by the company it keeps" (Firth, 1957, p. 11). Examining large quantities of texts makes it possible to place words (and phrases) in a multidimensional space which encapsulates meaning, but also part-of-speech, word form, etc. (Gärdenfors, 2004; Mikolov, Chen, et al., 2013). Moreover, it is possible to map these spaces to one another across languages. Such mappings can produce high-quality information about the most similar words in different languages (Chen & Cardie, 2018; Conneau et al., 2017; Joulin et al., 2018; Mikolov, Le, et al., 2013). I use the mapping process proposed by Joulin et al. to map individual language spaces to one another (2018).[2]

The greatest disadvantage of even the very best word-level translation for text analysis purposes is not that the resulting translation produces ungrammatical and stilted text: for most analyses that is not an issue. Instead, the problem is that for each source language word we need to select a single target language word without knowing anything about the context in which the source language word occurs. This is particularly problematic for words with multiple meanings (polysemy). For example, for 'sink', do we choose a translation for one of its verb meanings ('descend', or 'go down below a surface'), or for its main noun meaning (a 'fixed basin with a

[1] In fact, Colab's usage and idling limits make even this unrealistic, and it would likely take longer. A paid Colab Pro subscription lessens but does not eliminate this problem.

[2] For this paper I use the pre-mapped Wikipedia-based word embeddings made available by Facebook at https://fasttext.cc/docs/en/aligned-vectors.html (accessed 2022-04-30), but applying the mapping to other languages is straightforward and gives analogous results.

water supply and a drain')? Whichever we choose, it is certain to be hilariously wrong at least part of the time.

In embeddings, the problem of polysemy leads to word placements located somewhere between the various individual meanings of a word, with the proximity to each individual meaning affected by the comparative frequency of that meaning among all uses of the word. Thus, a set of word embeddings trained on, say, a plumber's manual will locate 'sink' much closer to the 'fixed basin' meaning than will embeddings trained on accounts of war-time naval battles. In general purpose embeddings of the type used here, the location will be somewhere inbetween. Whether this is a problem for a particular application is an empirical question.

Fortunately, when translating texts within a particular domain, it is trivial to substitute preferred translations for particular terms. Thus, to translate a plumber's manual using our general purpose dictionary, we could simply supply our own domain-specific list of plumbing-related translations, in which 'sink' is always translated to the target language version of the 'fixed basin' meaning. However, as I show below, even applying a general-purpose translation dictionary without such supplementation to the fairly specific context of the European Parliament produces high quality results. It would be straightforward to improve those further by substituting specific translations of EU-related terms, such as those available in the European Union terminology dataset (https://iate.europa.eu/home). I do not do so here, since not all domains have similar multilingual glossaries available to substitute in. However, as a general rule it is advisable to verify the correct translation of key domain-specific terms in the dictionary and to override the general purpose translations if necessary.

I now turn to some specific challenges associated with generating a translation dictionary from aligned word embeddings. In principle, translation is a simple matter of finding a word's location in the source language space and identifying the closest word to that location in the target language space. The standard metric for finding the closest word is cosine distance (equal to the vector dot product for vectors normalized to length 1). However, this metric fails to take into account that word locations are not evenly distributed across the vector space. As Radovanovic et al. have shown, high dimensionality inevitably produces 'hubs', which are nearest neighbours of many words (2010). More significantly, the distribution of words across vector space varies by language: a hub in one language will not necessarily be a hub in another.

This makes the cosine distance metric less helpful, and produces undesired results when mapping across spaces. In particular, it turns out that words will tend to be closer to words in another language that have a similar *frequency*, but not necessarily a similar meaning. Moreover, the problem is worse for more frequent words, which are often among the ones for which translation errors will have the largest impact. Schnabel et al. show that this pattern is driven by the algorithms that generate the word embeddings, rather than by the "intrinsic properties of natural

language" (2015, p. 10). Fortunately, this also means that it is a problem for which we can make adjustments.

Accordingly, I use the improved distance metric proposed in 2017 by Conneau et al., which adjusts for the degree of 'hubness' of various candidate neighbours (2017). This cross-domain similarity local scaling (CSLS) adds two adjustments to the standard cosine metric: one for the average distance to the nearest target-language neighbours of the source-language word (Rtarget(Wsrc)), and one for the average distance to the nearest source-language neighbours of the target-language translation candidate (Rsource(Wtgt)). The CSLS metric is:

$$CSLS(Wsrc, Wtgt) = 2cos(Wsrc, Wtgt) - Rtarget(Wsrc) - Rsource(Wtgt)$$

I follow Conneau et al. in using 10 as the number of nearest neighbours to consider (2017, p. 4). The impact of using the CSLS distance metric can be dramatic: words that are not even among the 25 closest neighbours using 'standard' cosine distance regularly become the closest neighbour under CSLS. More importantly, the top candidates using CSLS are virtually always better translations for the source word than the top candidates using cosine distance are.

Even using this improved distance metric, translation quality can still be fairly low, due to some confounding features of most embedding spaces. To help mitigate the negative effects of these features, I apply several additional heuristics in order to improve translation quality further. These heuristics fall into two categories: those that aim to identify the most appropriate embedding vector for a given source language word, and those that select the most appropriate translation from among the nearest target language neighbours of the source embedding vector in the aligned embedding space.

1. Identifying the appropriate embedding vector(s) for a source language word.

Word-level translation approaches frequently ignore words that do not appear in the source embeddings (Long et al., 2017). Even though the embedding vocabularies used are often quite large — some contain a million terms or more — many of the included terms are misspellings, variations on a word stem (conjugations etc.), and proper names, so the effective embedding size is considerably smaller. When unknown words are encountered, they are often simply replaced by an out-of-vocabulary marker in the translated text. This is not helpful for real-world applications.

One approach to this problem would be to draw upon the ngram (sub-word) encoding approached pioneered by fastText (Bojanowski et al., 2016; see also Sennrich et al., 2015) to generate new embedding vectors for out-of-vocabulary words based on sub-word character sequences. However, in practice a straightforward set of heuristics almost always produces better

vectors. First, if the word contains capital letters, the program checks whether a vector exists for the lowercased version of the word. Similarly, if the word contains diacritical marks (accents, etc.), the program checks whether a vector exists for a version of the word with all such marks removed. Next, if the word is common in the target language, the program assumes the word has been borrowed from that language and does not require translation. This happens regularly when the target language is English.

The program also implements a rudimentary form of spell-checking: if any single-character edit to a word produces a fairly common source-language word, it substitutes the translation for that word. Further, the program tries word segmentation: if a word can be split into pieces that are each fairly common words in the source language, it translates them separately. This is particularly helpful for languages with many compound words, such as Germanic languages. Finally, for long words (8 letters or more) the program checks whether stemming, or removing the last 1-4 letters would produce a source-language word; if so, it uses the translation for that word. If none of these options produce a translation, the source word more often than not is a proper name, for which a straight copy is the appropriate 'translation.'

Each of these options is coupled with a threshold, in terms of word frequency rank. Table 1 lists the thresholds. Thus, if a word is accepted as borrowed from the target language only if it is among the top 5,000 most common words in that language. All of these thresholds have been arrived at inductively, by looking at frequency rankings for cases where an option should not be accepted. For instance, spell-checking suggestions that are not in the top 50,000 most common source language words likely indicate that the original word is simply a rare word itself, or else a proper name, rather than a misspelled word.

Word frequency rank threshold	**Value**
Target language rank high enough to assume target language word	5,000
Ibid. for lower-cased version of word	4,000
Source language rank of accent-stripped word	50,000
Ibid. for lower-cased version of word	40,000
Source language rank of spell-check suggestion	50,000
Ibid. for lower-cased version of word	40,000
Source language rank of spell-check suggestion for word ending	75,000
Ibid. for lower-cased version of word	60,000
Sum of source language ranks of word segments	75,000
Ibid. for lower-cased version of word	60,000
Source language rank of stemmed word	300,000
Ibid. for lower-cased version of word	300,000

Table 1. Thresholds for translation heuristics.

For each word translated, we also keep track of the way it was translated. This makes it straightforward to change settings and update the translations, as needed. For example, in trying

to segment words, the program by default adopts a minimum segment length of 2. However, in Dutch, participles often end in 'ende' — for example, 'vliegende' (flying). Unfortunately, "ende" can be decomposed into two of the most common words in Dutch, 'en' and 'de' ('and' and 'the', respectively). To see whether many words are thus erroneously segmented (they are not), we could set the minimum segment length to 3 and produce new translations for only those words translated using segmentation.

To establish whether a given word is more or less common, as required in the heuristics outlined above, word frequency data are needed. The main source for these data is the python module *wordfreq*, by Luminoso (Speer et al., 2018), which has frequency data for the top 300,000 words or more for most of the languages used in this paper. For Danish, *wordfreq* does not have such extensive frequency data. Instead, I generate a frequency dictionary based on more than 1 million Danish newspaper articles, from the newspaper *Politiken*, 1997-2017.

2. *Identifying the appropriate target language word*

Standard word embeddings are generated from large quantities of texts automatically collected. The automatic collection process inevitably includes some texts in other languages, causing words in the wrong language to be included in the embedding. As a result, the Wikipedia-based fastText vectors for West-European languages contain numerous words that use non-Latin letters (Chinese, Arabic, etc.). For example, the closest neighbour to the Dutch word 'consequent' ('consistent', in English) in the English model is 舘 (Japanese kanji for palace). To filter out such words, I ignore any translation candidates that include characters not used in the target language.

In addition, word embeddings also contain vectors for misspelled words. The prevalence of proper names in texts makes it difficult to identify misspelling with certainty. However, whenever a source-language word is reasonably common, but the target-language translation candidate proposed is dramatically less so (because misspelled), I ignore that candidate, unless all other candidates are similarly rare.

Translation dictionary contents

The words in the *wordfreq* module provide the initial list of words to include in the translation dictionary. In addition, I include all the words in texts included in the test dataset, which is drawn from debates in the European Parliament. For each word translated, the translation dictionary often grows by more than one word, for instance when a word is translated by looking up the translation for its lower-case version, or when a compound word's individual segments are translated. As a result, the eventual size of the translation dictionaries varies considerably from one language to the next.

Table 2 shows the current size of the translation dictionaries into English for the 10 languages included so far. In addition to the total size, it shows the percentage of words that were directly translated, either in their original or lower-cased version, as well as the percentage that were translated as compound words. For most languages, these two categories account for 80-90% of all word translations. The remainder is translated through stemming, spelling adjustments, or by simply copying the word from the source to the target language, on the assumption that it is either a proper name or borrowed from that language. The appendix gives full details on the relative contribution of different translation heuristics to the translation dictionary.

Language	**# words translated**	**% direct translations**	**% segmented**
DA - Danish	30,124	98.64	0.09
DE - German	1,580,681	76.89	9.46
ES - Spanish	731,951	74.62	5.99
FI - Finnish	340,460	50.28	17.76
FR - French	805,326	77.37	7.29
IT - Italian	714,085	78.42	6.39
NL - Dutch	1,230,467	56.37	22.25
PL - Polish	187,007	83.61	4.35
PT - Portuguese	200,589	89.85	2.34

Table 2. Translation dictionary information

Validation data

In order to validate translation quality, it is essential to have a corpus of parallel texts in the source and target languages. The most widely used such corpus for European languages is *Europarl*, which provides parallel text in 21 official languages of the European Union, drawn from the legislative record of the European Parliament (Koehn, 2005). Legislative debates in the European Parliament (EP) are, by law, translated into the primary official national language of each member state by professional translators. As such, the translation is of a very high quality.

I use release v7 of the corpus, which includes speeches through November 2011. More specifically, I use *EuroparlExtract*, a cleaned and parsed version of the corpus which makes it easier to identify the original language of any speech (Ustaszewski, 2019; see also Graën et al.,

2014). This is of interest because we can compare whether our translation dictionary's performance differs depending on whether the source was originally in that language, or is, instead, already the result of a translation process. For example, the parallel corpus for Dutch and English will contain some original Dutch speeches, but also speeches translated into Dutch from any of the other languages used in the EP. It seems possible, for example, that our translation method will do better at translating an original English speech back into English from the Dutch into which it was translated by a human translator than it will at translating an original Dutch speech into English for the first time.

I analyze translation from 9 different European languages into English, selecting the languages of the five largest EU member states: German, French, Italian, Spanish, and Polish, plus the national languages of four smaller member states representing different language groups: Portuguese (Romance), Dutch (Germanic), Danish (Nordic, or North Germanic), and Finnish (Uralic). All languages studied are written using the Latin alphabet, albeit with varying sets of diacritical markers that extend the number of distinct letters we may encounter.

I compile three different corpora. The *EuroparlExtract* corpus stores texts by source & target language, one speech per file. Each line in the file, generally corresponding to a sentence, has source text and target text, separated by a tab. Due to the corpus creation process for *EuroparlExtract*, the source text for a given speech can appear with small variations depending on the target language, either at the sentence or the speech level. That makes it problematic to compare translations directly to one another. In the first two corpora I collect, therefore, I use only speeches or sentences whose source language contents are the same across all the target languages.

The first corpus, *Fullspeeches*, includes all complete speeches for which the source language appears identically across all 10 target languages in our dataset. This produces parallel corpora across all nine languages to be translated into English, along with a 'gold standard' corpus of texts that were either originally in English or translated into that language by professional translators. The second corpus, *Filteredspeeches*, includes all sentences that appear identically across all 10 languages in the dataset, aggregated into the speeches they appeared in. This corpus includes more speeches, but the average speech length is shorter (since sentences that vary are excluded). The third corpus, *Sourcespeeches*, includes only sentences whose original language is the source language, along with their translation into English. This corpus eliminates possible biases introduced by prior translation from another language, but it is (obviously) not parallel across the nine languages.

Table 3 shows, for each language, the corpus sizes in speeches, total words, and distinct words.[3] Table 4 provides the same information for the single language corpora, plus the ratio of distinct words in the source language to that in English. That ratio always exceeds 1, likely due to two factors: translation tends to simplify language, and English is a language with fewer word forms (conjugations, declensions, etc.) than most. For this reason, English is often an easier language to translate into than out of.

Language	**Full speeches (n = 46,139)**	**Filtered speeches (n = 50,661)**
DA - Danish	8,943,659 (135,914)	8,511,459 (133,510)
DE - German	9,188,586 (156,240)	8,735,527 (153,011)
ES - Spanish	10,420,425 (80,951)	9,900,725 (79,940)
FI - Finnish	6,608,905 (288,704)	6,268,182 (282,206)
FR - French	10,295,323 (68,236)	9,777,279 (67,547)
IT - Italian	9,454,217 (87,648)	9,005,119 (88,495)
NL - Dutch	10,032,644 (114,362)	9,552,203 (113,869)
PL - Polish	8,154,166 (158,755)	7,766,268 (158,861)
PT - Portuguese	9,925,111 (89,234)	9,433,333 (88,452)

Table 3. Word count info for *Fullspeeches* and *Filteredspeeches* corpora.
Total words across all speechs, with number of distinct words in parentheses.

Language	**Speeches**	**Total words**	**Distinct words**	**Distinct word ratio (source/English)**
DA - Danish	2,414	631,365	32,702	1.85
DE - German	21,374	5,796,590	136,055	2.77
ES - Spanish	9,171	3,092,418	52,382	1.63
FI - Finnish	3,089	578,767	80,845	3.95
FR - French	19,586	6,245,683	65,977	1.32

[3] Words with different capitalization are considered distinct. Many distinct words are proper names, so the count of distinct words is not a measure of the vocabulary size of 'actual' (i.e. not proper name) words in a corpus or language.

IT - Italian	11,419	2,857,803	55,805	1.55
NL - Dutch	9,886	2,965,638	70,382	1.89
PL - Polish	4,933	736,423	58,929	2.73
PT - Portuguese	11,926	2,634,029	50,375	1.65

Table 4. *Sourcespeeches* corpus information.

Validation analysis

There is no doubt that word-level translation produces ungainly translations. Figure 1 shows the same original Danish sentence, as translated by the European Union's professional human translators, by Google Translate, and using the word level translation dictionaries introduced here. Finally, the fourth entry shows the word level translation from the Italian translation produced by the European Parliament's professional translators.

Danish original
Kernen i de problemer, vi beskæftiger os med her, er den demografiske ubalance i verden.

EU human translation
At the core of the problems that we are concerned with here is the demographic imbalance in the world.

Google Translate
At the heart of the issues we are dealing with here is the demographic imbalance in the world.

OPUS-MT
The core of the problems we are dealing with here is the demographic imbalance in the world.

Word-level translation, from Danish original
Core in ones problems we employs ourselves with where is the demographical imbalance in world

Word-level translation, from professional Italian translation
The center of problems of including there whaaaa occupying in this headquarter c is it imbalance demographic in world

Figure 1. Translations compared.

While both of the word-level translations are clunky and difficult to read, it is important to note that they do include the three substantive key words in the sentence: "demographic", "imbalance", and "world". Interestingly, too, the word-level translations retain the word "problems", as does the EU's official translation, whereas Google substitutes "issues". The direct word-level translation also retains the word "core" (as do the EU's official translation and the translation using OPUS-MT), while Google substitutes "heart" and the word-level translation from Italian replace it by "center". In short, in terms of correctly translating key words that might

be of relevance in automated text analysis, the word level translation of this example performs quite well — arguably even better than Google Translate, if we care about translating the important words as exactly as possible.

To validate the translations overall, I perform three separate automated text analyses on each of the three corpora outlined above. In each case, I pool the human-translated and machine-translated texts into a single corpus, run the analysis, and assess the correlation of the results at the level of individual texts, by original language. Here, I present the results for the *Sourcespeeches* corpus, which includes only speeches in their original language. These represent a more realistic proxy for likely applications of the translation dictionary, but the results for different languages are not fully comparable since the source texts differ. Results for the other two corpora, *Fullspeeches* and *Filteredspeeches*, do permit direct comparison. Fortunately, the overall patterns are highly similar across all three corpora. Results for the translations of the latter two corpora appear in the appendix.

1. Sentiment analysis

I use the sentiment analysis method *MultiLexScaled*, a high-performing lexicon-based approach (van der Veen & Bleich, 2021), which builds on eight widely-used SA lexica, and calibrates them against a representative corpus of representative newspaper texts from the United Kingdom. Figure 2 presents two different measures of the comparison between professional human translation and word-level translation. The first cluster of bars shows the percentage of texts, by language, for which the sentiment polarity (positive or negative) matches. Since we are also often concerned with relative gradations in sentiment (more or less positive; more or less negative), the second cluster shows the correlation between the sentiment scores for human and word-level translation, respectively.

The languages are clustered by family: four Romance languages, then three Germanic ones, followed by Finnish and Polish on their own. The dotted gray line in the chart at 0.75 represents a level of coding agreement that is often used as a threshold in both automated and manual coding. As is clear from the figure, the word-level translation comfortably exceeds this threshold for each language as regards sentiment polarity, with the Romance languages apparently easiest to translate correctly at the word level. Interestingly, the German automated translation performs weakest in terms of sentiment gradations correlating, even though English is itself a Germanic language. Average performance on the binary measure is 0.82; for correlation it is 0.77. Results for the other two corpora, which are larger but also include texts that are not originally in the source language, are very similar: the average correct fraction is 0.83 for both, and the average correlation is 0.78 (*FullSpeeches*) and 0.75 (*FilteredSpeeches*). Full results can be found in the appendix.

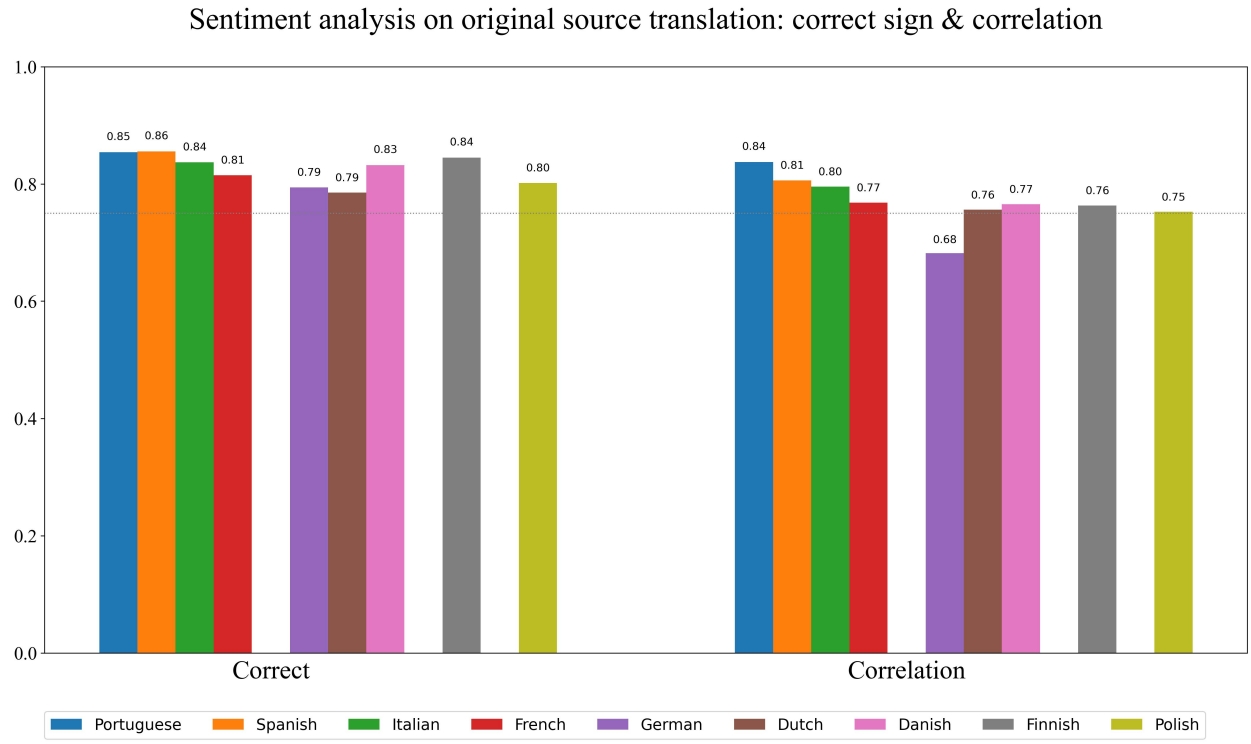

Figure 2. Sentiment analysis performance.

2. Emotions in text

To measure emotions in text, I use the WM_emotion lexicon, which outperforms other published emotion lexica (including the most widely used, LIWC) across a range of benchmark corpora (cite). The lexicon contains word lists for each of the 8 emotions in Plutchik's well-known wheel of emotions: anger, anticipation, disgust, fear, joy, sadness, surprise, and trust (2001). I use the lexicon to count the number of words in each emotion category in each speech, both in its human-translated form and in the machine-translated version. This test is more difficult than the sentiment analysis test, since the individual emotion dictionaries range in size from roughly 200 words (for fear) to roughly 500 words (for trust), compared to thousands of words in the sentiment analysis lexica.

To compare human translation against machine translation, I compare the eight emotion counts produced for each version of a text in two ways. First, I consider the set of eight values as a single vector, and measure the cosine similarity between the two vectors. In addition, I compare the emotions individually, calculating the correlation between the human translation counts and the machine translation counts for each, and then averaging those eight correlation values to get an overall measure. The results are shown in figure 3.

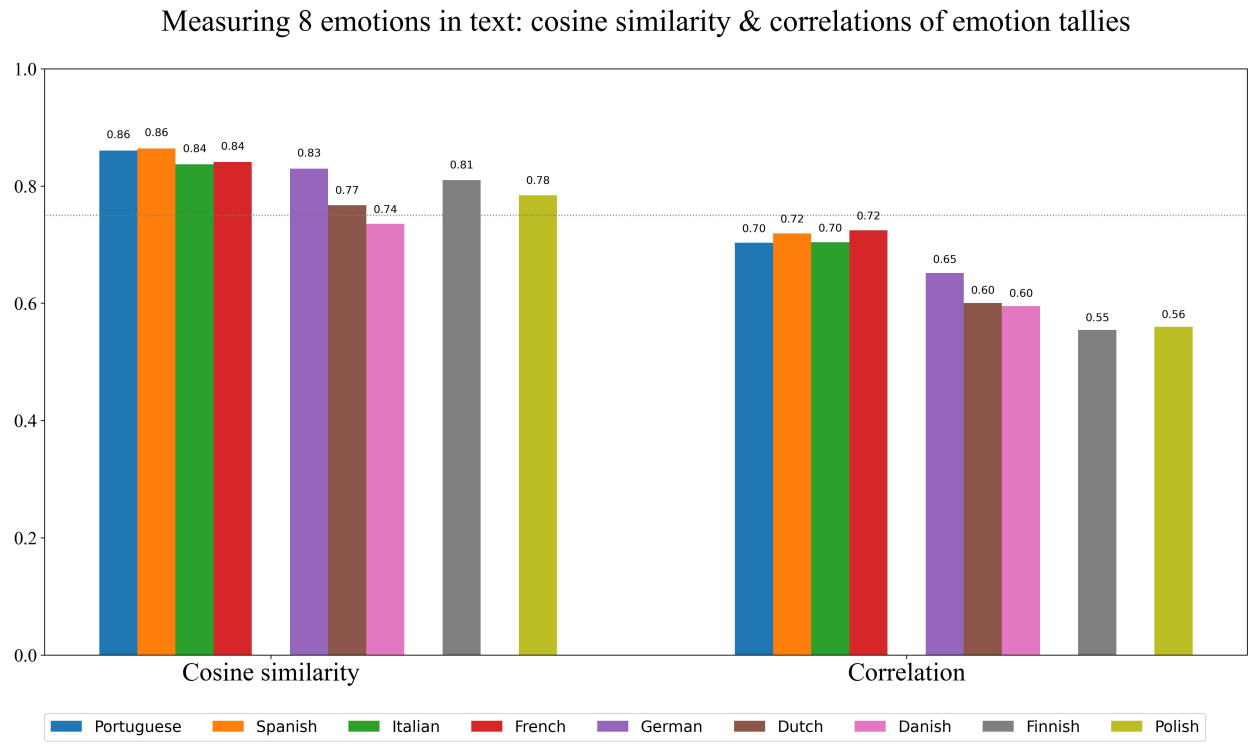

Figure 3. Emotion measurement performance.

Again, the overall similarity between emotion measurements on the human translations and those on the word-level translations is quite high. For individual emotions, correlations are a bit lower. As noted, this is likely due in part to the comparatively small dictionary sizes for these emotions. More generally, it is simply very difficult to identify specific emotions in texts (cite). In the present case, the greatest differences between the human and the machine translations (i.e. the lowest correlations) occur for anger, disgust, and surprise; in contrast, fear is apparently the emotion most robust to machine translation. Correlation data broken down by individual emotion appears in the appendix.

3. Topic modeling

I run a topic model on a pooled corpus containing, for each text or speech, the EU's official translation into English, along with the automated translation of the source language, or for the *Fullspeeches* and *Filteredspeeches* corpora, each of the 9 non-English versions of the speech The resulting corpora contain either 2 or 10 versions of each text. For each corpus, I run a topic modeling analysis, using Non-Negative Matrix Factorization (NMF) (Lee & Seung, 1999), selecting the optimal number of topics by measuring the average topic coherence across all topics in the model using corpus-specific embeddings (cf. Greene & Cross, 2017).

For the *Sourcespeeches* corpus, the optimal number of topics is 25. The top 6 words for each topic are shown in table 5. NMF topic models almost always contain a few topics that aggregate

common non-substantive features of texts, such as standard text headers, stopwords, etc. In the case of legislative debates, various formalities and procedural terms also get included here. As a result of the translation process, such features are comparatively more common: 9 of the 25 topics fall into this category. In table 5, these non-substantive topics, as judged from their top words, are greyed out. They are not included in the similarity calculations reported below.

Topic	**Top 6 words**
	believe, need, madam, way, time, thank
	lord, everybody, sure, pretend, proper, kind
	ramble, monsieur, ends, saw, discrepancy, president
EU institutions & activities	council, parliament, commission, proposal, treaty, decision
General politics	peace, government, political, international, situation, country
	vice, needs, al, ussr, whatsoever, lies
	lord, sole, president, mi, turns, kidding
	written, aim, bound, united, members, kind
Legislative debate & voting	vote, debate, place, statements, written, rule
EU member states	member, states, national, state, legal, eu
Trade & extra-EU relations	countries, trade, eu, agreement, agreements, world
	honorable, sole, breaks, president, relationship, doing
Resolutions & voting	voted, favor, report, resolution, vote, voting
	ones, rapporteur, pretty, exceptionally, entering, chairman
Regional & cohesion policy	development, regional, regions, policy, program, cohesion
Budget	budget, financial, budgetary, eur, funds, fund
Economics & welfare	social, economic, growth, employment, europe, stability
Energy	energy, nuclear, climate, renewable, emissions, gas
Fisheries	fishing, fisheries, fishermen, fish, sector, fleet
Amendments & voting	amendment, oral, amendments, voting, vote, paragraph
Legislative deliberations	item, committee, behalf, report, statement, affairs
	dear, amending, considered, chairman, determination, arise
EU markets	products, market, consumers, health, food, directive
Rights	rights, human, fundamental, respect, freedom, charter
Women & children	women, men, equality, violence, children, equal

Table 5. Topics in *Sourcespeeches* corpus.

As with the emotion analysis, I assess the similarity between the human and machine translations in two ways: across all topics, and by topic (averaged across all topics to produce a single number). Figure 4 shows the results. On average cosine similarity across the 16 topics, all nine languages score 0.75 or higher. In terms of correlations at the individual topic level, all four Romance languages do so, along with Dutch and Polish.

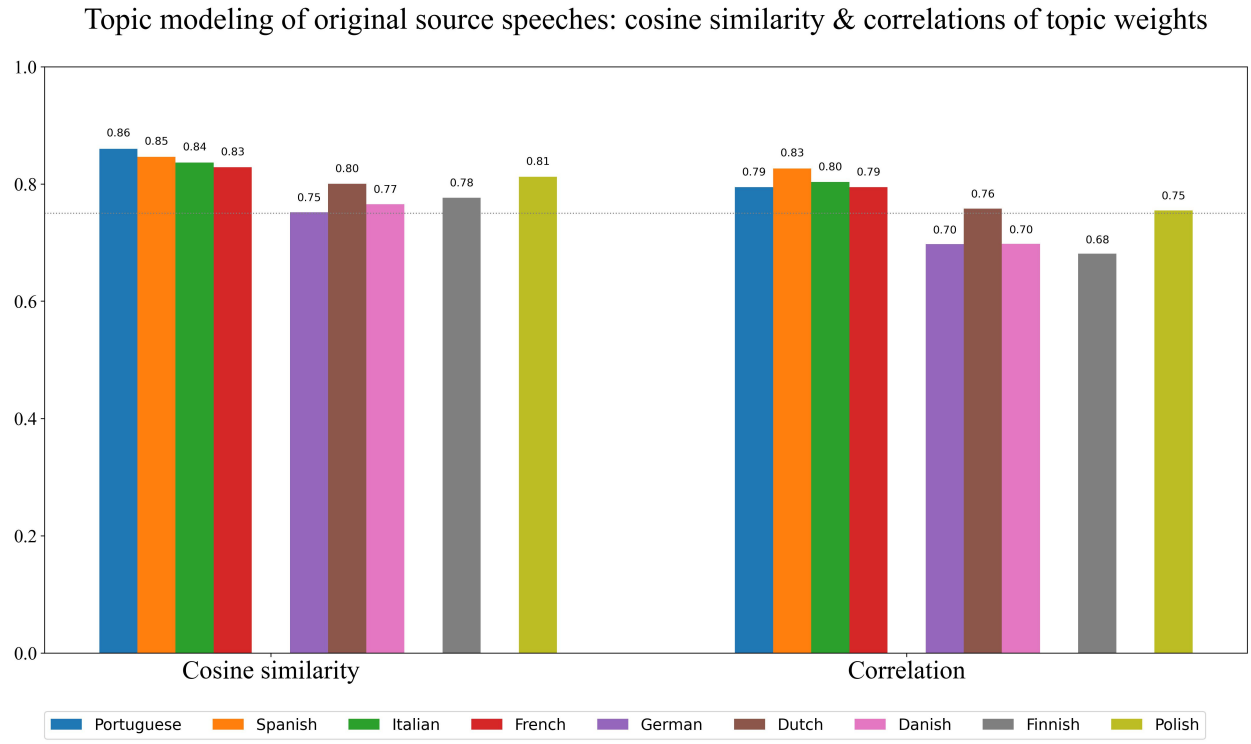

Figure 4. Topic modeling performance, source language speeches

It is possible that some topics are particularly associated with specific languages. For example, the Polish fishing sector is rather smaller than is the case for some of the other countries included here. To verify that the results shown in figure 4 are not driven by language-specific topics, figure 5 displays the analogous results for a topic model run on the pooled *Fullspeeches* corpus. For this corpus, the optimal number of topics was 16, of which 8 are substantive, and are listed in table 6 (a full listing, including non-substantive topics appears in the appendix). Figure 5 shows the associated performance levels.

Topic	**Top 6 words**
Economy	economic, social, crisis, financial, development, growth
Rights	rights, human, country, international, freedom, democracy
EU institutions & activities	commission, council, parliamentx, day, order, proposal
Voting	report, voted, favor, written, vote, eu
Voting & debating	voting, article, debates, closed, 142, 149
Women & children	women, men, equality, violence, children, social
Products	products, consumers, market, protection, health, agricultural
Energy	energy, nuclear, climate, efficiency, emissions, renewables

Table 6. Substantive topics in *Fullspeeches* corpus.

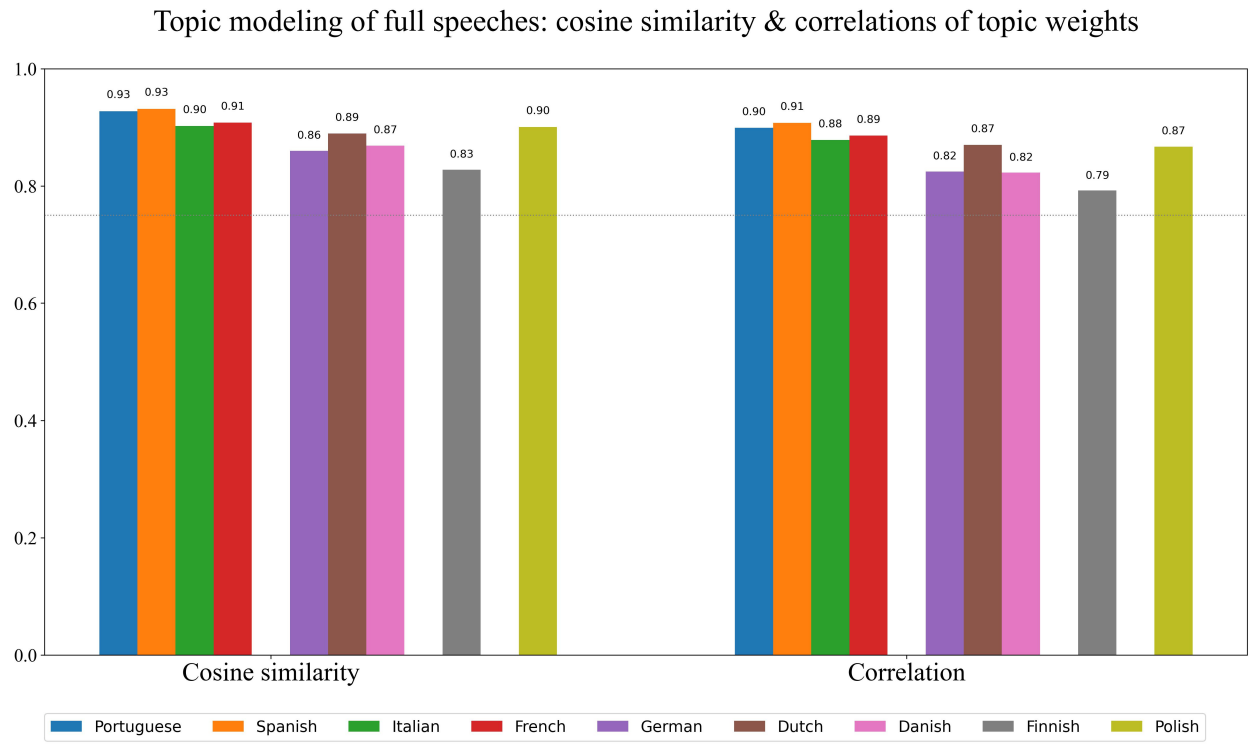

Figure 5. Topic modeling performance, full speeches

The match between the human and the machine translations is notably higher here, both overall and at the level of individual topic weights. Some of the increased performance is likely due to the less fine-grained topic demarcations (16 topics total, as compared to 25 for the *Sourcespeeches* corpus). Nonetheless, the level of agreement between the two sets of translations is very high, eliminating concerns that the results in figure 4 were biased upwards by the presence of language-specific topics.

Conclusion & future extensions

When resources and time permit, there is no doubt that neural machine translation is to be preferred over word-level translation. However, resource and time often do not permit, and in those cases, the availability of high-quality word-level translation dictionaries can be invaluable. In addition, scholars may wish to run preliminary analyses on multilingual corpora before deciding whether to invest the necessary time or money. Others still may have laboriously developed a particular set of coding categories (for instance in the form of dictionaries) which have been validated in English, and may want to compare the scores for English-language texts on those categories to those of texts originally in other languages.[4]

[4] Note that translating a dictionary itself into another language may be an inferior substitute: as noted, it is generally easier to translate correctly into than out of English.

The present paper makes two important contributions on this front. First it outlines a method to automatically generate high-quality word-level translation dictionaries from aligned word embeddings. The dictionaries used in this paper are available online and can be freely downloaded. In addition, they are regularly updated and expanded. Second, the paper performs a number of comparisons between professional human translation and word-level translation using the dictionaries introduced here, which demonstrate that the latter produce comparable results on standard text analysis tasks. The comparisons presented here should assuage fears that the ungainliness of word-level translations implies they must also produce poorer or incompatible results when used as inputs for automated text analyses.

Automated translation steadily continues to improve in quality, but the highest quality translations will continue to require money, resources, or time for the foreseeable future. Moreover, for many of the analyses social scientists might want to conduct, it is not obvious that incremental improvements in translation quality make much of a difference for the eventual findings. Accordingly, there will continue to be a need for high-quality word-level translation. Hopefully the dictionaries and method introduced here will help address that need.

References

Bojanowski, P., Grave, E., Joulin, A., & Mikolov, T. (2016). *Enriching Word Vectors with Subword Information*. https://arxiv.org/abs/1607.04606

Conneau, A., Lample, G., Ranzato, M., Denoyer, L., & Jégou, H. (2017). Word Translation Without Parallel Data. *ArXiv:1710.04087 [Cs]*. http://arxiv.org/abs/1710.04087

de Vries, E., Schoonvelde, M., & Schumacher, G. (2018). No Longer Lost in Translation: Evidence that Google Translate Works for Comparative Bag-of-Words Text Applications. *Political Analysis*, *26*(4), 417–430. https://doi.org/10.1017/pan.2018.26

Graham, Y., Haddow, B., & Koehn, P. (2019). Translationese in Machine Translation Evaluation. *ArXiv:1906.09833 [Cs]*. http://arxiv.org/abs/1906.09833

Greene, D., & Cross, J. P. (2017). Exploring the Political Agenda of the European Parliament Using a Dynamic Topic Modeling Approach. *Political Analysis*, *25*(1), 77–94. https://doi.org/10.1017/pan.2016.7

Joulin, A., Bojanowski, P., Mikolov, T., Jegou, H., & Grave, E. (2018). Loss in Translation: Learning Bilingual Word Mapping with a Retrieval Criterion. *ArXiv:1804.07745 [Cs]*. http://arxiv.org/abs/1804.07745

Junczys-Dowmunt, M., Grundkiewicz, R., Dwojak, T., Hoang, H., Heafield, K., Neckermann, T., Seide, F., Germann, U., Aji, A. F., Bogoychev, N., Martins, A. F. T., & Birch, A. (2018). Marian: Fast Neural Machine Translation in C++. *ArXiv:1804.00344 [Cs]*. http://arxiv.org/abs/1804.00344

Klein, G., Hernandez, F., Nguyen, V., & Senellart, J. (2020). *The OpenNMT Neural Machine Translation Toolkit: 2020 Edition*. *1*, 8.

Klein, G., Kim, Y., Deng, Y., Nguyen, V., Senellart, J., & Rush, A. M. (2018). OpenNMT: Neural Machine Translation Toolkit. *ArXiv:1805.11462 [Cs]*. http://arxiv.org/abs/1805.11462

Lee, D. D., & Seung, H. S. (1999). Learning the parts of objects by non-negative matrix factorization. *Nature*, *401*(6755), 788–791. https://doi.org/10.1038/44565

Long, Z., Utsuro, T., Mitsuhashi, T., & Yamamoto, M. (2017). Translation of Patent Sentences with a Large Vocabulary of Technical Terms Using Neural Machine Translation. *ArXiv:1704.04521 [Cs]*. http://arxiv.org/abs/1704.04521

Plutchik, R. (2001). The Nature of Emotions: Human emotions have deep evolutionary roots, a fact that may explain their complexity and provide tools for clinical practice. *American Scientist*, *89*(4), 344–350.

Radovanovic, M., Nanopoulos, A., & Ivanovic, M. (2010). Hubs in Space: Popular Nearest Neighbors in High-Dimensional Data. *Joournal of Machine Learning Research*, *11*, 2487–2531.

Schnabel, T., Labutov, I., Mimno, D., & Joachims, T. (2015). Evaluation methods for unsupervised word embeddings. *Proceedings of the 2015 Conference on Empirical Methods in Natural Language Processing*, 298–307. https://doi.org/10.18653/v1/D15-1036

Sennrich, R., Haddow, B., & Birch, A. (2015). Neural Machine Translation of Rare Words with Subword Units. *ArXiv:1508.07909 [Cs]*. http://arxiv.org/abs/1508.07909

Speer, R., Chin, J., Lin, A., Jewett, S., & Nathan, L. (2018). *Wordfreq v2.2 (by Luminoso Insight)*. Zenodo. https://doi.org/10.5281/zenodo.1443582

van der Veen, A. M., & Bleich, E. (2021). *Automated sentiment analysis for the social sciences: A domain-independent, lexicon-based approach*.