Proceedings of the 12th Conference on Language Resources and Evaluation (LREC 2020), pages 4190–4196
Marseille, 11–16 May 2020

Multilingual Corpus Creation for Multilingual Semantic Similarity Task

Mahtab Ahmed[1], Chahna Dixit[2], Robert E. Mercer[1], Atif Khan[2], Muhammad Rifayat Samee[1], Felipe Urra[1]
[1]University of Western Ontario, [2]Messagepoint Inc.
mahme255@uwo.ca, chahna.dixit@messagepoint.ai, mercer@csd.uwo.ca, atif.khan@messagepoint.ai, msamee@uwo.ca, furra@uwo.ca

Abstract

In natural language processing, the performance of a semantic similarity task relies heavily on the availability of a large corpus. Various monolingual corpora are available (mainly English); but multilingual resources are very limited. In this work, we describe a semi-automated framework to create a multilingual corpus which can be used for the multilingual semantic similarity task. The similar sentence pairs are obtained by crawling bilingual websites, whereas the dissimilar sentence pairs are selected by applying topic modeling and an Open-AI GPT model on the similar sentence pairs. We focus on websites in the government, insurance, and banking domains to collect English-French and English-Spanish sentence pairs; however, this corpus creation approach can be applied to any other industry vertical provided that a bilingual website exists. We also show experimental results for multilingual semantic similarity to verify the quality of the corpus and demonstrate its usage.

Keywords: Multilingual Corpus, Multilingual Semantic Similarity, Web Crawling

1. Introduction

Semantic similarity, one of the important natural language processing (NLP) tasks, aims to measure the distance between two given content pieces in terms of their meaning. Traditionally, WordNet-based similarity measures such as Lin, Resnik, Jiang and Conrath (Budanitsky and Hirst, 2006) as well as statistical approaches including Latent Semantic Analysis (LSA) (Landauer et al., 2013) and Pointwise Mutual Information (PMI) (Zhao et al., 2014) have been used to solve this problem. Recently with the advent of deep learning, the use of deep neural networks has gained popularity in solving this task; for example Siamese recurrent networks (Mueller and Thyagarajan, 2016) and convolutional neural networks (Shao, 2017). However, a major factor affecting the success of deep networks is the availability of substantially large and good quality corpora (Kiros et al., 2015; Devlin et al., 2019).

The most popular benchmark dataset for semantic similarity is the Semantic Textual Similarity (STS) dataset from SemEval tasks. The latest STS17 dataset (Cer et al., 2017) includes monolingual as well as cross-lingual sentence pairs for English, Arabic and Spanish languages. Nonetheless, the STS corpus requires a classification score ranging from 0 to 5 measuring the degree of similarity between the sentence pairs. We approach multilingual semantic similarity as a binary classification problem which has required us to collect a large corpus of our own based on the domain and language requirements of our application.

The collection of an entirely new and large corpus in itself is a challenging task; more specifically, textual data for NLP problems require human expertise and domain knowledge of the application. Above all, the acquisition of a multilingual corpus also demands some amount of linguistic knowledge. This leads to an increasing interest in developing an automated or semi-automated approach for building a multilingual corpus. Several corpus creation approaches have been published focusing on multiple languages and application domains. Papavassiliou et al. (2018) proposed a web crawler to acquire parallel language resources for European languages. Soares et al. (2018) developed a parallel corpus of scientific articles in English, Portuguese and Spanish languages by first acquiring documents from the Scielo database (Packer, 2009) and then aligning sentences from document pairs of different languages. Few other approaches exist, but in all of these works the generated corpus is meant to be utilized for machine translation. Apart from this, existing techniques focus on curating similar sentence pairs, but rarely talk about dissimilar sentence pair generation.

Bilingual sentence alignment lies at the heart of collecting similar pairs for a multilingual corpus. Maligna, a bilingual sentence alignment tool (Jassem and Lipski, 2008) does this by using statistical machine translation and a few sentence alignment algorithms to align sentences from document pairs. The tool is mainly used to align text for a machine translation dataset. While the corpus for machine translation requires perfect alignment among the sentence pairs, this is not true for the semantic similarity task since we are not looking for an exact translation of a sentence with another.

Considering all of the aspects discussed above for multilingual corpus creation specific to the semantic similarity problem, in this work, we describe a semi-automated approach to build a large corpus of English-French and English-Spanish sentence pairs that can be used for the multilingual semantic similarity task. The approach is based on scraping documents from bilingual websites and aligning the document pairs at the sentence and/or paragraph level. We have considered websites from government, insurance, and banking domains; but the advantage of this approach is that it can be applied for any language and domain or industry that has a website with bilingual content. We also plan to open-source the collected multilingual corpus for use by other researchers.

2. Architecture

Multilingual semantic similarity as a binary classification task requires a dataset consisting of bilingual sentence pairs labelled as either semantically similar or dissimilar. For simplicity, we will term the similar sentence pairs as positive samples and dissimilar pairs as negative samples. In this section, we provide detailed information on collecting the positive and negative samples for the corpus. The positive sample selection is a semi-automated approach that involves crawling multiple websites followed by bilingual sentence alignment along with an additional filtering process. It is to be noted that, these positive sample pairs are obtained from the same webpage of two different languages. On the other hand, the hypothesis for negative samples is that the sentence pairs should have similar topics determined by some automatic means but talk about a different aspect of this topic. Hence, the negative sentence pairs are formed by sampling from different webpages having a similar topic. To do this, we utilize a well-known topic modelling algorithm, LDA (Blei et al., 2003) and a sentence representation model, Open-AI GPT (Radford et al., 2018) on top of the positive samples. Table 1 shows some examples of positive and negative sentence pairs.

Sentence pairs	Label
We will get back to you by next week	Positive
We will contact you soon	
We will get back to you by next week	Positive
Nous vous contacterons la semaine prochaine	
You must pay your taxes	Negative
Ontario has high tax rate	

Table 1: Positive and Negative Sentence Pair Examples

2.1. Positive Sample Selection

The positive sample selection approach consists of four main steps: data crawling, HTML parsing, text translation and text alignment. Each of these steps is explained in detail in the following subsections.

2.1.1. Data Crawling

In the first step we give the base URL of a bilingual (or multilingual) website of interest as input to a web crawler built using a Python library called *Scrapy* (Kouzis-Loukas, 2016). *Scrapy* is a fast high-level framework that crawls websites to extract structured data. The web crawler finds all the URLs from a given webpage URL and crawls each of those URLs recursively to extract the data. This process goes on in an iterative fashion where the input to a particular iteration is the list of URLs obtained from the previous iteration. We run the crawler for as long as no new webpages are being crawled.

For a given webpage URL, the key point is finding the corresponding parallel webpage URL in the counterpart language. This can be searched by finding a pattern in the HTML code of several webpages of that website. The crawler outputs several HTML files where each HTML file corresponds to a single webpage. In the end, these HTML files undergo post-processing to delete the webpages that do not have a parallel webpage in the counterpart language. We denote the parallel sets of HTML files as H_{l1} (HTML files for language $l1$) and H_{l2} (parallel HTML files for counterpart language $l2$), where H_{l1} and H_{l2} have an equal number of files after post-processing.

2.1.2. HTML Parsing

We use the Python library *inscriptis* (Weichselbraun et al., 2016) to extract all of the text content from the HTML files. The extracted text is then split into lines where each line can be a word, a sentence or a paragraph. We then discard those lines that do not contain at least one alphabetic character as well as the lines containing just one word. Next, a text file is generated for each HTML file which contains the parsed and clean raw text from that HTML. In order to build the corpus, we retain only those pairs of parallel text files that have equal numbers of lines. The reason for this step is that our text alignment approach is based on the line order of the file. More details on the alignment process can be found in Subsection 2.1.4.. We term T_{l1} as the set of text files corresponding to H_{l1} and T_{l2} as the set of text files from H_{l2}, where T_{l1} and T_{l2} have an equal number of files but may not be the same as the number of files in H_{l1} and H_{l2} due to the refinement process.

The HTML tags in the parallel HTML files can also be leveraged to extract more text content and remove additional noise of headers, footers, titles, etc. from the webpage. We experimented with extracting text content based on the *class* attribute of parallel * * tags which helped to retain more files when applying post-processing based on the length of text files.

2.1.3. Text Translation

Our text alignment approach works on parallel text files in the same language. So, any one set from the parallel set of text files must be translated to another language. We translate all of the text files in any other language into English using the Python library mtranslate (Aliès, 2016) which implements the Google Translate API. Now, if $l1$ represents the English language and $l2$ represents any language other than English, then each text file t^k_{l1} in T_{l1} will correspond to two parallel files – t^k_{l2} from T_{l2} in the counterpart language and $t^k_{l2'}$ from $T_{l2'}$ which consists of text files from T_{l2} translated into English. Here, k represents the k^{th} file in the set of text files.

2.1.4. Text Alignment

The alignment of text is the most important step in the framework of selecting positive samples. The approach is based on the hypothesis that the contents of two parallel bilingual webpages appear in somewhat the same order. So most of the parallel text files should be aligned; however, there will be exceptions in some cases. Hence, we devise the text alignment approach in such a way that the alignment check for a particular pair of text files is line-based and one-to-one. This means that a line at a given position in t^k_{l1} is checked against a single line at the same position in $t^k_{l2'}$.

We use word frequency-based cosine distance as a distance measure between each line in the files t^k_{l1} and $t^k_{l2'}$. This

basic measure seems to work well in aligning semantically similar content pairs. The positions (or indices) of line pairs with cosine distance greater than 0.6 are recorded as being misaligned. The set of indices for misaligned lines I^k is refined further such that if a particular index in the set does not have a consecutive misaligned line, then that index is discarded from I^k. The intuition behind this step is that the translation may have affected the cosine distance to record it as misaligned.

Finally, the files having empty I^k are considered to be aligned and the remaining files are aligned manually based on the indices in I^k. The manual alignment of files involves rearranging certain lines or discarding lines that do not have a match in the corresponding parallel file. The manual alignment is only done for the files that have the number of misaligned lines under a certain threshold. The aligned set of parallel text files containing semantically similar positive sample pairs are denoted as P_{l1} and P_{l2}. Here, $p^k_{l1} \in P_{l1}$ and $p^k_{l2} \in P_{l2}$ are the k^{th} parallel files obtained after alignment of files t^k_{l1} and t^k_{l2}. Each corresponding line pair from the parallel files is considered as a positive sample pair.

2.2. Negative Sample Selection

In this subsection we explain our negative sample selection approach which is applied over each language pair and domain individually. For the three domains i.e., government, insurance, and banking, we divide our aligned parallel files into three sets: English-French Government **EN-FR-G**, English-French Insurance Banking **EN-FR-IB**, and English-Spanish Insurance Banking **EN-ES-IB**. A detailed description of these partitions is given in Section 3..

Our negative sample selection approach starts by training a range of unsupervised LDA (Blei et al., 2003) models on P_{l1} where $l1$ is constrained to be the English language. Each file p^k_{l1} in P_{l1} is considered as a single document D. The LDA model with the maximum coherence score is chosen as the best topic model. Fig. 1 shows the coherence score vs. number of topics plot for **EN-FR-G**, **EN-FR-IB** and **EN-ES-IB** where the optimal number of topics are 74, 17 and 41, respectively. We use the following parameters for training LDA: *random_state*=100, *update_every*=1, *chunksize*=100, *passes*=300, *alpha*=$auto$, *per_word_topics*=$True$.

Using the best LDA model, we represent each English document as a document vector which is a probability distribution over all the topics. Then, for an input sentence S, its dominant topic T is obtained according to this topic distribution. Next, we use a pretrained OpenAI-GPT model (Radford et al., 2018) to get the vector representation $M(s)$, where s represents sentences from all of the English documents. We then extract a set of documents A having the same topic T, and collect the sentences (and their multilingual counterparts) having cosine similarity with input sentence S in the range 0.8-0.9. The cosine similarity between the two sentences is calculated from the vector representations $M(.)$ of those sentences. Based on our definition of the negative samples, we choose the similarity threshold range to be 0.8-0.9; which gives us samples that belong to a similar topic. However, this threshold range can be adjusted based on the application requirements.

Algorithm 1: Negative sample selection with LDA-LM

Pretrained LDA model: L
Pretrained OpenAI-GPT model: M
Input English document: D
Topic of D according to L: T
Set of negative samples: N
List of documents with same topic as T: A
Number of sentences to be selected: n
Input sentence from document D: S
$N \leftarrow \emptyset$
for $i \leftarrow 0$ **to** n **by** 1 **do**
  $x \leftarrow$ NULL
  foreach *document* d **in** A **do**
    foreach *sentence* s **in** *document* d **do**
      if $0.80 < Cosine(M(S), M(s)) < 0.90$ **then**
        $x \leftarrow s$
        break
      end
    end
    if $x \neq NULL$ **then**
      break
    end
  end
  $N \leftarrow$ multilingual counterpart of x
end

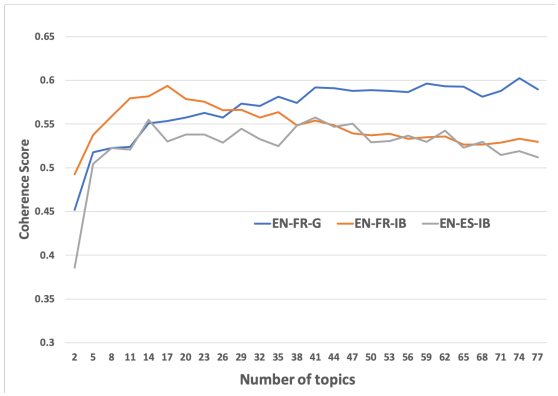

Figure 1: Topic coherence score vs number of topics.

Following the above mentioned steps, we select n sentences for each S and then create n negative sentence pairs in the multilingual space by pairing S with the appropriate multilingual counterparts of each of these sentences. Algorithm 1 makes precise the above steps. For our experiments we choose the value of n to be 10. This yields a sufficient number of negative samples for the corpus. However, not all the samples are used to build the corpus. In the end, the negative sentence pairs are sampled in order to create a balanced dataset with respect to the total number of positive sentence pairs.

3. Corpus Details

We scraped 11,156 bilingual webpages pairs in total, out of which approximately 9.25% were discarded based on the filtering and alignment process described in Subsections 2.1.2. and 2.1.4.. Hence, 10,124 text file pairs were used to create the positive and negative sentence pairs.

The final corpus consists of 351,334 English-French (EN-FR) sentence pairs and 53,826 English-Spanish (EN-ES)

Language	Sentence 1	Sentence 2	Label
EN-FR	There are minimum and maximum permissible withdrawals from the plan each year.	Les retraits sont soumis à des minimums et à des maximums annuels admissibles.	Positive
	At the eye of a hurricane there is a clam area of blue sky.	Dans l'oeil d'un ouragan, il y a une zone calme de beau temps.	Positive
	Guide T4002, Business and Professional Income	Formulaire T4A, État du revenu de pension, de retraite, de rente ou d'autres sources	Negative
	What can be deducted from an employee's pay cheque?	Quand l'employeur doit-il verser l'indemnité de congé annuel?	Negative
EN-ES	To focus on the love and fun a pet can bring, instead of the extra cost, all pet parents should consider purchasing pet insurance from a reputable, caring company.	Con el fin de enfocarse en el amor y la alegría que una mascota puede ofrecer, y no en los costos adicionales, todos los "papás" de mascotas deberían pensar en comprar un seguro de mascotas de una compañía de reputación que se preocupe por sus clientes.	Positive
	Don't stress if you lose track of your phone—all mobile wallet transactions require the verification you set up, like a fingerprint scan.	No se estrese si pierde su teléfono, todas las transacciones de la billetera móvil requieren la verificación que usted haya establecido, como una huella digital.	Positive
	Use these tips to get your bike in top shape for the new riding season.	Si los carros deportivos son una pérdida total o son robados, normalmente cuesta más reemplazarlos.	Negative
	All coverages are subject to the terms, provisions, exclusions, and conditions in the policy itself and in any endorsements.	Ésta es sólo una descripción general de las coberturas de los tipos de seguros disponibles y no representa una declaración de contrato.	Negative

Table 2: Examples from collected multilingual corpus

Dataset	Train	Validation	Test
EN-FR-G	195,303	48,826	61,033
EN-FR-IB	29,546	7,389	9,237
EN-ES-IB	34,447	8,613	10,766

Table 3: Sentence pair counts for dataset partitions

sentence pairs summing up to a total of 405,160 sentence pairs with 202,580 sentence pairs for each class – positive and negative. Some examples from the collected corpus are shown in Table 2.

As mentioned before, we created our multilingual corpus by scraping from 5 different bilingual websites. Three had content in English and French, while the remaining two were in English and Spanish. The topics of these website contents were government, insurance, and banking. In order to do an extensive evaluation, we divided the entire corpus based on language pairs and domain. The government domain was only available for English-French pair, so we created one dataset – **EN-FR-G**. The insurance and banking domains were available for both English-French and English-Spanish pairs. So we created two more datasets using these – **EN-FR-IB** and **EN-ES-IB**.

For each of the three datasets, we used the positive and negative sample pairs, described earlier, to create a balanced 10-fold training and validation partition for doing 10-fold cross validation experiments. We also created a test set for testing.

The dataset partition details are given in Table 3. In order to verify the quality of our corpus we have evaluated our model on a benchmark dataset that has been supplemented with our corpus. We chose the well known Microsoft Research Paraphrase Corpus (MSRP) where the task is to do paraphrase identification (Dolan et al., 2004). Because of the way we prepared our corpus, it aligns well with this kind of task. The original MSRP dataset has $5,801$ sentence pairs, $4,076$ in the training set and $1,725$ in the test set. Adding our corpus to the MSRP training set shows an increase in performance on the MSRP test set. We hypothesize that this indicates that our corpus is of good quality.

4. Evaluation Experiments

In this section, we present a thorough analysis of all the evaluation experiments that we did to validate our corpus. We first describe the model architecture which we used to solve the multilingual semantic similarity task. Following this, we explain the training details of the model along with its hyper-parameter settings. We also present the detailed results obtained with our experiments and compare the transfer performance of our selected model with some of the top performing models on the MSRP dataset.

4.1. Model Architecture, Parameters and Training Details

We have chosen to use InferSent (Conneau et al., 2017a), an LSTM based model, to compute the representations of a pair of sentences, a and b, and then compare the representations for an underlying task. The model first traverses each sentence as a sequence of T words $\{x_t\}_{t=1,\ldots,T}$ from both left to right and right to left and generates two hidden representations at each time step $\overrightarrow{h_t}, \overleftarrow{h_t}$ $\forall t \in [1, \ldots, T]$. During input, it considers the vector representation of each word (x_t) in the sentence from a pre-trained word embed-

Model	Validation set Accuracy		
	Mean	Max voting	Avg. voting
EN-FR	95.41 ± 0.39	95.76	95.97
EN-ES	96.35 ± 0.57	97.07	97.22
EN-FR-ES	95.03 ± 0.24	95.40	95.59

Table 4: 10-fold cross validation performance of different models (mean includes standard deviation)

Model	Test set accuracy		
	en-fr	en-es	en-en (MSRP)
EN-FR	95.64	95.91	76.05
EN-ES	87.15	97.25	75.07
EN-FR-ES	94.91	98.34	76.00

Table 5: Cross corpus performance (Accuracy). Rows indicate training language pairs, and columns indicate testing language pairs

ding model.

$$\begin{aligned} \overrightarrow{h_t} &= \overrightarrow{\text{LSTM}_t}(x_1, \ldots, x_T) \\ \overleftarrow{h_t} &= \overleftarrow{\text{LSTM}_t}(x_1, \ldots, x_T) \\ h_t &= [\overrightarrow{h_t}, \overleftarrow{h_t}] \end{aligned} \tag{1}$$

After this, the model employs a max (or mean) pooling block to summarize the hidden states in one dense representation.

$$h = \text{maxpool}(h_1, \ldots, h_T) \tag{2}$$

The next steps are to infer the similarity between the two representations (h_a, h_b) using standard matching methods and to project the resultant vector into the space of classes y (which is two in our case) through a series of fully connected layers as follows

$$x = (h_a, h_b, |h_a - h_b|, h_a * h_b) \tag{3}$$

$$P(y|\mathbf{X}) = \sigma(\mathbf{W}_1\sigma(\mathbf{W}_2 x + \mathbf{b}_2) + \mathbf{b}_1) \tag{4}$$

Finally, the model is trained by optimizing a task specific loss function as follows

$$H(p, q) = -\sum_{i=1}^{n} Q(y_i) \log(P(y_i)) \tag{5}$$

The LSTM hidden state dimension is set to 600. Multilingual word vectors are initialized with the 300 dimension MUSE embeddings (Conneau et al., 2017b) and are not updated during training. To smooth the update, the gradients are divided by B^2 where B is the batch size which is set to 512. The learning rate is reduced by a factor of 2 if $\sqrt{\sum_{i=1}^{k} \|\nabla\theta_i^2\|}$ is more than a threshold, which is 5 for our experiments. We use Adam as the optimization algorithm and the dropout in the classification layer is set to 0.5. The number of topics parameter is described in Subsection 2.2..

4.2. Results and Analysis

We train three different models with different combinations of training data depending on the language pairs and domain. The models **EN-FR** and **EN-ES** use **EN-FR-G** and **EN-ES-IB** datasets respectively, whereas the **EN-FR-ES** model uses all three datasets. Table 4 shows the 10-fold cross validation performance of all of these models. We summarize the performances over all the folds in three different ways. Firstly, we report the mean accuracies along with the standard deviation over all the folds for all three models. Following this, we report the results of the ensemble experiment where we use max voting and average voting as our ensemble methods. It can be seen that the average voting achieves the better performance among all of these methods getting 95.97%, 97.22% and 95.95% accuracy for **EN-FR**, **EN-ES** and **EN-FR-ES**, respectively.

Table 5 reports the cross corpus performance of the three models (models are in uppercase and datasets are in lowercase). We have not included **en-fr-es** for test purposes because samples in this dataset are taken from the English-French (**en-fr**) and English-Spanish (**en-es**) pairs. The performance scores are reported over the test set that we create for each of these datasets as shown in Table 3. It is to be noted that the models **EN-FR**, **EN-ES** and **EN-FR-ES** are trained on the English-French (**en-fr**), English-Spanish (**en-es**) and English-French-Spanish (**en-fr-es**) datasets, respectively. We have chosen to use the best model on each of these datasets out of the 10 models that we create during the 10-fold cross validation. It can be seen that the **EN-FR** model shows very good performance over **en-es** (95.91%) and it is doing even better than its own test set (95.64%). When tested on **en-fr**, the performance of the **EN-ES** model drops with respect to **en-es** being the test set; the relatively smaller size of training data for **EN-ES** as compared to **EN-FR** can be one of the reasons for this performance drop. When trained on **en-fr-es** the performance of **EN-FR-ES** on the two language pairs compare well. We also report the performance of the models when tested on MSRP which is a **en-en** corpus. We believe that this good cross corpus performance is because the word embeddings are aligned in the same semantic space.

Model	Accuracy
InferSent (Conneau et al., 2017a)	74.46
LSTM (Conneau et al., 2017a)	70.74
BiGRU Last Encoder (Conneau et al., 2017a)	70.46
Tree LSTM (Tai et al., 2015)	73.50
ConvNet Encoder (Zhao et al., 2015)	73.96
Ours (Transfer + Finetuning)	**76.05**

Table 6: Performance comparison on the MSRP dataset against some existing top performing models.

Table 6 reports the performance of the model on the MSRP task compared to some of the existing top performing models. As we can see, InferSent trained on the MSRP training set from scratch yields an accuracy of 74.46% (Conneau et al., 2017a), whereas transferring the weights from the model pretrained on our dataset gives an accuracy of 76.05%. It is to be noted that we are also doing better than Tree LSTM (Tai et al., 2015) (73.50% accuracy) which uses additional parse information and ConvNet Encoder (73.96%) which uses a complex and expensive convolution operation over multiple channels.

Table 7 shows the models' predictions on a few examples

Dataset	Sentence 1	Sentence 2	GT	Pr
EN-FR	The Cannabis Act proposes many rules that would protect youth from accessing cannabis.	Le projet de loi sur le cannabis prévoit de nombreuses dispositions pour empêcher les jeunes d'avoir accès au cannabis.	1	1
	The authorized health care practitioner's licence information	Numéro de téléphone et adresse électronique de la personne morale	0	0
	What can be deducted from an employee's pay cheque?	Quand l'employeur doit-il verser l'indemnité de congé annuel?	0	0
EN-ES	Use window sheet kits	Usa kits de aislamiento para ventanas	1	1
	It will only take a minute and won't impact your credit score	Díganos quién es y qué le gusta, para ver qué ofertas están	0	0
	List out your debt	Fíjate un presupuesto semanal, empezando el lunes	0	0

Table 7: Example predictions from the test set. **GT**: ground truth, **Pr**: predicted.

from our dataset. It can be seen in **EN-FR** that the two negative sentences talk about the same topics as their counterpart English sentences, but the contents differ. In **EN-ES**'s second pair, the English sentence talks about a credit score while the Spanish sentence talks about some offers which are somehow related. Here in the third pair, the English sentence talks about debt whereas the Spanish sentence talks about budget, which are not exactly related but somehow gets used in the same context. This justifies our hypothesis of choosing topic related negative examples.

Our discussion on the experimental results shows that the semantic similarity models trained using the collected multilingual corpus perform well across different languages and domains. It is important to understand that like any other curated dataset, this corpus may have some amount of noise in terms of alignment. However, the results of transfer learning on the MSRP benchmark dataset verifies the quality of the dataset.

5. Conclusion

In this paper, we develop a multilingual corpus for doing the multilingual semantic similarity task. We investigate this similarity problem as a binary classification task. To obtain the positive examples, we adopt web crawling of bilingual sentence pairs followed by a set of careful preprocessing steps to align them. We focus on websites in the government, insurance, and banking domain to collect English-French and English-Spanish sentence pairs. To create the bilingual sentence pairs of the negative class, we propose an algorithm utilizing LDA and OpenAI-GPT. Using this algorithm, we can create synthetic non-similar bilingual sentence pairs, where the participating entities talk about the same topic with some differing content. Our corpus creation approach can be applied to any other industry vertical provided that a bilingual website exists. To evaluate the quality of the corpus, we create a pre-trained multilingual version of InferSent and show that we obtain better transfer learning performance over a well known public dataset – MSRP.

6. Acknowledgements

This research was supported by Mitacs through the Mitacs Accelerate program. We also acknowledge the helpful comments provided by the reviewers.

7. Bibliographical References

Aliès, A. (2016). mtranslate. https://github.com/mouuff/mtranslate.

Blei, D. M., Ng, A. Y., and Jordan, M. I. (2003). Latent Dirichlet allocation. *Journal of Machine Learning Research*, 3:993–1022.

Budanitsky, A. and Hirst, G. (2006). Evaluating WordNet-based measures of lexical semantic relatedness. *Computational Linguistics*, 32(1):13–47.

Cer, D., Diab, M., Agirre, E., Lopez-Gazpio, I., and Specia, L. (2017). SemEval-2017 task 1: Semantic textual similarity multilingual and crosslingual focused evaluation. In *Proceedings of the 11th International Workshop on Semantic Evaluation (SemEval-2017)*, pages 1–14.

Conneau, A., Kiela, D., Schwenk, H., Barrault, L., and Bordes, A. (2017a). Supervised learning of universal sentence representations from natural language inference data. In *Proceedings of the Conference on Empirical Methods in Natural Language Processing*, pages 670–680.

Conneau, A., Lample, G., Ranzato, M., Denoyer, L., and Jégou, H. (2017b). Word translation without parallel data. *arXiv preprint arXiv:1710.04087*.

Devlin, J., Chang, M.-W., Lee, K., and Toutanova, K. (2019). Bert: Pre-training of deep bidirectional transformers for language understanding. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)*, pages 4171–4186.

Dolan, B., Quirk, C., and Brockett, C. (2004). Unsupervised construction of large paraphrase corpora: Exploiting massively parallel news sources. In *Proceedings of the 20th International Conference on Computational Linguistics*, pages 350–356.

Jassem, K. and Lipski, J. (2008). A new tool for the bilingual text aligning at the sentence level. In *Proceedings of 16th International Conference on Intelligent Information Systems*, pages 279–286.

Kiros, R., Zhu, Y., Salakhutdinov, R., Zemel, R. S., Torralba, A., Urtasun, R., and Fidler, S. (2015). Skip-thought vectors. In *Proceedings of the 28th International Conference on Neural Information Processing Systems - Volume 2*, pages 3294–3302.

Kouzis-Loukas, D. (2016). *Learning Scrapy*. Packt Publishing Ltd.

Landauer, T. K., McNamara, D. S., Dennis, S., and Kintsch, W. (2013). *Handbook of Latent Semantic Analysis*. Psychology Press.

Mueller, J. and Thyagarajan, A. (2016). Siamese recurrent architectures for learning sentence similarity. In *Proceedings of the Thirtieth AAAI Conference on Artificial Intelligence*, pages 2786–2792.

Packer, A. L. (2009). The SciELO open access: A gold way from the south. *Canadian Journal of Higher Education*, 39(3):111–126.

Papavassiliou, V., Prokopidis, P., and Piperidis, S. (2018). Discovering parallel language resources for training mt engines. In *Proceedings of the Eleventh International Conference on Language Resources and Evaluation (LREC 2018)*, pages 3795–3798.

Radford, A., Narasimhan, K., Salimans, T., and Sutskever, I. (2018). Improving language understanding by generative pre-training. https://s3-us-west-2.amazonaws.com/openai-assets/researchcovers/languageunsupervised/language_understanding_paper.pdf.

Shao, Y. (2017). HCTI at SemEval-2017 task 1: Use convolutional neural network to evaluate semantic textual similarity. In *Proceedings of the 11th International Workshop on Semantic Evaluation (SemEval-2017)*, pages 130–133.

Soares, F., Moreira, V., and Becker, K. (2018). A large parallel corpus of full-text scientific articles. In *Proceedings of the Eleventh International Conference on Language Resources and Evaluation (LREC 2018)*, pages 3459–3463.

Tai, K. S., Socher, R., and Manning, C. D. (2015). Improved semantic representations from tree-structured long short-term memory networks. In *Proceedings of the 53rd Annual Meeting of the Association for Computational Linguistics*, pages 1556–1566.

Weichselbraun, A., Göbel, M., and Odoni, F. (2016). inscriptis. https://github.com/weblyzard/inscriptis.

Zhao, J., Zhu, T., and Lan, M. (2014). ECNU: One stone two birds: Ensemble of heterogenous measures for semantic relatedness and textual entailment. In *Proceedings of the 8th International Workshop on Semantic Evaluation (SemEval 2014)*, pages 271–277.

Zhao, H., Lu, Z., and Poupart, P. (2015). Self-adaptive hierarchical sentence model. In *Proceedings of the Twenty-Fourth International Joint Conference on Artificial Intelligence*, pages 4069–4076.