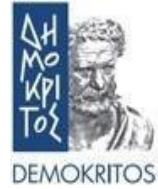




ΠΑΝΕΠΙΣΤΗΜΙΟ ΠΕΙΡΑΙΩΣ
UNIVERSITY OF PIRAEUS



Argumentative Sentence Classification using Transfer Learning Across Languages

by

Panagiotis Tamvakidis

Submitted
in partial fulfilment of the requirements for the degree of Master of
Artificial Intelligence
at the
UNIVERSITY OF PIRAEUS

Athens, July 2021

Argumentative Sentence Classification using Transfer Learning Across Languages

Panagiotis Tamvakidis

MSc. Thesis, MSc. Programme in Artificial Intelligence

University of Piraeus, NCSR “Demokritos”, July 2021

Copyright © 2021 Panagiotis Tamvakidis. All Rights Reserved.

Author: Panagiotis Tamvakidis

II-MSc “Artificial Intelligence”

Athens, July 2021

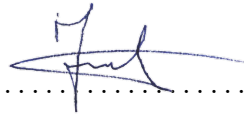
Approved by the examination committee

(Signature)



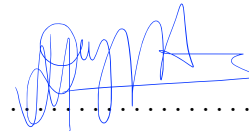
.....
Georgios Petasis
Researcher

(Signature)



.....
Georgios Giannakopoulos
Researcher

(Signature)



.....
Maria Dagioglou
Researcher

**Argumentative Sentence Classification using
Transfer Learning Across Languages**

by

Panagiotis Tamvakidis

Submitted to the II-MSc “Artificial Intelligence” on July 2021
in partial fulfillment of the
requirements for the MSc degree

Acknowledgments

First of all I would like to give special thanks to Mr. Petasis Georgios for his support and contribution. His help was decisive through these academic years. Additionally, I would like to thank Mr. Giannakopoulos Georgios and Ms. Dagioglou Maria for their comments and recommendations.

Also, I would like to give special thanks to my family, my friends Evi, Apostolis, Klainti and our dog Casey who were there across these two years.

To my family.

Περίληψη

Η Μεταφορά μάθησης (“Transfer Learning”) είναι μια πρακτική που χρησιμοποιείται συνήθως για να γίνουν οι εργασίες μηχανικής μάθησης γρηγορότερες και πιο επιτυχημένες. Αυτή η πρακτική μπορεί επίσης να είναι χρήσιμη για ανάλυση κειμένου και τη μηχανική μάθηση. Το “Argument Mining” η αλλιώς ‘Εξορυξη Επιχειρηματολογίας’ είναι μια κατηγορία επεξεργασίας φυσικής γλώσσας που μπορεί να χρησιμοποιηθεί η ‘Μεταφορά μάθησης’ (“Transfer Learning”). Το μεγαλύτερο μέρος της έρευνας και της ανάπτυξης συμβαίνει συνήθως στην αγγλική γλώσσα και αυτό το φαινόμενο μπορεί να βοηθήσει στη λήψη γνώσης από την Αγγλική γλώσσα για να χρησιμοποιηθεί για άλλες γλώσσες σε πρακτικές μηχανικής μάθησης και βαθιάς μάθησης.

Αυτή είναι μια πρακτική που θα χρησιμοποιηθεί για τη σχετική εργασία. Η αναγνώριση επιχειρήματος σε προτάσεις με την εφαρμογή τεχνικών μεταφοράς μάθησης. Μια πρόταση πρόκειται να περιέχει επιχειρήματα όταν ένας ισχυρισμός, προκείμενη η συμπεράσμα είναι επιχειρήματα. Η κύρια ιδέα της μελέτης μας, βασίζετε στα contextual embeddings τα οποία έχουν εκπαιδευτεί στην αγγλική γλώσσα και πρόκειται να ευθυγραμμιστούν με την χρήση παράλληλου dataset με στόχο την δημιουργία ελληνικών embeddings για να κάνουν τις προβλέψεις σε ελληνικές προτάσεις. Αυτή η τεχνική που ονομάζεται “Language Distillation” (Απόσταξη Γλώσσας) [1] και σε αυτή τη σχετική εργασία χρησιμοποιείται με μια ποικιλία από embeddings . Το σύνολο δεδομένων των παράλληλων προτάσεων από τη γλώσσα πηγής (Αγγλικά) και τη γλώσσα στόχο (Ελληνικά) είναι το κύριο όπλο για να γίνει αυτό το είδος της μεταφοράς μάθησης.

Τα σύνολα δεδομένων που χρησιμοποιήθηκαν είναι το Essays corpus στην πρωτότυπη και τη μεταφρασμένη του μορφή στα ελληνικά, καθώς και οι παράλληλες προ-

τάσεις που αναφέρθηκαν από τις ομιλίες TEDex 2020. Η προετοιμασία των δεδομένων ήταν επίσης ένα σημαντικό βήμα προκειμένου να μετατραπούν τα δεδομένα σε μορφή πρότασης με την κλάση επιχειρήματος ή μη επιχειρήματος. Χρησιμοποιήθηκε επίσης πρακτική αύξησης δεδομένων, δεδομένου ότι ο όγκος των κλάσεων δεν ήταν όμοιος. Η προσέγγιση μας βασίζεται στους Transformers[2] και χρησιμοποιεί τα μοντέλα BERT [3], SBERT [4] και XLM-Roberta [5] σε συνδυασμό με μοντέλα βαθιάς μάθησης που παράγει την τελική πρόβλεψη.

Abstract

Transfer learning is one practice that is commonly being used for making machine learning tasks quicker and more successful. This practice can be also useful for text analysis and machine learning. “Argument mining” is one of the natural language processing tasks that “Transfer Learning” can be used. Most of the research and development for machine learning tasks happens in English language and this phenomenon can help for taking that kind of knowledge to use it for other languages in machine learning and deep learning tasks using “Transfer Learning”.

Transfer Learning practices is also going to be used in this work. Making argument identification in sentences by applying transfer learning technics. A sentence is going to be argumentative when contains a claim or premise. The main idea is that the contextual embeddings which have been trained in English language are going to be aligned to the Greek model embeddings in order to make the predictions in Greek sentences. This technique is called Language Distillation [1] and in this related work has been used with a variety of embeddings. Parallel corpus dataset that contains sentences from source language (English) and target language (Greek) is the main weapon in order to make that kind of transfer learning.

Datasets that were used are the Essays corpus in the original and its translated form in Greek as well as the parallel sentences from TEDex 2020 talks. Data preparation was also one important step in order to transform the data into a sentence form with label of argumentative or not. Data augmentation practice was also used since volume of classes was imbalanced. The transformer based approach that took place in that thesis uses BERT [3], SBERT [4] and XLM-Roberta [5] models in relation of a deep learning model in order to produce the final prediction.

Contents

List of Tables	iii
List of Figures	iv
List of Abbreviations	vi
1 Introduction	1
1.1 Problem description	1
1.2 Thesis structure	3
2 Related Work	5
2.1 Transformers	5
2.1.1 Attention Use Cases	8
2.2 Unsupervised Cross-lingual Representation Learning at Scale (XLM Roberta)	9
2.3 Making Monolingual Sentence Embeddings Multilingual using Knowledge Distillation	10
2.4 Sentence Embeddings using Siamese BERT-Networks (Sentence-BERT)	13
2.5 Useful Metrics	15
2.5.1 Euclidian distance	15
2.5.2 Manhattan Distance	15
2.5.3 Pearson Correlation	16
2.5.4 Spearman Correlation	16

3	Proposed Approaches	17
3.1	Datasets and Data Preparation	17
3.1.1	TED 2020 - Parallel Sentences Corpus	18
3.1.2	Essays Dataset and Transformation	18
3.2	Transformers Library Experiment	20
3.3	Sentence Embeddings alignment (Transfer Learning)	22
3.4	Model Training in English and Baseline Prediction in Greek	24
3.5	Final Prediction on Greek Sentences	26
4	Results and Discussion	27
4.1	Experimental Settings Overview	27
4.2	Language Distillation Experiments	28
4.2.1	Language Distillation Results	28
4.3	Transformers Library Results	30
4.4	Model Training with English dataset and Greek Argumentative Prediction	30
4.4.1	Model Training in English dataset	31
4.4.2	Greek Sentence Prediction with non Greek Aligned Embeddings	32
4.4.3	Greek Sentence Prediction with Aligned Language Models	34
5	Conclusions and Future Work	39

List of Tables

3.1	Teacher - Student Models	22
4.1	Results for 8 Epochs of training	29
4.2	Results for 5 Epochs of training	29
4.3	First Small deep learning Model Results on English Dataset (Default Transformer)	31
4.4	Second Bigger deep learning Model Results on English Dataset (Default Transformer)	32
4.5	Small Deep learning Model (Figure 3.4) Results on Greek Dataset - Default Model Prediction	33
4.6	Big Deep learning Model (Figure 3.5) Results on Greek Dataset - Default Model Prediction	33
4.7	Prediction with Small Network and 5 epochs aligned embeddings	34
4.8	Prediction with Bigger Network and 5 epochs aligned embeddings	35
4.9	Prediction with Small Network and 8 epochs aligned embeddings	35
4.10	Prediction with Bigger Network and 8 epochs aligned embeddings	35
4.11	All results of the Greek Sentence Predictions	37

List of Figures

2.1	The Transformer architecture.[2]	6
2.2	Scaled Dot-Product Attention.[2]	7
2.3	Attention Function.[2]	7
2.4	Attention Function.[2]	8
2.5	Attention Function.[2]	8
2.6	Attention Function.[1]	11
3.1	CONLL Structure	19
3.2	Sentences per Class	20
3.3	Tokens Count of Sentences X axis: Tokens of sentences , Y axis : Density	21
3.4	Small Neural Network	25
3.5	Big Neural Network	25
4.1	Transformers Results English Sentences Classifier	30

List of Abbreviations

ML	Machine Learning
BERT	Bidirectional Encoder Representations from Transformers

LIST OF ABBREVIATIONS

Chapter 1

Introduction

1.1 Problem description

The subject we are going to address is called “Argumentative Sentence using Transfer Learning Across Languages”. The goals that we are going to study is the Embedding Alignment Across languages which tries to make two different sets of embeddings that have been created from two languages, similar. The other task, that is going to be the second goal of our study will be to Classify successfully Greek argumentative sentences without any training on Greek datasets. The main objective is to transfer knowledge from the already English pre-trained language models and use them to classify sentences that contain arguments in the Greek language. To achieve such a demanding task we used Transformer’s[2] technology by taking advantage of the embeddings that such a model can produce. These embeddings are getting aligned from English language to Greek by using parallel datasets on these two languages and a method which called language distillation. The two datasets that have been used are the essays dataset [6] which gives all the argument related information for the classification task and the TED-Ex [7] talks dataset which provides all the parallel translated sentences for the embedding alignment (language distillation) task.

Additionally, major role for our work had the “Sentence Transformers”[4]. In general we could say that Sentence Transformers are like a framework that gives

the ability to take the standard pretrained Transformer models and modify them in a sentence manner. By saying that, according to [4] by adding siamese and triplet network structures to the pretrained models we are able to derive semantically meaningful sentence embeddings that can be compared using cosine-similarity. As a result, besides Sentence Bert pretrained model which is the initial “sentence” language model, we can make other “sentence” language models like XLM-Roberta [5] and BERT base [8]. Using that “framework” we are able to compare embeddings of the two languages and take metrics such as euclidian or manhattan distances with robustness. All the experiments that we are examining in our work are taking place with the Multilingual BERT [8], XLM-Roberta [5] and BERT base [8] models. The aforementioned models are used on top of different architectures of deep learning models. Our approach tries to achieve the following:

- Learn a model for argument mining on a language where annotated datasets are available (i.e. English). Using for example the essays corpus, and add-model embeddings, when can learn such a model in a supervised setting.
- Obtain aligned embeddings between the source language (i.e. English), and a target language where no annotated datasets exists (i.e Greek).
- Use the aligned embeddings to encode documents-sentences encoded in the target language (i.e. Greek).
- Apply the argument mining model trained on the source language, on the encoded through aligned embeddings documents, to obtain classification results in the translated essays corpus.

At the end of the experiments, we compare the results between the aligned embeddings and the embeddings that had no fine tuning in the Greek sentence classification task. This comparison results, will provide our final metrics which are going to show how efficient was the embedding alignment transfer learning task. This approach is been analyzed further in Chapter 3 which is the proposed approach part.

1.2 Thesis structure

Regarding the thesis structure, the related work is provided in the next section. It gives an overview of all the research that is been made on relation with the task we are going to work on. This is the Chapter 2 and we give information regarding the Transformers in general, XLM-Roberta model, Language distillation and the Sentence-BERT. Chapter 3 contains the proposed approach and the processing steps involved with its application. How we did the data preparation, what architectures of deep learning model we had, how we did the embedding alignment, what kind of metrics was used and other similar questions in going to answered. Chapter 4 is going to provide all the results of the experiments we made. Last but not least, we are going to have the concluding Chapter 5 and it will contain an overview as well as future work of our experiments.

Chapter 2

Related Work

This particular chapter is going to provide information for research areas such as Deep learning, Transfer Learning, Natural Language Processing, Transformers and Argument mining. These areas are closely related with the current work and further analysis is going to be provided for each of the them in separate sections. Historical context and research experiments are some of the analyses is going to be made.

2.1 Transformers

Transformers[2], are consist of the attention mechanism, dispensing with recurrence and convolutions entirely. The best performing models also connect the encoder and decoder through an attention mechanism. Model architecture eschewing recurrence instead of relying entirely on an attention mechanism to draw global dependencies between input and output. The Transformer allows for significantly more parallelization and can reach a new state of the art in translation quality. The model architecture is defined by a encoder-decoder structure. The input sequence of the encoder is continuous representations (X_1, \dots, X_n) which are mapped to $Z = (Z_1, \dots, Z_n)$. At the end decoder takes the Z as input and provides the output (Y_1, \dots, Y_n) .

To be more specific, the Encoder is created by six layers ,which every one of these six have two more sub layers. The first sublayer is a multi-head self-attention mechanism and the second sub layer is a fully connected layer “feed forward network”.

These two layers have residual connection which is followed by a normalization layer and the output has an embedding of dimension equals to $d = 512$. From the other side is the Decoder. Decoder also consisted from six stacked layers and it adds one third sublayer in the two sublayers of the Encoder. The third layer performs multi-head attention over the output of the encoder stack. The three layers follows the same approach with the Encoder, residual connection followed by a normalization layer. In addition, the self-attention sub-layer, has been modified in order to prevent positions from attending to subsequent positions. This type of masking, combined with fact that the output embeddings are offset by one position, ensures that the predictions for position i can depend only on the known outputs at positions less than i . “Figure 2.1”

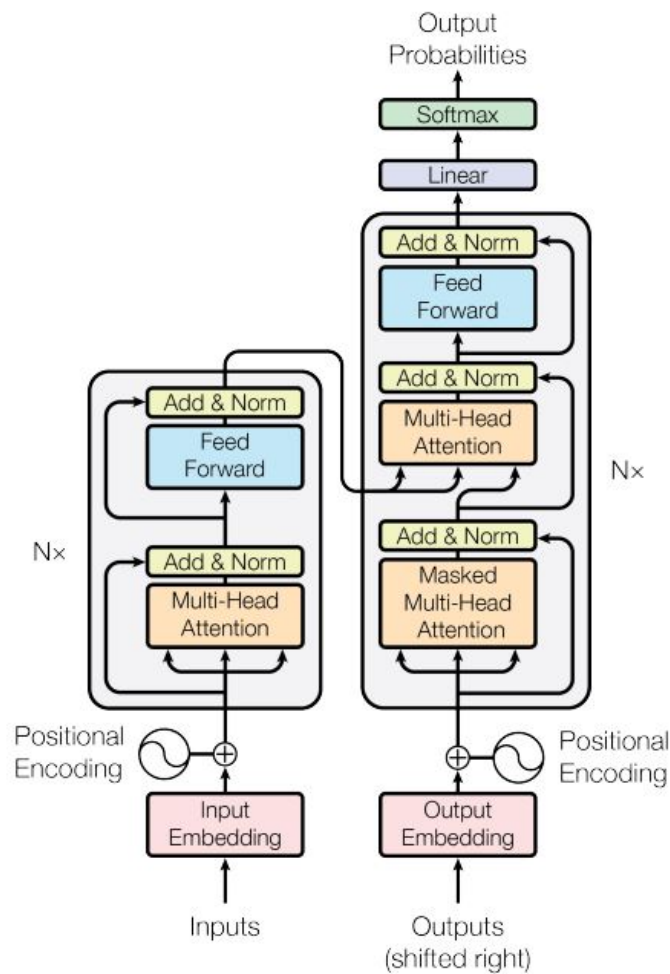


Figure 2.1: The Transformer architecture.[2]

As mentioned above, Attention is a very important component of the Transformer Architecture [2]. Attention function maps a query and a set of key-value pairs as vectors to the output which is computed as weighted sum of the values. Scaled Dot-Product Attention is also another practice which has as input the queries and keys of dimension d_k and the values of the dimension d_u . “Figure 2.2”

Scaled Dot-Product Attention

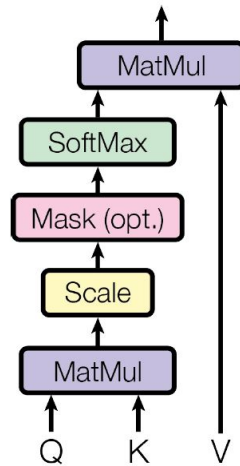


Figure 2.2: Scaled Dot-Product Attention.[2]

To be more specific, the attention function is been calculated for a set of queries simultaneously, packed together into a Matrix Q and keys, values as K, V divided by $\sqrt{d_k}$. Below you can find the equation Scaled Dot-Product Attention. “Figure 2.3”

$$\text{Attention}(Q, K, V) = \text{softmax}\left(\frac{QK^T}{\sqrt{d_k}}\right)V$$

Figure 2.3: Attention Function.[2]

In order to get the maximum benefit of the queries, values and key it is proposed that its better to linearly project them instead of just perform a single attention function (d_k, d_u) . For each of these queries values and keys that have been performed the attention function in parallel. This projections of queries, values and keys are concatenated and then are projected one time as the “Figure 2.4” that follows.

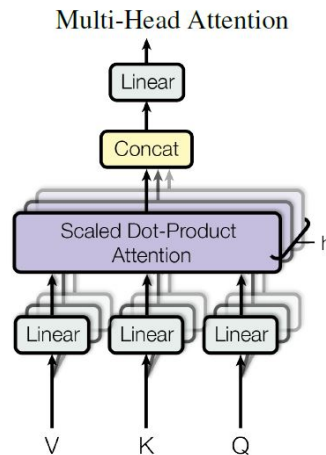


Figure 2.4: Attention Function.[2]

Multi-head attention that is depicted in Figure 2.5 allows the model to get information from all the representations in different positions.

$$\text{MultiHead}(Q, K, V) = \text{Concat}(\text{head}_1, \dots, \text{head}_h)W^O$$

where $\text{head}_i = \text{Attention}(QW_i^Q, KW_i^K, VW_i^V)$

Figure 2.5: Attention Function.[2]

2.1.1 Attention Use Cases

Multi-head Attention is used:

- Encoder - Decoder attention Layers. This application of Attention feeds the queries from the previous decoder layer as well as the memory keys and values have as starting point the encoders output. It is a simulation encoder-decoder attention mechanism and all positions in a decoder can reach all positions.
- Self - Attention layers in encoder which all keys, values and queries come from the same position. This position is the output of the previous layer in the encoder. [9], [10], [11]
- Self - Attention layers in decoder, provide the same approach as mentioned in the encoder. This layers are make possible to the decoder to attend all

positions in the decoder. Specifically, it is mandatory to prevent the leftward information flow in the decoder to preserve the auto-regressive property. This practice is used in the section of scaled dot-product that have been mentioned earlier in previous sections.

2.2 Unsupervised Cross-lingual Representation Learning at Scale (XLM Roberta)

There are many Transformer-based[2] masked language models. One of them is XLM-RoBERTa [5]. It is a pretrained multilingual language model which leads to significant performance in comparison to other multilingual models. The training dataset of that model is consisted of a hundred languages and it provides very good results when it comes to cross lingual classification, sequence labeling and question answering. In addition, Facebook’s Artificial Intelligence team has focused on the trade offs between high-resource and low-resource languages and the impact of language sampling and vocabulary size. After making multiple experiments, the result was that more languages leads to an optimum cross-lingual performance on low-resource languages up to a point and after that particular point the model performance starts to be less efficient for both monolingual and cross-lingual experiments. The overall results lead to out-perform multilingual BERT (mBERT) [8] on cross-lingual classification by up to 23% accuracy on low-resource languages as well as other cross-lingual and monolingual tasks.

To elaborate further, for the training part of the language model, it uses the XLM approach [12]. The training data are monolingual and the Transformer model [2] is used. For each language streams are created and the prediction of the masked tokens is happening. These samples batches of streams are selected like in Cross Lingual language model pretraining article [12]. The models that have been trained are the XML- R_{Base} and the XLM-R. Their parameters are $L = 12$, $H = 768$, $A = 12$, 270M and $L = 24$, $H = 1024$, $A = 16$, 550M respectively while the size of the vocabulary was 250k with a full soft max.

XLM-R model is trained with one hundred languages. The dataset comes from

Wikipedia. It is also trained with Hindi and traditional Chinese which are not popular languages for natural language processing tasks. In all the experiments, English, French, German, Russian, Chinese, Swahili and Urdu have been included for classification and sequence labeling evaluation benchmarks. The reason why is that these families of languages include both low and high resource languages.

The comparison to the XLM-R comes when it's been evaluated. There are 4 evaluations tasks. The first is the Cross-lingual Natural Language Inference (XNLI) and it consist of 15 languages with ground truth dev and test sets. For this approach the model is been trained in the other 14 non English languages while the evaluation is getting place on cross-lingual transfer from English to other languages. The second task is the Name Entity Recognition (NER). The experiments for NER was based on CoNLL-2002 [13] and CoNLL-2003 [14] which contain the English, Dutch, Spanish and German languages. In order to evaluate the model in multilingual learning, first it had to be trained in English and then evaluated in cross-lingual transfer. The last two task were Cross-lingual Question Answering [15] and GLUE Benchmark [16].

2.3 Making Monolingual Sentence Embeddings Multilingual using Knowledge Distillation

Transfer learning is a practice that provide many solutions when it comes to cross-lingual tasks. This task is been achieved with sentence embeddings by using knowledge distillation[1]. Main idea of that work is to extend existing monolingual models to new languages as multilingual models. This can be achieved by mapping the translated sentence in the vector space that has the starting sentence.

The main idea of the aforementioned research is this of teacher model M and the student model \hat{M} which tries to align its embeddings from the Teacher. Their output is a sentence embedding. These models need a set of parallel translated sentences $(s_1, t_1), \dots, (s_n, t_n)$ where the s is the source language and the t is the translation of the source. With these sentences the student model \hat{M} got trained in order to have $\hat{M}(s_i) \approx M(s_i)$ and $\hat{M}(t_i) \approx M(s_i)$ [1] using mean square loss . In that approach the \hat{M} student model gains knowledge from the teacher M and learns a multilingual

sentence embedding space, this is why this method is called multilingual knowledge distillation. The student model \hat{M} learns:

- Vector spaces are aligned across languages, i.e., identical sentences in different languages are close. [1]
- Vector space properties in the original source language from the teacher model M are adopted and transferred to other languages. [1]

Regarding the training part of the models, parallel translated columns are vital, s_i and t_i represent the source language and translated sentences respectively. As mentioned before a very important task is the one that student model \hat{M} got trained in order to $\hat{M}(s_i) \approx M(s_i)$ and $\hat{M}(t_i) \approx M(s_i)$ [1] for a minibatch B and the mean-squared loss has to be minimized as follows:

$$\frac{1}{|B|} \sum_{j \in B} [(M(s_i) - \hat{M}(s_j))^2 + (M(s_i) - \hat{M}(t_j))^2]$$

Regarding the student and teacher models architecture, both models can have different networks. In any task the student will try to learn the representation of the teacher. Below is been illustrated of how the teacher and student models work together with the mean-squared loss function in order to achieve a good alignment [1]. This specific example uses sentences of German and English as translated and the source languages respectively.

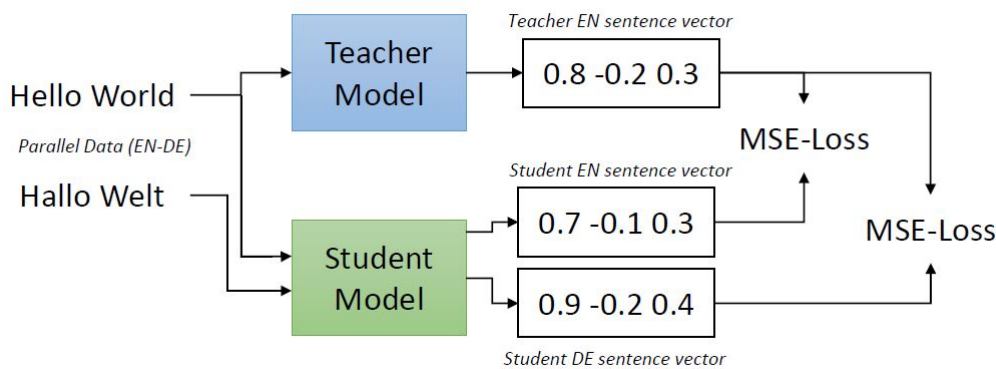


Figure 2.6: Attention Function.[1]

According to [1], many datasets has been used as training data. These datasets contain paraller sentences, some of them are: GlobalVoices, TED2020, NewsCom-

mentary, Tatoeba, WikiMatrix, JW300 [17], UNPC [18] as well as bilingual dictionaries like MUSE[19] and Wikititles. The initial experiments did took place with XLM-R as student and SBERT [4] with 20 training epochs, 64 as batch size and learning rate equals to $2e-5$. The experiments that took place had many combinations of teacher-student models as well as with practices that have no parallel data. Two experiments of them use parallel datasets and the language distillation of teacher-student approach. The first with SBERT-nli-stsb[4] as teacher and mBERT, DistilmBERT, XLM-R as student models. The second SBERT-paraphrases [4] as teacher and XLM-R[2] as student. The experiments that use the aforementioned approaches are the following:

- Multilingual Semantic Textual Similarity (STS). The goal is to create embeddings and assign a score of similarity for 2 sentences. The sentence embeddings are produce the Spearman’s rank correlation between the computed score and the gold score. Distillation practice produce state of the art results compared to older models like LASER, mUSE, LaBSE. XLM-R as student has the better results and this is one of the approaches that is going to be used in our experiments. SBERTparaphrase was the teacher model that contributed to that results.
- BUCC - Bitext Retrieval: This particular task is concerned with the identification of a pair of translated sentences that are located in two different corpora. It is used the BUCC bitext retrival code from LASER [20] and scoring function from “Margin based Parallel Corpus Mining with Multilingual Sentence Embeddings” [21]. Parallel sentences are going to be extracted for four different languages from the BUCC dataset. Train is used to find the threshold for the score function and the sentences above that threshold are parallel. SBERTparaphrase and XLM-R achieved state of the art results.
- Tatoeba-Similarity Search: This experiment took place for the low resource languages. Tatoeba test setup used for evaluation from the LASER[20]. 1000 English-aligned sentence pairs of various languages contained in the particular dataset[1]. Evaluation is done by finding for all sentences the most similar

sentence in the other language using cosine similarity [1]. For this particular experiment, SBERT-nli-stsb got distilled as teacher and the student was the XLM-R, where results were much better than the LASER model.

2.4 Sentence Embeddings using Siamese BERT-Networks (Sentence-BERT)

Sentence BERT [4] is the modified version of the BERT pretrained model. Actually, with the sentence embeddings frame work that this work provides, one is able to make his “Sentence” Transformer. Specifically, the addition that BERT has is the “siamese and triplet network structures, which derives semantically meaningful sentence embeddings that can be compared using cosine-similarity” [4]. This change provides a new BERT, which is available to became better in tasks such as similarity comparison, clustering and information retrieval. Also, SBERT, by using the siamese network architecture, has the ability to have fixed size vectors. As a result, the comparison of two sentences can happen by measuring the cosine similarities or taking distances such as Euclidian or Manhattan. Another big advantage of the SBERT is that the measures that is able to use, are performed fast and efficient in the current hardware technology.

Regarding the SBERT fine-tuning, NLI data was used with state of the art results by out performing many other practices. The SBERT model creates an output which came from a polling operation and modifies models such as BERT and RoBERTa with a fixed size of sentence embeddings. Architecture of the model is close related with the dataset and it is closely depends on that. For [4] research, 3 different approaches have been used. First one is the “Classification Objective Function” on which the trainable weight $W_t \in \mathbb{R}^{3 \times k}$ get multiplied with the concatenated sentence Embeddings $|u - v|$ as follows:

$$O = \text{softmax}(W_t(v, u | v - u))$$

The second approach is “Regression Objective Function” which actually calculates between two sentence the cosine similarities and as loss function is been used the mean-squared-error.

Last but not least, the third objective that is been tested is the “Triplet Objective Function”. For this task the input is an “anchor sentence α , a positive sentence p , and a negative sentence n , triplet loss tunes the network such that the distance between a and p is smaller than the distance between a and n ” [4].

$$\max(\|s_\alpha - s_p\| - \|s_\alpha - s_n\| + \epsilon, 0)$$

All the above are the sentence embeddings distances of $\alpha/n/p$ and margin ϵ . The metric for that example is the Euclidian distance while the margin ϵ “ensures that s_p is at least ϵ closer to s_a than s_n ”.

As for the training of the SBERT model, the datasets that have been used are the SNLI [22] and the Multi Genre NLI [23] datasets. The first one, is been consisted from 570000 sentences with their annotations (contradiction, entailment,neutral) while the second one contains 430000 sentences. In addition, the evaluation of the model has two phases. One with Semantic Textual Similarity tasks and one with the SentEval toolkit[24]. To start with Semantic Textual Similarity tasks, the number of evaluation tasks were four, as below:

- Unsupervised STS: SBERT gets evaluated without specific Semantic Textual Similarity data and the datasets are used have as labels between 0 and 5 on the semantic relatedness between sentence pairs. Mostly in all datasets, SBERT outperforms other state of the art approaches.
- Supervised STS: This experiment uses STS benchmark (STSb) [25] dataset which has 8,628 sentences pairs that categorized with 3 labels captions, news, and forums. SBERT is been optimized by a regression objective function. While the prediction task is running, it is been computed the cosine-similarity between the sentence embeddings.
- Argument Facet Similarity: The corpus of that experiment is the Argument Facet Similarity [26]. This dataset is based on sentence annotations with topics of gun control, gay marriage and death penalty. These annotation have values that are numbered from 0 to 5. When an annotation has the smaller value 0, it means that the topic of sentence is different while the greatest value 5 means that is completely equivalent.

- Wikipedia Sections Distinction: Wikipedia specific experiment which use a dataset, which created from [27]. This dataset has made smooth classes form paragraphs from wikipedia assuming that all sentences in a paragraph will be thematically closer. The dataset has “sentence triplets: The anchor and the positive example come from the same section, while the negative example comes from a different section of the same article” [27]. SBERT is getting trained with 1.8 Million training triplets and the evaluation happens with 222,957 triplets.

2.5 Useful Metrics

2.5.1 Euclidian distance

The Euclidean distance [28] between two objects for mathematics, “is the length of a line segment between the two points. It can be calculated from the Cartesian coordinates of the points using the Pythagorean theorem, therefore occasionally being called the Pythagorean distance”. When the distance is between objects and not points then Euclidian distance defined to be the smallest distance among pairs of points from the two objects. The distance of two points q , p with coordinates (q_1, q_2) and (p_1, p_2) is calculated as below:

$$d(p, q) = \sqrt{(q_1 - p_1)^2 + (q_2 - p_2)^2}.$$

2.5.2 Manhattan Distance

The Manhattan distance [29] or Taxicab distance between two vectors (city blocks) is equal to the one-norm of the distance between the vectors. The distance function (also called a “metric”) involved is also called the “taxi cab” metric. The Manhattan distance, d_1 , between two vectors p , q in an n -dimensional real vector space is been calculated as below:

$$d_1(p, q) = |p - q| = \sum_{i=1}^n |p_i - q_i|$$

2.5.3 Pearson Correlation

The Pearson correlation [30] evaluates the linear relationship between two continuous variables. A relationship is linear when a change in one variable is associated with a proportional change in the other variable. For example, you might use a Pearson correlation to evaluate whether increases in temperature at your production facility are associated with decreasing thickness of your chocolate coating.

2.5.4 Spearman Correlation

The Spearman correlation [31] evaluates the monotonic relationship between two continuous or ordinal variables. In a monotonic relationship, the variables tend to change together, but not necessarily at a constant rate. The Spearman correlation coefficient is based on the ranked values for each variable rather than the raw data. Spearman correlation is often used to evaluate relationships involving ordinal variables. For example, you might use a Spearman correlation to evaluate whether the order in which employees complete a test exercise is related to the number of months they have been employed.

Chapter 3

Proposed Approaches

This Section is going to describe the datasets and the models that we are going to use for the classification of the sentences that are argumentative or not. The approach that is going to be used is the one of the “Making Monolingual Sentence Embeddings Multilingual using Knowledge Distillation” [1]. It is going to use Multi Layer Perceptrons with Transformers such as multilingual Bert, XLM-R BERT. The train - alignment of the teacher student models is going to use the TEDex Talks [7] dataset which has parallel sentences of English and Greek. In addition, for prediction and evaluation we use after some data preparation the essays dataset [6] and the translated essays dataset which is provided by Spyridon Spyliopoulos. In a nutshell we are going to provide a full analysis on how we train a classifier in English language and how we use the weights of the model to make the final classification to the Greek sentences with the use of the aligned transformers.

3.1 Datasets and Data Preparation

This specific section is going to provide all the information and the process that was followed in order to prepare and transform the data, which is going to be used for the training and evaluation for all the experiments. First, we are going to further analyse the TED 2020 dataset which is used for the transfer learning part and cross-lingual embedding alignment. In addition, for the second part of the dataset’s section the Essays dataset will be presented as well as all the important

points for the data preparation.

3.1.1 TED 2020 - Parallel Sentences Corpus

In this section we are going to describe the structure of the TED 2020 dataset and how it is going to be used for this particular work. This dataset is also used for [1] but since it also contains Greek translation is quite useful for us. It contains 16424 talks in English while 3781 of them are translated in Greek. To elaborate further the number of the parallel sentences that exist between English and Greek language is 266861. To be more specific, the English sentences are 427436 and 266861 are also translated in Greek.

With the use of this dataset we are able to implement the transfer learning, as proposed by the state-of-art method our approach is based on. The dataset is been loaded and it has been iterated sentence by sentence to get the mean square error loss of each pair of the parallel Greek-English sentences. As mentioned above that dataset plays a critical role for the outcome and the results of all the experiments.

3.1.2 Essays Dataset and Transformation

Essays dataset [32] is going to be used for the training of the deep learning model that is going to make the classification task, which is to recognize the argumentative and non-argumentative sentences. The structure of the initial dataset contains pairs of files, annotation and text files which are named the same and they have different file type. Each text file contains parts of clear text and sentences in English language. Annotation files provide a mapping that contains the tags for each token of the text. These tokens are tags with “Arg-B” when the token is the first argumentative component and with “Arg-I” when it is an argument component. The non argumentative tokens are tagged with “O”. The key components of our transformation is the “Claim” tag and the “Premise” tags. The Claim tag which is also included in the dataset annotation file is the central component of an argument as it is mentioned in the [32] and the Premise tags are reasons to justify or refute the Claim.

In order to make the transformation we use Claim tags to set the class as 1 which means that the sentence is argumentative or as 0 which is translated as non argumentative. The transformation process has two steps. The first one, is the transformation to the CONLL format [13]. According to [32] the brat tool has been used for annotating the documents. CONLL datasets are structured line by line with in which they contain a token. For our specific task, the dataset had the token and the tag of a Claim, Premise or O as shown in Figure 3.1. The brat tool contains a tool that converts annotations into the CoNLL format, which however had to be modified in order to support Greek. The issue was that the tokenizer wasn't able to identify the Greek words. As a result the tool by itself was returning a flat file with all the words concatenated as a single character. What we had to do was to change the default tokenizer with one that was able to work with Greek. The tokenizer that provide the solution was the one from the nltk [33] python library. Below the result of the first stage of the data transformation is depicted.

0	580	581	,
0	581	583	τα
0	583	589	παιδιά
B-Claim	589	596	μπορούν
I-Claim	596	598	να
I-Claim	598	604	μάθουν
I-Claim	604	607	για
I-Claim	607	620	διαπροσωπικές
I-Claim	620	630	δεξιότητες
I-Claim	630	633	που
I-Claim	633	638	είναι
I-Claim	638	648	σημαντικές
I-Claim	648	651	στη
I-Claim	651	661	μελλοντική
I-Claim	661	664	ζωή
I-Claim	664	668	όλων
I-Claim	668	671	των
I-Claim	671	678	μαθητών
I-Claim	678	679	.

Figure 3.1: CONLL Structure

The second part of the data transformation was to create a file with all the sentences with labels. These labels will provide us the type of the sentences (argumentative, non-argumentative). To achieve these results was developed a custom python script. This particular script was able to iterate over all the tokens one by one, identify if the tokens are tagged as Claim or Premises and separate all the

sentences as well. As mentioned previously, the way to identify if a sentence is argumentative is depended on Claim tag. If Claim tags are contained then the sentence will be tagged as argumentative. Last but not least, upsampling was also applied to the dataset. The main reason behind upsampling were that the two classes was imbalanced (Figure 3.2). Initially the dataset had 5327 sentences, 4097 argumentative and 1230 non-argumentative. Non-argumentative are much less than the argumentative ones and upsampling was used to duplicate randomly some of them. The number of sentences now has been changed to 5707. All the new 380 sentences which have duplicated had non-argumentative class. We didn't add more duplicated sentences in the dataset because non-argumentave sentences were already small in number. Our main idea was to replicate 1/3 of the sentences and we only replicate only 380 sentences which is the 32% of the total number of the non-argumentative sentences.

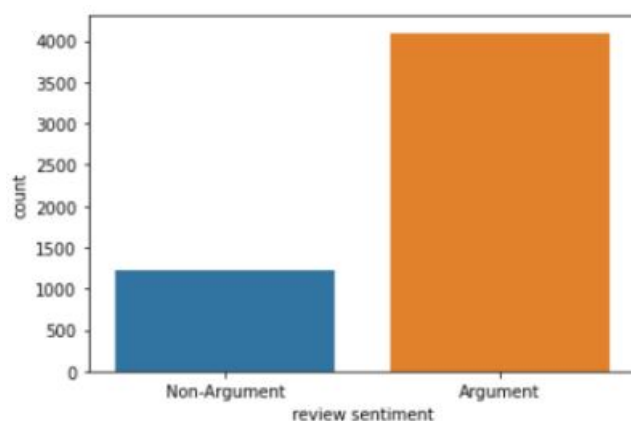


Figure 3.2: Sentences per Class

3.2 Transformers Library Experiment

In this section, all the experiments and the deep learning models were made with the help of the Transformers library from hugging face repository within the pytorch framework. Following that way a classifier was made and it was combined with Transformer model on top.

To start with, the first experiment was implemented on top of the 'bert-base-multilingual-cased' language model from the Transformers library. This pretrained

language model generates embeddings (vectors) for each sentence of the dataset. In order to provide and estimate an ideal sequence length of tokens that we are going to have as input we are making a distribution chart Figure 3.3. X axis has the token count of the sentences and the Y axis has the density. Based on the results we set the sequence length as 70 tokens.

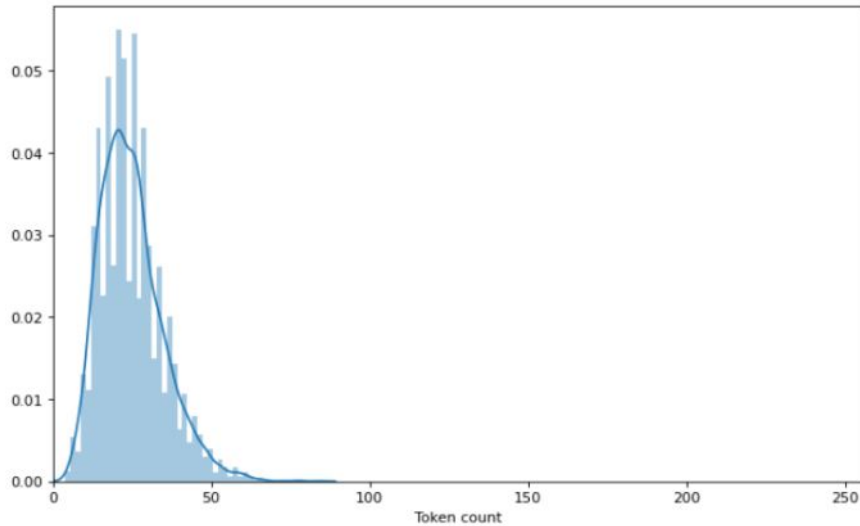


Figure 3.3: Tokens Count of Sentences X axis: Tokens of sentences , Y axis : Density

The 'bert-base-multilingual-cased' tokenizer by default encodes the tokens and gives the inputs ids as tag in each token. These are the first inputs of the neural network on which the Transformer located on top. After the transformation, the development of the neural network is taking place. At this specific point is important to be mentioned that the development for this classifier didn't extended further since it was used other libraries which are going to give us more flexibility our experiments. This method is going to be analyzed further in next sections . As a result, a simple and not that "deep" neural network was made to get trained and classify the argumentative or non-argumentative sentences.

Regarding the model architecture, we first apply the multilingual Bert pretrained model from the Transformers library as the first layer. Afterwards, it has been added one dropout layer with probability of 30%. This means that, by using Bernoulli distribution with the probability that has been mentioned, with the same probability the inputs of the tensor will be zeroed. On the next and final layer it is applied a linear fully connected layer of the pytorch library with a sigmoid activation function.

Sigmoid activation function is used because we have binary classification as it is already has been mentioned. The data of the training are the 80% of the entire dataset while the rest 20% has been used for testing.

To elaborate further, we would like to provide some more information regarding the hyper parameter tuning of the model. To start with, the data loaders that had been used for this training of the model were the default of the Transformers with a batch size of 2. The learning rate was 1e-05 (0.00001) and the number of epochs was 8. As mentioned previously, this particular model is not the one that we are going to make the final experiments, as a result there wasn't such an experimentation with the hyper parameter tuning. The reason why that this model exist is just to have it as a yardstick of the next experiments in order to compare the Transformer library, which this experiment developed and the Sentence Transformer library that is going to be used in all the following experiments.

3.3 Sentence Embeddings alignment (Transfer Learning)

This section presents the main approach of this work, whose implementation is based on the sentence transformers library [34]. This specific work consists from two core parts. The first one, is the development of the classifier that is used on top of the Transformer models while the second is the language distillation part which does the alignment of the embeddings. What follows is a step by step analysis on how the language distillation works and the Teacher - Student logic of models for our specific case. The Teacher - Student models that were used were the below combinations:

Combination Number	Teacher Model	Student Model
1	bert-base-nli-stsb-mean-tokens	bert-base-multilingual-cased
2	bert-base-nli-stsb-mean-tokens	xlm-roberta-base
3	bert-base-nli-stsb-mean-tokens	bert-base-cased

Table 3.1: Teacher - Student Models

For all the models shown in “Table 3.1” the same hyper-parameters were used

for the training part in order to have a straight forward comparison. In addition, the TED Talks dataset it is used in all experiments for aligning embeddings across languages. The number of sentences is 266861. These sentences are getting trained for 5 and 8 epochs with a batch size of 32 records. Batch size of 32 was the maximum number that we were able to use since for bigger batches we faced memory issues and the training process stopped due to memory allocation errors. Also, for the training of the model we limit the input length of the characters that are going to be part of the sentence as 250. This means that the input of the model is going to handle a quite good length of sentences. Last but not least it is important to be mentioned that the learning rate that has been used for this specific task is $2e-5$ which is the one that [1] suggests.

Data preprocessing part plays a significant role. The reason why is that before the training of the model that is going to give us the multilingual embedding model we have to calculate the below metrics:

- Mean Squared Error (MSE), which actually measures the euclidean distance between teacher and student embedding
- The translation evaluation which is a function of the sentence Transformers library that compares two translated sentences, for example GR-sentence EN-sentence and returns the accuracy in both directions.
- Embedding similarity evaluator which is a function of sentence Transformers library which evaluates the similarity of the embeddings by calculating the Spearman and Pearson rank correlation in comparison to the gold standard labels. The metrics are the cosine similarity as well as euclidean and Manhattan distance. The returned score is the Spearman and the Pearson correlations.

This comparison is happening for each sentence and actually, it plays major role for the transfer learning part. With the use of sentence embeddings library we are able to use the aforementioned evaluators, while the model is getting trained from the parallel English - Greek dataset and they can provide important guidance that can help the training process to evaluate models and save the best model on the disk.

The scores are returned in a regular basis during the training in order to have a good overview about how the performance during training. The evaluation results will be analyzed further in the fourth chapter that is going to provide all the results of all the tasks. The distance metrics for comparing the embeddings after the language distillation is the Euclidian and Manhattan distances.

3.4 Model Training in English and Baseline Prediction in Greek

The idea behind training a classifier using the default embeddings (BERT, mBERT, XLM-Roberta) without any hyper parameter tuning and the English Essays [32] dataset was to have a model with trained weights to use for the Greek aligned Transformer's prediction as well as baseline metrics without having done any transfer learning. By doing that practice, we will have a comparison point to observe how strong it was the alignment of the Teacher - Student approach and if the transfer learning worked efficiently for all the Transformer language models. Consequently, this mean that if the Greek aligned model has better performance results than the initial language model then the whole process has succeed.

To elaborate further, it should be noted that that no training happens to the classifier which is using the Greek aligned embeddings or other training with the Greek datasets. As a result, we entirely rely on the performance of the trained English classifier and the Teacher - Student alignment to be successful for our final Greek argumentative prediction. All the training happens with the English dataset and it is combined with the aligned embeddings. In addition, the whole methodology is going to be the same for all the language models that we will be working on. For this part of our work, we are going to develop a classifier with two different architectures. These architectures will be used across implementations of the English and Greek classifiers. The reason why, is that, in order to reuse the learned models without issues, the deep learning's model architectures have to be exactly the same.

Regarding the training of the model, we are going to do a deep dive analysis for the two architectures and their hyper-parameters. To begin with, below you can

find the two architectures that has been implemented and evaluated:

- The first model (Figure 3.4) is a simple and not that “deep” learning model. Actually, from a design perspective is the same that has been developed for the “Transformer Library Experiment” approach. The main difference for this particular model is the type of Transformers, where the transformer has been replaced with a sentence transformer [34]. Using the Sentence Transformers, we start with the Multilingual BERT, BERT and XLM-R pretrained models as the first layer in all three different experiments performed. One dropout layer with probability of 30% has been added. Last but not least, the output layer follows, with the sigmoid activation function.
- The second architecture (Figure 3.5) that is been used, is a slightly more complex one. It starts again with the Transformer layer. Then follows a linear fully connected layer with 768 input neurons and 1000 output neurons. Two fully connected layers are taking this output: the first one has input as 1000 and output 500 neurons while the other one has 500 input and 60 output neurons. Then for regularization purposes we are adding a dropout layer with 0.1 probability. Last but not least we have another fully connected layer with 60 and 10 neurons as input and output respectively and another dropout layer with 0.1 probability. At last is the output classification layer with a sigmoid activation function having 10 neurons as input and 1 as output.

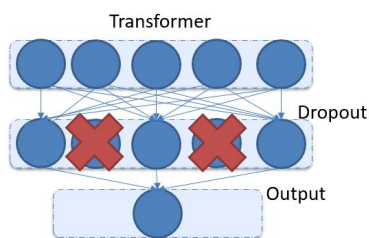


Figure 3.4: Small Neural Network

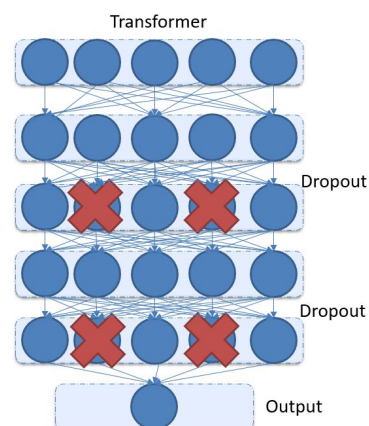


Figure 3.5: Big Neural Network

Both neural network architectures have trained for 200 epochs with batch size of 1024. The train dataset has 4566 records while the test set 1141. Also, the optimizer that is been used is ADAM and the loss function is Binary Cross Entropy. The reason why we use sigmoid activation function is that we want to produce one output that it will represent if the sentence is argumentative or not. After the aforementioned procedure we save the weights of the trained model in a local directory. These weights are going to be used in the next section on which we are going to predict the argumentative class of the Greek sentences.

As last step of that section, we are making baseline predictions using the default Transformers on the Greek dataset. With that way we are taking the minimum results that a language model can give and we will going to compare them afterwards with the Greek aligned ones.

3.5 Final Prediction on Greek Sentences

This section is the one who is going to provide us the final results. These results are purely created by transfer learning techniques as mentioned previously without doing any training on Greek datasets and sentences. The major task of these part is to make the prediction by combining all the previous work. To start with, in that section we are using the Greek dataset. We are going to predict 6444 sentences if they are argumentative or not. A very important step is to use exactly the same deep learning architecture with the models that have been trained in the English sentences. The reason why, is that we need to use the trained and saved weights of the previous section. Theoretically, these weights are going to work good enough because we are going to use on top of the deep learning model the Greek aligned transformers. Since those embeddings are trained to give very similar vectors for the Greek language comparing them with the standard transformers we are expecting to see better results using them. In the next chapter that is following we are going to have an extending discussion for all the results of the methods that has been used in the current chapter.

Chapter 4

Results and Discussion

4.1 Experimental Settings Overview

In this chapter we are going to present our experimental setting, and the results. The tasks and results that are going to be presented in the specific Chapter are:

- Language distillation results (4.2.1 section). This section is going to provide all the metrics and results of the transfer learning part of our work. It presents how all the Transformer models have been aligned between the Greek and English languages of the TED talks dataset and we present the embedding distances correlation for 8 and 5 epochs of training alignment in tables 4.1 and 4.2 respectively.
- Transformers Library Results (4.3 section), presents the results of the training we did with the “transformers” library instead of the “sentence transformer” library and the evaluation results are presented in Figure 4.1. In this section it is used the essays dataset
- Model Training with English dataset and Greek Argumentative Prediction (4.4 section). This section is going to present the training and evaluation results of the English classifier (4.4.1 section), which uses the sentence transformer library and essays dataset. We provide results from the two deep learning architectures (Tables 4.3, 4.4) that we call them as “small” and “big” and their architectures are depicted on 3.4 and 3.5 figures. Then follows the 4.4.2

section “Greek Sentence Prediction with non Greek Aligned Embeddings” on, which we are going to predict Greek argumentative sentences of the translated essays dataset with the non Greek align initial pretrained embeddings (4.5 and 4.6 Tables with evaluation results). The final section (4.4.3) is “Greek Sentence Prediction with Aligned Language Models”, we show and discuss the results of the predictions the aligned embeddings of 4.2 section on top of the trained model of the 4.4.1 section.

4.2 Language Distillation Experiments

In this section we are going to provide the results and metrics regarding the Student-Teacher models as well as how they interact to each other. For all the experiments we made we had as Teacher model the 'bert-base-nli-stsb-mean-tokens' which is used on [1]. In addition, as mentioned in Chapter 3, we trained all the language models with 5 and 8 epochs. Regarding the hyper-parameter tuning, because of system memory capacity we were able to have as maximum number of batch equals to 32. Learning rate across all the experiments was 0.002 with a max capacity of sentence characters as 250. The metrics used are Manhattan and the Euclidian distance. The tables 4.1 and 4.2 are showing the distances on the final training epoch of the Teacher - Student model alignment.

4.2.1 Language Distillation Results

The question we are trying to answer in this specific section is “how successful can be the embedding alignment between Greek and English languages”. The training results of the Language distillation process are closely depended on the relationship of the Teacher and Student models. Thus, the models results vary in different ways. The tables 4.1 and 4.2 contain the training results of the Distillation practice. We depict the Manhattan and Eudlidian distance by calculating the Pearson and Spearman correlation coefficients. To start with multilingual BERT (bert-base-multilingual-cased) model, we are able to observe that after the fifth epoch of training the Pearson and Spearman correlations of the distances are getting increased

slightly faster than the other models with the final result at the eighth epoch, which again seems to be better compared to others. Secondly, we have the XLM-Roberta model (xlm-roberta-base). For that specific model, no major differences were observed between the training epochs. Results after eight epochs of training had in general better results than what we had on the fifth epoch but the correlation results by itself are not that great. The reason why is that we had not that great correlation scores as 4.1 and 4.2 tables show. The last model we used was BERT (bert-base-cased), whose performance is similar to XLM-Roberta model.

Results for 8 and 5 Epoch training:

Num	Student Model	Manhattan Dist.		Euclidian Dist.	
		Pearson	Spearman	Pearson	Spearman
1	bert-base-multilingual-cased	0.2538	0.236	0.2497	0.2204
2	xlm-roberta-base	0.1622	0.1635	0.1386	0.1391
3	bert-base-cased	0.0962	0.0872	0.1081	0.1091

Table 4.1: Results for 8 Epochs of training

Num	Student Model	Manhattan Dist.		Euclidian Dist.	
		Pearson	Spearman	Pearson	Spearman
1	bert-base-multilingual-cased	0.2055	0.2019	0.2023	0.211
2	xlm-roberta-base	0.1494	0.1547	0.1333	0.1438
3	bert-base-cased	0.1361	0.1462	0.153	0.1542

Table 4.2: Results for 5 Epochs of training

What follows from the two previous tables is that the multilingual BERT model has the higher correlation score in the final training epochs while it also has a steady growth on regards to the correlation that Manhattan and Euclidian distances provide across all passing epochs. While XLM-Roberta and base BERT models seem to not have a great correlation growth until the eight epoch. We can say that since XLM-Roberta has a different architecture and has higher capacity (more parameters) than BERT model, it is expected to exhibit a behaviour similar to the one we observed in the alignment experiment. In addition, we need to mention that indeed the training of the language models helped to have more similar embeddings between the Greek and the English datasets for the multilingual BERT specifically. In contrary, for the

base BERT and XLM-Roberta the difference after the training was so small that we cannot say that it will be efficient. In conclusion, the answer we have for our question of this section is that we are able to align Greek and English embeddings with moderate results.

4.3 Transformers Library Results

Initially when we started to build the classifier for the English sentences, we develop our initial model with the transformers libraries. This architecture, substituted with architectures based on the “sentence transformers” library, which were more suitable implementation wise. Although, this specific classifier was able to predict the English argumentative sentences with the below metrics (“1” symbolize the argumentative class while “0” the non argumentative).

	precision	recall	f1-score	support
0	0.79	0.61	0.69	249
1	0.89	0.95	0.92	816
accuracy			0.87	1065
macro avg	0.84	0.78	0.80	1065
weighted avg	0.87	0.87	0.87	1065

Figure 4.1: Transformers Results English Sentences Classifier

With neural network architecture of the “3.4 Figure” we were able to score a 87% accuracy something that is pretty close with the results we are also taking for the English sentence prediction by using the Sentence Transformer library.

4.4 Model Training with English dataset and Greek Argumentative Prediction

This part of our work will provide the results of all the classification tasks. Also, we are going to present the performance of all combinations of experiments performed with the default and the aligned language models that we trained on top of the deep learning model. At first, we evaluated the performance on the argument classification task with the default, pre-trained by their authors, embeddings in order to be able to compare them with our trained and aligned embeddings. The baseline

results and the classifier training on the English dataset is going to be analyzed for both the deep learning architectures (Figures 3.4, 3.5) we discussed in the previous Chapter.

4.4.1 Model Training in English dataset

It was really important for our work to develop classifiers which are able to predict argumentative English sentences using the already pretrained not Greek aligned Transformers. Their weights are going to be used on the prediction of the Greek argumentative sentences with the use of the Greek aligned Transformers (Section 4.2) on top. The training was performed 200 epochs with a train set of 4566 sentences and a test set 1141 sentences. The optimizer that is used is Adam, the learning rate was 0.01 and Binary Cross Entropy as loss function. The first table (Table 4.3) shows the evaluation results of the model that was trained in the English dataset which contains English sentences with argumentative or non argumentative label. This training took place with the deep learning model of Figure 3.4 which has a dropout layer and as output a single linear layer with sigmoid activation function. The results located on table 4.3 below:

Transformer	Label	Precision	Recal	F1 Score	Accuracy
bert-base-multilingual-cased	Non Arg.	79%	59%	68%	84%
	Arg.	86%	94%	90%	84%
xlm-roberta-base	Non Arg	79%	58%	67%	84%
	Arg.	95%	84%	89%	84%
bert-base-cased	Non Arg.	76%	61%	68%	84%
	Arg.	86%	92%	89%	84%

Table 4.3: First Small deep learning Model Results on English Dataset (Default Transformer)

Then follows the table that contains the results of the second deep learning model of Figure 3.5, which has five layers and two dropout layers as it have been mentioned in the previous Chapter. Also for this experiment we have the same learning rate as 0.01 and Binary Cross Entropy as loss function. The evaluation results are the following:

Transformer	Label	Precision	Recal	F1 Score	Accuracy
bert-base-multilingual-cased	Non Arg.	78%	73%	75%	87%
	Arg.	90%	92%	91%	87%
xlm-roberta-base	Non Arg	74%	70%	72%	85%
	Arg.	88%	90%	89%	85%
bert-base-cased	Non Arg.	73%	74%	73%	85%
	Arg.	90%	89%	90%	85%

Table 4.4: Second Bigger deep learning Model Results on English Dataset (Default Transformer)

After running these training experiments we are able to observe that there is no major differences in the output metrics between the two deep learning model architectures. Additionally, we can say that for the second training part on, which is used the deep Multi-Layer Perceptron model of Figure 3.5 we have slightly better accuracy with the multilingual BERT among the other models. Also, in both experiments we observe that the accuracy is almost the same for the small deep learning architecture of Figure 3.4.

4.4.2 Greek Sentence Prediction with non Greek Aligned Embeddings

Our next experiment is to take the baseline predictive metrics for all the Transformer models that we use. This step is important for our related work because it sets the baseline performance of each model. Consequently, we used the hugging face models through sentence transformer’s “framework” on top of the two deep learning model architectures that have been discussed previously (small, Figure 3.4 and big, Figure 3.5). Below you can find the baseline accuracy of the Transformer models without any fine-tuning (Table 4.5). These results are coming from the deep learning model that has a drop out layer and the linear output layer with 200 epochs of training.

From the outputs we have (Table 4.5), the multilingual BERT has the worst accuracy while the base BERT and the XLM-Roberta have much better performance. XLM-Roberta is the model with the best accuracy from all the experiments we made with the no fine-tuned embeddings. This model is already trained in many languages

Teacher Model	Student Model	Greek Sentences	Accuracy
bert-base-nli-stsb-mean-tokens	bert-base-multilingual-cased	6444	58.12%
bert-base-nli-stsb-mean-tokens	xlm-roberta-base	6444	75.96%
bert-base-nli-stsb-mean-tokens	bert-base-cased	6444	73.81

Table 4.5: Small Deep learning Model (Figure 3.4) Results on Greek Dataset - Default Model Prediction

according to [2] and the results we have verify what the initial work [2] mentions about the XLM-Roberta. What is been mentioned in [2] is that the XLM-Roberta model outperforms other Transformers like multilingual BERT something that is also happens the results above (Table 4.5). In addition, results suggest that base BERT exhibiting better results in comparison to multilingual BERT. Since base BERT as a model is trained specifically in English datasets while the multilingual one has been trained in other 104 languages, initially we were expecting to have a better result with the multilingual Bert. But it seems that with a small neural network (Figure 3.4) as we have for the current experiments the multilingual embeddings are trained to have a better average result for multiple languages. Specifically, for Greek, base BERT is a model with better accuracy according to Table 4.5 results.

Prediction results of the second deep learning model (Figure 3.5) follows after 200 epochs of training (Table 4.6):

Teacher Model	Student Model	Greek Sentences	Accuracy
bert-base-nli-stsb-mean-tokens	bert-base-multilingual-cased	6444	67.52%
bert-base-nli-stsb-mean-tokens	xlm-roberta-base	6444	75.46%
bert-base-nli-stsb-mean-tokens	bert-base-cased	6444	73.68%

Table 4.6: Big Deep learning Model (Figure 3.5) Results on Greek Dataset - Default Model Prediction

Again, in this deep learning architecture we are able to observe that the XLM-Roberta and the base BERT models had almost same accuracy while XLM-Roberta has the best. Multilingual BERT model on the other side, exhibited a significant improvement from the accuracy perspective by closing the big gap of the first small deep learning model that we used. This shows that multilingual BERT is not a strong model to classify Greek argumentative sentences by itself. We expect an improvement in its performance when we are going to use it with the aligned em-

beddings.

4.4.3 Greek Sentence Prediction with Aligned Language Models

This Section is going to present and analyse the results of the final set of experiments, which is the Argumentative prediction of Greek sentences. For this part our work we are loading the Greek aligned embeddings (Section 4.2.1) that we have done the training as well as we are going to use the trained weights of the English trained classifier (Section 4.4.1). To start with, we provide the results of the small deep learning classifier (Figure 3.4) with the one drop out layer and the five epoch trained aligned model.

Transformer Model	Align Epochs	Greek Sentences	Accuracy
bert-base-multilingual-cased	5	6444	65.812%
xlm-roberta-base	5	6444	73.395%
bert-base-cased	5	6444	31.396%

Table 4.7: Prediction with Small Network and 5 epochs aligned embeddings

Our observations from that experiments is that multilingual BERT had a significant improvement in performance. The accuracy on predicting the Greek argumentative sentences with the Greek aligned embeddings got improved from 58.12 (Table 4.5) to 65.81 which is something that suggests that the Greek sentence alignment for that model works with good results. On the other hand, BERT base results suggest, for this particular model, that the alignment was not very successful. 31.396% is low, and the main reason for this result is the deep learning architecture (as also suggested by the results presented in table 4.8). This means that the alignment was not the only reason why BERT base exhibited this behaviour. Also, XLM-Roberta seems to have close accuracy with the results that the no fine-tuned embeddings provided on top of the same model (Table 4.5).

The second step of our experiments, does the same process using the same aligned Transformers on top of the other deep learning architecture (Figure 3.5) which consisted of five linear and two dropout layers. Prediction result shown below (Table

4.8).

Transformer Model	Align Epochs	Greek Sentences	Accuracy
bert-base-multilingual-cased	5	6444	73.754%
xlm-roberta-base	5	6444	74.545%
bert-base-cased	5	6444	66.723%

Table 4.8: Prediction with Bigger Network and 5 epochs aligned embeddings

A bigger deep learning model with more parameters like this seem to give a pretty good boost to the multilingual BERT model. Again we see that the accuracy started at 67.52% (Table 4.6) from the non aligned Greek embedding big model and after the alignment we an accuracy of 73.754% (Table 4.8). XLM-Roberta seems to have a stable accuracy again without having any accuracy changes. Finally, the BERT base model had a great improvement in comparison to the accuracy that had on the small deep learning model (Figure 3.4) and it seems that this specific model depends a lot to the neural network that follows in order to have a good performance.

Next part of the experiments is going to take place with the same exact models and the one change that we are going to make is the aligned model. The following experiments apply embeddings that have been aligned for eight epochs of training instead of five. First we start again with the small neural network of Figure 3.4.

Transformer Model	Align Epochs	Greek Sentences	Accuracy
bert-base-multilingual-cased	8	6444	69.056%
xlm-roberta-base	8	6444	74.323%
bert-base-cased	8	6444	37.33%

Table 4.9: Prediction with Small Network and 8 epochs aligned embeddings

Results of the big neural network of Figure 3.5.

Transformer Model	Align Epochs	Greek Sentences	Accuracy
bert-base-multilingual-cased	8	6444	74.11%
xlm-roberta-base	8	6444	73.941%
bert-base-cased	8	6444	72.949%

Table 4.10: Prediction with Bigger Network and 8 epochs aligned embeddings

Getting all the prediction results of the 8 epoch alignment we are able to point out that again multilingual BERT had an improvement. Now with the small neural network we have a progress from 65.812% (Table 4.7) to 69.056% (Table 4.9). Additionally, we also had a slight improvement for multilingual BERT from accuracy perspective in the bigger neural network (Figure 3.5) from 73.754% (Table 4.8) to 74.11% (Table 4.10). BERT base model on the other hand, had also improvement in comparison to the five epoch trained embeddings. For the small neural network (Figure 3.4) we had an accuracy impact from 31.396% (Table 4.7) to 37.33% (Table 4.9) and again we are able to observe how much this model depends on the neural network. On the bigger neural network the improvement was from 66.723% (Table 4.8) to 72.949% (Table 4.10). Regarding the XLM-Roberta we almost have same results for all the experiments and the neural network architectures. On Table 4.11 we present the summary results in relation with the baseline prediction of the non aligned embeddings. Non aligned predictions have the “No” in alignment column and the neural networks are named as “small” for Figure 3.4 and “big” Figure 3.5 network architectures.

What follows from the preceding experiments, is that the multilingual BERT was a model that language distillation worked good enough with all the architectures we made and all the results we had were far better than the baseline prediction results. While, for base BERT model we have to point out that after the alignment task the model wasn't able to provide respective results without having a deep learning model with a good number of layers. Thus, base BERT had average results only when was used with our big neural network. Besides that, base BERT alignment wasn't that efficient and on top of the big neural network was able to provide almost the same results with the baseline model without any improvement. This is something that was expected since on alignment training the model wasn't able to adapt it's embeddings on the training epochs. Last but not least, XLM-Roberta had almost the same results from accuracy point of view in all the experiments. Again XLM-Roberta while training has not that great improvement on regards of the distance correlation. Actually, XLM-Roberta seems to be not that effective on the embedding alignment task in general.

Transformer Model	Alignment	Epochs	Network	Accuracy
bert-base-multilingual-cased	No	NA	small	58.12%
bert-base-multilingual-cased	No	NA	big	67.52%
bert-base-multilingual-cased	Yes	5	small	65.812%
bert-base-multilingual-cased	Yes	5	big	73.754%
bert-base-multilingual-cased	Yes	8	small	69.056%
bert-base-multilingual-cased	Yes	8	big	74.11%
xlm-roberta-base	No	NA	small	75.96%
xlm-roberta-base	No	NA	big	75.46%
xlm-roberta-base	Yes	5	small	73.395%
xlm-roberta-base	Yes	5	big	74.545%
xlm-roberta-base	Yes	8	small	74.323%
xlm-roberta-base	Yes	8	big	73.941%
bert-base-cased	No	NA	small	73.81%
bert-base-cased	No	NA	big	73.68%
bert-base-cased	Yes	5	small	31.396%
bert-base-cased	Yes	5	big	66.723%
bert-base-cased	Yes	8	small	37.33%
bert-base-cased	Yes	8	big	72.949%

Table 4.11: All results of the Greek Sentence Predictions

Chapter 5

Conclusions and Future Work

In this work we tried to solve a Transfer Learning problem between languages. Our goal was to retrieve knowledge from English pretrained Transformer models in order to be able to classify argumentative sentences of Greek language. In order to achieve that we trained Greek embeddings from pairs of parallel translated sentences that contain the same sentence in English and Greek as the [1]. In order to feed the data for the embedding alignment training, we had to do some preparation and give the argumentative and non argumentative class to each sentence. Since we used the Essays dataset which has all the argumentative parts of a sentence, we were able to do that kind of data preparation in a sentence perspective.

As mentioned our first part of this work was to make the embedding alignment or else Language distillation. For that part we trained 3 different Transformer models the multilingual BERT , the XLM-Roberta and the base BERT for two times, one was with 5 epochs and the other was for 8 epochs. Our metrics was the Euclidian and Manhattan distances and we are able to see the Teacher (Source Embeddings) and Student (Target Embeddings) model correlation to get slightly higher at each passing epoch. Our next step was to make classifiers in order to use them in cooperation with these Transformer models. First, we did the training and prediction in English sentences in order to get that trained model and use it to classify Greek sentences with the aligned embeddings on top. Also, we got the base line prediction accuracy for Greek sentences using the default Transformer models and the two deep learning

architectures that we developed which are located under the Transformer.

The final step was to make the prediction with our aligned models. We did the process with all three models that have been mentioned. What we observed is that the multilingual BERT was the only model, which we could say that worked successfully since after the alignment task had much better accuracy than the initial no fine-tuned embeddings. For the two other models we saw no major differences, a result that is been verified also from the alignment results since the distance's Pearson and Spearman correlations didn't had any major changes.

What follows from the preceding discussion is that we successfully achieved our initial goals. First we observe that the distance correlation of the multilingual BERT got higher throughout the alignment epochs. These results bring us to the point to say that we aligned with good results one combination of embeddings across languages. Additionally, after the prediction we made with the aligned embeddings we obtained better results from what we had as baseline for the multilingual Bert model. Consequently, again for that specific transformer we successfully observed that the Classification of Greek argumentative sentences without any training on Greek datasets worked with good results.

Future directions can be to train classifiers with the translated Essays dataset in order to have an overview about upper limits of the training metrics. One important implementation that we did not had the opportunity to research, because of time limitations, is the training and evaluation on the English dataset with the aligned multilingual embeddings that we trained. With that way, we are going to be aware of, if the already English embeddings that are contained after the alignment have the ability to be the same accurate with their initial ones that have not been aligned. Thus, we will be able to know if our approach worked only to make the Greek embeddings better and not to make the already existing English embeddings worse. Last but not least, according to the alignment results we got, XLM-Roberta could be trained for more epochs to see if the correlation of the distances is going to change.

References

- [1] Nils Reimers and Iryna Gurevych. Making monolingual sentence embeddings multilingual using knowledge distillation, 2020.
- [2] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin. Attention is all you need. In I. Guyon, U. V. Luxburg, S. Bengio, H. Wallach, R. Fergus, S. Vishwanathan, and R. Garnett, editors, *Advances in Neural Information Processing Systems*, volume 30. Curran Associates, Inc., 2017.
- [3] Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 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, Minneapolis, Minnesota, June 2019. Association for Computational Linguistics.
- [4] Nils Reimers and Iryna Gurevych. Sentence-bert: Sentence embeddings using siamese bert-networks, 2019.
- [5] Alexis Conneau, Kartikay Khandelwal, Naman Goyal, Vishrav Chaudhary, Guillaume Wenzek, Francisco Guzmán, Edouard Grave, Myle Ott, Luke Zettlemoyer, and Veselin Stoyanov. Unsupervised cross-lingual representation learning at scale, 2020.
- [6] Christian Stab and Iryna Gurevych. Parsing argumentation structures in persuasive essays. *Computational Linguistics*, 43(3):619–659, September 2017.
- [7] TED Talks. Ted talks 2020. In *TED Talks parallel translated datasets*, December 2020.

- [8] Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. Bert: Pre-training of deep bidirectional transformers for language understanding, 2019.
- [9] Oriol Vinyals, Łukasz Kaiser, Terry Koo, Slav Petrov, Ilya Sutskever, and Geoffrey Hinton. Grammar as a foreign language. In C. Cortes, N. Lawrence, D. Lee, M. Sugiyama, and R. Garnett, editors, *Advances in Neural Information Processing Systems*, volume 28. Curran Associates, Inc., 2015.
- [10] Jonas Gehring, Michael Auli, David Grangier, Denis Yarats, and Yann N. Dauphin. Convolutional sequence to sequence learning. In Doina Precup and Yee Whye Teh, editors, *Proceedings of the 34th International Conference on Machine Learning*, volume 70 of *Proceedings of Machine Learning Research*, pages 1243–1252. PMLR, 06–11 Aug 2017.
- [11] Leon Bergen, Dzmitry Bahdanau, and Timothy J. O’Donnell. Jointly learning truth-conditional denotations and groundings using parallel attention. *CoRR*, abs/2104.06645, 2021.
- [12] Guillaume Lample and Alexis Conneau. Cross-lingual language model pretraining, 2019.
- [13] Erik F. Tjong Kim Sang. Introduction to the CoNLL-2002 shared task: Language-independent named entity recognition. In *COLING-02: The 6th Conference on Natural Language Learning 2002 (CoNLL-2002)*, 2002.
- [14] Erik F. Tjong Kim Sang and Fien De Meulder. Introduction to the CoNLL-2003 shared task: Language-independent named entity recognition. In *Proceedings of the Seventh Conference on Natural Language Learning at HLT-NAACL 2003*, pages 142–147, 2003.
- [15] Patrick Lewis, Barlas Oğuz, Ruty Rinott, Sebastian Riedel, and Holger Schwenk. Mlqa: Evaluating cross-lingual extractive question answering, 2020.
- [16] Alex Wang, Amanpreet Singh, Julian Michael, Felix Hill, Omer Levy, and Samuel R. Bowman. Glue: A multi-task benchmark and analysis platform for natural language understanding, 2019.

-
- [17] Preslav Nakov and Alexis Palmer, editors. *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics: Tutorial Abstracts*, Florence, Italy, July 2019. Association for Computational Linguistics.
- [18] Michał Ziemski, Marcin Junczys-Dowmunt, and Bruno Pouliquen. The United Nations parallel corpus v1.0. In *Proceedings of the Tenth International Conference on Language Resources and Evaluation (LREC'16)*, pages 3530–3534, Portorož, Slovenia, May 2016. European Language Resources Association (ELRA).
- [19] Alexis Conneau, Guillaume Lample, Marc’Aurelio Ranzato, Ludovic Denoyer, and Hervé Jégou. Word translation without parallel data, 2018.
- [20] Mikel Artetxe and Holger Schwenk. Massively multilingual sentence embeddings for zero-shot cross-lingual transfer and beyond, 2019.
- [21] Mikel Artetxe and Holger Schwenk. Margin-based parallel corpus mining with multilingual sentence embeddings. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 3197–3203, Florence, Italy, July 2019. Association for Computational Linguistics.
- [22] Samuel R. Bowman, Gabor Angeli, Christopher Potts, and Christopher D. Manning. A large annotated corpus for learning natural language inference. In *Proceedings of the 2015 Conference on Empirical Methods in Natural Language Processing*, pages 632–642, Lisbon, Portugal, September 2015. Association for Computational Linguistics.
- [23] Adina Williams, Nikita Nangia, and Samuel Bowman. A broad-coverage challenge corpus for sentence understanding through inference. In *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long Papers)*, pages 1112–1122, New Orleans, Louisiana, June 2018. Association for Computational Linguistics.
- [24] Alexis Conneau and Douwe Kiela. Senteval: An evaluation toolkit for universal sentence representations, 2018.
- [25] Daniel Cer, Mona Diab, Eneko Agirre, Inigo Lopez-Gazpio, and Lucia Specia. Semeval-2017 task 1: Semantic textual similarity multilingual and crosslingual
-

- focused evaluation. *Proceedings of the 11th International Workshop on Semantic Evaluation (SemEval-2017)*, 2017.
- [26] Amita Misra, Brian Ecker, and Marilyn Walker. Measuring the similarity of sentential arguments in dialogue. In *Proceedings of the 17th Annual Meeting of the Special Interest Group on Discourse and Dialogue*, pages 276–287, Los Angeles, September 2016. Association for Computational Linguistics.
- [27] Liat Ein Dor, Yosi Mass, Alon Halfon, Elad Venezian, Ilya Shnayderman, Ranit Aharonov, and Noam Slonim. Learning thematic similarity metric from article sections using triplet networks. In *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers)*, pages 49–54, Melbourne, Australia, July 2018. Association for Computational Linguistics.
- [28] Euclidian distance. In *Euclidian Distance website wikipedia*.
- [29] Manhattan distance. In *Manhattan Distance website wikipedia*.
- [30] Pearson correlation. In *Pearson Correlation website wikipedia*.
- [31] Spearman correlation. In *Spearman Correlation website wikipedia*.
- [32] Christian Stab and Iryna Gurevych. Parsing argumentation structures in persuasive essays, 2016.
- [33] NLTK Org. Nltk. In *NLTK Library website*.
- [34] Nils Reimers. Sentence transformers. In *Sentence Transformers Library website*, 2019.