Task 8: Multilingual News Article Similarity

TEAM ABCD: Arpit Dwivedi[*], Pravartya Dewangan[†], Parakh Agarwal[‡], Tias Mondal[§], Pawan Goyal[¶]
IIT Kharagpur, West Bengal, India

1 Task Description

The objective of this project is developing the language processing tool to efficiently obtain the similarity between different news articles of different languages based on multiple factors including geography, entities, time, narrative, style and tone. The problem statement is defined based on the Codalab Competition, Multilingual News Article Similarity and we are obtaining data from the competition dataset itself. Here, the similarity is obtained pairwise on a 4-point scale. The data constitutes over 7 languages namely, English, Spanish, German, Polish, Turkish, Arabic and French, having both cross-lingual pairs as well as monolingual data. There are over 4964 pairs of news articles in the mentioned languages.

2 Dataset

Dataset was created using extensively trained human annotators (¿20 paid students) were used on a detailed codebook and with dozens of examples to gain high inter-annotator agreement on the task (GWET's AC1 of 0.84). The annotation task consisted of carefully reading (4-5 min) each of the two news articles in a pair and selecting the OVERALL similarity score. The score is shown in Figure 1.

Other sub-dimensions are also provided that annotators have rated, so as to inform better model building, which are shown in Figure 2.

For extracting the dataset, cached links were used, further, multiple different channels were required to obtain all the dataset from the source. A

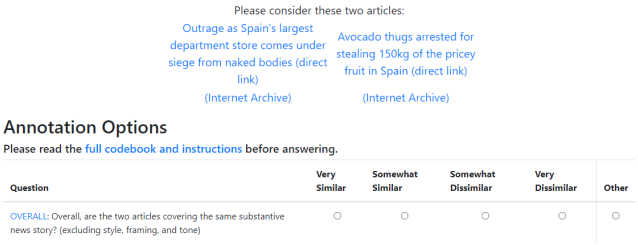

Figure 1: Overall Evaluation Topics

pair of articles with id 0123456789_9876543210 was be stored in output_dir/89/0123456789.json and output_dir/10/9876543210.json respectively. The json file contains additional information extracted from the page using the package newspaper3k. Thus, the final extracted output contains title, images links, text, title, description as well summary among others. These information can be used to further refine the network.

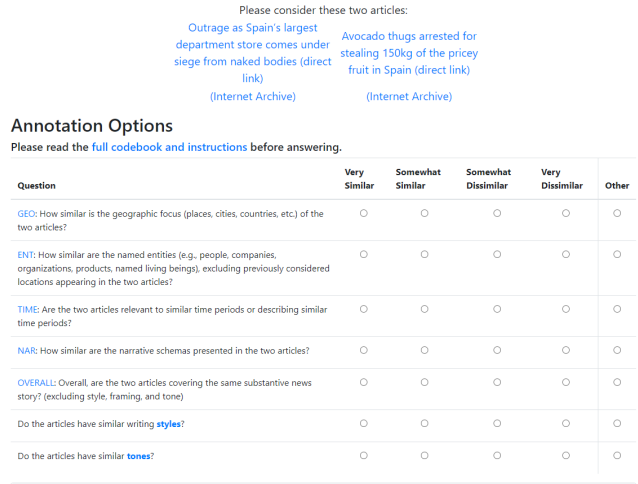

Figure 2: All Topics for Evaluation

[*]Roll No.: 17EC35005
[†]Roll No.: 17EC35041
[‡]Roll No.: 17EC35016
[§]Roll No.: 17EC35043
[¶]Supervising professor

3 APPROACH

3.1 Data Pre-processing

We perform pre-processing on the news articles to ensure that the base models get access to clean and processed text inputs. This includes removing punctuation, whitespaces, stopwords and performing lemmatization. We don't lower the case as we use a cased mBERT model.

3.2 Model Architecture

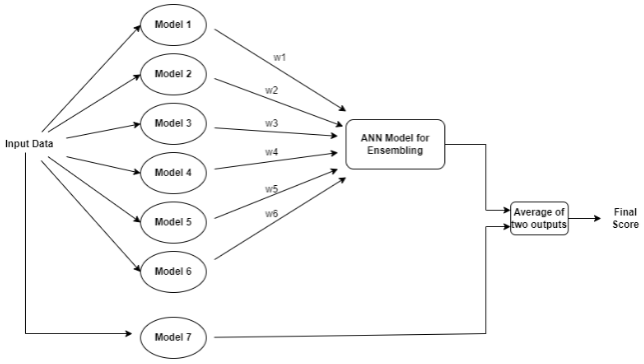

Figure 3: Model Architecture

For the final model we create an ensemble using stacked generalisation or stacking. We train six separate models that predict each of the given labels viz. "Geography", "Entities", "Time", "Narrative", "Style" and "Tone" given two input news articles using a mBERT model. These models constitute our Level-0 or Base models. Then we train an artificial neural network consisting of two hidden layers that takes as input the predictions of the above six models and outputs overall score S1. The ANN serves as a Level-1 or Meta model that builds upon the predictions of the base models. Then we train another mBERT model that directly predicts the overall score S2 from the input news article pair. The final "Overall" score is calculated as the mean of S1 and S2.

4 EXPERIMENTS

4.1 Baseline Approaches

In our baseline models 4, we have tried both Cross-encoding and Bi-encoding for sentence pair scoring.

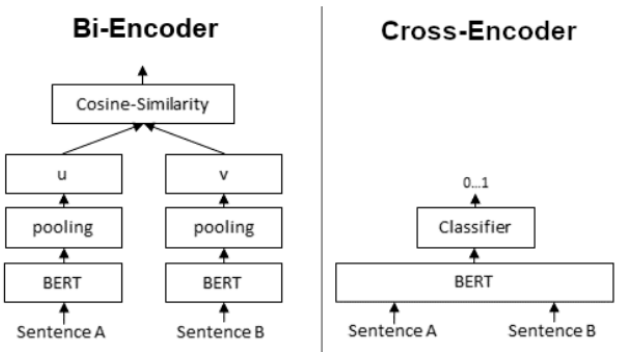

Figure 4: Baseline Approach

In bi-encoders, we pass the sentences A and B to a BERT[1][2][3][4] model independently which results in the sentence embeddings u and v after pooling. Pooling is done to reduce the sentences to a fixed size vector irrespective of their length. We have used mean pooling in our work. These sentence embedding[5] can then be compared using cosine similarity.

In contrast, for a Cross-Encoder, we pass both sentences simultaneously to the BERT network. The result is then passed through a classifier to produce an output value between 0 and 1 indicating the similarity of the input sentence pair. Since the overall score is from 1 to 4 we subtract 1 and divide by 3 to obtain the ground truth label.

The BERT which is used for performing semantic similarity is XLM-RoBERTa-Base[6]. XLM-RoBERTa is a multilingual version of RoBERTa. It is pre-trained on 2.5TB of filtered CommonCrawl data containing 100 languages.This model is primarily aimed at being fine-tuned on tasks that use the whole sentence to make decisions. For fine tuning we make use of 4938 news article pairs and evaluate the performance on 1680 pairs. The result is reported as the Pearson's correlation coefficient[7] between the ground truth labels in the test set and the predicted outputs by the model.

4.2 Variation of models and loss functions

Since the cross-encoder was performing better than bi-encoder we have used the former for all our future work. After getting the baseline results, we have experimented with few different loss func-

tions on the mBERT along with XLM-RoBERTa-Base and XLM-RoBERTa-Large models.
The mBERT is an advanced version of BERT, it is a multilingual variant of BERT that has been trained on and can be used with 104 languages. When mBERT was built, data from all 104 languages was combined. As a result, mBERT understands and is aware of word relationships in all 104 languages at the same time. XLM-RoBERTa models on the other hand are pre-trained on 2.5TB of filtered CommonCrawl data containing 100 languages. We have experimented with both XLM-RoBERTa-base as well as XLM-RoBERTa-large. The base model has 125M parameters with 12-layers, 768-hidden-state, 3072 feed-forward hidden-state, 8-heads whereas the large model has 355M parameters with 24-layers, 1027-hidden-state, 4096 feed-forward hidden-state, 16-heads.
In addition we experimented with different hyper-parameters such as learning rate, optimizer, scheduler to obtain the best possible results.

4.3 Final Model

We train an ensemble model as described in the Approach section and take the average of this score and the output from Model 7 (see Figure 3) to predict the Overall score.

5 RESULTS

5.1 Baseline Results

We estimate the Overall Similarity between two pairs of news stories. The similarity ratings are compared with the gold standard ratings using Pearson's correlation[7].

Approach Type	**Pearson's Coefficient**
Bi-Encoder	0.1289
Cross-Encoder	0.4831

Table 1: Pearson's correlation coefficient for baseline models

We clearly see from 1 that the cross encoder has performed better than the bi-encoder. This can be attributed to the fact that Cross-encoders perform full (cross) self-attention over both input texts while their counterparts, Bi-encoders perform self-attention over each input separately. Therefore the cross encoders are capable of learning much more than bi-encoders due to richer interactions.

5.2 Variation of models and losses

The Pearson's correlation with the ground truth labels on the evaluation dataset for different models and loss functions are shown in table 2.

Model Name	**MSE Loss**	**L1 Loss**
mBERT	0.7325	0.7198
XLM-RoBERTa-Base	0.5915	0.4276
XLM-RoBERTa-Large	0.3188	0.2234

Table 2: Pearson's correlation coefficient for different setups

5.3 Final Model Results

We now report the Pearson's Correlation Coefficient obtained for different base models between the predicted outputs and the actual labels in the evaluation dataset. The outputs are then passed through the meta ANN model and this output is then averaged with the output of Model 7 (as shown in Figure 3) to get the "Overall" score.

Label	**Pearson's Coefficient**
Geography	0.5875
Entities	0.7308
Time	0.4848
Narrative	0.7144
Style	0.5367
Tone	0.4901

Table 3: Pearson's correlation coefficient for Base/Level-0 models

The Pearson's correlation coefficient between the predicted score and the ground truth is **0.702**.

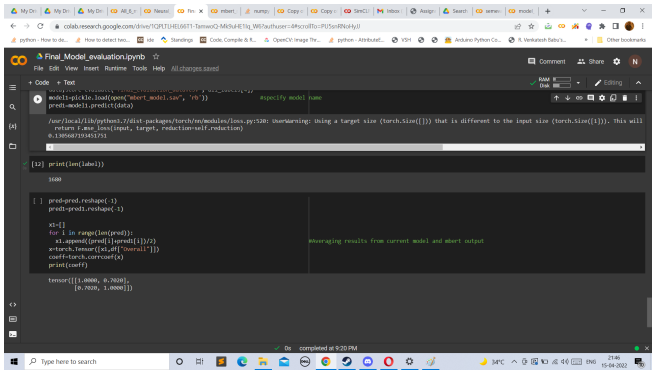

Figure 5: Final Pearson's Correlation Coefficient

6 Individual Contributions

6.1 Arpit Dwivedi - Literature review and Code Compilation

6.2 Pravartya Dewangan - Training of Ensemble Model

6.3 Parakh Agarwal - Training of Base Models

6.4 Tias Mondal - Training of different Cross-Encoder configurations

7 Future Steps

Different data augmentation techniques can be used to ensure better utilisation of the limited data available for fine-tuning. Further, the results could be analysed to understand the specific areas where the model is lacking and how that could be improved upon.

References

[1] J. Devlin, M. Chang, K. Lee, and K. Toutanova, "BERT: pre-training of deep bidirectional transformers for language understanding," *CoRR*, vol. abs/1810.04805, 2018.

[2] N. Reimers and I. Gurevych, "Sentence-bert: Sentence embeddings using siamese bert-networks," *CoRR*, vol. abs/1908.10084, 2019.

[3] S. Humeau, K. Shuster, M. Lachaux, and J. Weston, "Real-time inference in multi-sentence tasks with deep pretrained transformers," *CoRR*, vol. abs/1905.01969, 2019.

[4] H. Xiao, "bert-as-service." `https://github.com/hanxiao/bert-as-service`, 2018.

[5] Y. Yang, D. Cer, A. Ahmad, M. Guo, J. Law, N. Constant, G. H. Ábrego, S. Yuan, C. Tar, Y. Sung, B. Strope, and R. Kurzweil, "Multilingual universal sentence encoder for semantic retrieval," *CoRR*, vol. abs/1907.04307, 2019.

[6] A. Conneau, K. Khandelwal, N. Goyal, V. Chaudhary, G. Wenzek, F. Guzmán, E. Grave, M. Ott, L. Zettlemoyer, and V. Stoyanov, "Unsupervised cross-lingual representation learning at scale," *CoRR*, vol. abs/1911.02116, 2019.

[7] W. Kirch, ed., *Pearson's Correlation Coefficient*, pp. 1090–1091. Dordrecht: Springer Netherlands, 2008.