# Using Large Language Models in Translationese Classification

Student: Baciu Daniel Mihai
Scientific Coordinator: Lect. Dr. Bogdan Dumitru

University of Bucharest

July 10, 2023

# Table of Contents

# The problem addressed

The need to improve translationese classification and the potential of large language models (LLMs) to enhance translation detection accuracy.

As two languages can not be perfectly mapped with each other $\rightarrow$ translated text and its original can not be perfectly matched.

# Table of Contents

# The proposed solution

In the process of obtaining the best accuracy, I initially passed all the texts through the BERT model which was set in test mode, that is, backward() was not done through the network. Then I trained several Neural Networks (NN) to see which ones presented the best accuracy. The best models from Experiment 1 were used further, in Experiment 2.

# Table of Contents

# Technologies used

Technologies used:

- Python
- PyTorch
- BERT
- Hugging Face

# Table of Contents

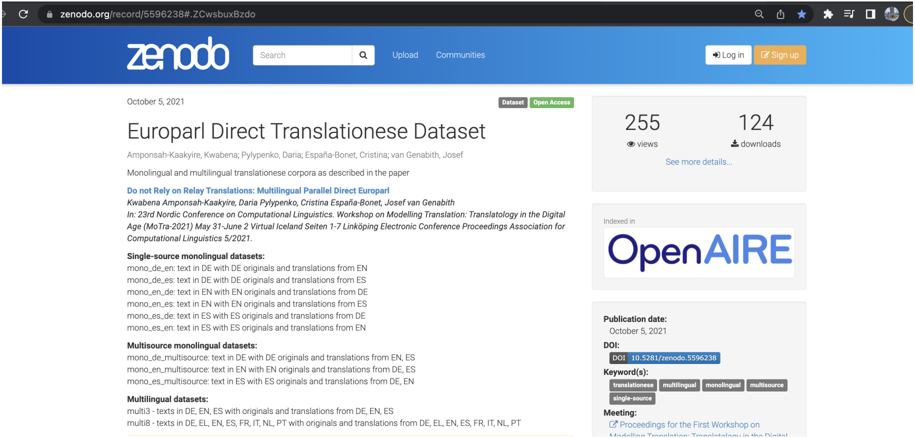


Figure 1: Europarl Direct Translationese Dataset

# Dataset

| Language of Origin | Filename | Rows | Label=1 | Label=0 |
|---|---|---|---|---|
| german | mono_en_de_dev.tsv | 6336 | 3168 | 3168 |
| german | mono_en_de_test.tsv | 6344 | 3172 | 3172 |
| german | mono_en_de_train.tsv | 29580 | 14790 | 14790 |
| spain | mono_en_es_dev.tsv | 6336 | 3168 | 3168 |
| spain | mono_en_es_test.tsv | 6344 | 3172 | 3172 |
| spain | mono_en_es_train.tsv | 29580 | 14790 | 14790 |

Figure 2: Dataset

# Table of Contents

Experiment 1 consisted of finding the best model for fine-tuning. The tested models exhibited variations in both the type and quantity of layers employed, as well as various activation functions. The number of epochs was the same for all, this being 50. The learning rate took values from the following array $[4 * 10^{-5}, 4 * 10^{-4}, 2 * 10^{-4}, 10^{-4}, 6 * 10^{-3}, 4 * 10^{-3}, 2 * 10^{-3}, 10^{-3}]$. The batch size remained constant at 32.

# Technical implementation
## Experiment 1


Figure 3: Experiment 1 - Results for Model 1

# Technical implementation

## Experiment 1

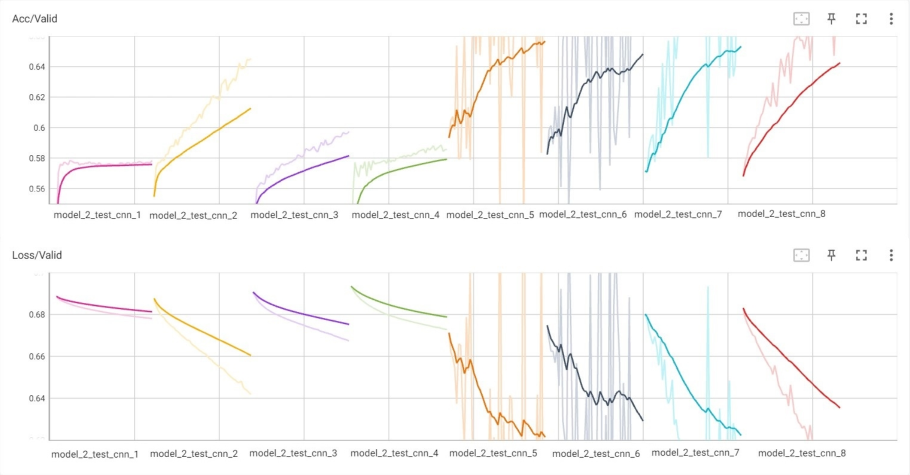


Figure 4: Experiment 1 - Results for Model 2

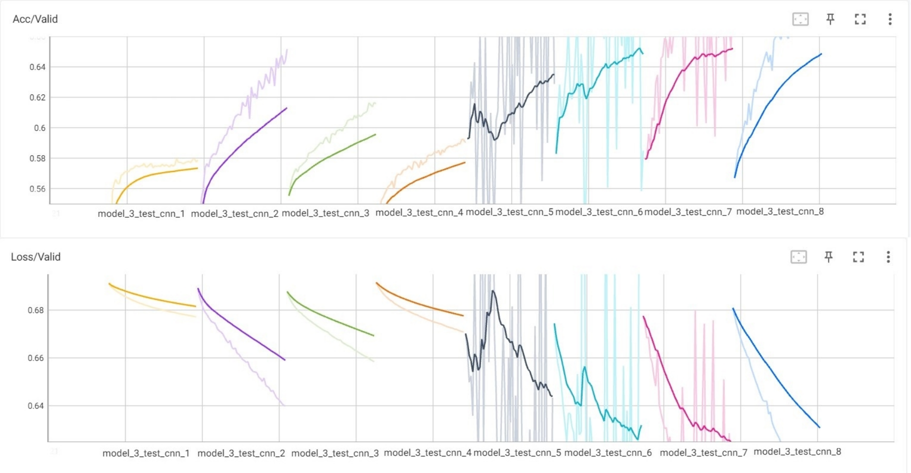


Figure 5: Experiment 1 - Results for Model 3

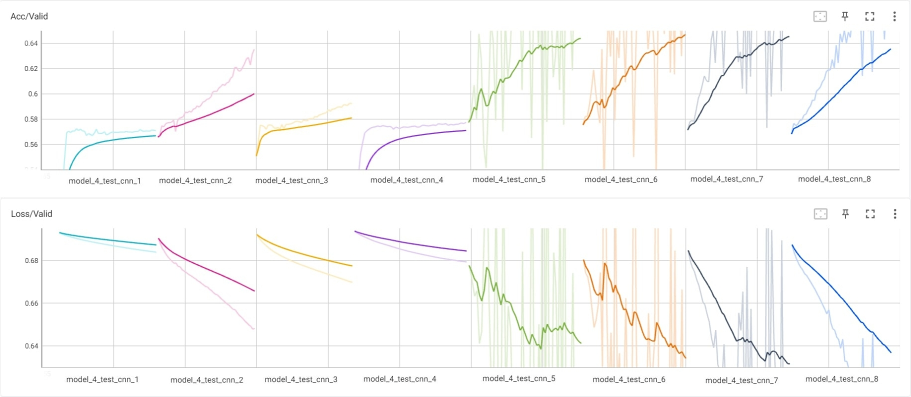


Figure 6: Experiment 1 - Results for Model 4

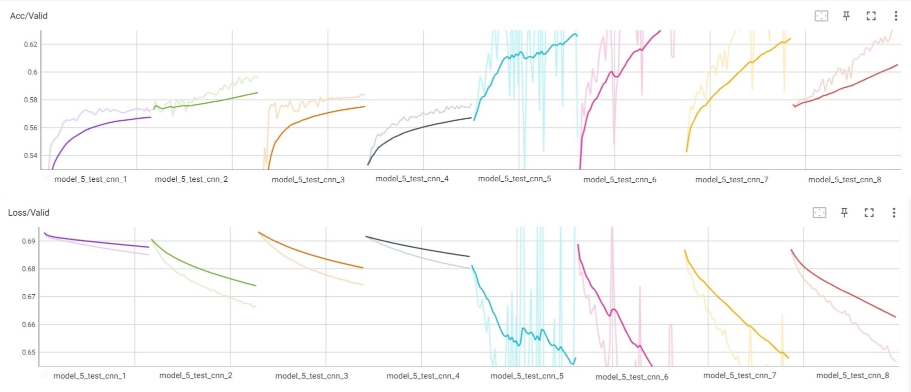


Figure 7: Experiment 1 - Results for Model 5

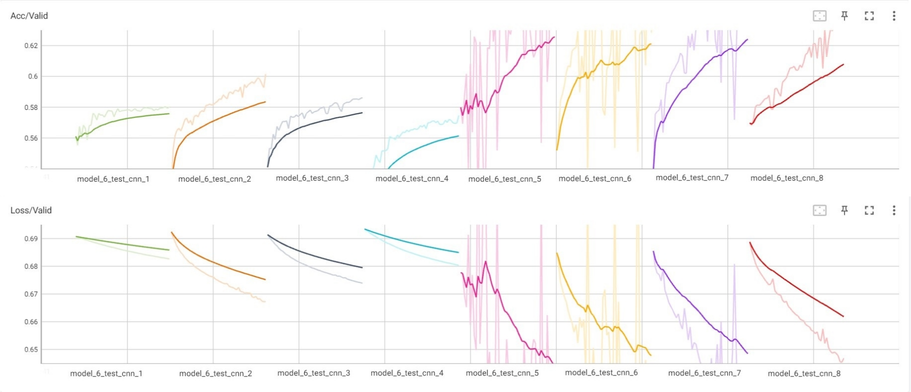


Figure 8: Experiment 1 - Results for Model 6

# Technical implementation

## Experiment 2

As already noticed, models 2, 3 and 4 had high accuracy, so we will use them to do fine tuning when we retrain the BERT model as well. The learning rate took the values: $4 * 10^{-5}, 2 * 10^{-5}$. The batch size was kept at 32.

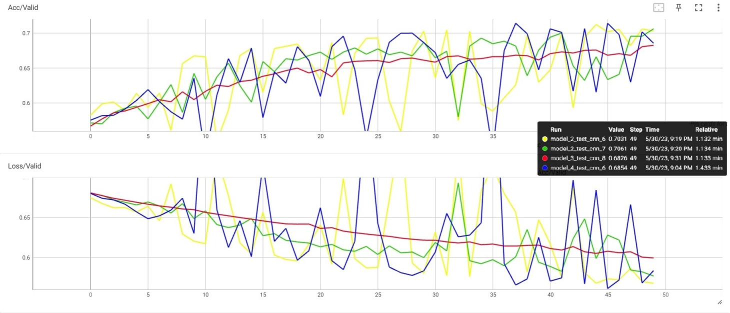


Figure 9: Experiment 1 - Results for Model 2, 3 and 4

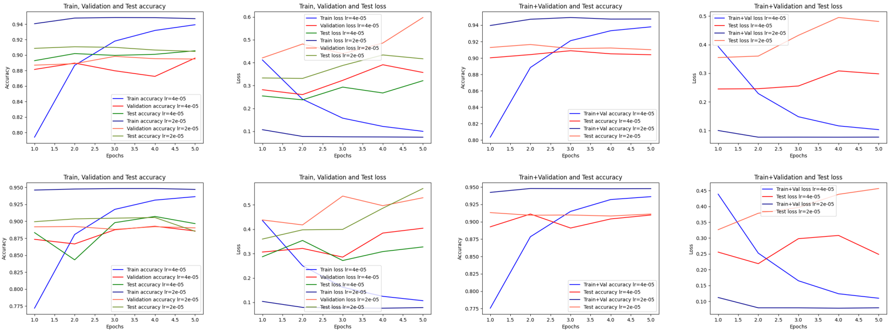


Figure 10: Results for Model 2 with and without BERT activation. Tested on Validation(left) and on Test(right)

# Technical implementation

## Experiment 2

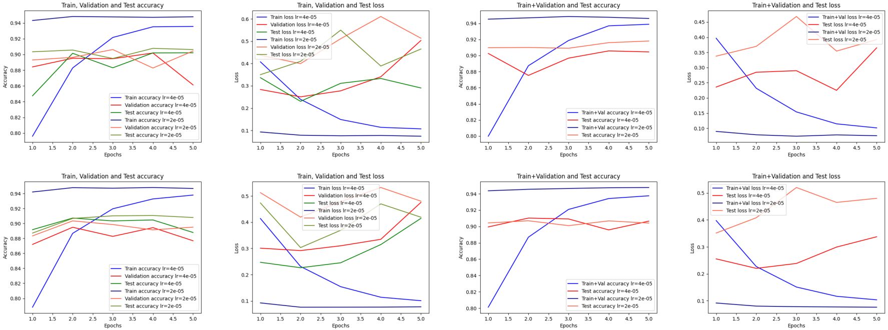


Figure 11: Results for Model 3 with and without BERT activation. Tested on Validation(left) and on Test(right)

# Technical implementation

## Experiment 2

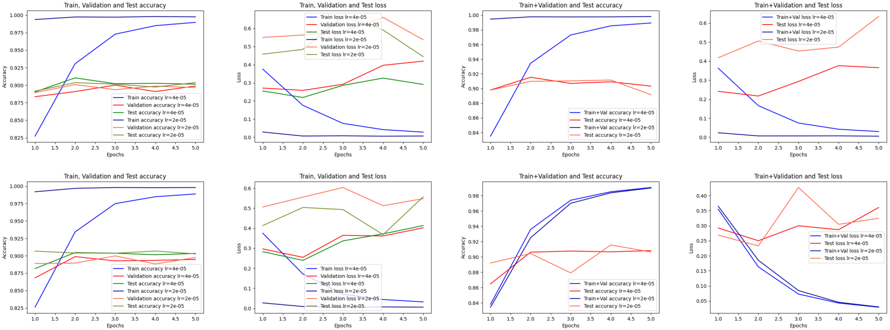


Figure 12: Results for Model 4 with and without BERT activation. Tested on Validation(left) and on Test(right)

# Table of Contents

# Results

## Experiment 1

| Model number | Accuracy | Loss |
| --- | --- | --- |
| Model 1 | 71.07% | 0.5611 |
| **Model 2** | **71.23%** | 0.5691 |
| **Model 3** | **71.4%** | 0.5682 |
| **Model 4** | **71.42%** | 0.5611 |
| Model 5 | 68.58% | 0.5994 |
| Model 6 | 69.03% | 0.6034 |

Table 1: BERT fine-tunning results: Experiment 1

# Results

## Experiment 2

| Model no. | Acc on Train | Loss on Train | Acc on Test | Loss on Test |
|---|---|---|---|---|
| DE-EN dataset | | | | |
| Model 2-A | 94.935% | 0.103 | 91.646% | 0.36 |
| Model 2-B | 94.785% | 0.11 | 91.315% | 0.326 |
| Model 3-A | 94.846% | 0.101 | 91.803% | 0.392 |
| Model 3-B | 94.771% | 0.103 | 91.031% | 0.408 |
| Model 4-A | 99.825% | 0.03 | 91.535% | 0.474 |
| Model 4-B | 99.067% | 0.03 | 91.567% | 0.361 |
| ES-EN dataset | | | | |
| Model 2-A | 95.097% | 0.102 | 91.992% | 0.325 |
| Model 2-B | 95.005% | 0.1 | 92.04% | 0.384 |
| Model 3-A | 94.963% | 0.102 | 92.055% | 0.409 |
| Model 3-B | 94.944% | 0.099 | 92.323% | 0.365 |
| Model 4-A | 99.833% | 0.03 | 91.992% | 0.404 |
| Model 4-B | 99.805% | 0.028 | 92.229% | 0.358 |
| A: BERT not activated, B: BERT activated | | | | |

Table 2: BERT results: Experiment 2

# Results

## Compared results

| Dataset | Model | Accuracy |
|---|---|---|
| DE-EN | *BERT** | **92.4%** |
| DE-EN | Model 3-A | 91.8% |
| ES-EN | *BERT** | 91.4% |
| ES-EN | Model 3-B | **92.32%** |
| *BERT**: BERT best result from paper [3] | | |

Table 3: Compared results

# Table of Contents

# Practic Applications of our model

The findings from this study have several practical applications that can benefit various stakeholders in the field of translation and language processing. Here are some practical applications of a BERT model trained to detect English translationese classification based on the research:

- Translation Quality Assessment: as an objective tool for translated-text quality assessment
- Translator Training and Feedback: as an educational tool for translator training programs
- Translation Memory Optimization: can receive suggestions or warnings when translationese patterns are detected in segments
- Machine Translation Improvement: can be integrated into machine translation(MT) systems to enhance their output quality

# Table of Contents

# Future Research Directions

- Cross-Linguistic Analysis: Extending the study to include a broader range of languages and language pairs can help uncover language-specific characteristics and identify common translationese patterns across different linguistic backgrounds.
- Addressing Ethical Concerns: As LLMs continue to grow and expand, ethical considerations become crucial. Future research should tackle diagnosis bias worries, fairness, and more importantly, transparency in translationese classification to ensure responsible and equitable use of these models.
- Using bigger models: For example, this paper is used bert-base-uncased which has 110M parameters. For future work, we may want to use bert-large-uncased which has 340M parameters and may be better to classify translationese

# Table of Contents

# Conclusion

By pursuing these future research directions, the field of translation studies can further leverage the capabilities of large language models and advance our extended comprehension of translationese, ultimately benefiting translation professionals and improving cross-linguistic communication.

# Table of Contents

# Bibliography I


[1] Marius Popescu. "Studying Translationese at the Character Level.". In: Jan. 2011, pp. 634–639.

[2] "Progress in Machine Translation". In: *Engineering* 18 (July 2021). DOI: 10.1016/j.eng.2021.03.023.

[3] Daria Pylypenko et al. "Comparing Feature-Engineering and Feature-Learning Approaches for Multilingual Translationese Classification". In: *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing*. Association for Computational Linguistics, Nov. 2021. DOI: 10.18653/v1/2021.emnlp-main.676. URL: https://aclanthology.org/2021.emnlp-main.676.

# Bibliography II


[4] Towards Data Science Rani Horev. *BERT Explained: State-of-the-Art Language Model for NLP.* https://towardsdatascience.com/bert-explained-state-of-the-art-language-model-for-nlp-f8b21a9b6270. Accessed on 10/05/2023.

[5] Ashish Vaswani et al. *Attention Is All You Need.* 2017. arXiv: 1706.03762 [cs.CL].

Thank You for Your Attention!