NLP CS Kaggle Challenge: Language Identification

GROUP 5

Bouhadida	**El Barhichi**	**Guelfane**	**Meziany**	**Yakhou**
Malek	Mohammed	Abdelaziz	Imane	Yousra

Abstract

This paper presents our approach to developing a robust multilingual language classifier. We began with thorough exploratory data analysis and rigorous data cleaning, then evaluated a broad set of models—from TF-IDF-based and FastText to advanced Transformer architectures such as BERT, mBERT, DeBERTaV3, mT5, XLM-RoBERTa, and RemBERT. Our analysis highlights both strengths and limitations of each method, revealing that an ensemble strategy surpasses any single model in classification performance. This final ensemble voting system achieved top rankings, demonstrating the value of combining complementary models for language identification.

1 Introduction

Language identification is the foundation of numerous natural language processing (NLP) tasks and becomes more complex in the context of multilingual data. The primary goal of this project is to develop a robust classifier capable of identifying the language of each text snippet in a diverse dataset. We were provided with labeled training data and unlabeled test data (hosted on Kaggle) to evaluate and compare different modeling approaches.

Our contributions are threefold:

(1) A thorough dataset analysis addressing anomalies and label inconsistencies.

(2) A suite of baseline models and fine-tuned large language models.

(3) An ensemble approach that significantly boosts overall performance.

Our implementation is publicly available here.

2 Solution

2.1 Data Preprocessing and Cleaning

The original dataset contained 190,600 samples spanning 389 distinct language labels. However, exploratory data analysis (EDA) revealed several issues, notably severe class imbalance, with sample counts per class ranging from as few as 1 to over 1,500. Additionally, we identified problematic instances, such as duplicate text samples assigned inconsistent labels. Another significant anomaly was the presence of a class labeled `'nan'`, corresponding to a Chinese dialect, which Kaggle mistakenly interpreted as missing values.

To address these anomalies, we applied rigorous preprocessing steps. First, we removed samples labeled `nan`, reducing the dataset size to 190,100 samples across the remaining labels. Duplicate entries with conflicting labels were either corrected or removed to ensure data integrity. Further cleaning involved removing noise-inducing elements like URLs, user mentions (`@user`), and hashtags (`#word`). Finally, we normalized textual data through lowercasing and Unicode normalization, resulting in a cleaner and more reliable dataset suitable for robust model training and evaluation.

2.2 Baseline Models

Our initial experiments evaluated baseline models to establish performance benchmarks. We used `TF-IDF` representations, transforming text data into sparse feature vectors with character-level and word-level n-grams, and evaluated classical classifiers including `Logistic Regression` and `SGD Classifier`. These baseline models achieved accuracy in the range of **74–75%**, providing a foundational benchmark for subsequent experiments.

2.3 FastText with Hyperparameter Tuning

Building upon our baseline experiments, we explored `FastText`, known for efficiently handling multilingual data through n-gram embeddings. To maximize performance, we employed a comprehensive grid search with stratified cross-validation, optimizing parameters such as learning rate, epochs, embedding dimensions, and n-gram sizes. This optimization raised our accuracy to around **79%**,

surpassing baseline methods and indicating the importance of hyperparameter tuning.

2.4 Advanced Transformer-Based Approaches

To boost performance, we fine-tuned several transformer models using different hyperparameter settings. We found that some architectures performed best on a fully cleaned dataset (e.g., BERT-base, DeBERTa-v3, XLM-RoBERTa), while others (e.g., mBERT and mT5) benefited from a lighter preprocessing pipeline. Briefly, our experiments included:

- **BERT (base)**: An encoder-only model (110M parameters) pre-trained with MLM and NSP, excelling on cleaned data.
- **Multilingual BERT (mBERT)**: Pre-trained on 104 languages, yielding robust performance with minimal preprocessing.
- **DeBERTa-v3**: With 180M parameters and enhanced attention mechanisms, it provided strong semantic understanding.
- **XLM-RoBERTa**: A multilingual model (270M parameters) trained on 100+ languages, achieving high accuracy after tuning.
- **RemBERT**: Optimized for multilingual representation, performing well with fine-tuning.
- **mT5**: A text-to-text model (580M parameters) where we prepend "identify language:" to each input, generating the language code; it performed best with a lightly preprocessed dataset

Each model was fine-tuned using cross-entropy loss and tailored training parameters (learning rate, number of epochs, batch size) optimized for its architecture, achienving validation accuracies ranging from **85% to 88%**. This diverse experimental setup informed our subsequent ensemble voting strategy, where model predictions were combined to leverage their complementary strengths.

2.5 Ensemble Voting Strategy

While individual models demonstrated strong classification capabilities, we observed that their performance varied across different language classes. To leverage the complementary strengths of each model, we implemented an **ensemble voting** mechanism. In this scheme, each model casts a vote for the predicted language. In cases of ties, the prediction from the highest-performing model is selected. This strategy improved overall accuracy by mitigating individual model weaknesses, especially for minority languages.

3 Results and Analysis

The results for the models discussed in the previous section are presented in Table 1. The metrics chosen to compare these models are the accuracy obtained on a common test set (10% of the available data) (i.e., Best Validation Score) and the accuracy of the model on the Kaggle final test set (i.e., Best Kaggle Score):

Model	Best Validation Score	Best Kaggle Score
TF-IDF + SGD	0.7536	0.7432
FastText	0.7820	0.7946
mT5	0.8186	0.7931
BERT	0.8386	0.8267
mDeBERTa	0.8451	0.8629
mBERT	0.8800	0.8656
XLM-Roberta	0.8771	0.8710
RemBERT	0.8769	0.87445
Ensemble Method	-	**0.8950**

Table 1: Performance of different models

The **ensemble approach** further boosts performance, achieving the best Kaggle score of 0.8950. By combining the strengths of different models, it reduces misclassification errors and enhances robustness across various language classes. This improvement demonstrates the effectiveness of ensemble learning in mitigating the weaknesses of individual models and leveraging their complementary strengths.

4 Conclusion

In this work, we developed a robust multilingual language identification system for 389 languages by combining thorough data preprocessing, extensive baseline experimentation, and the fine-tuning of advanced transformer-based models. Our initial baselines (TF-IDF and FastText) provided a solid foundation, while the transformer models (including BERT, mBERT, DeBERTa-v3, XLM-RoBERTa, RemBERT, and mT5) achieved significantly higher accuracies, ranging from 85% to 88% on validation data. By integrating these diverse models using an ensemble voting strategy—prioritizing the best-performing predictions—we boosted overall performance, culminating in a final Kaggle score of 89.5% and a top ranking.

References

[1] Hyung Chung, Minsik Lee, et al. Rembert: Improved multilingual representation by resetting bert. *arXiv preprint arXiv:2012.15723*, 2020.

[2] Alexis Conneau, Kartik Khandelwal, Naman Goyal, Vishrav Chaudhary, Guillaume Wenzek, Francisco Guzmán, Édouard Grave, Armand Joulin, and Veselin Stoyanov. Unsupervised cross-lingual representation learning at scale. In *Proceedings of ACL 2020*, pages 8440–8451, 2020.

[3] Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. Bert: Pre-training of deep bidirectional transformers for language understanding. *arXiv preprint arXiv:1810.04805*, 2018.

[4] Thomas G. Dietterich. Ensemble methods in machine learning. In *Multiple Classifier Systems*, pages 1–15. Springer, 2000.

[5] Jérémy Fix, Stephane Vialle, Rémi Hellequin, Claudine Mercier, Patrick Mercier, and Jean-Baptiste Tavernier. Feedback from a data center for education at centralesupélec engineering school. In *2022 IEEE International Parallel and Distributed Processing Symposium Workshops (IPDPSW)*, pages 330–337, 2022.

[6] Pengcheng He, Xiaodong Liu, Jianfeng Gao, and Weizhu Chen. Deberta: Decoding-enhanced bert with disentangled attention. In *International Conference on Learning Representations*, 2021.

[7] Armand Joulin, Edouard Grave, Piotr Bojanowski, and Tomas Mikolov. Bag of tricks for efficient text classification. In *International Conference on Learning Representations*, 2017.

[8] Lin Xue et al. mt5: A massively multilingual pre-trained text-to-text transformer. *arXiv preprint arXiv:2110.07396*, 2021.