Team ISCL_WINTER at SemEval-2023 Task 12:AfriSenti-SemEval: Sentiment Analysis for Low-resource African Languages using Twitter Dataset

Alina Hancharova and **John Wang** and **Mayank Kumar**
Eberhard Karls University of Tübingen

Abstract

This paper presents a study on the effectiveness of various approaches for addressing the challenge of multilingual sentiment analysis in low-resource African languages. The study focuses on Task 12 of the SemEval-2023 Competition, which aims to promote interest in these languages and develop efficient models for their analysis. The approaches evaluated in the study include Support Vector Machines (SVM), translation, and an ensemble of pre-trained multilingual sentimental model methods. The paper provides a detailed analysis of the performance of each approach based on experimental results. In our findings, we suggest that the ensemble method is the most effective with an F1 Score of 0.68 on the final testing. This system ranked 19 out of 33 participants in the competition.

1 Introduction

There is a growing interest in using AI for various natural language processing (NLP) tasks such as sentiment analysis, machine translation, and hateful content detection in African languages. However, most of these languages do not have curated datasets available for developing such AI applications. Recently, there have been individual and funded initiatives to create datasets for African languages, but research is needed to determine the suitability of current NLP techniques and the development of techniques to maximize the applications of such datasets. Subtask B, Task 12 of shared task the SemEval-2023 Competition Muhammad et al. (2023b) focus on 12 African languages and is designed to strengthen their further development by including languages such as Hausa, Yoruba, Igbo, Nigerian Pidgin, Amharic, Tigrinya, Oromo, Swahili, Algerian Arabic dialect, Kinyarwanda, Twi, Mozambique Portuguese, and Moroccan Arabic/Darija. In this study, the goal is to tackle the problem of sentiment analysis in African languages, which are under-represented in natural language processing (NLP) research. To address this challenge, we employed an ensemble strategy using deep language pre-trained models. These models were individually able to handle the languages present in our dataset, which included 12 African languages.

1.1 Competition Details

The competition is divided into 3 sub-tasks. In the first task, a language is chosen and sentiment analysis is performed. These languages are Hausa, Yoruba, Igbo, Nigerian Pidgin, Amharic, Algerian Arabic, Moroccan Arabic/Darija, Swahili, Kinyarwanda, Twi, Mozambican Portuguese, and Xitsonga (Mozambique Dialect), Setswana, TisiZulu, Xitsonga (South African Dialect). Languages are related to Afro-Asiatic, Niger-Congo, and Semitic families. These languages are low resource and diverse. In the second task, the participant works on the combination of the 12 languages database. The third subtask, unlabelled tweets in two African languages (Tigrinya and Oromo), and zero-shot sentiment analysis are performed. For the competition participation, our team chose the second task.

1.2 Dataset Description

The AfriSenti dataset is a set of tweets, written in one of the African languages for sentiment analysis. The dataset involves tweets labeled with three sentiment classes (positive, negative, and neutral). Each tweet is annotated by three annotators following the annotation guidelines in Muhammad et al. (2023a).

1.3 Approach Overview

The described system is focused on the second task, which does sentiment analysis in all 12 languages. Our main contribution is based on an ensemble consisting of pre-trained language sentiment analyzers. Also, we experiment with different approaches like

Proceedings of the The 17th International Workshop on Semantic Evaluation (SemEval-2023), pages 1085–1089

SVM and translations. The final ensemble uses Roberta-based Barbieri et al. (2020), Bert-mini Bhargava et al. (2021), and a model trained on MFAQ dataset Bruyn et al. (2021). All models are pre-trained and run with the same metrics.

2 Background

2.1 Languages

The large variety of languages poses an interesting problem. African languages comprise a wide variety of languages that are not necessarily related, unlike the languages of Europe. Languages such as Hausa and Arabic belong to the Afro-Asiatic language family, while Swahili is a Bantu language, which belongs to the Niger-Congo family. Mozambique Portuguese is a language spoken in Africa due to colonialism and is a member of the Indo-European family. Nigerian Pidgin is another non-typical language for NLP pre-processing.

2.2 Earlier works

To our knowledge, the AfriSenti-SemEval task has not been previously presented. However, it is related to shared tasks on sentiment analysis focusing on Arabic dialects from African countries such as Algerian Arabic and Tunisian Arabic. Previous research in this area includes studies by various authors, however, they were not consulted when designing our approach. Only their models were used, which includes Muhammad et al. (2023a), Bruyn et al. (2021), Barbieri et al. (2020) and Bhargava et al. (2021).

3 System Overview

Descriptions of approaches that were attempted are given here.

3.1 Ensemble Approaches

The main approach was an ensemble method. Ensemble methods use multiple models and combine them to get better results than any individual.

The particular ensemble method is hard voting, which entails summing the predictions for each class label and predicting the class label with the most votes. This approach has the advantage that it is simple to implement and does not require probabilities.

Initially, the ensemble contained three models, which were later expanded to five models.

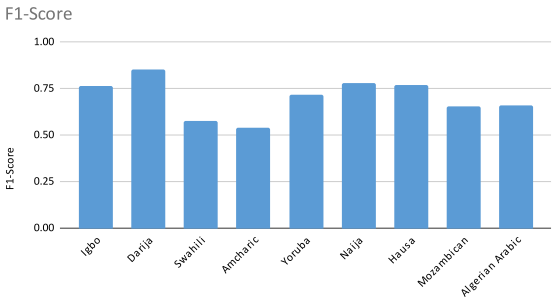

Figure 1: Per Language breakdown for the ensemble of five models.

3.2 English Approach

Two reasons stand for translating the dev dataset into English. First, the majority of models that were chosen for an ensemble were trained in English or European language datasets. Second, we want to manually check the result random slice and classify the dataset. For the translation, the Google Translate library was used. Additionally, since some of the data was in both English and another language, translating to English may yield better results, since it would be in one language. After checking the random slice of the translated tweets we found some sentences meaningless or not translated. Examples such as *a spec and half sef* seems to contain an untranslated lexeme and some such as *I want to come in, to 'smiti' lips* have unclear meaning.

3.3 Arabic Approach

An alternative to English was the Arabic language, which is distantly related to a few languages. This was the reason to use the model by Ali trained in Arabic to improve results for a group of languages that are presented in Arabic script (Darija and Algerian Arabic). Also, since the classifier for Arabic performed decently, we wanted to try a variant of the translation approach before giving up on translation. The result of the performance in those languages led to the translation of the multilingual dataset into Arabic and then the translated dataset was applied to the ensemble.

3.4 Support Vector Classifier

In the first trial with Hausa language data, the F1 scores for negative and positive tweets were around 0.6, while the neutral class achieved only 0.2. Subsequently, we applied downsample and stop words

	No Emojis		Emojis	
	recall	precision	recall	precision
positive	0.74	0.65	0.84	0.8
neutral	0.68	0.8	0.68	0.7
negative	0.72	0.66	0.7	0.71
overall	0.71		0.74	

Table 1: Emoji Analysis

list parameters, but the model's performance decreased compared to its performance without these parameters. We got similar problems on our run with the random forest model and the boost model.

4 Experimental setup

We mostly modified the included starting code [1] Muhammad et al. (2023b). The main file used was the Jupyter Notebook, which contained a set-up code that would retrieve a model from Hugging Face and train it on the data. We ran the notebook on Google Colab. While we left most of the hyperparameters unchanged, we did adjust the `learning_rate` parameter, which had been incorrectly used in the original code at Codalab, in the cell of applying data into the train set.

5 Results

The initial ensemble consisted of `Davlan/afro-xlmr-mini` by Alabi et al. (2022), `prajjwal1/bert-tiny` by Bhargava et al. (2021), `clips/mfaq` by Bruyn et al. (2021), it resulted in an F1 Score of 0.687.

For final the version, the initial ensemble was expanded with `finiteautomata/bertweet-base-sentiment-analysis` by Pérez et al. (2021), `cardiffnlp/twitter-roberta-base-sentiment-lates` by Barbieri et al. (2020).

Running the ensemble, we obtained the following results: The performance of the ensemble during the development stage was a 0.715 F1 Score, and 0.68 during the testing stage, respectively. To analyze the performance in detail, we conducted quantitative research by comparing precision and recall for each label (depending on the existence of emojis inside tweets) and presenting the F1 score for each language. Based on the dev set and the predictions of our ensemble, we analyze and collect statistics for each language. We discovered that the performance of languages like Igbo, Hausa, and Amharic did not change compared to the `Davlan/afro-xlmr-mini` model result for a single language. On the other hand, languages like Darija showed an increase of 19%, while the results of other languages changed by less than 1-6%. The results of the F1 Score on each language illustrate in Figure 1 Additionally, we tested the theory that emojis have a significant influence on the results. Since tweets with emojis comprise only 1/3 of the whole set, we determined the proportion of wrongly and correctly predicted tweets according to the presence of emojis in the tweet. The results on proportion show that tweets in emojis are predicted better than without. The results are presented in Table 1. It is also seen in a training experiment in section 6.

The experiments with translations resulted in the following data. The English translation approach performs lower with a result 0.559 F1 Score compare to 0.62 of the `Davlan/afro-xlmr-mini` model (a default start-kit model).

The `Davlan/afro-xlmr-mini` F1 Score for Darija is 0.66 and for Algerian Arabic is 0.53. These results are among the lowest for this model in presented languages. To increase results, the Arabic sentimental analyses model was chosen. The model by Ali trained in Arabic performed higher in certain languages than the `Davlan/afro-xlmr-mini`. For Darija language F1 Score is 0.87 and for Algerian Arabic, F1 Score is 0.70. As a result, we attempted to translate the languages into Arabic and then used the Arabic model to make predictions. For this Google Translate library was used. The model performed worse (F1 Score:0.32) than our ensemble on the multilingual data set without translation (F1 Score: 0.715). Another idea was to include the model in the ensemble and make the decision of this model on Arabic tweets is the most preferable and ignore its decisions on other languages. The result was a 0.70 F1 Score compared to the 0.715 F1 Score of the ensemble result.

6 Conclusion

Experimenting with various approaches, we came for several ideas that may impact the performance. In our view, the most challenging fact is a lack of data material. Probably the increase in examples and more specific description of parameters to mod-

[1] https://github.com/afrisenti-semeval/afrisent-semeval-2023

	SVM	afro-xlmr-mini	Ensemble
Darija	-	0.53	0.85
Najia	-	0.76	0.78
Hausa	0.6	0.75	0.77
Swahili	-	0.52	0.57
Amharic	-	0.53	0.53

Table 2: F1 Score of the baseline model compare to the ensemble

els like SVM and Random Forest might make it more accurate.

Our approach worked best when we used the ensemble approach. Through an extensive number of experiments on individual languages, we discovered that our system had better performance than the baseline model provided in the starter kit on languages with Arabic script with the best-performing languages such as Darija, Najia, and Hausa, and the worst performing languages were Swahili and Amharic. The reason for this is unknown. The F1 Score of these languages is presented in Table 2 The lower performance on Arabic scripts highlights the potential of using the Arabic language-based pre-trained model as a base for the sentiment analysis in African languages and the importance of considering language relationships when developing NLP models for under-represented languages. We also found that striping the tweets of the emojis cause a loss of about 2 percent in the accuracy of prediction.

The attempt to translate the datasets into one language is reasonable, in the output you get a massive amount of data that can be classified efficiently. However, it does not seem to work with African languages. Translating a part of the dataset in English we found most of the tweets meaningless as there is a heavy amount of loss of data while translating the tweets, that could be a reason we got especially lower results trying to classify the tweets on translations. Nonetheless, we acknowledge that using more sophisticated translation libraries may improve performance. Also, we should mention that we did not use any lexicons and cleaning data techniques that might potentially raise the accuracy. The most efficient idea, on our point, was a voting ensemble. Meanwhile, it has advantages and disadvantages. With the ensemble, we achieve the highest score on development data (0.72). Second, comparing the results, we have noticed that the ensemble predicts significantly better languages with Arabic script (an increase of 19% compare to the default model performance shown for the Darija language). However, tuning and running those models take a lot of time and computational resources. In the future, it will be a good idea to find the more productive solutions – that we found - for each language and combine them with simple language classifiers.

References

Jesujoba O. Alabi, David Ifeoluwa Adelani, Marius Mosbach, and Dietrich Klakow. 2022. Adapting pre-trained language models to African languages via multilingual adaptive fine-tuning. In *Proceedings of the 29th International Conference on Computational Linguistics*, pages 4336–4349, Gyeongju, Republic of Korea. International Committee on Computational Linguistics.

Ammar Alhaj Ali. Ammar-alhaj-ali/arabic-marbert-sentiment. https://huggingface.co/Ammar-alhaj-ali/arabic-MARBERT-sentiment.

Francesco Barbieri, Jose Camacho-Collados, Luis Espinosa-Anke, and Leonardo Neves. 2020. TweetEval:Unified Benchmark and Comparative Evaluation for Tweet Classification. In *Proceedings of Findings of EMNLP*.

Prajjwal Bhargava, Aleksandr Drozd, and Anna Rogers. 2021. Generalization in NLI: Ways (not) to go beyond simple heuristics.

Maxime De Bruyn, Ehsan Lotfi, Jeska Buhmann, and Walter Daelemans. 2021. MFAQ: a multilingual FAQ dataset.

Shamsuddeen Hassan Muhammad, Idris Abdulmumin, Abinew Ali Ayele, Nedjma Ousidhoum, David Ifeoluwa Adelani, Seid Muhie Yimam, Ibrahim Sa'id Ahmad, Meriem Beloucif, Saif M. Mohammad, Sebastian Ruder, Oumaima Hourrane, Pavel Brazdil, Felermino Dário Mário António Ali, Davis David, Salomey Osei, Bello Shehu Bello, Falalu Ibrahim, Tajuddeen Gwadabe, Samuel Rutunda, Tadesse Belay, Wendimu Baye Messelle, Hailu Beshada Balcha, Sisay Adugna Chala, Hagos Tesfahun Gebremichael, Bernard Opoku, and Steven Arthur. 2023a. AfriSenti: A Twitter Sentiment Analysis Benchmark for African Languages.

Shamsuddeen Hassan Muhammad, Idris Abdulmumin, Seid Muhie Yimam, David Ifeoluwa Adelani, Ibrahim Sa'id Ahmad, Nedjma Ousidhoum, Abinew Ali Ayele, Saif M. Mohammad, Meriem Beloucif, and Sebastian Ruder. 2023b. SemEval-2023 Task 12: Sentiment Analysis for African Languages (AfriSenti-SemEval). In *Proceedings of the 17th International Workshop on Semantic Evaluation (SemEval-2023)*. Association for Computational Linguistics.

Juan Manuel Pérez, Juan Carlos Giudici, and Franco Luque. 2021. pysentimiento: A Python toolkit for sentiment analysis and socialnlp tasks.