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

Shashank Rathi
PICT, India
shashankrathi2@gmail.com

Siddhesh Pande
PICT, India
siddheshspande@gmail.com

Harshwardhan Atkare
PICT, India
atkareharsh@gmail.com

Rahul Tangsali
PICT, India
rahuul2001@gmail.com

Aditya Vyawahare
PICT, India
aditya.vyawahare07@gmail.com

Dipali Kadam
PICT, India
ddkadam@pict.edu

Abstract

This paper presents a summary of our findings obtained on sentiment analysis of 3 African languages among the 17 languages mentioned in the shared task. We carried out a sentiment analysis on Hausa, Yoruba, and Swahili. The models used here for the mentioned task were logistic regression, SVM, RandomForest, and mBERT along with a few data-preprocessing and oversampling techniques. The performance of the models used was evaluated by considering weighted average and macro average F1 scores as metrics. The best set of scores obtained on the languages Hausa, Yoruba and Swahili are (76.53, 76.55), (74.83, 73.15) and (57.79, 48.59) respectively.

1 Introduction

Most languages are becoming digitally accessible as a result of the growing usage of the Internet and social media platforms. This enables numerous artificial intelligence (AI) applications that make it possible to do operations like sentiment analysis, machine translation, and the detection of offensive information. 2,058 or 30 percent of all surviving languages, according to UNESCO (2003), are African languages. The majority of these languages, however, lack curated datasets for creating these AI applications. The Lacuna Fund is one recent example of a financed program that aims to buck this trend and provide such datasets for African languages. To maximise the use of these datasets, new natural language processing (NLP) techniques must be developed, and research is needed to ascertain whether the present approaches are appropriate in both cases. The shared task covers 17 African languages namely Hausa, Yoruba, Igbo, Nigerian Pidgin from Nigeria, Amharic, Tigrinya, and Oromo from Ethiopia, Swahili from Kenya and Tanzania, Algerian Arabic dialect from Algeria, Kinyarwanda from Rwanda, Twi from Ghana, Mozambique Portuguese from Mozambique and Moroccan Arabic/Darija from Morocco(Muhammad et al., 2023b).

The major system strategies used here include using some traditional classifier models like logistic regression, SVM, and RandomForest. They were used through various pipelines. These pipelines also included a vectorizer and over-sampling tools like ROS and SMOTE. One more result per language was obtained using the mBERT model. All the results obtained were compared and some conclusions can be drawn based on that analysis. The evaluation metrics used in this shared task were average-macro and average-weighted F1 scores.

Since in the shared task, Afrisenti SemEval 2023 we had to deal with African languages it helped us gain familiarity with handling multi-lingual data. We explored and researched different models and data-preprocessing techniques available for such kind of sentiment analysis tasks.

2 Background

Here, we have used the dataset provided in the shared task itself. Thus, the input contains tweets to be classified in respective languages. We have used a vectorizer in our pipeline for data-preprocessing purposes. The output is the classification of that particular tweet into 'positive', 'negative', or 'neutral' sentiments. The tracks we participated in include languages Hausa (track 1), Swahili (track 8), and Yoruba (track 2).

3 System Overview

In this shared task, we have used the following models and algorithms:

3.1 Logistic Regression

A supervised classification algorithm is essentially what logistic regression is. For a given collection of features (or inputs), X, in a classification issue, the target variable (or output), y, can only take discrete

Proceedings of the The 17th International Workshop on Semantic Evaluation (SemEval-2023), pages 1161–1165
July 13-14, 2023 ©2023 Association for Computational Linguistics

values(Cox, 1958). Only when a decision threshold enters the picture does logistic regression transforms into a classification technique. Precision and Recall levels play a significant role in determining the threshold value. It is based upon the sigmoid function. This Logistic Regression classifier was combined with oversampling techniques like ROS and SMOTE by creating a pipeline. This model when combined with SMOTE produced the best overall performance for Swahili language.

3.2 Random Forest

Random forest is an ensemble technique capable of handling both regression and classification tasks(Ali et al., 2012). The fundamental idea behind this is to mix numerous decision trees to determine the final output rather than depending just on one decision tree. As its primary learning models, Randomforest uses a variety of decision trees. Row and feature sampling are done at random from the dataset to create sample datasets for each model. It is known as Bootstrap. In a classification problem, the majority voting classifier is used to determine the final output. The mean of every output is the final result in the regression problem. This is called Aggregation.In this classifier, we set the n-estimators parameter to 100. This Random Forest ensemble classifier was combined with oversampling techniques like ROS and SMOTE by creating a pipeline.

3.3 Support Vector Machine

Both classification and regression tasks can be performed using supervised learning algorithms found in Support Vector Machines, or SVMs. Due to its resilience, it is frequently used to resolve classification challenges. The data points in this technique are initially shown in an n-dimensional space. The algorithm then employs statistical techniques to identify the best line that divides the different groups represented in the data(Hearst et al., 1998). Here, we have used SVC() classifier for our classification task. Here, we have selected "linear" kernel for the classifier. This classifier was combined with oversampling techniques like ROS and SMOTE by creating a pipeline. This model when combined with SMOTE produced the best overall performance for Yoruba language.

3.4 mBERT

BERT stands for Bidirectional Encoder Representations from Transformers(Vaswani et al., 2017). In order to pre-train deep bidirectional representations from unlabeled text, it simultaneously conditions on both the left and the right context(Devlin et al., 2019). As a result, state-of-the-art models for a variety of NLP applications can be developed using the pre-trained BERT model with just one additional output layer. Half of the secret of BERT's success lies in this pre-training phase. This is due to the fact that as a model is trained on a big text corpus, it begins to grasp the more intricate details of how the language functions. This information serves as a kind of all-purpose NLP toolkit. mBERT is a version of BERT that may be used with 104 different languages(Libovický et al., 2019). When mBERT was developed, information from all 104 languages was combined. As a result, mBERT simultaneously comprehends and is aware of the links between words in all 104 languages.

Here, we have used the ktrain API for the implementation of mBERT. The Keras deep learning software framework includes the library called ktrain that may be used to create, train, test, and deploy neural networks(Maiya, 2020). (As of version 0.7, ktrain leverages TensorFlow's tf.keras rather than Keras alone.) With just a few lines of code, ktrain, which was inspired by the fastai library, enables you to easily estimate an optional learning rate by providing a learning rate finder. It also allows you to employ learning rate schedules as well as fast and easy-to-use pre-canned models for the classification of text data.

4 Experimental Setup

4.1 Data Splits

We have used the dataset provided in the shared task(Muhammad et al., 2023a). We used the 'language-train.tsv' datasets for training the models, and the 'language-dev.tsv' datasets for validation purposes and generated predictions using mentioned models on 'language-test.tsv' for respective languages.

4.1.1 Data-split sizes

Language	Train	Dev	Test
Hausa	14172	2677	5303
Yoruba	8552	2090	4515
Swahili	1810	453	748

4.2 RandomOverSampler

A single instance may be chosen more than once because Random Oversampling involves selecting random instances from the minority class with replacement and augmenting the training data with multiple copies of this instance(Moreo et al., 2016). We have used the RandomOverSampler pipeline which consisted of a vectorizer, a classifier, and the RandomOverSampler. Here, we set the hyperparameter random-state to 777.

4.3 SMOTE

SMOTE is an algorithm that adds artificial data points to the actual data points to accomplish data augmentation(Bowyer et al., 2011). SMOTE can be viewed as an improved form of oversampling or as a particular data augmentation procedure. We created a pipeline for SMOTE which consisted of a vectorizer and a classifier. Here, we set the hyperparameter random-state to 777.

4.4 TF-IDF

Term Frequency Inverse Document Frequency is referred to as TF-IDF. This is a widely popular algorithm that converts text into meaningful numerical representations that can be used to fit machine prediction algorithms(Sammut and Webb, 2010). We have used this vectorizer in the pipeline by tuning it to unigrams and bigrams. The max-features parameter was set to 100,000.

4.5 Evaluation Metrics

The evaluation metrics used in this task are weighted-average and macro-average F1 score. The harmonic mean of recall and precision is referred to as the F1-score for a more balanced summary of model performance(Yedidia, 2016).

4.5.1 macro-average F1

The arithmetic mean, also known as the unweighted mean, of all the per-class F1 scores is used to calculate the macro-averaged F1 score, also known as the macro F1 score. Regardless of the support values, all classes are treated equally by this method.

4.5.2 weighted-average F1

The weighted-averaged F1 score is determined by averaging all of the per-class F1 scores while accounting for the support of each class. The term "weight" essentially refers to the share of support for each class in relation to the total worth of support.

5 Results

Table-1 contains the two types of f1-scores i.e. weighted and macro for the various models on which test data of the Hausa language was tested. From the data we see that multilingual BERT performed best among four models (some with different types of sampling) used for testing by giving a value of 76.53 and 76.55 percentage on weighted and macro types of f1-score respectively.

Table-2 contains the two types of f1-scores i.e. weighted and macro for the various models on which test data of the Swahili language was tested. From the data we see that logistic regression (SMOTE) performed best among four models (some with different types of sampling) used for testing by giving a value of 57.79 and 48.59 percentage on weighted and macro types of f1-score respectively.

Table-3 contains the two types of f1-scores i.e. weighted and macro for the various models on which test data of the Yoruba language was tested. From the data we see that SVC (SMOTE) performed best among four models (some with different types of sampling) used for testing by giving a value of 74.83 and 73.15 percentage on weighted and macro types of f1-score respectively.

It is hereby informed that we only submitted mBERT based results in the competition and rest of the models were added later. On the basis of submitted results, we ranked 24th in Hausa, 31st in Yoruba and 29th in Swahili tracks of subtask-1.

6 Conclusion

Thus, we implemented various traditional and transformer models for sentiment analysis of African languages. We had different types of data preprocessing techniques in case of traditional methods and analyzed their performance. It was observed that in case of a balanced dataset, deep learning based models like mBERT will be more efficient. If the data is imbalanced, traditional classification and oversampling methods produce better performance. In future, we plan to train our models using on a dataset built by data scraping and data crawling. We also plan to try different tokenizers for better preprocessing.

Limitations

The main limitation of this paper is data preprocessing. We have used a very limited number of data

Model	**weighted-F1**	**macro-F1**
Logistic Regression (without oversampling)	71.44	71.50
Logistic Regression (RandomOverSampler)	70.84	70.91
Logistic Regression (SMOTE)	71.14	71.21
SVC (without oversampling)	72.07	72.13
SVC (RandomOverSampler)	71.82	71.88
SVC(SMOTE)	71.92	71.99
RandomForest (without oversampling)	71.35	71.46
RandomForest(RandomOverSampler)	71.03	70.99
RandomForest(SMOTE)	71.28	71.07
mBERT	**76.53**	**76.55**

Table 1: Hausa

Model	**weighted-F1**	**macro-F1**
Logistic Regression (without oversampling)	72.80	70.88
Logistic Regression (RandomOverSampler)	73.41	71.81
Logistic Regression (SMOTE)	73.13	71.57
SVC (without oversampling)	74.78	73.04
SVC (RandomOverSampler)	74.53	72.83
SVC(SMOTE)	**74.83**	**73.15**
RandomForest (without oversampling)	64.66	61.04
RandomForest(RandomOverSampler)	68.46	66.00
RandomForest(SMOTE)	66.69	62.83
mBERT	50.00	48.56

Table 2: Yoruba

Model	**weighted-F1**	**macro-F1**
Logistic Regression (without oversampling)	52.43	33.79
Logistic Regression (RandomOverSampler)	56.85	46.75
Logistic Regression (SMOTE)	**57.79**	**48.59**
SVC (without oversampling)	54.57	47.15
SVC (RandomOverSampler)	55.75	42.27
SVC(SMOTE)	56.58	44.12
RandomForest (without oversampling)	51.28	34.08
RandomForest(RandomOverSampler)	52.27	38.64
RandomForest(SMOTE)	52.20	31.03
mBERT	47.55	28.87

Table 3: Swahili

preprocessing methods during our submissions. We were not able to implement any kind of data scraping or data crawling techniques, which ultimately resulted in reducing our overall accuracy. Also a few more models like AfriBERT and some other pre-trained models for African languages are yet to be implemented on this dataset along with trying various tokenizers in the pipeline.

References

Jehad Ali, Rehanullah Khan, Nasir Ahmad, and Imran Maqsood. 2012. Random forests and decision trees. *International Journal of Computer Science Issues(IJCSI)*, 9.

Kevin W. Bowyer, Nitesh V. Chawla, Lawrence O. Hall, and W. Philip Kegelmeyer. 2011. SMOTE: synthetic minority over-sampling technique. *CoRR*, abs/1106.1813.

David R Cox. 1958. The regression analysis of binary sequences. *Journal of the Royal Statistical Society: Series B (Methodological)*, 20(2):215–232.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2019. Bert: Pre-training of deep bidirectional transformers for language understanding.

M.A. Hearst, S.T. Dumais, E. Osuna, J. Platt, and B. Scholkopf. 1998. Support vector machines. *IEEE Intelligent Systems and their Applications*, 13(4):18–28.

Jindrich Libovický, Rudolf Rosa, and Alexander Fraser. 2019. How language-neutral is multilingual bert? *CoRR*, abs/1911.03310.

Arun S. Maiya. 2020. ktrain: A low-code library for augmented machine learning. *CoRR*, abs/2004.10703.

Alejandro Moreo, Andrea Esuli, and Fabrizio Sebastiani. 2016. Distributional random oversampling for imbalanced text classification. In *Proceedings of the 39th International ACM SIGIR Conference on Research and Development in Information Retrieval*, page 805–808, New York, NY, USA. Association for Computing Machinery.

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.

Claude Sammut and Geoffrey I. Webb, editors. 2010. *TF–IDF*, pages 986–987. Springer US, Boston, MA.

Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N. Gomez, Lukasz Kaiser, and Illia Polosukhin. 2017. Attention is all you need.

Adam B. Yedidia. 2016. Against the f-score.