Available online at www.ifejost.org

IFE JOURNAL OF SCIENCE AND TECHNOLOGY

Vol 3. No. 1 (2019) 12–25

Design Issues in Sentiment Analysis for *Yorùbá* Written Text

O. Abegunde, A. R. Iyanda* and D. O. Ninan
Computer Science and Engineering Department, Obafemi Awolowo University, Ile-Ife, 220005, Nigeria

*Corresponding author: abiyanda@oauife.edu.ng
Tel.: +2348030459953

Abstract

Abstract. Sentiment Analysis (SA) is an exciting and important field in Artificial Intelligence combining Human Language Processing, Machine Learning and Psychology. It is a means of understanding a user's opinion about an event. The goal of SA is to get opinion expressed in implied text, targets of the opinion and reason for the opinion. Conversely, a great number of research efforts are dedicated to English language data, while a countless share of information is obtainable in other languages as well but none yet for *Yorùbá*. This work examines the design issues with respect to automating SA for standard *Yorùbá* language. The process of SA which includes data cleaning, data annotation etc. is highlighted. The structure of the *Yorùbá* text is described and a text corpus design for *Yorùbá* sentiment analysis system is presented. The outcome of this work provided suitable requirements for the design.

Keywords: Design issues; Opinion mining; Sentiment analysis; Yorùbá language

Introduction

Sentiment Analysis (SA) is the study of how opinions, emotions and attitudes are expressed in a language. It is a means of extracting and evaluating subjective information from small or large datasets. It is one of the fields of Natural language processing that has an application in human language processing and decision making. SA can be featured to be positive or negative opinion expressed in a language with some of its applications in automatic detection of whether a review is positive or negative on an entity implied. SA has become a tool being used in the repertoire of social media analysis extracts information by companies such as; political analysts and marketers or online sales (Taboada, 2016).

In addition, SA system is a crucial part of the natural language processing to determine peoples' opinion about an event. The way *Yorùbá* language is expressed on daily basis requires a system to analyse the opinion of every individual because it is a language whose terms can easily be misinterpreted. *Yorùbá* language can express some terms full of confusion that will probably lead to explanation by other people that have more knowledge about the language. It is impossible to have a good-working SA system without having a tool that generates correct detection and classification when presented with sentences. Thus, it is important to employ the recent advancement in technology to detect opinion which will increase understanding of spoken or written word in *Yorùbá* language.

The *Yorùbá* language (YL) is a tonal language which was introduced into writing in the year 1832 with Rev. Samuel Crowther as one of the main facilitators (Adewole *et al.*, 2017). YL is a language that belongs to one of the major well-spoken languages in Africa and it is one of the three main Nigeria languages alongside Hausa and Igbo. YL is a native language of the *Yorùbá* people that is situated in South-Western Nigeria (Lagos, Oyo, Ogun, Ondo, Ekiti, Osun and parts of Kwara and Kogi states) (Iyanda and Odejobi, 2015) and it is well-spoken in Nigeria as well as the neighbouring countries such as Togo, Republic of Benin, Sierra Leone and Ghana.

Description of Yorùbá text

Yorùbá alphabet comprises 18 consonants (*b, d, f, g, gb, h, j, k, l, m, n, p, r, s, ṣ, t, w, y*) and 7 vowels (*a, e, ẹ, i, o, ọ, u*). This language comprises a diagraph, *gb* (a consonant written with two letters) which is not common in other languages (Iyanda 2014). Five nasal vowels exist in the language by adding "*n"* with the oral vowels "*a, ẹ, i, ọ, u*" to give "*an, ẹn, in, ọn, un*" and also, one syllabic nasal (*n*) exists There are no consonants clusters in *Yorùbá* language as appeared in English language but consonant clusters from borrowed words are re-syllabified, mostly through vowel insertion. For example; buredi (bread), gilasi (glass), etc. The structure of *Yorùbá* sentence can be of two types: mono-clausal which represents simple sentence and multi-clausal which can be combined with conjunction or disjunction. The system of sound in *Yorùbá* can be used to create many words that can form infinite linguistic patterns. In *Yorùbá* sound system, there are three sets of phonemes which are: consonants, vowels and tone with syllables as the basic unit of sound. YL consists of five syllable types: oral vowels (V), nasal vowels (Vn), syllabic nasals (N), combination of consonants and oral vowels (CV) as well as combination of consonants and nasal vowels (CVn) (Odejobi, 2007). As a toner language, the *Yorùbá* system uses different pitch patterns to differentiate each of the words or grammatical forms of the words.

Given the way *Yorùbá* people use their language to express their views on events, instructions and responses, the available sentiment detection system in other languages like (English, Chinese and Hindi) cannot be applied based on the orthography of the language.

The development of an automatic sentiment detection system is required for the language, hence there is a need to draw out the challenging issues as it relates to the design.

Related Works

Islam *et al.* (2016) worked on sentimentality extraction of Bengali language using Naive Bayes classification model. The dataset used in the work was retrieved from Facebook social media. Above 1000-positive comments and 1000-negative comments were collected for the train sets, where 500-comments was used as the test set. Data processing was performed on the collected corpus. The features used are unigram and bigram. The extracted features were fed into the Naive Bayes algorithm to generate the data model.

Kumar *et al.* (2017) worked on the extraction of emotions from the multilingual Text using advanced framework for detection of emotions of users in multilanguage text data. Three domain areas were chosen (Political election, Health-care, and Sports) for real-time empirical study. The technique used in the data collection was Rich Site Summary (RSS) feeds through the headline news from Twitter. Text preprocessing was carried out on the tweets such as normalization, tokenization, stop-words removal, stemming, lemmatization, etc.

Masdisornchote (2015) presented sentiment analysis framework in sentence-level implicit opinions for the Thai language. The framework used consists of three modules for data preparation (Knowledge, Construction and Data preprocessing). The experiment was conducted in one of the mobile leading domains for data collection and all reviews were collected from Siamphone website. The total number reviews are 1090 sentences. The data were analyzed using pattern-matching approach.

Su *et al.* (2017) presented idiomatic expressions for Chinese sentiment analysis which focused on the improved precision and performance of the emotion classifier. A web crawler with Plurk search API was applied to collect Chinese short text messages from the Plurk platform. The collected messages end with emotion icons with positive or negative. The data collected for the trained phase was 52694 sentences and 1000 sentences for the test phase. The collected data were segmented with Jieba and bigram feature extraction approach used was applied to the datasets. Naive Bayes classifier was applied on the feature-sets.

Basiri and Kabiri (2017) presented sentence-level sentiment analysis in Persian. The study addresses the problem of resource scarcity. SPerSent which contains 150000 sentences was used for binary classification. Lexicon-based method was applied manually on the corpus to build the sentiment words. The feature selection conducted in the study used occurrence filter and stop-word filter. The selected feature-sets were applied to the Naive Bayes algorithm.

Mining opinions and sentiments from natural language is challenging, because it requires a deep thoughtful of the explicit and implicit, regular and irregular, and syntactical and semantic language rules. Sentiment analysis researchers scuffle with NLP's unresolved problems: co-reference resolution, negation handling, anaphora resolution, named-entity recognition, and word-sense disambiguation (Cambria, 2013).

Challenging issues in the design of SA system as applied to Yorùbá language

The quality of SA systems could be improved by addressing some of the design issues which are confronting sentiment analysis. These are:

- Word Sense Disambiguation: Language often encounters a problem in word sense disambiguation. The correct meaning of a word based on the context needs to be extracted because a word can be different in meanings based on the domain. For example: *ìwọ̀n kékeré* (Little amount) can be positive in terms of work and become negative in terms of money.
- Comparisons: Determining the polarity of a comparative sentence can be a challenge. For example: *ẹ̀rọ ìléwọ́ yìi dára ju ti tiẹ̀ lọ* (This mobile phone is better than his own). This review has positive word *dára* which means better, but the author's idea of the object is unable to find out which is the key piece of information in the comparative review.
- Negations: If negation is not well handled it can give a wrong result completely. For example: *irin iṣẹ́ yìi dára sùgbọ́n kò kì ń pẹ́ bàjẹ́* (The tool is good but it doesn't last long). This review shows positive polarity but it also expresses negation which changes the effect completely.
- Sarcasm: Another interesting challenge can be in the identification of sarcasm and to express the emotion in the text at standard level. This term is most common among *Yorùbá* people. For example, *Sé mo lè jòkòó? Jòkòó!!!* (Can I sit? sit!!!). The word *jòkòó* actually signifies that the person is not permitted to sit.

Other types of challenge in SA may come due to the nature of the problem. In fact, some negative sentiments may be expressed in a sentence without the use of any negative words. For example: *ìmọ̀ ti ara rẹ̀ nìkan ni ó mọ̀* (He is selfish). Besides, there is a thin line between whether a phrase expresses sentiment or the statement is objective in nature (i.e. no sentiment expressed). One of the difficult tasks in SA is to determine the opinion holder in a text and how certain is the opinion (subjective probability). SA greatly depends on the domain of the data, hence while applying sentiment analysis to a particular problem, data must be obtained from that domain to be able to get the expected result.

Methods

The research corpus was built from four different domains which include: government parastatal, schools, health sectors, and marketing sector. About 1039 sentences were created from several online and offline sources such as; dictionaries, Bible, social media and awayoruba blog. The data that represent the human feelings show the behavioural

characteristics or attitude about an event in the specified domain. The corpus was manually annotated at sentence level only, with two classes of polarity (positive and negative) considered and verified by the human experts.

The datasets were made up of 639-negative and 400-positive records with the remaining (491) being objectives sentences. To increase the performance of the system, all objectives sentences were discarded, because they express facts about an event with no sentiment. The set of feature used in this work shows a balance between expressiveness and compactness of the feature set. In this case, subjective content which comprises all the sentiment sentences annotated from the dataset was considered for analysis.

The data without correct diacritics were processed using *Tákàdá. Tákàdá* is a text processing tool for typing *Yorùbá* language using the correct orthographic items (tone marks and under dots). The datasets were split into the proportion of 80% for training sets and 20% for testing sets. Table 1 shows sample data.

Data preparation and processing

Often times, data collected is not always in a usable format therefore, it is important that the right data is fed into the machine learning algorithms. In this regard, the study ensured that the data was converted to a useful scale format with all the meaningful features extracted from the collected corpus and preprocessing as well as data cleaning were performed on the data before being fed into the machine learning algorithms. The algorithms used in this study are Logistic Regression (LR), Naive Bayes (NB) and Support Vector Machine (SVM).

Feature engineering techniques were applied to the dataset which made machine learning algorithm able to give a good result. Furthermore, the study employed the use of data iterations, exploration, and analysis for the performance of the classifier for preparing the data. This allows for removal of noise, handling of missing values, isolating outliers and correcting data inconsistencies. Iteration is an important aspect of machine learning because

Table 1: Simple sentence that expresses sentiment

S/N	Positive Sentence	Negative Sentence
1	*ọ̀mọ̀wé wa ni ilé ọgbà wa kò sí ẹlẹgbẹ́ wọn ní ìlú yìí.* (There are scholars in our school, and there is none like them in this city).	*ọjọ́ wo ni a ó bọ́ kúrò ní ọwọ́ àwọn ọ̀ṣ́ọ̀hàn tí mbá ilẹ̀ wa fàá.* (When are we going to be delivered from the thieves troubling us in this city).
2	*ohun tí àwọn ọlọ́jà wa ń tà ní ìsisìyí wúni lórí lọ́pọ̀lọpọ̀.* (What our marketters are selling are so impressive).	*ìbàjẹ́ ọjà lóhún tà ó kàn tilẹ̀ pọ̀ lójú ni.* (She's is selling quantity and not quality)
3	*ìwa rẹ̀ ni ìwúlò ni àwùjọ.* (His good attitude is important for the society)	*nǹkan tó burú ni kí ẹnu àwọn àgbá ìlú ma kò àti ṣe nnkan èyẹ sí ìlú yìí mba mí lẹ́rù.* (It's a bad thing for the elders not to be united, I am even scared to do good thing in this land)
4	*adúpẹ́ pé akò ba ti ara wa tì ní ilẹ̀ yí.* (We appreciate our effort towards the success recorded in this land.)	*ọmọ ìgboro ni, ó kàn n fi ọjà bojú ni.* (He is a street boy, he uses his market as a cover)
5	*modúpé mò sì tún júbà fún olùrànlọ́wọ́ wa .* (I thank and appreciate all our supporters)	*ìwà ìbàjẹ́ ti wá dàṣà ní àárín wa àwọn ajẹgúdújẹrá n pọ̀ si.* (Evil act has become a normal thing in our society, embesslers are multiplying).

as models are being exposed to new data, they are able to adapt. They learn from previous computations to produce reliable, replicable choices and refined model.

Removing noise and outliers from Yorùbá data

Data is noisy if it contains incorrect or wrong records or outliers, although record and observation are used interchangeably in machine learning application. According to Vadehra (2015), outliers are like dictionary type with entity-pair that can throw the whole classifier or training model off-base. An outlier is not a false or void value, but it can lead to a high value of false positives and false negatives. Outliers are data that do not fit in with the other data in the corpus, for example sarcastic statements were considered as outlier in this study.

According to Vadehra (2015), noise garbage consists of void or null information that is not relevant to the dataset, if the datasets have those errors, data preprocessing and imputation

will be applied. Any ASCII character or other languages apart from *Yorùbá are* considered to be noise in the datasets. For example, *"Ó yà mí lẹ́nu nígbà tí mo rí* problem *rẹ́"*- "problem" in this statement is a noise.

An outlier would not make the model fail but might have a faulty result some or most of the times based on the number of observations and the number of outliers while noise will nearly or certainly fail the model most times. The data collected online that contain non-*Yorùbá* words, special characters, *Yorùbá* words that have elongation i.e. repetition of symbols were preprocessed. Firstly, all non- *Yorùbá* characters that may have HTML links and tags (e.g. html weblinks, emojis, numbers, characters-!!!, @, # etc.) were removed from the text being part of the unwanted data. If these are present in the dataset, our algorithm will be fed with inaccurate or wrong data and it will affect the performance of our model.

The second treatment deals with special characters. The characters included in the text are sometimes used to express emotion and sentiment towards an event, such as smiley faces where the other types may be present as a mistake. The user may try to repeat some letters in a word to show the feelings he or she has towards an event. For example: *"inú wa dùn gaaaaaaan ni fún isẹ́ ribiribi tí ẹgbẹ́ṣe láàrin wa"* (We are so glad for the work you have done in our midst). With this statement, user shows his feelings towards an event by repeating letter *'a'*, but the word cannot be overlooked as it is being represented by the user because this might have negative effect on data processing.

Data preparation techniques

For data to be well presented in the way machine learning algorithm can be applied, there is a need to apply some basic concepts used in natural language processing to filter the data for best performance. The following basic techniques were applied in the text preprocessing.

- Tokenization: For the computer to process natural language, there is a need to classify words that constitute a string of characters because the meaning of a text generally depends on the relations of words present in the sentence. The initial step undertaken in NLP pipeline is tokenization which the splitting of text into smaller units corresponding to the words of the language on which this study is based. There are some inbuilt NLTK library that help achieve this, e.g. WordTokenizer, sen tokenize, PunkSentence-Tokenizer, StanfordTokenizer, etc. wordTokenizer that deals with breaking the sentence into words in order to separate the stopwords and as well make frequency count of the unique words was used in this study in addition with the developed algorithm. Fig. 1 shows the sample of the tokenized datasets. The selected algorithms (LR, NB and SVM) trained on these data to produce the model.
- *Stop words removal:* Removing stop-words is one of the commonly used processing steps across various NLP applications. The basic idea is to remove all common words that occur frequently in a corpus (Schrauwen, 2010). There are several words in *Yorùbá* that carry little or no meaning in a sentence, but are really common e.g. *èmi* (me), ìwọ

(you), *òun* (him), *sí* (to) etc. The Natural language toolkit (NLTK) does not come with stop-words in *Yorùbá* though it contains stop words for other languages. Therefore, *Yorùbá* stop words generated by Asubiaro (2013) were manually added to the NLTK

1. ['ọjà', 'tí', 'ó', 'n', 'tà', 'kò', 'dára', 'bótiwù', 'kó', 'pọ̀', 'tó']
2. ['rúdurùdu', 'ni', 'ayé', 'rẹ̀', 'ṣe', 'rí']
3. ['ìwà', 'wọn', 'kan', 'ó', 'tún', 'korò']
4. ['ókú', 'tẹyintẹyín', 'lẹ́hìn', 'ìwòsàn', 'tí', 'ó', 'ti', 'gbà', 'nígàbtí', 'ìlú', 'kò', 'dẹrùn', 'fún', 'n', 'ní', 'ìgbàtì', 'ó', 'wà', 'láyé']
5. ['obìnrin', 'sọ́', 'ìwà', 'nù']
6. ['ni', 'ojojúmọ́', 'ni', 'ìjà', 'máa', 'n', 'sẹlẹ̀', 'ní', 'àwùjọ', 'wa']
7. ['bọ', 'tilẹ̀', 'jé', 'pé', 'òrọ̀', 'náà', 'máa', 'n', 'dá', 'ìjà', 'sílẹ̀', 'láàrín', 'ara', 'wà']
8. ['oní', 'ìlara', 'òdì', 'ni', 'àwọn', 'Yorùbá']
9. ['wọ́n', 'fi', 'ìyónú', 'wọn', 'hàn', 'sí', 'mi', 'pé', 'òrọ̀', 'aṣáájú', 'kò', 'dàbí', 'ìṣọ̀kan', 'Yorùbá', 'kì', 'íṣe', 'tuntun', 'sì', 'wà', 'mọ']
10. ['Òfo', 'ijó', 'kejì', 'ọjà', 'ni', 'àríyèlé', 'àwọn', 'olùtánjú', 'wa', 'ní', 'orílè', 'èdè', 'yìí']
11. ['akánjú', 'jẹ', 'ayé', 'kò', 'tilẹ̀', 'ro', 'ẹ̀yìn', 'ọ̀la']
12. ['àbá', 'má', 'tilẹ̀', 'kó', 'owó', 'dànù', 'sí', 'ilé̩ìwòsàn', 'nígbàtí', 'àwọn', 'tó', 'wà', 'níbẹ̀', 'kò', 'sí', 'ounkan', 'tí', 'wọ́n', 'mọ̀', 'ni', 'pàtó']
13. ['ìbẹ̀rẹ̀', 'ijọba', 'yìí', 'wu', 'àwọn', 'ènìyàn', 'ní', 'ọ̀pọ̀lọpọ̀', 'ṣùgbọ́n', 'ó', 'ṣeni', 'ní', 'àánú', 'pé', 'wọ́n', 'já', 'ọ̀pọ̀', 'ènìyàn', 'kulẹ̀']
14. ['owó', 'tó', 'yẹ', 'kí', 'á', 'fi', 'sọ', 'ilé̩ìwòsàn', 'wa', 'di', 'nílá', 'ni', 'wọ́n', 'ti', 'kó', 'lọ', 'sí', 'ilẹ̀', 'òkèèrè', 'fún', 'ìmọ̀', 'tara', 'wọn']
15. ['kòsí', 'ọjà', 'gidi', 'mọ́']
16. ['ọ̀làjú', 'dé', 'bá', 'wa', 'ní', 'àlejò', 'pẹ̀lú', 'inira']
17. ['bí', 'ojú', 'ṣe', 'n', 'là', 'si', 'ni', 'ó', 'yẹ', 'kí', 'ìlú', 'máa', 'gbòrò', 'si']
18. ['òmìnira', 'da', 'omi', 'ìnira', 'mọ́', 'wa', 'ní', 'ọwọ́']
19. ['adára', 'níta', 'má', 'dára', 'ni', 'ìlú', 'ni', 'ìwà', 'tí', 'ọ̀pọ̀lọpọ̀', 'olórí', 'wa', 'ní', 'ẹ̀tàn', 'pọ̀', 'nínú', 'ìwà', 'wọn']
20. ['agbọ́', 'tẹni', 'má', 'fi', 'tì', 'kan', 'ṣe', 'mà', 'ni', 'wọ́n']
21. ['agbọ́', 'tẹni', 'ma', 'fi', 'ti', 'kan', 'ṣe', 'mà', 'ni', 'wọn']
22. ['ọ̀pọ̀', 'nínú', 'àwọn', 'dókítà', 'yí', 'ni', 'kò', 'mọ', 'iṣẹ́', 'rẹ̀', 'dájú']
23. ['ìbádára', 'tó', 'bá', 'jẹ́', 'pé', 'ìrònú', 'wà', 'nínú', 'oun', 'kan', 'tí', 'àwọn', 'olórí', 'wa', 'bá', 'n', 'ṣe']
24. ['kò', 'ní', 'wá', 'bó', 'ṣe', 'pàdánù', 'èrò', 'ìwòsàn', 'ọ̀fẹ́', 'ti', 'ìjọba', 'gbé', 'kalẹ̀', 'àìmọ̀', 'nǹkan', 'sì', 'n', 'da', 'àwọn', 'ènìyàn', 'láàmú']
25. ['ọ̀pọ̀', 'àwọn', 'òsìsẹ́', 'ilé̩ìwòsàn', 'ni', 'wọn', 'kò', 'fi', 'taratara', 'ṣisẹ́', 'wọn', 'bó', 'tilẹ̀', 'jẹ́', 'wípé', 'ijọba', 'ún', 'gba', 'ìyànjú', 'tirẹ̀']

Figure 1: Word tokenization example library

Typically, articles and pronouns are normally classified as stop words. For example: *kan, ẹyọkan, díẹ̀, náà* and pronouns such as *èmi, ìwọ, òun, `awa, àwọn, tèmi, ìwọ, tirẹ, tiwọn, tiwa*.

- *Feature extraction:* This is the process of converting what is essentially a list of words into a feature that can be used by the classifier. For usage with the NLTK classifiers, the dataset was converted into dictionary (dict) style features sets to make each word become a key with the True value. A bag of words was constructed for the present feature set from all the words of an instance. Algorithm 1 shows the feature extraction and selection process. This reads the encoded text data and classifies them into positive and positive sets. There are two columns in the dataset, the first one consists of the target variables while the second column is the opinion collected from users. The second column was the variables found in the problem set that helps to build the model. Fig. 2 shows the system extracted features from the datasets. The classification was done by learning from labelled features sets which are a key-value pair that maps each feature names to the feature values. The feature names are words and the values are “True”. Algorithm 2 shows the process involved in polarity detection from the user. The process begins with

the user entering the *Yorùbá* sentence into the system. If the user enters other languages aside *Yorùbá*, an error is raised telling the user to enter the correct language. The sentence is converted into "dict" style feature sets and is fed into the classifiers. The

Algorithm 1: Feature extraction and selection process

```
Data: Opinion Corpus for Yorùbá
Result: Extracted Features From the Yorùbá
special treatment of the first line
begin
    Function format_sentence(sentence)
        words = word_tokenize(sentence)
        tokens = set(words) - set(stopwords)
        /* diction word is assigned to each tokens          */
        dicWords = word.lower():True for word in tokens
        return dicWords
    /* empty list of positive & negative list               */
    negdata, podata =[],[]
    /* create annonymous function for negdata data          */
    Function open(data, encoding) as neg
        read_negdata = csv.read(neg, delimeter=',')
        for data in read_data do
            negdata.append([format_sentence(data[0]), 'negative'])
    /* create annonymous function for posdata data          */
    Function open(data, encoding) as pos
        read_posdata = csv.read(pos, delimeter=',')
        for data in read_data do
            posdata.append([format_sentence(data[0]), 'positive'])
```

sentence is classified based on the opinion it expressed and the percentage probability is calculated.

Discussion

The developed system takes in a *Yorùbá* sentence from the user, the features are extracted automatically by the system, and the sentiment, as well as accuracy of the sentence, are determined by the system. Fig. 3 shows the interface of the implemented system which provides a means by which the user can interact with the system. For example, the user enters a sentence *ìlú yìí dára púpọ̀ kódà mo fẹ́ máa gbé ibẹ̀* (this city is good, I will like to live therein) as shown in Fig. 3, the system extracted the feature and fitted it into the

models. The result shows “positive” which implies that the sentence entered by the user has “positive” opinion.

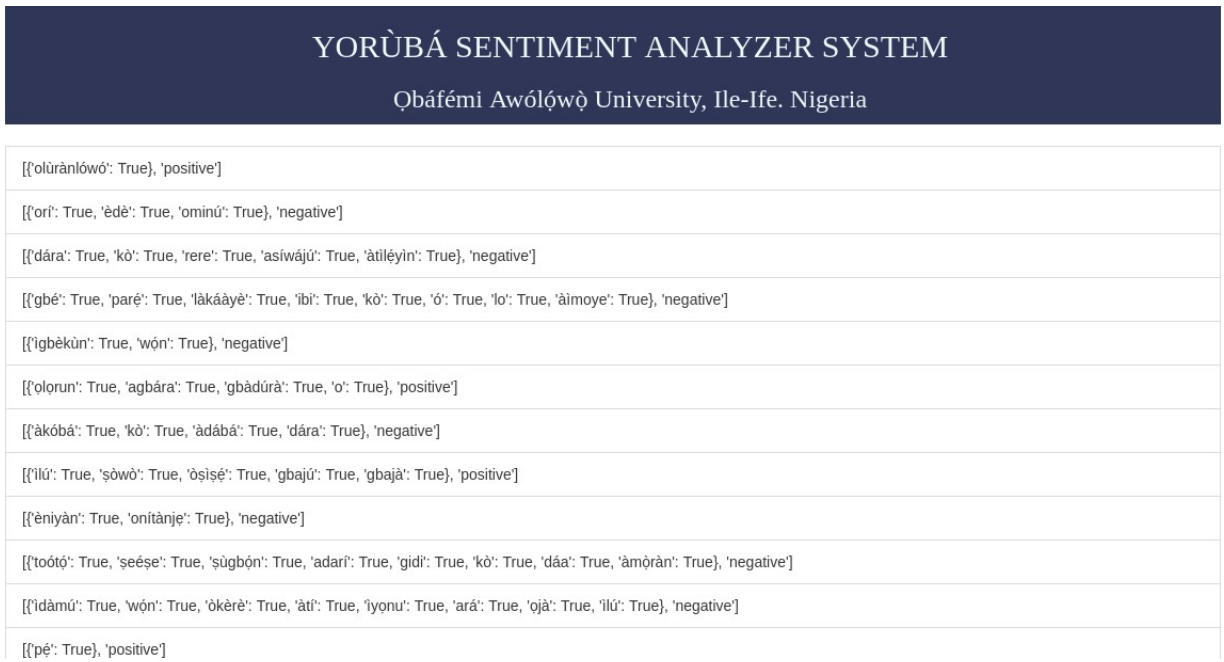

Figure 2: Screenshot for feature extracted by the system

Encountered Issues

- Language problem**:** Human language is very difficult to understand despite computer understanding of it. In opinion mining, language is one of the major resources that are required. Unlike English with a large amount of resources, *Yorùbá* is well spoken in West Africa but with limited corpus available and tools for processing them. Therefore, the corpus was manually created from dictionaries, Bible, social media and awayoruba blog.
- Annotation problem**:** obtaining high-quality annotations most time requires following the instruction given, but this can be true for simple annotation tasks. In a case where one record (instance) is labelled as positive or negative and the statement is neither of the two, if the annotator wrongly classifies its polarity, it will have effect on the dataset. Sometimes some words may be greatly associated to negative and to positive when used by the opinion holder; this requires an expert knowledge to determine where the opinion belongs.
- Natural language processing (NLP): the application of NLP on *Yorùbá* corpus requires an extra work because *Yorùbá* is a language with diacritics, consequently, all the words needed to be encoded. In the natural language toolkit library, there is no corpus containing any *Yorùbá* words, sentences or stopwords. Therefore, all these need to be manually added. English and some other languages have a lot of organized corpus from

the NLTK library. Most of the tools required for language processing (POS taggers, Stemmers, Lemmatizers, WordNet, etc) were inclusive but none was available for

Algorithm 3: Polarity detection process for *Yorùbá* sentence

```
Data: Yorùbá sentence containing opinion
Result: Positive or Negative opinion
/* From the input sentence                                    */
begin
    result = ''
    lang = detect(inputSentence)
    if lang == English then
        return 'language not recognize'
    else
        result = lang
    Function format_sentence(sentence)
        words = word_tokenize(sentence)
        tokens = set(words) - set(stopwords)
        /* diction word is assigned to each tokens            */
        dicWords = word.lower():True for word in tokens
        return dicWords
    /* Instantiate the machine learning classifiers           */
    /* Fit the extracted feature into the machine learning
       classifiers                                            */
    classifierResult = classifier.classify(result)
    accuracy = classifier.prob_classify(result).prob(classifierResult) * 100
    return jsonify ('result':classifierResult,'accuracy':accuracy)
```

Yorùbá language. Therefore, the generated stopwords (Asubiaro, 2013) were added to the library for easy processing of the language.

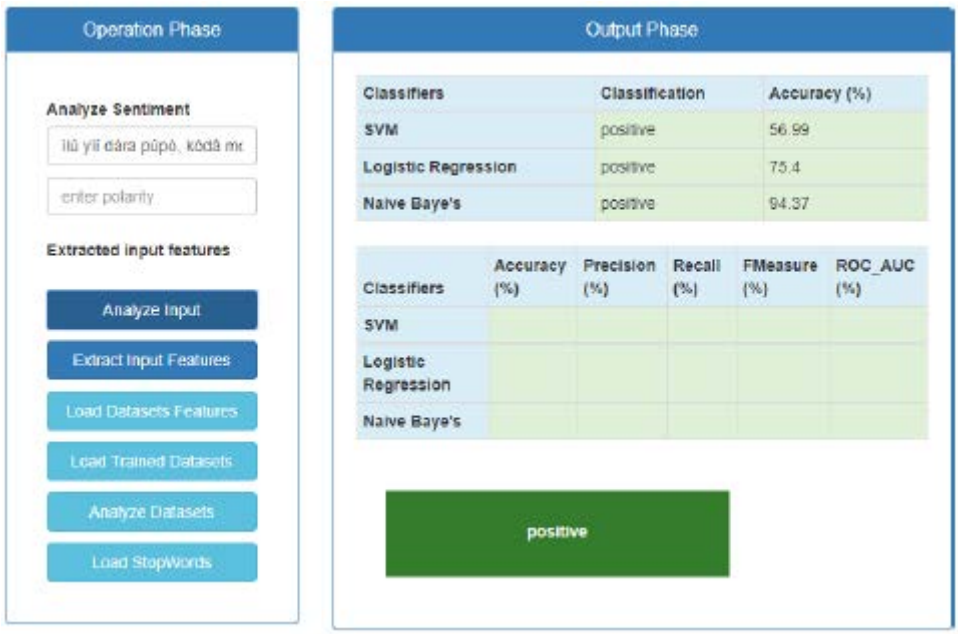

Figure 3: Screenshot for the *Yorùbá* sentiment analyzer system

- Fake Opinion: this kind of opinion can mislead the readers by providing deceitful information relating to any event or business. This issue often occurs among *Yorùbá* people where it is easy to express negative or positive comments on a particular event when it is neither (i.e. using positive sentiment in a situation where it does not apply and vice versa). It is like an irony. This can lead to wrong analysis.

Conclusions

Sentiment Analysis (SA) is an exciting and important field in Artificial Intelligence that combines Human Language Processing, Machine Learning and Psychology and it is used to get opinion expressed in implied text, targets of the opinion and reason for the opinion. Attempts have been made to develop SA system for other languages such as English language but none yet to address *Yorùbá* language. In this paper we outlined the fundamentals of this concept and gave a detailed progress report on our ongoing research. An account of the data analysis performed was given and the process underlying SA for *Yorùbá* language was examined. Some other design issues were as well discussed. This work provides adequate requirements such as choosing the right data that expresses sentiments, applying text cleaning and pre-processing, annotating the dataset into positive and negative opinions, choosing the right algorithm, splitting the datasets into training and testing sets and feeding the algorithms with the train set) for the SA design and also

provides the foundation for research and development in automatic SA for standard *Yorùbá* language.

References

Adewole, L. B., Adetunmbi, A. O., Alese, B. K., and Oluwadare, S. A. (2017). Token Validation in Automatic Corpus Gathering for Yoruba language. *FUOYE Journal of Engineering and Technology,* 2(1). http://engineering.fuoye.edu.ng/journal/index.php/engineer/article/download/85/pdf. (Accessed: March 2018).

Asubiaro, T. V. (2013). Entropy-based generic stopwords list for Yoruba texts. *International Journal of Computer and Information Technology*. 2(5):1065-1068.

Basiri, M. E. and Kabiri, A. (2017). Sentence-level sentiment analysis in Persian. 3rd *International Conference on Pattern Recognition and Image Analysis (IPRIA 2017),* pp. 84-89. IEEE.

Cambria, E., Schuller, B., Xia, Y., & Havasi, C. (2013). New avenues in opinion mining and sentiment analysis. *IEEE Intelligent Systems,* 28(2): 15-21.

Islam, M. S., Islam, M. A., Hossain, M. A., and Dey, J. J. (2016). Supervised approach of sentimentality extraction from bengali facebook status. *19th International Conference on Computer and Information Technology*, pp. 383-387. IEEE.

Ìyàndá, A. R. (2014). *Design and Implementation of a Grapheme-to-Phoneme Conversion System for Yorùbá Text-to-Speech Synthesis*. PhD thesis, Obafemi Awolowo University, Ile-Ife, Nigeria.

Ìyàndá, A. R. and Odéjobí, O. A. (2015). Design issues in automatic grapheme-to-phoneme conversion for standard Yorùbá, *Research in Computing Science,* 90:195–205.

Kumar, J. V., Shishir, K., and Lawrence, F. S. (2017). Extraction of emotions from multilingual text using intelligent text processing and computational linguistics. *Journal of Computational Science* 21:316-326.

Masdisornchote, M. (2015). A sentiment analysis framework in implicit opinions for Thai language. *41st Annual Conference of the IEEE Industrial Electronics Society, IECON 2015*, pp. 000357-000361. IEEE.

Odéjobí, O. A. (2007). A quantitative model of Yorùbá speech intonation using stem-ml. *INFOCOMP Journal of Computer Science,* 6(3):47–55.

Schrauwen, S. (2010). Machine learning approaches to sentiment analysis using the dutch netlog corpus, *Computational Linguistics and Psycholinguistics Technical Report Series, CTRS-001,* July 2010.

Su, Y.-J., Huang, H.-W., and Hu, W.-C. (2017). Using idiomatic expression for Chinese sentiment analysis. *10th International Conference on Ubi-media Computing and Workshops (UbiMedia)*, 2017, pp. 1-4. IEEE.

Taboada, M. (2016). Sentiment analysis: an overview from linguistics, *Annual Review of*

Linguistics 2: 325–347.

Vadehra, A. (2015). What is the basic difference between noise and outliers in data mining?https://www.quora.com/What-is-the-basic-difference-between-noise-and-outliers-in-Data-mining, (Accessed: May, 2018).