ISSN No:-2456-2165

# N-Gram and K-Nearest Neighbour Based Igbo Text Classification Model

Ifeanyi-Reuben Nkechi J.[1], Odikwa Ndubuisi[2], Ugwu Chidiebere[3]
[1]Department of Computer Science, Nnamdi Azikiwe University, Awka, Nigeria
[2]Department of Computer Science, Abia State University, Uturu, Nigeria
[3]Department of Computer Science, University of Port Harcourt, Nigeria

**Abstract:- The evolution in Information Technology has gone a long way of bringing Igbo, one of the major Nigerian languages evolved. Some online service providers report news, publish articles and search with this language. The advancement will likely result to generation of huge textual data in the language, that needs to be organized, managed and classified efficiently for easy information access, extraction and retrieval by the end users. This work presents an enhanced model for Igbo text classification. The classification was based on N-gram and K-Nearest Neighbour techniques. Considering the peculiarities in Igbo language, N-gram model was adopted for the text representation. The text was represented with Unigram, Bigram and Trigram techniques. The classification of the represented text was done using the K-Nearest Neighbour technique. The model is implemented with the Python programming language together with the tools from Natural Language Toolkit (NLTK). The evaluation of the Igbo text classification system performance was done by calculating the recall, precision and F1-measure on N-gram represented text. The result shows text classification on bigram represented Igbo text has highest degree of exactness (precision); trigram has the lowest level of precision and result obtained with the three N-gram techniques has the same level of completeness (recall). Bigram text representation technique is extremely recommended for any text-based system in Igbo. This model can be adopted in text analysis, text mining, information retrieval, natural language processing and any intelligent text-based system in the language.**

***Keywords:-*** *Igbo Language; Text Classification; Text Mining; K-Nearest Neighbour; N-Gram; Similarity Measure.*

## I. INTRODUCTION

The management of valuable information that are unknown in textual documents has created concerns to Information Technology (IT) experts [1]. It is hard and too expensive to seek for the useful information hidden in the unstructured textual documents of web pages, news articles and others. This is faced with lack of sophisticated text analysis model for discovery new information in their unstructured manner [2]. The need for systematic organization and management of free available online documents for proper utilization and decision making is emphasized in [3]. As a result of this, text mining and data mining have gained great interest with the intense of converting the data into valuable facts [4]. In [5], it is approximated that about 500 companies lose about $12 billion in value annually because of their inability of exploiting unstructured textual data and this practically implies that performing text analysis may increase the organization's competitive advantage.

Text Mining (TM), gotten from the meaning textual data mining, is called a knowledge extraction from text. Discovering or extracting knowledge, ideas from text simply means extracting interesting but hidden patterns or trends from textual documents. Text mining is a quite novel research that has created great concern on researchers, due to a continuous increase of electronic text. It is also has seen to be of great business values. This is proved by [6] statement: "As the mainly natural means of saving information is textual data, textual data mining is assumed to comprise a higher profitable potential than data mining. It is pointed that a research done shows that more than 70% of the business information is hold in a textual document. TM involves difficult tasks because of the unstructured nature of the textual data; one has to perform these tasks in order to put the text in a structured format before analysis can be performed on it [7]. Text mining adopts models and algorithms from machine learning, artificial intelligence, data mining, information retrieval and natural language processing to extract knowledge from the text automatically [8] [9]. Text mining involves many fields such as text summarization, text classification, entity extraction, clustering, semantic and sentiment analysis [10]. This work focuses on text classification. Text classification is a process of giving predefined classes to unstructured textual documents [11] [12].

The advancement of Information Technology (IT) has gone a long way of bringing in Igbo, one of the three Nigerian major languages evolved [13]. It has grown to the extent one can use (Windows 7 and above operating system), create documents, reports news online, search and publish articles online with this language. As there is vast increase in information stored in text format of this language, there is need for an intelligent text-based system for proper management of the data [14]. It is necessary to have means to organise and manage the data generated with these languages effectively. These needs have motivated for this research to develop an improved model to classify Igbo textual documents for proper organisation and management.

In addition, text mining is gaining popularity as a result of increase in textual documents in different languages in the world. Most text mining models majorly deals on processing foreign languages (like English, French, Latin, Chinese, Arabic, Spanish, and Japanese) documents. Little or no research has been done to apply the text mining techniques on Igbo textual document.

## II. OVERVIEW OF IGBO LANGUAGE

According to [15], language can be defined as a means of interaction and communication between people sharing common policies and code, in terms of symbols. In Nigeria, Igbo language is one of the 3 (three) main languages (Hausa, Yoruba and Igbo). It is mainly spoken by the eastern Nigerian people [13]. Igbo language has many dialects. Standard Igbo is used for this work because it is the Igbo that are majorly used for formal communication. Formal (Standard) Igbo is made up of thirty-six (36) alphabets. These include: a, b, ch, d, e, f, g, gb, gh, gw, h, i, ị, j, k, kw, kp, l, m, n, nw, ny, ṅ, o, ọ, p, r, s, sh, t, u, ụ,v, w, y, and z. The Igbo character set consist of 8 (eight) vowels and 28 (twenty-eight) consonants. The consonant alphabets are “b, ch, d, f, g, gb, gh, gw, h, j, k, kw, kp, l, m, n, nw, ny, ṅ, p, r, s, sh, t, v, w, y, z” while the vowels alphabets are “a, e, i, ị, o, ọ, u, ụ”. In Igbo character set, nine consonants letters are digraphs: “ch, gb, gh, gw, kp, kw, nw, ny, sh” [16].

The Igbo language is a language with two outstanding tones (high and low). Igbo is a language that forms majority of its words by the concatenation of two or more words. It is agglutinative language in nature [15].

## III. RELATED WORKS

Some research connected to the work were studied and reviewed as follows:

[17] proposed an improved way of classifying Arabic text using Kernel Naive Bayes (KNB) classification method to solve the issue of non-linearity in Arabic text categorization. Performance evaluation on the classification result on the collected dataset of topic mining Arabic showed the proposed KNB classifier is more effective than other classifiers.

The system evaluation of five often used feature selection methods (Chi-square, Correlation, GSS Coefficient, Information Gain and Relief F) on Arabic text categorization is studied in [18]. The grouping of feature selection methods based on the average weight of the features was adopted. The experimental result on Support Vector Machine classification model and Naïve Bayes proved that best classification results were gotten when feature selection is done using Information Gain technique.

The work in [19] surveyed three feature selection techniques (filter, wrapper and embedded) and their effects on text classification. The survey proved that filter method should be adopted if the result is needed in lesser time and for large dataset; and wrapper method should be adopted if the accurate and optimal result is needed. It was also observed that the performance of different algorithms differ according to the data collection and desires.

[20] compared the performance of various text classification systems in diverse ways using feature extraction without stemming and with stemming on Arabic text. Many text classification techniques such as K-Nearest Neighbour, Decision Tree, and Naïve Bayesian were used. The outcome showed the classification correctness for Decision Tree and Naïve Bayesian method is superior to K-Nearest Neighbours.

A model for classification of the emotion of the memes (images that contains both text and image) associated with the COVID-19 is proposed in [21]. Image and Text sentimental Analysis methodologies are adopted for the research. OCR and YOLO techniques are proposed for the classification of the memes. The model performance evaluation is proposed to be measured by using accuracy and precision.

[22] proposed a new system for text classification based on Binary Particle Swarm Optimization and Reduced Error Pruning Tree- BPSO/REP-Tree hybrid. The Binary Particle Swarm Optimization – BPSO is used for the feature selection procedure and the “Reduced Error Pruning Tree - REP” is used for the classification process. BBC-Arabic dataset is used for the experiment.

An approach for computing range-based rules from numerical data to develop categorization and characterization models is carried out in [23]. Their experimental results show the proposed approach performs better than the commonly used rule mining methods.

In all the related works reviewed, little or no work on this research focus has been done on Igbo, one of the Nigerian major languages. This paper resolved challenges that may hinder the effectiveness of mining textual data in an Igbo language; developed a system that extracts useful features in Igbo text for the classification of the document.

## IV. MATERIALS AND METHODS

This section discusses the processes (figure 1) involved in developing and implementing efficient and robust text classification model that extracts features from Igbo textual corpora for the classification of the documents based on predefined categories. The system uses K-Nearest Neighbour model based on similarity measurement to automatically classify Igbo text documents.

The tasks in this system are:

- Igbo Textual Documents Collection
- Igbo Text Pre-processing
- Text Representation
- Feature Selection
- Text Classification
- System Performance Evaluation

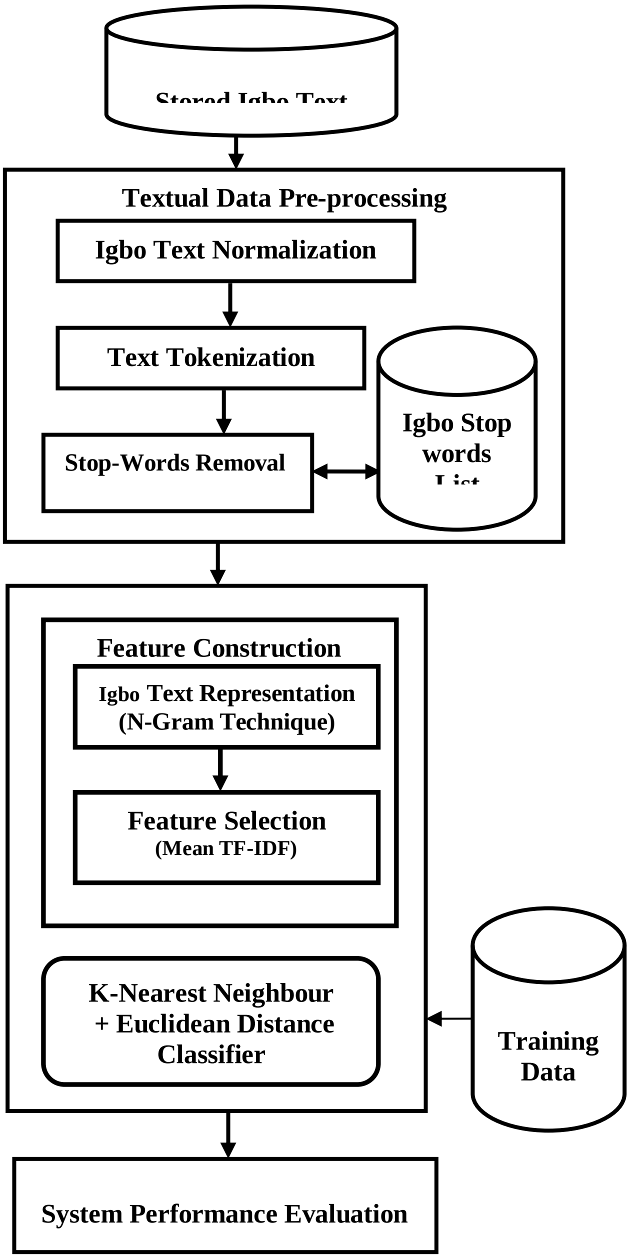

Stored Igbo Text
Textual Data Pre-processing
Igbo Text Normalization
Text Tokenization
Stop-Words Removal
Igbo Stop words List
Feature Construction
Igbo Text Representation (N-Gram Technique)
Feature Selection (Mean TF-IDF)
K-Nearest Neighbour + Euclidean Distance Classifier
Training Data
System Performance Evaluation

Fig. 1: Architecture of Igbo Text Classification System

### A. *Igbo Text Collections*

The implementation of the Igbo Text Classification system begins with the gathering of Igbo text documents. Igbo language is one of the languages that employ non-ASCII character sets. Unicode model was used for its text extraction and processing. This requires UTF-8 encoding [13]. UTF-8 uses numerous bytes and represents group of Unicode characters. The text extraction and processing is achieved with the means of decoding and encoding as shown in Figure 2. Decoding converts Igbo text files into Unicode while encoding write Unicode to a file and converts it into a suitable encoding [24]. A sample of an Igbo text is displayed in figure 3.

The Igbo text documents used for the work was gathered from the following sources:

- Igbo Online Radio – Igbo Online News Reports (www.igboradio.com).
- Catholic Rex Igbo Publications - Catholic Weekly Bulletin.
- Microsoft Igbo Language Software Localization Data.
- Igbo Story Books.

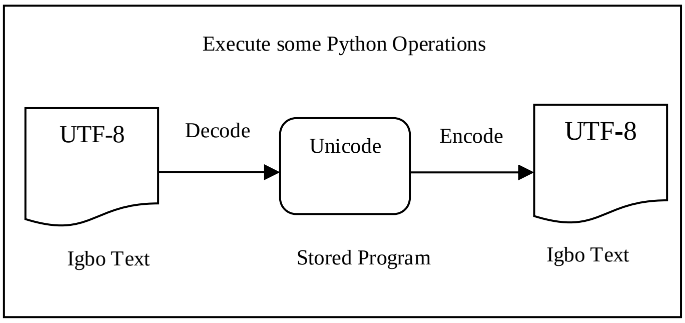


Fig. 3: Display of Igbo Text

> Ihụ akwụkwọ Weebụ ọhụrụ ahụ na-agba mbọ ịmepe otu saịtị na Intaneètị̀. Ị chọrọ ịnye nke a ohere? Kwụnye diisk ike ǹsìebeọzọ ma ọ bụ ọdọnnyà USB mpịọkụ gị. Mgbe ị pịkịnyechara ya, i ga e lenyeanya n'ebe Mwefe Mfe Windows chekwara faịlị gị gasị.

### B. *Igbo Text Pre-processing*

This module converted unstructured Igbo text into a more comprehensible and structured format ready for next processing [25]. The text pre-processing covers text normalization, Igbo text tokenization and Igbo stop-words removal.

### C. *Text Normalization*

In this process, the Igbo textual document is transformed to a format that makes its contents consistent, convenient and full words for an efficient processing. All text cases are converted to lower cases. The diacritics and noisy data are removed. The noisy data is assumed to be data that are not in Igbo dataset and can be from numbers, currency, date, time and other special symbols.

Algorithm 1 outlines the procedures followed in performing normalization in the collected Igbo textual documents.

Algorithm 1: Algorithm for Normalization of Igbo Text

Data-in: Igbo Text Document, Non-Igbo Standard Data/Character list

Result: Normalized Igbo Text

Steps:

1. Convert all text cases to lower cases.
2. Eliminate diacritics. Characters like ụ̄, ụ̀, and ụ́ contains diacritics called tone marks.
3. Eliminate non-Igbo character.
4. For every word in the Text Document:

i. If the word is a digits (0, 1, 2, 3, 4, 5, 6, 7, 8, 9) or contains digits then the word is not useful, remove it.

ii. If the word is a special character (“, =, +, ), (, $,^,&, >,<,% and others) or non-Igbo character, get rid of it.

iii. If the word is joined with hyphen like “ụlọ-akwụkwọ”, “na-eri”, then take away hyphen and disconnect the words. For example, the following word “ụlọ-akwụkwọ” will be “ụlọ” and “akwụkwọ”, forming two dissimilar words.
iv. If the word contains apostrophe like n’ụlọ akwụkwọ then get rid of the apostrophe and separate the words. For example “n’ala eze, after normalization will be three words “n”, “ala” and “eze”.

### D. *Tokenization of Igbo Text*

Tokenization is the means of separating Igbo text into a series of distinct words. These words are known as tokens.

Algorithm 2 outlines the procedures followed in performing tokenization exercise in the normalized Igbo textual documents.

Algorithm 2: Algorithm to tokenize the Igbo text
Data-in: Normalized Igbo text
Result: Tokenized Igbo Text
Steps:

1. Make a TokenHolder.
2. Insert to the TokenHolder when token is found.
3. Disconnect letters or words between “-”; check if the letter or word equals any of the following: “ga-”, “aga-”, “n’”, “na-”, “ana-”, “ọga-”, “ịga-”, “ọna-”, “ịna-”. For example, the strings: “na–ese”, “aga-eche”, “na-eme” will be disconnected and written as “na”, “-”, “ese”, “aga”, “-”, “eche”, “na”, “-”, and “eme”.
4. Separate character or word(s) following n with apostrophe “n’ ”. For example, the strings: “n’aka”, “n’ụlọ egwu” will be disjointed and written as “n”, “aka”, “n”, “ụlọ” and “egwu” tokens.
5. Eliminate diacritics. The grave accent (` ), or acute accent (´ ) in a word will be removed. For instance, if these words ìhè and ájá appear in a corpus, the accent will be removed, tokens ihe and aja will be taken.
6. Any word separated with a whitespace is considered a token.
7. Any single word that ends with dot (.) or semicolon (;), or comma (,) or colon (:) or exclamation mark (!) or question mark (?), is considered as a token.

### E. *Removal of Igbo Stop-words*

Stop-words are most frequently used functional words in a language that usually carry no useful information [26]. There is no exact number of stop-words which every Natural Language Processing (NLP) toolkit is expected to have. This process eliminates the stop-words in Igbo text. Figure 4 displays some of the Igbo stop-words.

ndị, nke, a, i, ị, o, ọ, na, bụ, m, mụ, ma, ha, ụnụ, ya, anyị, gị, niine, nile, ngị, ahụ, dum, niile, ga, ka, mana, maka, makana, tupu, e, kwa, nta, naanị, ugbua, olee, otu, abụọ, atọ, anọ, ise, isii, asaa, asatọ, iteghete, iri, anyi, ndị, a, n', g', ụfọdu, nari, puku, si, gara, gwa, ihi, dịka

Fig. 4: Sample of Igbo Stop-words List

In the developed Igbo text classification model, a stop-word list is generated and saved in a system. This is automatically loaded to the system whenever the system is in operation. The model assumed any Igbo word with less than three length characters do not carry useful information. These are considered as stop-words and are also eliminated in this process [27].

Algorithm 3 outlines the procedures followed in performing removal of stop-words in the tokenized Igbo textual documents.

Algorithm 3: Removing Igbo Stop-Words Algorithm
Data-in: Igbo Tokenized Text
Result: Igbo Stop-Word Free Text
Steps:

1. Insert the stop-word file.
2. Change the loaded stop-words to small letters.
3. Scan each word in the formed TokenList.
4. For each word *t* є TokenList of the file
   i. Verify if *t*(TokenList) is in Igbo stop-word list
   ii. Yes, delete *t*(TokenList) from the Token List
   iii. Decrease tokens total
   iv. Shift to the next *t*(TokenList)
   v. No, shift to the next *t*(Token List)
   vi. End Repetition Loop
   vii. Move to the next job in the pre-processing stage

### F. *Text Representation*

Text representation is a process of selecting suitable features to represent a textual document [28]. This is one of the challenges that have to be settled for a successful research in any text-based system in any natural language like Igbo [13]. The way in which text is represented contributes a lot to the performance of its application in a system [29]. According to [12], text representation with vector space model results to high dimensional data and likely contains many irrelevant features that may affect the performance of a text-based system.

Compounding is a common style of word creation in Igbo language. Igbo vocabulary consists of many compound words because of its agglutinative nature. So many Igbo keywords and features are in phrasal form [13]. The meaning of a whole is not the same to the meaning of a part. Considering the compounding structure of Igbo words, N-gram model is adopted for the text representation in this research.

### G. *Feature Selection*

Feature selection is a process of choosing relevant features from a textual document to be used for a text-based task. This is put in place to reduce the dimension feature space of a system to improve its performance. It involves the identification of relevant features to be used in the system without affecting its accuracy [30]. This process will serve as a filter; muting out irrelevant, unneeded and redundant attributes / features from Igbo textual data to boost the performance of the system. Improving the feature selection will improve the system performance. The goal of feature selection is summarised in threefold:

i. Reducing the amount of features;
ii. Focusing on the relevant features; and
iii. Improving the quality of features used in the system process.

The Mean Term Frequency-Inverse Document Frequency (Mean TF-IDF) model is adopted for the feature selection in this system.

### H. *Text Classification*

Text Classification is a research area in text mining that automatically allocates one or more predefined classes to textual documents based on their contents [12]. The inculcation of Igbo language in the operation of Information Technology has given rise to the generation of large amount of Igbo textual data. The large document size of the text makes the classification of the documents very complex and time consuming. As the quantity of the text is increasing rapidly, this initiated the trend of automatically classifying the text. The classification of the Igbo text is based on K-Nearest Neighbour technique.

### I. *K-Nearest Neighbour (KNN) Classification Technique*

KNN text classification model classifies test document regards to the k- nearest training documents in the documents set. KNN is a good model for text classification [31]. It is chosen because it is a simple and effective means of classifying text. The KNN algorithm works with three major parameters:

1. Similarity / Distance metric: The distance metric is adopted to compute the difference between two data instances in order to measure the similarity. The distance metric in calculated by getting the distance between input test document instance and training document set instances. Choice of distance metric acts vital task in the efficient and effective performance of the proposed text classification model. The Igbo text classifier uses the Euclidean distance metric to compute the distance between two neighbours.
2. K-Value Selection: K-value represents the neighbourhood size. This is one of the input parameters used to determine the class.
3. Computing the Class Probability: The assignment of a data instance into a class is simply based on voting. This is illustrated using figure 3.16.

Given a test document *t* to classify, k-NN model positions the text document’s neighbours amidst the training documents. It then uses the document class of the k nearest / most similar neighbours to guess the class of the test document.

The procedure of the proposed KNN text classifier based on similarity measurement is shown in algorithm 4.

Algorithm 4: K-Nearest Neighbour Classifier
Procedure: Find the class label
Data-in: k-value, the number of nearest neighbours; Z – Testing data set; T – Training data set;
Result: C, Label set of testing data set

```
1. Enter Training data file
2. Enter Testing data file
3. Execute pre-processing task
4. Choose relevant features
5. For each zi in Z and each ti in T do

Calculate d(zi,ti) based on distance measure
6. Determine the similarity or dissimilarity based on the computed distance d.
7. Determine the k-value.
8. To decide whether the document belongs to a class;
   i.    Assign C = { }, a set or list that holds the class labels
   ii.   For each zi in Z and each ti in T do
   iii.      Neighbours (zi) = { }
   iv.       if │Neighbours(zi) │ < k then
   v.            Neighbours (zi) = closest (zi,ti) ∪ Neighbours (zi)
   vi.       End if
   vii.  if │Neighbours(zi) │ = k then
   viii. C = test Class (Neighbours (zi) ∪ C
   ix.   End for

9. Exit Classifier
```

In the algorithm 4, a text document is given a class label by votes of its neighbours. For instance, assuming K = 1, then classifier will assign the document to label of its most similar neighbour. The Neighbours (vi) returns the k-nearest neighbours of vi; and closest (vi,wi) returns the closest element of wi in vi.

The choice of k-value will be dependent on the number of neighbours, distance metric and decision rule. The decision for choice of k is heavily dependent on the actual distribution of v and w. Figure 5 shows the illustration on how the KNN performs its classification.

In figure 5, the green circle represents the test instance and is to be grouped either into blue square class or into red triangle class regards to the k-value. For example if k-value = 3 (solid outline circle), the green circle is categorized into red triangle instance class label due to existence of 2 triangle instances and 1 square instance in internal circle. If k-value =5 (dashed outline circle), the green circle is categorized into blue square instance class due to existence of 3 blue square instances and 2 red triangle instances in the external circle.

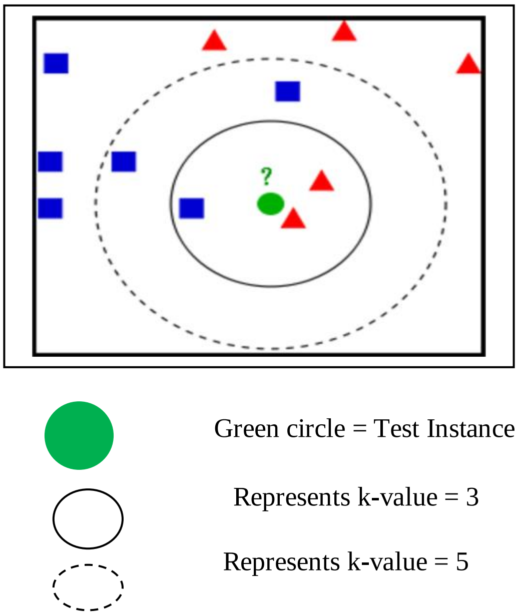


Fig 5: Illustration of KNN Algorithm [32]

### *J. Document Similarity Measurement*

A document similarity measurement reflects the degree of closeness or separation among the documents [33]. Similarity score is defined to determine the similar documents. Different features of the documents are quantified and similarity algorithm is employed across the features to get the similarity score between the documents.

Euclidean distance metric is used to produce the similarity scores between the documents. The documents that have minimal similarity score are likely to be more similar while those with maximal similarity score are likely to be dissimilar. The text documents in the same class will appear to be most similar neighbours.

Definition: If A = ($a_1$, $a_2$, ... $a_n$) and B = ($b_1$, $b_2$, ... $b_n$) are two points in n-dimensional Euclidean space, then the distance (d) from A to B or B to A is given by the formula:

$$d^2(A,B) = d^2(B,A) =$$
$$(a_1-b_1)^2 + (a_2-b_2)^2 + \ldots (a_n-b_n)^2 \quad \ldots\ldots\ldots\ldots\ldots\ldots \quad 1$$

### *K. System Performance Evaluation*

The system performance is evaluated by computing the precision, F1-measure and Recall. Precision is defined as the quotient of total TPs and sum of total TPs and FPs. Precision point is known to as a point of correctness.

$$\boldsymbol{Precision} = \frac{\mathbf{TP}}{\mathbf{TP+FP}} \quad \ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots \quad 2$$

Recall of the classification system is described as the quotient of total TPs and sum of total TPs and total FNs. Recall level measures completeness.

$$\boldsymbol{Recall} = \frac{\mathbf{TP}}{\mathbf{TP + FN}} \quad \ldots\ldots\ldots\ldots \quad 3$$

F1-Measure is single function that joins recall and precision points. When the F1-measure is high, it means that the overall text classification system is high.

F1-Measure =
(2 * Recall * Precision) / (Recall + Precision) ……… 4
= 2TP / (2 TP + FP + FN) ………. 5

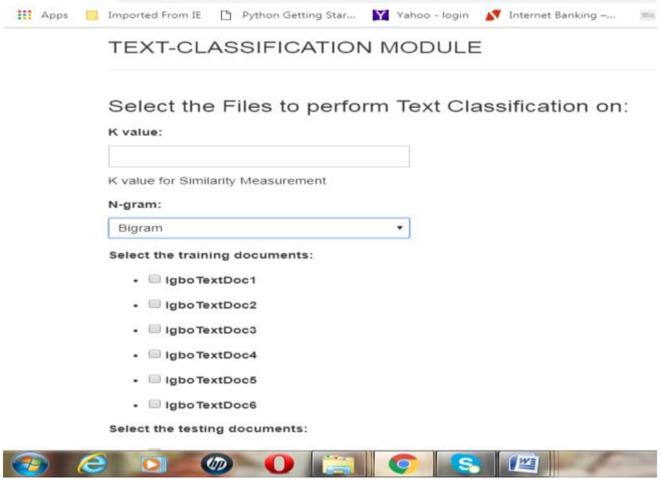
In summary, computation of precision, recall and f1-measure required four input parameters: TP, FP, TN and FN.

i. TP - total of text documents accurately allotted to document class.
ii. FP - total of text documents wrongly allotted to document class.
iii. FN - total of text documents wrongly rejected from document class.
iv. TN - total of text documents correctly rejected from document class.

These parameters are input to the evaluator. They are obtained from the classification result. Figure 9 displays the performance measure module of the Igbo text classification system.

## V. EXPERIMENT

This entails the means of putting into operational all the theoretical design of the proposed Igbo text classification model. The Igbo text classification system is implemented with Python and tools from Natural Language Toolkit (NLTK).



TEXT-CLASSIFICATION MODULE

Select the Files to perform Text Classification on:

K value:

K value for Similarity Measurement

N-gram:

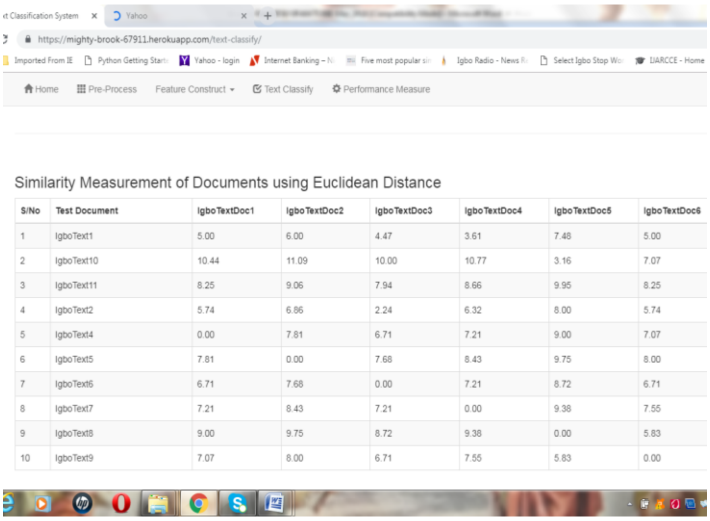
Bigram

Select the training documents:

- □ IgboTextDoc1
- □ IgboTextDoc2
- □ IgboTextDoc3
- □ IgboTextDoc4
- □ IgboTextDoc5
- □ IgboTextDoc6

Select the testing documents:

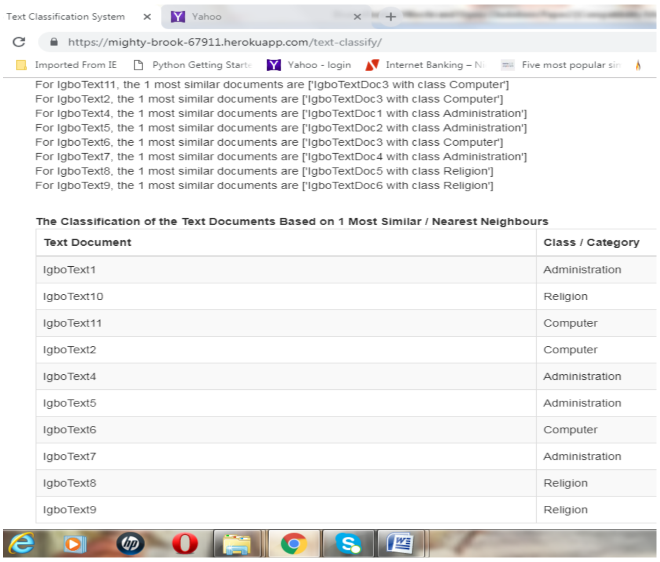
Figure 6: Text Classification Module of the System



Similarity Measurement of Documents using Euclidean Distance

| S/No | Test Document | IgboTextDoc1 | IgboTextDoc2 | IgboTextDoc3 | IgboTextDoc4 | IgboTextDoc5 | IgboTextDoc6 |
|---|---|---|---|---|---|---|---|
| 1 | IgboText1 | 5.00 | 6.00 | 4.47 | 3.61 | 7.48 | 5.00 |
| 2 | IgboText10 | 10.44 | 11.09 | 10.00 | 10.77 | 3.16 | 7.07 |
| 3 | IgboText11 | 8.25 | 9.06 | 7.94 | 8.66 | 9.95 | 8.25 |
| 4 | IgboText2 | 5.74 | 6.86 | 2.24 | 6.32 | 8.00 | 5.74 |
| 5 | IgboText4 | 0.00 | 7.81 | 6.71 | 7.21 | 9.00 | 7.07 |
| 6 | IgboText5 | 7.81 | 0.00 | 7.68 | 8.43 | 9.75 | 8.00 |
| 7 | IgboText6 | 6.71 | 7.68 | 0.00 | 7.21 | 8.72 | 6.71 |
| 8 | IgboText7 | 7.21 | 8.43 | 7.21 | 0.00 | 9.38 | 7.55 |
| 9 | IgboText8 | 9.00 | 9.75 | 8.72 | 9.38 | 0.00 | 5.83 |
| 10 | IgboText9 | 7.07 | 8.00 | 6.71 | 7.55 | 5.83 | 0.00 |

Figure 7: Similarity Measurement of Igbo Documents using Euclidean Distance

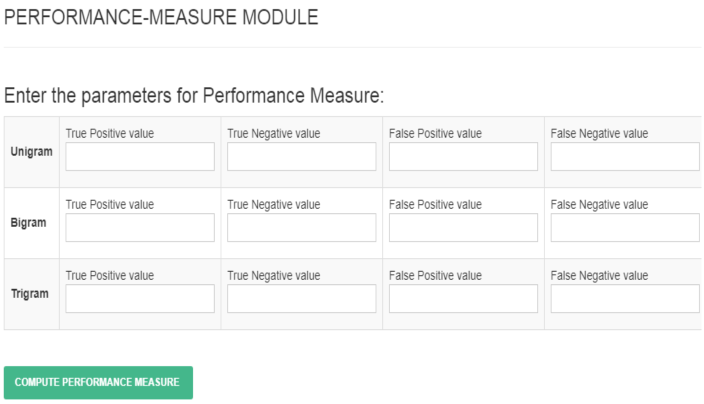


```
For IgboText11, the 1 most similar documents are ['IgboTextDoc3 with class Computer']
For IgboText2, the 1 most similar documents are ['IgboTextDoc3 with class Computer']
For IgboText4, the 1 most similar documents are ['IgboTextDoc1 with class Administration']
For IgboText5, the 1 most similar documents are ['IgboTextDoc2 with class Administration']
For IgboText6, the 1 most similar documents are ['IgboTextDoc3 with class Computer']
For IgboText7, the 1 most similar documents are ['IgboTextDoc4 with class Administration']
For IgboText8, the 1 most similar documents are ['IgboTextDoc5 with class Religion']
For IgboText9, the 1 most similar documents are ['IgboTextDoc6 with class Religion']
```

**The Classification of the Text Documents Based on 1 Most Similar / Nearest Neighbours**

| Text Document | Class / Category |
|---|---|
| IgboText1 | Administration |
| IgboText10 | Religion |
| IgboText11 | Computer |
| IgboText2 | Computer |
| IgboText4 | Administration |
| IgboText5 | Administration |
| IgboText6 | Computer |
| IgboText7 | Administration |
| IgboText8 | Religion |
| IgboText9 | Religion |

Figure 8: Igbo Text Classification System Result on Bigram Represented Text when k =1

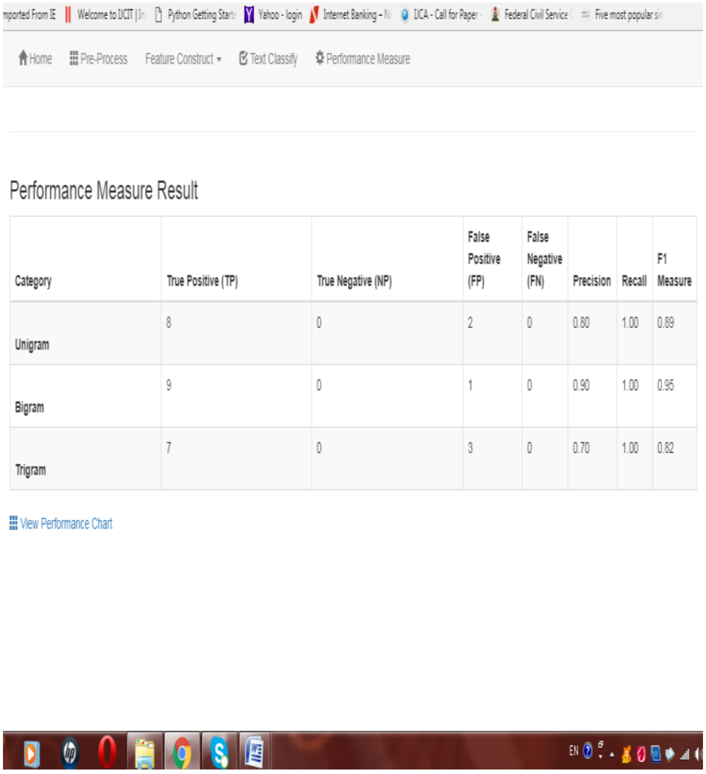


PERFORMANCE-MEASURE MODULE

Enter the parameters for Performance Measure:

| | | | | |
|---|---|---|---|---|
| Unigram | True Positive value | True Negative value | False Positive value | False Negative value |
| Bigram | True Positive value | True Negative value | False Positive value | False Negative value |
| Trigram | True Positive value | True Negative value | False Positive value | False Negative value |

Figure 9: Igbo Text Classification System Performance Module



Performance Measure Result

| Category | True Positive (TP) | True Negative (NP) | False Positive (FP) | False Negative (FN) | Precision | Recall | F1 Measure |
|---|---|---|---|---|---|---|---|
| Unigram | 8 | 0 | 2 | 0 | 0.80 | 1.00 | 0.89 |
| Bigram | 9 | 0 | 1 | 0 | 0.90 | 1.00 | 0.95 |
| Trigram | 7 | 0 | 3 | 0 | 0.70 | 1.00 | 0.82 |

Figure 10: Performance Measure Result

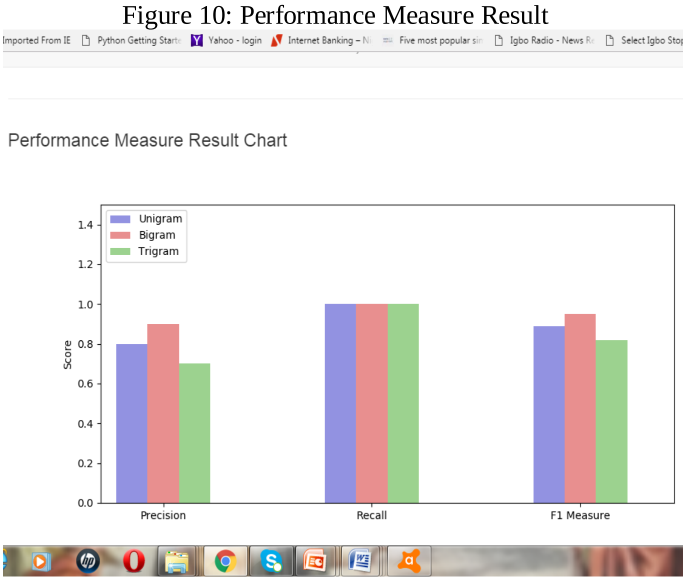


Figure 11: Igbo Text Classification System Performance Measure Result Chart

Table 1: Summary of Unigram, Bigram and Trigram Text Classification result

| Text Document | Unigram | Bigram | Trigram |
|---|---|---|---|
| IgboText1 | Administration | Administration | Administration |
| IgboText10 | Religion | Religion | Religion |
| IgboText11 | Computer | Computer | Administration |
| IgboText2 | Religion | Computer | Religion |
| IgboText4 | Administration | Administration | Administration |

| Text Document | Unigram | Bigram | Trigram |
|---|---|---|---|
| IgboText5 | Religion | Administration | Administration |
| IgboText6 | Computer | Computer | Administration |
| IgboText7 | Administration | Administration | Administration |
| IgboText8 | Religion | Religion | Religion |
| IgboText9 | Religion | Religion | Religion |

## VI. RESULT ANALYSIS

Figure 6 displays the Text classification module of the system. The K-value and N-gram model are the input parameters required to supply before classification can be done. The value of K determines the number of most similar / nearest documents to consider when assigning the class label / name. The document(s) to classify is selected from the testing documents set.

Figure 7 shows the display of result obtained when similarity measurement is performed on Igbo text documents using Euclidean distance metric function. Figure 8 is a display of Igbo Text Classification System Result on Bigram Represented Text when k =1.

Figure 10 and figure 11 shows the classification performance measure result chart respectively. The result shows that the recall, precision and F1 for unigram are 1.00, 0.80 and 0.89 respectively. The recall, precision and F1 for bigram are 1.00, 0.90 and 0.95 respectively. The recall, precision and F1 for Trigram are 1.00, 0.62 and 0.82 respectively. Recall measures the degree of completeness. The result shows Igbo text classification on the text represented with the three models (unigram, bigram and trigram) has the same level of recall (completeness). This means all the text documents that were given to the classifier, were given a label name. Precision measures the degree of exactness. The classification with bigram has highest degree (0.90) of exactness (precision) while trigram has the lowest degree (0.62) of exactness. F1 measures the classification system accuracy by taking into consideration the precision and recall to calculate its value. F1-measure is at its finest score (value) at 1 and at its worst at 0. Bigram represented text classification has the highest value (0.95) of F1 while Trigram has the lowest value (0.82).

Table 1 gives the summary of classification output on Unigram, Bigram and Trigram represented texts. A total of 10 testing documents are used for the experiment. In Unigram, eight documents are correctly assigned a class label while two are incorrectly assigned a class label. In bigram, 9 documents are correctly assigned a class label while one is incorrectly assigned. In trigram, 7 documents are correctly assigned a class label while 3 are incorrectly assigned a class label.

## VII. CONCLUSION

In this work, an improved model for representation and classification of Igbo texts using N-gram and KNN model based on the similarity measure is developed and implemented. This model richly represented Igbo text using N-gram model of length 3 (Unigram, Bigram and Trigram) considering the compound nature of the Igbo words. Relevant features were selected with Mean TF-IDF technique. The text classification was performed on the selected features using K-Nearest Neighbour technique based on similarity measure. The similarities between the Igbo text documents are measured with Euclidean distance metric.

The performance was measured by computing the classification accuracy of Unigram, Bigram and Trigram represented text. The result showed that the classification performed on bigram represented text has higher performance than unigram and trigram represented texts.

Igbo text classification model will be of high commercial potential value and is recommendable for adoption in the development of any intelligent text-based system in Igbo. More researchers are ought to be motivated by this work to develop interest to think out ways to bring Information Technology fully into the indigenous language to the profit of people and society.

## ACKNOWLEDGMENT

The authors desire to appreciate the reviewers of this paper. Your useful comments, suggestions, guidelines and contributions helped immensely in improving the quality of this work.

## REFERENCES

[1] M. Thangaraj, M. Sivakami, 2018. Text classification techniques: A literature review. Interdisciplinary Journal of Information, Knowledge, and Management. 13: 117-135.

[2] K. S. Dileep, S. Vishnu, 2013. Data security and privacy in data mining: research issues & preparation. International Journal of Computer Trends and Technology. 4(2):194 -200.

[3] R. Elhassan, M. Ahmed, 2015. Arabic Text Classification Review. International Journal of Computer Science and Software Engineering (IJCSSE). 4(1): 1-5.

[4] A. Roiss, O. Nazlia, 2015. Arabic text classification using K-nearest neighbour algorithm. The International Arab Journal of Information Technology. 12(2): 190 - 195.

[5] T. Joseph, 2013. Text analysis: A crucial part of enterprise data initiatives. Symposium on Technology and Society (ISTAS). 191 - 200.

[6] T. R. Mahesh, M. B. Suresh, M. Vinayababu, 2010. Text mining: advancements, challenges and future directions. International Journal of Reviews in Computing (IJRIC). 8(3): 61-65.

[7] R. Janani, S. Vijayarani, 2016. Text mining research: A survey. International Journal of Innovative Research in Computer and Communication Engineering. 4(4). 6564 – 6571.

[8] K. L. Sumathy, M. Chidambaram, 2013. Text mining: concepts, applications, tools and issues – an overview. International Journal of Computer Applications. 80(4):29 -32.

[9] G. Pratiyush, S. Manu, 2014. Data Mining In Education: A Review On The Knowledge Discovery Perspective. International Journal Of Data Mining & Knowledge Management Process (IJDKP). 4(5): 47 -60.

[10] P. A. De Maziere, M. Marc, V. Hulle, 2011. A clustering study of a 7000 EU document inventory using MDS and SOM. Expert Systems with Applications. 38(7): 8835–8849.

[11] R. Jindal, S. Taneja, 2015. A Lexical Approach for Text Categorization of Medical Documents. International Conference on Information and Communication Technologies (ICICT 2014). Procedia Computer Science. 46(2015 ):314 – 320.

[12] K. Kim, S. Y. Zzang, 2019. Trigonometric comparison measure: A feature selection method for text categorization. Data & Knowledge Engineering. 119 (2019): 1–21.

[13] N.J. Ifeanyi-Reuben, C. Ugwu, T. Adegbola, 2017. Analysis and representation of Igbo text document for a text-based system. International Journal of Data Mining Techniques and Applications (IJDMTA). 6(1): 26-32.

[14] N. J. Ifeanyi-Reuben, M. E. Benson-Emenike, 2018. An Efficient Feature Selection Model for Igbo Text. International Journal of Data Mining & Knowledge Management Process (IJDKP). 8(6):19 -33.

[15] O. Ndimele, 1999. A first course on morphology & syntax. ISBN 978-33527-3-3.

[16] I. E. Onyenwe, C. Uchechukwu, M. Hepple, 2014. Part –of-Speech tagset and corpus development for Igbo, an African language, The 8th linguistic annotation workshop, Dublin, Ireland, 93-98.

[17] R. Al-khurayji, A. Sameh, 2017. An Effective Arabic Text Classification Approach Based on Kernel Naive Bayes Classifier. International Journal of Artificial Intelligence and Applications (IJAIA). 8(6): 1 -10.

[18] A. Adel, N. Omar, A. Al-Shabi, 2014. A comparative Study of Combined Feature Selection Methods for Arabic Text Classification. Journal of Computer Science. 10(11): 2232 – 2239.

[19] P. Kumbhar, M. Mali, 2016. A Survey on Feature Selection Techniques and Classification Algorithms for Efficient Text Classification. International Journal of Science and Research (IJSR). 5(5): 1267 – 1275.

[20] G. Raho, R. Al-Shalabi, G. Kanaan, A. Nassar, 2015. Different Classification Algorithms Based on Arabic Text Classification: Feature Selection Comparative Study. (IJACSA) International Journal of Advanced Computer Science and Applications. 6(2):192 – 195

[21] S. Rajasree, R.Chinmaya, K.Dharsini Prabha, B. Indujapriya, 2021. Deep Learning Approach for COVID-19 Meme Categorization. International Journal of Innovative Science and Research Technology. 6(5): 1296 – 1299

[22] H. Naji, W. Ashour, M. Al Hanjouri, 2018. Text Classification for Arabic Words Using BPSO/REP-Tree. International Journal of Computational Linguistics Research. 9(1): 1 – 9.

[23] J. Shao, A. Tziatzios, 2018. Mining Range Associations for Classification and Characterization. Data & Knowledge Engineering 118 (2018) 92–106.

[24] S. Bird, E. Klein, E. Loper, 2009. Natural language processing with Python. Published by O'Reilly Media, Inc., 1005 Gravenstein Highway North, Sebastopol, CA 95472.

[25] [25] S. N. Arjun, P. K. Ananthu, C. Naveen, R. Balasubramani, 2016. Survey on pre-processing techniques for text mining. International Journal of Engineering and Computer Science. 5(6): 16875-16879.

[26] M. Harmain, H. El-Khatib, A. Lakas, 2004. Arabic Text Mining. College of Information Technology United Arab Emirates University. Al Ain, United Arab Emirates. IADIS International Conference Applied Computing. 2(2004): 33 -38.

[27] N. J. Ifeanyi-Reuben, C. Ugwu, 2018. An Enhanced Approach for Preprocessing Igbo Text. Nigeria Computer Society (NCS) 27th National Conference Proceedings - Digital Inclusion 2018. Vol. 29: 93 – 101.

[28] D. Shen, J. Sun, Q. Yang, Z. Chen, 2006. Text classification improved through multi-gram models," In Proceedings of the ACM Fifteenth Conference on Information and Knowledge Management (ACM CIKM 06), Arlington, USA. 672-681.

[29] D.L. David, 1990. Representation quality in text classification: An Introduction and Experiment. Selected papers from the AAAI Spring Symposium on text-based Intelligent Systems. Technical Report from General Electric Research & Development, Schenectady, NY, 1230.

[30] P. Divya, K. G. S. Nanda, 2015. Study on feature selection methods for text mining. International Journal of Advanced Research Trends in Engineering and Technology (IJARTET). 2(1): 11- 19.

[31] T. Bruno, M. Sasa, D. Dzenana, 2013. KNN with TF-IDF based framework for text categorization. 24th DAAAM International Symposium on Intelligent Manufacturing and Automation. Procedia Engineering. 69 (2014):1356 – 1364.

[32] K. Khushbu, 2013. Short text classification using KNN based on distance function. International Journal of Advanced Research in Computer and Communication Engineering. 2 (4): 1916 – 1919.

[33] S. M. Kavitha, P. Hemalatha, 2015. Survey on text classification based on similarity. International Journal of Innovative Research in Computer and Communication Engineering. 3(3):2099 – 2101.