Available Online at www.ijcsmc.com

International Journal of Computer Science and Mobile Computing

A Monthly Journal of Computer Science and Information Technology

ISSN 2320–088X
IMPACT FACTOR: 6.199

IJCSMC, Vol. 8, Issue. 12, December 2019, pg.46 – 58

An Enhanced Model for the Classification of Mined Data

Agi U.[1]; Onyejegbu L. N[2]; Onuodu F.E.[3]

[1, 2, 3]Department of Computer Science, University of Port Harcourt, Nigeria

[1] uchechukwuagi17@gmail.com; [2] nneka2k@yahoo.com; [3] gonuodu@gmail.com

Abstract— Text mining is the process of deriving high-quality information from text. It typically involves the process of structuring the input text, extracting previously unknown pattern and deriving patterns with the structured data. Text classification is the processing of classifying documents into pre-defined categories based on their contents. Unstructured data is typically text heavy and difficult to handle. In this work, we developed an enhanced model for mining and classification of textual data. The methodology used is Object Oriented System Development Methodology (OOSDM) in its approach. We used both K-Nearest Neighbour (KNN) Algorithm and Euclidean Distance Classifier for text mining and classification using data mining that requires fewer documents for training. Also, we employed the association rules from words to derive feature set from pre-classified textual documents and used Mean Term Frequency Inverse Document Frequency (Mean TF-IDF) Model for the feature Selection. The proposed system was implemented with C# Programming language and MySql connector was used to store the dataset in the database. The results of the text to speech from the software show that the model has a shift in pronunciation comparing to the human pronunciation of the natural language. This work could be beneficial to any organization that deals with language interpretations.

Keywords: Text mining, Text classification, Data mining, KNN, Euclidean Distance Classifier, Mean TF-IDF

I. INTRODUCTION

Data mining as a tool of extraction and management of useful information that are hidden in the huge quantity of textual documents has generated big concern to Information Technology (IT) professionals; together with its semi-structured and unstructured nature. It is difficult to explore the useful information present in the unstructured text documents of web pages, textual content management system, news articles, etc. This is faced with lack of sophisticated text analysis model for discovering new information in their semi-structure or unstructured manner.

Text mining, gotten from the meaning textual data mining is referred to as knowledge discovery from text. Discovering knowledge and ideas from text is extracting interesting but hidden patterns or trends from textual documents [1]. Text mining is regarded to be of high commercial values. This is proved by [2], which stated that Text Mining is the natural forms of storing information as text. Text mining is believed to have a commercial potential higher than that of data mining. In fact, a recent study indicated that 80% of a company's information is contained in textual documents. Text mining, involves a complex task because of the unstructured nature of the textual data. To put the text in a structured format before analysis can be performed on it a particular task must be carried out [3]. Text mining adopts models and algorithms from machine learning, data mining,

information retrieval and natural language processing to extract knowledge from the text automatically [4]. Vishwadeepak and Tripathi [5] defined text mining as an Intelligent Text Analysis.

Text mining incorporates mainly tools from data mining. Data mining, also referred to as the Knowledge Discovery in Databases (KDD), is the extraction of hidden, implicit and useful information from data in databases [1]. It involves the use of refined data analysis techniques to find out previously unknown, valid patterns and relationships in database. "These tools can include statistical models, mathematical algorithms, and machine learning methods. Machine learning methods are the algorithms that improve their performance automatically through experience, such as neural networks or decision trees" [6]. Text data mining techniques is useful in web search engine, marketing, manufacturing, process control, fraud detection, bioinformatics, electronic commerce and many others.

Text mining is a quite new research area, which has created much concern among researchers, due to the continuously increasing of electronic text. Text mining is a multidisciplinary research area, bringing together research studies from areas of data mining, natural language processing, machine learning, and information retrieval [7].

II. RELATED WORKS

Chowdhury, et al. [8] experimented and confirmed that the availability of online text is increasing rapidly. According to them, text is not costly, but information, in the form of knowing what classes a text belongs to, is costly. In their paper, they used association rule mining method to get feature set from pre-classified text documents and derived sub-set of features for the model using Naïve Bayes classifier.

Harmain, et al. [9] presented the architecture of Arabic text mining system as well as some issues involved in its mining task. This started with a preprocessing task, which converted the Arabic HTML documents to XML documents. The preprocessed text is then analyzed based on linguistic factors from the word level to the text level. The output of the analysis represented as a semantic network of the entities of the text and the relationships among them is established. This semantic network is then made available for some particular mining tasks.

Mofleh [10] investigated the problem of unstructured textual document using various data mining association rule-based technique. The association rule-based used are One Rule, rule induction (RIPPER), decision trees (C4.5), and hybrid (PART). The results of the research work showed that the hybrid (PART) approach achieved improved performance than others. This research was done on Arabic text. It is very necessary this data inform of text is managed efficiently.

Sabarin [11] presented a new and efficient pattern detection model that combines the processes of pattern deploying and pattern evolving, to improve the effectiveness of victimization and amend discovered patterns for locating important and interesting facts. Pattern mining regards to data mining involves searching for existing useful patterns in data.

Bhujade and Janwe [12] presented a text mining technique named EART (Extracting Association Rules from Text). The purpose of this technique is to extract association rules from textual documents collections automatically. It uses keyword features to find out association rules amongst keywords tagging the documents. EART system does not consider the word ordering but focuses only on the words and their statistical distributions in the documents. The system uses TF-IDF weighting scheme for selecting relevant keywords for the association rules generation. The system consists of three phases: Text Preprocessing, which involves text transformation, filtration, stemming and indexing of the documents; Association Rule Mining (ARM) phase, which involves applying their designed algorithm for generating association rules based on weighting scheme and finally Visualization phase, which involves visualization of the results. The experiment of the system was done and applied on Online WebPages and the extracted association rules contained important features.

Jafar, et al. [13] developed a model for an Arabic text extraction using a vector space model. Three different variations of Vector Space Models (Cosine, Dice, Jaccard) using K-Nearest Neighbour algorithms were investigated with their results compared. The performance of the system was tested, measured and it was confirmed that the Cosine performed better than Dice and Jaccard.

Saleh [14] evaluated the performance of Naïve Bayesian method (NB) and Support Vector Machine algorithm (SVM) on different Arabic data sets. Three evaluation measures, Recall, Precision, and F1were used as the bases of our comparison. The average of three measures obtained against Arabic data sets indicated that the SVM algorithm outperformed NB algorithm regards to F1, Recall and Precision measures. The work was based only on single labelled document.

Kavitha and Hermalatha [15] considered various methods of text categorization and feature selection based on similarity measure. They observed that similarity method of handling these textual documents in a real-valued method that quantifies the similarity between two items and shows the extent of closeness or separation of the target items.

III. MATERIALS AND METHODS

Jafar, et al [13] investigated an unstructured Arabic text documents with different variations of Vector Space Models (VSM) using KNN algorithm. The work classified Arabic text corpus into predefined categories based on their contents. The experiment was done on the Saudi data sets (Saudi Newspapers – SNP). The data consisted of some numbers of Arabic documents of different lengths (like: Culture, Economics, General, Information Technology, and Politics). They compared results obtained against Arabic text collections using K-Nearest Neighbor algorithm. Three different variations of Vector Space Models-VSMs (Cosine, Dice, Jaccard) using KNN algorithm were investigated. The average of three measures obtained against seven Arabic data sets indicated that the Cosine performed better than Dice and Jaccard. IDF weighting method was used and the system was implemented using JAVA.

Figure 1 shows the architecture of the existing system to classify an Arabic text. It started with an input of an Arabic textual data, which is then preprocessed. Its preprocessing task involves the following steps;

A. Input Data
Input can be done in different forms. From commands you enter from the keyboard to data from another computer.

B. Text Tokenization
This task converts the Arabic documents from sequences of characters into sequences of tokens.

C. Arabic Text Normalization
This is the process of transforming Arabic text into single canonical form that it might not have had before.

D. Non Arabic Text Filtration
The non-Arabic texts are filtered out.

E. Function Words Removal
The function words or stop words are removed. These are words that are not useful to the system like diacritics, special characters, punctuation marks, Arabic prefixes, pronouns and prepositions.

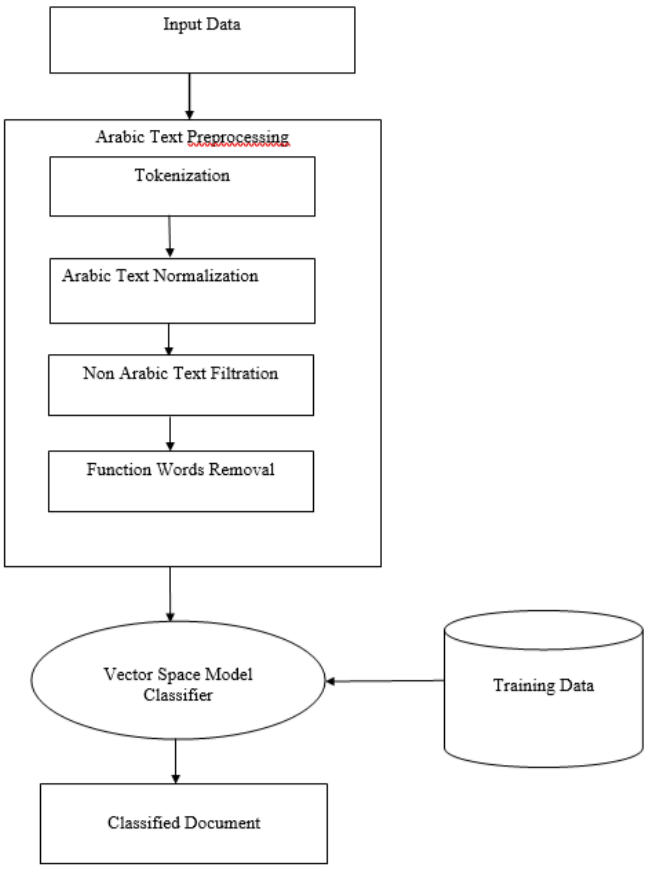

Fig. 1: Architecture of the Existing System (Source: Jafar et al, 2014)

A. *Algorithm of the Proposed System*

A hybrid algorithm will be used to combine desired features of each algorithm, so that the overall algorithm is better than the individual components. Specific properties of statistical distribution to determine centers with boundaries of each cluster will be employed. This algorithm is significantly less time consuming, efficient and convenient for finding clusters within the datasets.

Algorithm 1: Steps to Remove Discrete Words;

1. Read the stop word file
2. Convert all loaded stop words to lower case
3. Read each word in the created Token List
4. For each word (q) Token List of the document
5. Check if q (Token List) is in Language stop word list
6. Yes, remove q (Token List) from the Token List
7. Decrement tokens count
8. Move to the next q (Token List)
9. No, move to the next q (Token List)
10. Exit Iteration Loop
11. Do the next task in the pre-processing process

Algorithm 2: K-Nearest Neighbour Classifier

Procedure: Find the class label

Input: k-value; the number of nearest neighbours; V – Testing data set; W – Training data set;

Output: C, Label set of testing data set

1. Read Training data file
2. Read Testing data file
3. Perform pre-processing
4. Select relevant features
5. Compute the distance (d) metric in the data set.
6. Determine the similarity or dissimilarity based on the computed distance.
7. Determine the k-value.
8. To decide whether the document belongs to a class;
9. Assign C = { }, a set or list that holds the class labels
10. For each v in V and each w in W do
11. Neighbours (v) = { }
12. if $|$ Neighbours(v) $|$ < k then
13. Neighbours (v) = closest (v,w) $\cup$ Neighbours (v)
14. End if
15. if $|$ Neighbours(v) $|$ = k then
16. C = test Class (Neighbours (v) $\cup$ C
17. End for
18. Exit Classifier

B. *Architecture of the proposed system*

In the proposed system we added feature construction (feature representation and feature selection), K – Nearest Neighbour and Euclidean Distance Classifier to achieve our aim and objectives. We also, carried out Performance Evaluation by computing the parameters for precision, efficiency and effectiveness of the system as depicted in figure 2.

1) Text Representation

A classifier cannot directly interpret text. The raw text must first, be mapped into a compact representation. The choice of the representation varies across application and depends on what one considers the meaningful units of texts are. A feature can be as simple as a single token, or a linguistic phrase (Jeffery 2004). The proposed system will represent text in a single token, 2-tokens, 3-tokens (n-gram model, n =1, 2, 3) as opposed to the present system that represented its text only in single words. This is adopted to capture the issues of collocation and correlation in the corpus that were not considered in the present system.

2) *Feature Selection*

The feature selection is the process of selecting a subset of relevant features for use in the proposed system. It is used to handle the high dimension of data for effective text classification and it focuses on identifying relevant features without affecting the accuracy of the classifier. This module will serve as a filter; muting out irrelevant, unneeded and redundant attributes or features from the sample textual data to boost the performance

of the system. Feature selection is performed by keeping the words with highest score according to predetermined measure of the importance of the term. The goal of this feature selection is summarised in three fold;

(i) Reducing the amount of features;
(ii) Focusing on the relevant features; and
(iii) Improving the quality of features as well as the efficiency and performance of the proposed text classification model.

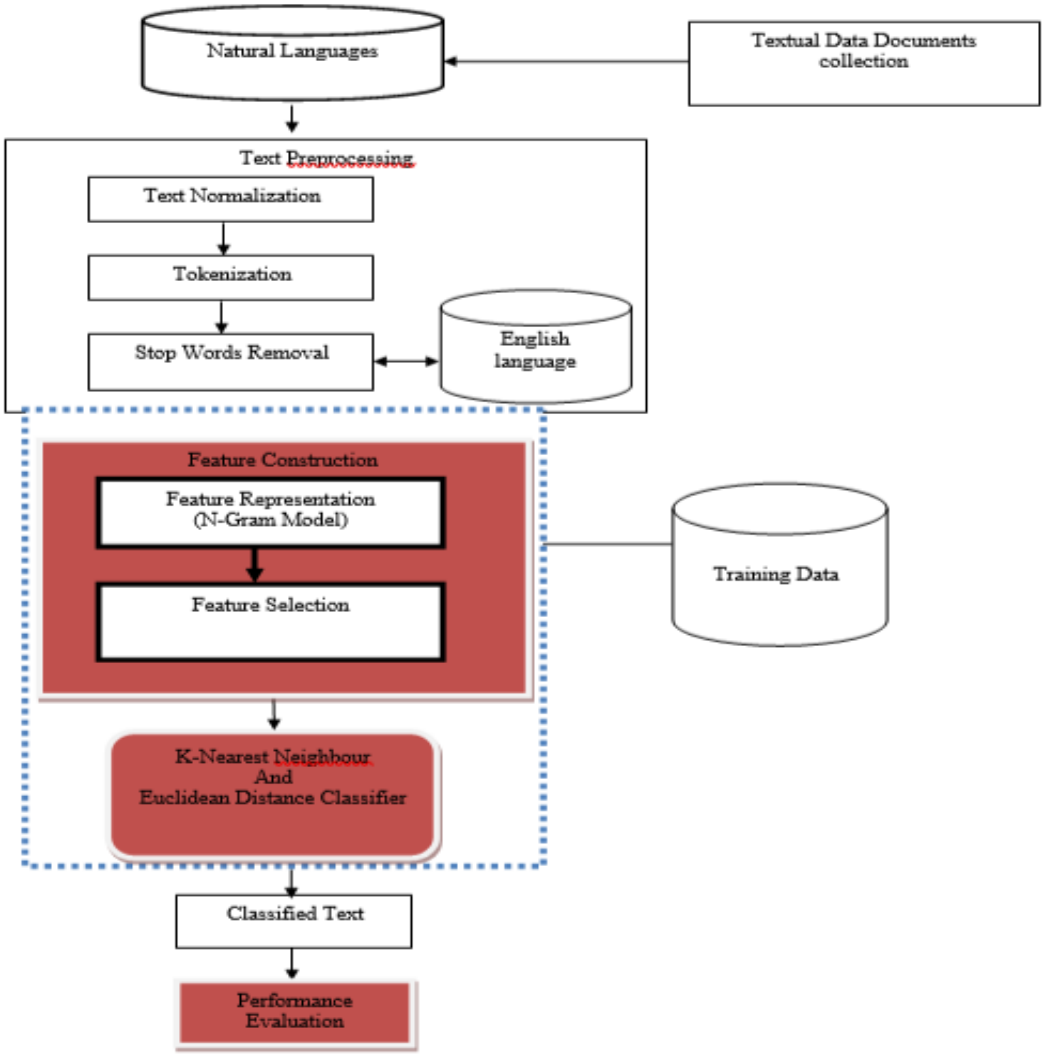

Fig. 2: Architecture of the Proposed System

TABLE I
DATASET OF THE PROPOSED SYSTEM

S/N	English text	Igbo text
1.	Money	Ego
2.	I am coming	Ana m abia
3.	When will you come	Olee mgbe I ga-abia
4.	My children are in school	Umu m no n ulo akwukwo
5.	We are here	Anyi no ebe a
6.	Please come here	Biko bia ebe a
7.	My name is Uche	Aha mbu Uche
8.	This is the beginning of success	Nke a bu mmalite nke mkpa
9.	It is complete	O zuru ezu
10.	Where are you	Ebee ka ino

C. Output/Input Specification

The input and output specification rely on spatial dataset using K-Nearest Neighbour and Euclidean Distance Classifier Algorithm. The input specifications are the data type used, attributes, data encapsulation etc, as shown in table II.

TABLE III
INPUT AND OUTPUT SPECIFICATION

FIELD NAME	FIELD TYPE	FIELD WIDTH	DESCRIPTION
Date of entry	Varchar	10	Date the user record enter the database
Type of language	Varchar	30	Initializing the type of language to be processed and transformed
Algorithm	Varchar	15	Identifies the algorithm used in transformation
Time used to access the site	Varchar	20	States the time that the users is accessing the site

The specification used in storing data is MySql database server. We used database storing specifications prior to the Software and Hardware requirements. The Software Requirements Specification (SRS) is a description of a software system to be developed. The Hardware specifications are technical descriptions of the computer`s components and its ability to perform a specified task. We followed an appropriate Software Requirement Specifications that can prevent system software failure and ensured our software system will interact with all internal modules, hardware, communicate with other programs and human user interactions

with wide range of real life applications. Figure 3 displays the home page of the program and the screen output shows how the textual data documents (language) are been translated into data mining language with a "speech function" as depicted in figure 4.

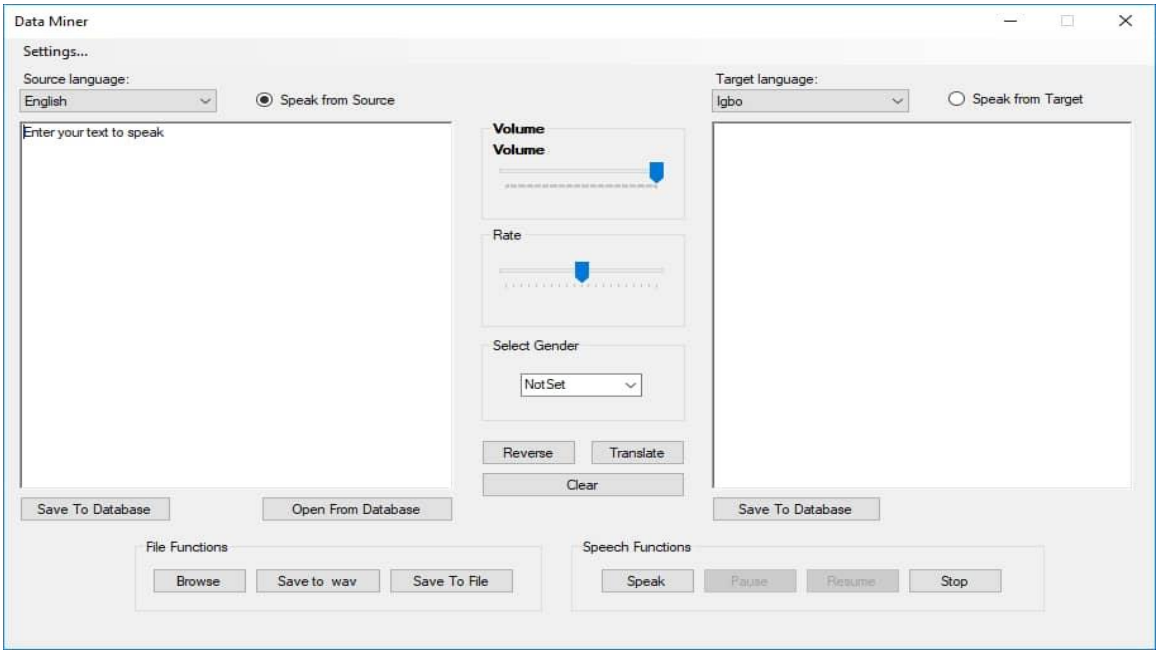

Fig 3. Home page of the Program

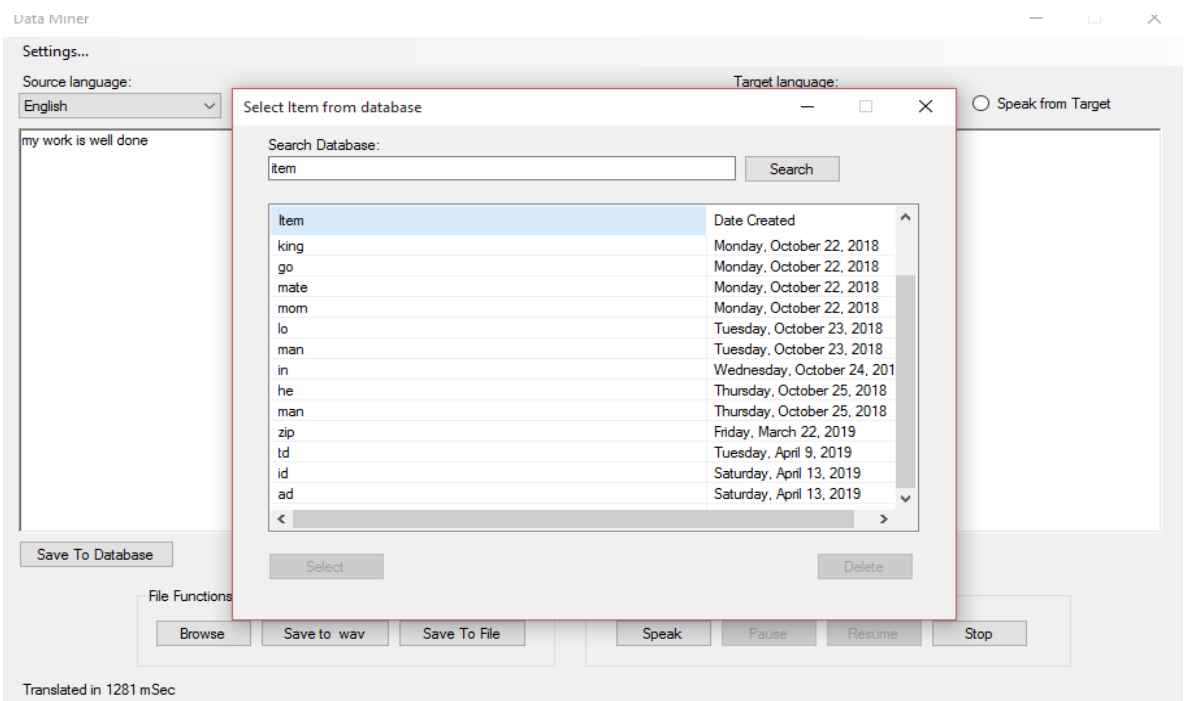

Fig 4. Data Analysis

D. How the Proposed System Works

The proposed system uses the source language which comprises of English and our native languages (Hausa, Igbo and Yoruba), mine it and then translate to the preferred or target language in the text editor, before saving

to either of the databases. Also, text or words can be imported or extracted from the databases in order to be mined and translated into the target language. The speech function enables the words translated to be vocal to the users for accurate pronunciation of the textual data.

E. *Pseudo-code of the Proposed System*

1. Source language comprises of English, Hausa, Igbo, and Yoruba

2. The "speak from source" enables the data miner to speak text written in the preferred language in the text editor.

3. *Save to Database*
   a. Enter your text in the text editor
   b. Click on save to database
   c. In the prompt, save to data/data label, type a filename e.g Uche
   d. Then, click save
   e. In the prompt record update, click ok

4. *To Open from Database*
   a. Click open from database
   b. Enter the file name in the search engine
   c. Scroll to the file name
   d. Click on it
   e. Again, click on select to launch the saved data in the text editor
   f. Click on delete, to delete the stored data from the database
5. *File functions*
   a. Click on browse, then filename
   b. Click on open
   c. The text document opens in the text editor

6. *Save to Wave*: it is an option to save file, before using the speech functions to speak
   a. Click on save to wave
   b. Enter the filename in the column provided, then save
   c. Go to document, locate the file
   d. Double click on the filename to import and speak

7. *Save to file*;
   a. From the text editor, enter your text
   b. Click on save to file, type file name
   c. Click on save

8. *To open data saved to file;*
   a. Go to document
   b. Locate and click on file name
   c. It will be displayed on notepad

To return back the saved file to the text editor
   d. Click on browse
   e. Click on file name
   f. It returns to text editor

9. *Speech function*:
   a. click on speech, it sounds out the text entered on the text editor

Pause: It interrupts the speech briefly
Return: To go back to the previous place
Stop: To cause the speech to end

10. *Volume*: to increase or reduce the level of sound produced in the data miner

11. *Rate*: to assign a standard or value to the data miner according to performance

12. *Select Gender*: it comprises of four options not set, male, female and neutral. Any option chosen determines the sound to be produced

13. *Reverse*: it will erase the translated text in the target language

14. *Translate*: it expresses the sense of words or text in another language

15. *Clear*: it will remove text from both source language, and target language

16. The target language comprises of English, Hausa, Igbo and Yoruba

17. *Speak from Target*: it enables the data miner to speak the translated (Target) language.

F. Mathematical Model

Let a document D be decomposed into individual sentences *D* = (S1; S2; ::: S*n*), where n is the number of sentences in the document *D*. Let *T* = (t1; t2; :::; *tm*) represents all the terms occurred in a document *D*, where *m* is the number of terms. Extractive summarization works by choosing a subset of the sentences in the original document. This process can be viewed as identifying the most salient sentences in a document, that give the necessary and sufficient amount of information related to the main theme of the document. To identify the most salient sentences in a document they must at first be represented in a corresponding form. As the model which expresses the feature of the sentence, the K-Nearest Neighbour and Euclidean Distance Classifier has been proposed, each sentence can be represented as a vector *Si = (wi1;wi2; :::;wim)* of features (the features are usually derived from the terms appearing in a document) with associated weights, where *wij* denotes the weight of term *tj* in the sentence *Si*.

A more advanced term weighting scheme is the *tf-idf* (term frequency - inverse document frequency). The *tf-idf* weighting scheme is a commonly used information retrieval technique for assigning weights to individual terms appearing in document.

This scheme aims at balancing the local and the global term occurrences in the documents:

$$wij = fij\ log\ \frac{n}{Nj}$$

(1) where: *fij* (local weight) { the number of occurrences of term *tj* in the sentence *Si* and *nj* denotes the number of sentences containing the term *t j*.

In formula (1) the log *³n/nj´*, which is very often referred to as the idf factor, accounts for the global weighting of term *tj*. Indeed, when a term appears in all sentences in the document, *nj = n* and thus the balanced term weight is 0, indicating that the term is useless as a sentence discriminator.

After obtaining the term weights of all terms, it's easy to apply traditional similarity measures like the cosine similarity to compute the similarity of two sentences. The cosine similarity between two sentences *Si* and *Sl* is defined as:

$$\mathrm{Cos}(Si;SI) = \frac{\sum_{j=1}^{m} \mathrm{wij\ w1j}}{\sqrt{\sum_{j=1}^{m} \mathrm{w2\ ij} \sum_{j=1}^{m} \mathrm{w2\ ij}}} \quad \mathrm{I, I-1, J, n}$$

(2) where *wij* is the term weight of the term *tj* in the sentence *Si = (wi1;wi2; :::;wim), i = 1; 2; :::; n, j = 1; 2; :::;m.*

To evaluate the importance of the sentences, we use classical IR precision and recall measures. To calculate the similarity measure of each sentence, it must at first be represented in a suitable form. In our method, a sentence *Si* is represented as bag-of-terms *Si = (t1; t2; ::: tmi)*, instead of the term-based frequency vector. Here *mi* denotes the number of terms in a sentence *Si*: The similarity between pair of sentences *Si* and *Sl* is evaluated to determine if they are semantically related. The similarity between sentences *Si* and *Sl* we define as

$$F(Si;SI) = 2P(Si;\frac{(\mathrm{SI})\mathrm{R(Si;\ S)}}{\mathrm{P(Si;\ SI + R(Si;\ SI)}}\ i\,6 =\ 1, 2;:::;\ n:$$

(3) Our approach to text summarization allows generic summaries by scoring sentences. Each text is

scored according to the formula. The relevance score of Si with regard to all sentences in a document D (based on F -measure), we compute as:

Fscore(Si) = X*n l* = 1 *l 6= i*

F(Si; Sl); i = 1; 2; :::; *n*:

(4) Since the main purpose is to show the effectiveness of the application of an F- measure as the similarity measure, that analogously we determine the relevance score of Si with regard to all sentences in a document D (based on cosine measure):

Cscore(Si) = ∑_(i-1)^n cos 〖(S1,Sj),i=1,2,n〗
(5)

Finally, prior to selection of sentences to generate a summary all sentences are ranked according to their relevance scores calculated from formulae (4) and (5), and a designated number of top-weight text are picked out to form the summary.

Thus the generation summary process consist the following steps:

- Decompose the document into individual sentences.
- Represent each text as bag-of-terms
- Using the formulae (1) and (2) described in for each pair of sentences *Si* and *Sl* compute the similarity measure.
- Using the formulae (4) and (5) for each sentence *Si*, compute the relevance score.
- Rank all text according to their relevance score.
- Starting with the text which has a highest relevance score the sentences add to the summary. If the compression rate (CR), which is defined as ratio of summary length to original document length, reaches the predefined value, terminate the operation; other- wise, continue the process adding of the sentences to the summary.

IV. DISCUSSION OF RESULTS

A. *Pre-processing*

Each document in the first group has been transformed to text format or you import the words from the document folders where the intended words is kept. Thereafter, it is translated into the target language. One of the major problems in text mining is that a document can contain a very large number of words. If each of these words is represented as a vector coordinate, the number of dimensions would be too high for the text mining algorithm. Hence, it is crucial to apply preprocessing methods that greatly reduce the number of dimensions (words) to be given to the text mining algorithm. Our system can apply several preprocessing methods to the original documents, like stemming and stop-words removal.

B. *Stemming*

(Removing word suffixes such as `ing', `ion', `s') consists of converting each word to its stem, that is natural form with respect to tag-of-speech and verbal/plural inflections .In essence, to get the stem of a word it is necessary to eliminate its suffixes representing tag-of-speech and/or verbal/plural inflections. We have used Porter's algorithm, originally developed for the English language.

C. *Stop Words*

(Insignificant words like `can', `in', `this', `from', `then', `or', `the', `by') are words that occur very frequently in a document. Since they are so common in many documents, they carry very little information about the contents of a document in which they appear.

Table III, IV, V respectively shows the statistics of the documents and the summarization results. We employ the standard measures to evaluate the performance of summarization, i.e. precision, recall and F-measure. We assume that a human would be able to identify the most important sentences in a document most effectively. If the set of sentences selected by an automatic extraction method has a high overlap with the human-generated extract, the automatic method should be regarded as effective. Assume that Sman is the manual summary and Sauto is the automatically-generated summary, the measurements are defined as;

$$P = \frac{Sman \cap Sauto}{Sauto}$$

(6.1)

$$R = \frac{Sman \cap Sauto}{Sauto}$$

(6.2)

$$F = \frac{2PR}{P + R}$$

Where;

P; is the precision, R; is the Recall, F; is the F measure, Sman; is the manual summary, Sauto; is the automatically generated summary.

F measure: is a measure of test's accuracy and is defined as the weighted harmonic mean of the Precision and Recall of the test.

Precision: also called Positive Predictive value, is the fraction of relevant instances among the retrieved instances.

Recall: also known as sensitivity is the fraction of relevant instances that have been retrieved over the total amount of relevant instances.

We used the F measure, to test the performance evaluation of the model classifier accuracy.

D. Automatic Summarization

It is the process of shortening a text document with software, in order to create a summary with the major points of the original document. Models that can make a logical and consistent summary take into account variables such as length, writing style and the arrangement of words and phrases to create well-formed sentences in a language. Automatic Summarization is part of machine learning and data mining, the main idea of Summarization is to find a subset of data which contains the information of the entire set.

TABLE III
STATISTICS OF THE DOCUMENTS AND THE SUMMARIZATION RESULTS

Document	**Number of words**						
	Documents	**Summaries created by summarizers**					
		Ms word		**Human**		**New model**	
		CR =		**CR =**		**CR =**	
		15%	**30%**	**15%**	**30%**	**15%**	**30%**
Doc1	3	5	7	9	4	7	6
Doc2	6	8	6	4	7	3	6
Doc3	4	5	9	3	6	6	2
Doc4	5	5	7	5	8	4	4
Avg	4.5	5.8	7.25	5.3	6.3	5	4

TABLE IIIV
ASSESSMENT CONTROL BETWEEN IGBO LANGUAGE AND THE TEXT2SPEECH OF CR=30%

Document	Overlap with the human-generated extracts								
	Documents	Summaries created by summarizers							
		Ms word			Human			New model	
		P	R	F	P	R	F	P	R	F
Doc1	3	0.10	0.10	0.10	0.12	0.12	0.12	0.28	0.20	0.30
Doc2	6	0.50	0.52	0.51	0.54	0.52	0.53	0.40	0.42	0.41
Doc3	4	0.13	0.15	0.17	0.14	0.16	0.15	0.22	0.26	0.24
Doc4	5	0.60	0.62	0.61	0.66	0.66	0.66	0.30	0.36	0.33
Avg	4.5	0.33	0.35	0.35	0.37	0.38	0.37	0.30	0.31	0.32

TABLE V
ASSESSMENT CONTROL BETWEEN IGBO LANGUAGE AND THE TEXT2SPEECH

Document	CR=15%			CR =30%		
Doc1	6.4 (+)	6.8(+)	6.6(+)	26.4 (+)	26.4 (+)	27.4 (+)
Doc2	10.4 (+)	16.0(+)	13.2 (+)	24.4 (+)	22.4 (+)	26.4 (+)
Doc3	16.4 (+)	18.4 (+)	17.4 (+)	13.8 (+)	13.4 (+)	13.6 (+)
Doc4	22.4 (+)	16.4 (+)	19.4 (+)	22.4 (+)	22.4 (+)	22.4 (+)
Doc5	36.4 (+)	36.4 (+)	36.4 (+)	16.4 (+)	16.4 (+)	16.4 (+)

V. CONCLUSION

The amount of text that is generated every day is increasing rapidly. This tremendous volume of mostly unstructured text cannot be simply processed and perceived by computers. Therefore, efficient and effective techniques and algorithms are required to discover useful patterns. Text mining is the task of extracting meaningful information from text, which has gained significant attentions in recent years. In this research, we described several of the most fundamental text mining tasks and techniques including text pre-processing and classification. In addition, we briefly explained the applications of text mining in some areas like, the retail and telecommunications industries, biomedical and health sectors etc.

Text classification is the process of classifying documents into predefined categories based on their content. It is the automated assignment of natural language texts to predefined categories. Text classification is the primary requirement of text retrieval systems, which retrieve text in response to a user query, and text understanding systems which transform text in some way such as producing summaries, answering questions or extracting data. Existing supervised learning algorithms automatically classify text and needs sufficient documents to learn accurately. This dissertation presents K-Nearest Neighbour (K-NN) Algorithm and Euclidean Distance Classifier for text classification using data mining that requires fewer documents for training. Instead of using word relation, we employed Association Rules from these words to derive feature set from pre-classified textual documents. As data mining can only uncover patterns actually present in the data, the target data set must be large enough to contain these patterns while remaining concise to be mined within an acceptable time limit.

A common source for data is a data mart or data warehouse. Pre-processing is essential to analyze the multivariate data sets before data mining. The target set is then cleaned to remove the observations containing noise and those with missing data.

ACKNOWLEDGEMENTS

I specially thank my supervisors Dr. (Mrs.) L.N. Onyejegbu and Dr. F.E. Onuodu for their supervision, inputs and interests in the completion of this research. I thank all the PG lecturers like Prof. P.O. Asagba, Dr. Chidiebere Ugwu, Dr. B. O. Eke, Dr. F. E. Onuodu for their instructions and contributions in my academic work. My profound gratitude goes to the PG Coordinator, Dr. F.E. Onuodu for his heartening and motivation.

I am most grateful to my dear wife, Mrs. Justina Uche-Agi for her fervent love, financial assistance and intrepidness. To my kids, Maxwell O. Agi, Wisdom O. Agi and Michelle H. Agi, their disturbances were my source of happiness, doggedness and strength.

I am indebted to my mother, Mrs. Augusta Agi and my mother in-law, Mrs. Violet Nwuzor for their love and courage.

To my brothers, Chimele, Chinedu, Chinwe and Oke, they are brothers in need and indeed.

Finally, to all my colleagues, it is not only a pleasure learning together but it was a knowledge and ideas well shared.

REFERENCES

[1] K. S. Dileep, and S. Vishnu, "Data Security and Privacy in Data Mining: Research Issues & Preparation," *International Journal of Computer Trends and Technology*, vol. 4, no. 2, pp. 194 -200, 2013.

[2] T. R. Mahesh, M. B. Suresh, and M. Vinayababu, Text Mining: Advancements, Challenges and Future Directions, *International Journal of Reviews in Computing*, vol. 3, no. 2, 2010.

[3] N. P. Falguni, and R.S. Neha, Text Mining: A Brief Surve, *International Journal of Advanced Computer Research*, vol. 2, no. 4, 2012.

[4] M. Davide, "*Graphical Models for Text Mining: Knowledge Extraction and Performance Estimation*," Ph.D Thesis, 2010.

[5] S. B. Vishwadeepak, and S.P. Tripathi, "Text Mining Approaches to Extract Interesting Association Rules from Text Documents," *International Journal of Computer Science Issues*, vol. 9, no. 3, 2012.

[6] W. S. Jeffrey, Data Mining: An Overview, Analyst in Information Science and Technology Policy Resources, Science, and Industry Division, 2004.

[7] F. Ronen, and S. James, "The Text Mining Handbook: Advanced Approaches to Analyzing Unstructured Data," Cambridge University Press. ISBN 0-521-83657, 2007.

[8] A.B. Deval, and R.V. Kulkarni, (2012). "Applications of Data Mining Techniques in Life Insurance," *International Journal of Data Mining & Knowledge Management Process*, vol. 2, no. 4, pp. 87-98, 2012.

[9] M. R. Chowdhury, A. S. Ferdous, N. Parvez, and S. M. Kamruzzaman, "Text Classification Using the Concept of Association Rule of Data Mining," *Proc. International Conference on Information Technology*, Kathmandu, Nepal, 2010.

[10] M. Harmain, E. Hazem, and L. Abdulrahamn, "*Arabic Text Mining*, College of Information Technology," United Arab Emirates University,United Arab Emirates, 2004.

[11] A. Mofleh, "*Arabic Text Categorization Using Classification Rule Mining,"* Department of Computer Science, Al Albayt University, vol. 6, no. 3, 2012.

[12] A.K. Sabarin, "Application of Pattern Mining Algorithm for Text mining," *International Journal of Research in Engineering & Advanced Technology*, vol. 1, no. 1, 2013.

[13] V, Bhujade, and N.J. Janwe, "Knowledge Discovery in Text Mining Technique Using Association Rules Extraction," Pp. 498-502, 2011.

[14] A. Jafar, A. Omar, H. Wael, K. Nidhal, E. Taha and A. Ali, "Vector Space Models to Classify Arabic Text," *International Journal of Computer Trends and Technology*, vol. 7, no. 4, pp. 2231-2803, 2014.

[15] A. Saleh, "Automated Arabic Text Categorization using SVM and NB," *International Arab Journal of e-Technology*, vol. 2, no. 2, 2011.

[16] S.M. Kavitha, and P. Hemalatha, "Survey on Text Classification Based on Similarity," *International Journal of Innovative Research in Computer and Communication Engineering*, Certified Organization, vol. 3, no. 3, 2015.

