Paper No: SC-20 **Smart Computing**

Keyword extraction from Tweets using NLP tools for collecting relevant news

Thiruni D. Jayasiriwardene*
Faculty of Information Technology
University of Moratuwa, Sri Lanka
thirudilushi@gmail.com

Gamage Upeksha Ganegoda
Department of Interdisciplinary Studies
Faculty of Information Technology
University of Moratuwa, Sri Lanka
upekshag@uom.lk

***Abstract*: Keywords play a major role in representing the gist of a document. Therefore, a lot of Natural Language processing tools have been implemented to identify keywords in both structured and unstructured texts. Text that appears in social media platforms such as twitter is mostly unstructured because of the character limitation. Consequently, a lot of short terms and symbols such as emoticons and URLs are included in tweets. Keyword extraction from grammatically ambiguous text is not easy compared to structured text since it is hard to rely on the linguistic features in unstructured texts. But when it comes to news on twitter, it may contain somewhat structured text than informal text does but it depends on the tweeter, the person who posts the tweet. In this paper, a methodology is proposed to extract keywords from a given tweet to retrieve relevant news that has been posted on twitter, for fake news detection. The intention of extracting keywords is to find more related news efficiently and effectively. For this approach, a corpus that contains tweet texts from different domains is built in order to make this approach more generic instead of making it a domain-specific approach. In fact, the Stanford Core NLP tool kit, Wordnet linguistic database and statistical method are used for extracting keywords from a tweet. For the system evaluation, the Turing test which has human intervention is used. The system was able to acquire an accuracy of 67.6% according to the evaluation conducted.**

Keywords: *Named Entity Recognition (NER), Natural Language Processing (NLP), Part of speech tagging, Stanford Core NLP, Wordnet corpus.*

I. INTRODUCTION

A keyword is a word that succinctly and accurately describes the subject or the aspect that identifies the subject mentioned in a document [1], plays a major role as an indicator of important textual information which spread among the people as soft documents or hard documents. In scientific communication, words are used to communicate information unambiguously while highlighting the words as keywords that are focused on the communicated topic. On the other hand, keywords are essential for the reader to get a quick idea of the information contained as text, in the meantime, it is important when searching for the associated information for a particular topic. Therefore, Natural Language Processing (NLP) tools have been built in order to extract keywords from different types of documents or sources. Scientific documentation contains structured text which enables natural language processing tools to work with linguistic and syntactical features of the language. Unstructured texts are commonly used in social media such as Twitter because of its character limitation. A tweet contains a maximum of 280 characters. Therefore, people tend to use words in an unstructured way. It doesn't contain linguistic features and sometimes provides ambiguity for the reader.

Several approaches such as the statistical approach, Rule-Based Approach, Machine Learning approach and Domain-specific approach [1, 2] exist to extract keywords from textual documents. Figure 1 depicts the classification of automatic keyword extraction.

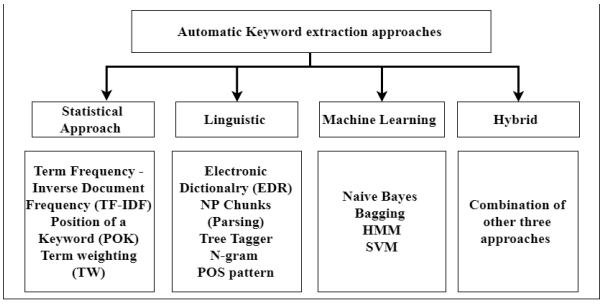

Fig. 1. Classification of automatic keyword extraction [2]

Method of extracting keyword differs based on the objective of the extraction; it could be carried out for text summarization, text categorization, information retrieval and question answering [2]. research had been carried out to extract keywords from unstructured text. It is not similar to extracting keywords from structured text such as abstracts and news articles due to lack of linguistic features, for instance, positioning of the word in a sentence and the number of occurrences of the word in the particular sentence. Unstructured texts are commonly used in social media because of the character limitation for a tweet post. Furthermore, it contains Abbreviations, links for websites, emoticons and images. Tweets are especially grammatically ambiguous so the accuracy of Part of Speech (POS) tagging may be less. Therefore, methods that are used to extract keywords from the structured text will not be accurate. Some of the statistical methods won't work on tweets as one word appears only once in many cases. But it can be used to filter 'most appeared words' in the corpus.

This paper proposes a methodology to extract keywords from a tweet to retrieve relevant news for a particular incident. This keyword extraction intends to collect associate news as much as possible. This paper is organized as follows. Related work for keyword extraction is overviewed in section II. The methodology is elaborated in section III and methods of evaluation are discussed in section IV.

II. Related Work

Research has been conducted on extracting keywords from grammatically ambiguous texts such as tweets using NLP tools. This approach is a combination of machine learning and rule-based approaches [3]. A corpus that contains domain-specific words that needed to be rejected had been built to remove the noises from the tweet texts. Machine learning-based Stanford Core NLP [4] POS tagging is used since it has the highest token accuracy. Penn tree notation is used to tag the tokens and Tregex [5] notation is used to structure the tree. Matching of rejected words was done based on the Levenshtein Distance; [6] when the distance between the word and the rejected word equals zero then the particular word is rejected and eliminated from the list of words. This process enables us to filter the final set of keywords consisting of Noun Phrases (NP) and Verb Phrases (VP) from the given tweet. These keywords are further filtered inside the second parser to find the other keywords which are not tagged as either NP or VP. This system is developed using java as a standalone desktop application.

An approach for keyword extraction was proposed to address the problems such as frequent usage of lexical variants and the high variance of cardinality presented in each tweet. In this approach, they had provided a keyword annotated data set of 1827 records and a system using unsupervised fashion. Brown clustering and continuous word vector approach have been integrated for unsupervised feature extraction in tweets [6]. Brown clustering along with hidden Markov model assumptions have been used to cluster lexical variants while generating a hierarchical structure of clusters. Structured skip-ngram goal function is proposed to use for extracting words as a vector [7]. In this approach, a method is included to learn and predict the number of keywords to be generated depending on the count of functional and non-functional words in the given tweet [3].

An attempt had been taken to extract keywords and generate headlines from the unstructured text using the hidden Markov model and other NLP tools such as POS tagging and clustering. In this approach, the first text preprocessing part had been done using the following steps. First, the input text is tagged using the Part of Speech tagging which implemented TreeTagger [8] a decision tree based on the probability. A binary decision tree is implemented recursively by using an improved version of ID-3 algorithm. A dataset of trigram is used and the probability was decided by traversing the tree. This tagging approach has achieved a higher level of accuracy than trigram tagger on PennTreeBank data. Then the tagged output was normalized and stemming was done, stop words were removed and merging similar content was done using WordNet. Then the keywords were extracted. The number of words to be extracted was reduced by merging the words with similar content [9].

A novel method has been invented to extract topical keywords from twitter. In this approach, topics are learnt from tweets to extract and organize key phrases using three standard steps known as Keyword ranking, candidate key phrase generation and key phrase ranking [10]. A modified topical page ranking method which introduces topic sensitive score propagation [11] was used to boost the performance. Candidate key phrase generation method combined with principle probabilistic phrase ranking method is proposed for keyword ranking. For topic discovery using twitter, a modified author-topic model named Twitter-LDA was used under the assumption that a single topic exists for the entire tweet [10]. The original topical page ranking method was modified by including context-free co-occurrence edge weight to rank words that have co-occurrence. A different probabilistic scoring function based on two hypotheses "a good key phrase should be closely related to the given topic" and “a good key phrase should be interesting and can attract users” was proposed and implemented. When extracting key phrases, length preference was incorporated to avoid meaningless key phrases. According to the evaluation of results, proposed context-sensitive page ranking performed better than the standard method as well as the probabilistic ranking method boosting the performance of keyphrase extraction [10].

TwittDict is a novel approach to extract semantically associated words with a target word from a corpus of tweets [12]. This approach addresses the problem of extracting semantically related key phrases from microblogs while solving the matter of language creativity and noise such as nonstandard verbs or symbolic expression [12]. This model first recognizes the topics that have been mentioned in the corpus and checks for the association of those topics for the given words. Then starts mapping to find the semantically associated words. It assumes that a word in a tweet cannot represent it semantic but a set of tweets that contains the same word can give the meaning of the word. Latent Dirichlet Allocation (LDA) [13] is a method of text mining used to represent a document as a set of topics and to identify the topics.

Some of the commonly used approaches to extract keywords from text processing along with NLP tools have been mentioned below. These approaches have been used to extract keywords from both structured and unstructured texts.

A simple statistical approach is rough and has a tendency to work without a training set. It focuses on the statistics obtained from non-linguistic features of the document where these insights can be used to generate keywords. N-gram statistical data can be used to filter the keywords inside the text. It is known as TF-IDF – Term Frequency – Inverse Document Frequency [1, 9], The main criteria is the frequency of occurrence. This statistical information can then be used to find the support and confidence of the word. Later, the keywords are inferred using the Apriori technique which is generally used for frequent itemset mining and association rule learning [2, 14, 15].

Linguistic features of the words are focused on keywords extraction and detection in text documents using the Linguistic approach. It consists of lexical analysis, syntactical analysis and discourse analysis. Electronic dictionary, tree tagger, Wordnet, n-grams, POS patterns are the main resources that are incorporated for lexical analysis [16]. Noun phrases, noun chunks are used as resources for syntactical analysis. This approach is more accurate and computationally intensive. But the limitation of this approach is that it requires domain knowledge [1, 15].

Keyword extraction can be seen in the angle of a learning problem. Therefore, a machine learning approach can be used along with manually annotated training data and training models. Training models such as support vector machine

(SVM) naïve Bayes, bagging and Hidden Markov model (HMM) are commonly used [2,14]. Keyword Generation Algorithm (KEA) is one of the most popular and accurate algorithms which is built upon this approach. The KEA algorithm first converts words into nodes, whenever two words are laid on the same sentence then a vector graph is created by connecting the two nodes. A number of edges is converted into scores and are clustered accordingly. after that cluster heads are treated as keywords and they are categorized into two categories, keyword or not a keyword using Bayes classification [9]. There are two methods that can be used; supervised learning and unsupervised learning [14].

A hybrid approach is a combination of supervised and unsupervised learning or heuristics such as position, length, layout feature of the words, etc. It is designed to extract the best feature from the above-mentioned approaches [2, 14].

A considerable number of algorithms have been described and used for keyword extraction in documents, C4.5 algorithm which is an extension of ID3 algorithm is used in the keyword extraction process. It includes decision tree statistical classification which is more suitable for balanced class attributes [17]. KEA is a keyword generation algorithm which builds upon Naïve Bayer's learning approach and statistical approach called TF-IDF. This algorithm is based on two lexical features named Term Frequency and the position of the keyword. If the occurrence of the word is high in a document there is a high probability for that word to be a keyword. There should be a large data set in order to train this model well [14, 18]. KEA++ is an extension of KEA keyword generating algorithm and it uses three linguistic features to select the keyword. It uses a thesaurus to link the synonyms of the words and filter the words with a high node degree. KEA++ uses structures-controlled vocabulary. The advantage of KEA++ is its use of controlled vocabulary which eliminates the occurrence of meaningless and incorrect word extractions, its performance is dependent on the control's vocabulary [19].

A method is proposed to extract keywords from tweets by using brown clustering which is clustered by brown corpus and continuous word vectors. The clustering algorithm attempts to find clusters with the maximum likelihood. A tweet can have any length from 1 to 280 characters. Therefore, the number of keywords to be extracted must be proportional to the word count in the tweet. This fact is considered in the approach presented in this paper. In the evaluation, precision and recall hold a higher value for the algorithm MAUI which consists of both Browns clustering and Word Vectors than other existing methods [14]. Another method to extract keywords has been proposed by using Backpropagation Neural Network [20]. Corpora that had been used to train the model consist of journal articles. Each and every word is encoded with a set of features such as term frequency. Inverse document frequency (IDF) is not used in this approach since it requires the analysis of the whole system. In order to finetune the system, Backpropagation Neural Network is used through C4.5 algorithm. According to the evaluation results obtained from this approach, BNN is found to predict keywords with 90.11% precision, 59.50% recall and 0.717 F-Measure [21]. A model is proposed using Recurrent Neural Network (RNN) containing two hidden layers to extract keywords using tweets. The first layer captures the information of keyword while the second layer extracts the key phrase by using information. Evaluation of results proves that this method performs better than the traditional state-of-the-art. [22].

III. METHODOLOGY

A. Dataset

A twitter data set that contains tweet text was obtained, in order to extract candidate keywords and for evaluation purposes. This data set is a combination of tweets from different domains such as health, sports, politics and education. More than 100,000 records from various news categories are available. The intention was to make the system work accurately in any domain. In other words, to make it a generic approach. Data were collected using the Twitter streaming API and directly downloaded from Kaggle.

B. Pre-processing

Fake news detection does not focus on a specific domain therefore, the corpus should contain tweets from different domains in order to find the candidate keywords from a tweet. There are existing corpora that contain twitter data but according to the experiments carried out, they do not contain enough tweets to find the candidate phrase. Therefore, it was decided to create a corpus by combining domain-specific corpora and some manually collected tweets using Twitter streaming API.

Data preprocessing plays a major role when creating a corpus because the proposed approach is implemented using unstructured text that is used on twitter. The process of data pre-processing carried out is depicted in figure 2.

A tweet consists of URLs, punctuations and emoticons which are needed to be removed in the phase of preprocessing. Furthermore, there are words such as pronouns, articles and prepositions that have been frequently used in natural language which must be filtered out at the preprocessing stage because the statistical approach used for keyword extraction takes the most frequent word in the text, so the impact of stop words must be removed. The list of stop words that is used for pre-processing is created by analyzing a previously used stop words list in NLP and some of the abbreviations that are commonly used in social media platforms such as LOL, GM, etc. Usually, those words are not used in tweets that contain news. But this step is included to avoid uncertainty, as the language of social media platforms is evolving and getting more informal according to the user base and the tweeter.

Then Lemmatization is carried out to convert a word to the base form. Otherwise, it will be difficult to identify the same word in a different format [9]. Stemming was not used because it leads to spelling errors and ambiguous words since it removes the last few letters of a word. Then strings are tokenized using NLTK library in order to continue with spelling corrections. There could be intentional or unintentional spelling errors in the tweet content. Python package pyspellchecker provides features to find misspelled words and suggest possible corrections. The word with the highest probability will be selected as the corrected version of misspelled words. After following the preprocessing steps, the corpus will be created.

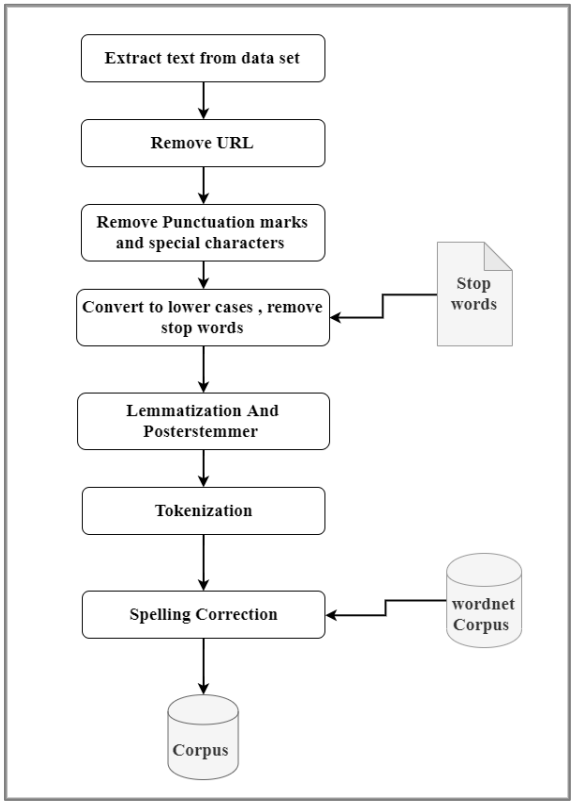

Fig. 2. Data pre-processing

C. Extracting keywords

Once the claimed tweet's text is given to the system it is preprocessed to extract the keywords. The preprocessing steps include removing URLs, special characters, punctuation and stop words. Then, lemmatization is done and the string is tokenized into words. This text might include words that are misspelled, thus spelling correction also has to be done. At the end of this preprocessing, the output will be a set of words containing the keywords.

After pre-processing the claimed tweet text, POS is needed to be done with the intention of using Named Entity Ranking (NER). It is required to apply POS tagging for the NER tagging. Even though there are libraries for POS tagging, every library will not give a high accuracy since it depends on the data set it was trained on. Since this research is on unstructured text, it is better to find a library which is trained on a twitter data [14] set. Stanford CoreNLP toolkit [23] is an extensible pipeline that facilitates natural language analysis and could be used for POS tagging as well as for NER tagging.

Figure 3 depicts the preprocessing steps for the text of the claimed tweet. After Applying the POS tagging, the NER must be applied. Stanford Core NLP toolkit facilitates labeling words into seven classes. Location, person, organization, money and percentage, date and time [24] are the classes of the model. When extracting keywords with the intention of collecting associate news to the given tweet text, location, person and organization play a major role in retrieving associate news that has been circulated in the social media platform (twitter). In fact, these seven classes are the most important keywords in a tweet.

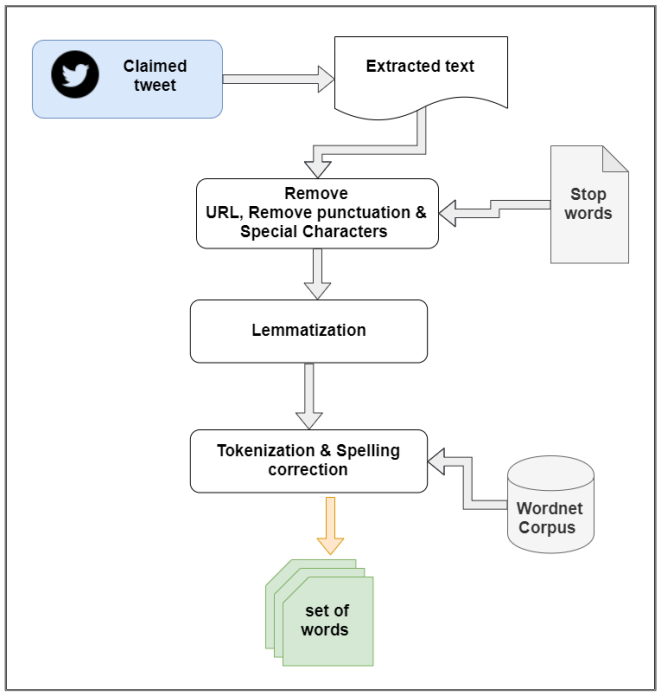

Fig. 3. Preprocessing for the text

Even though these essential keywords can be extracted from the given tweet using Stanford core NLP framework, some of the important keywords can be missed due to the unrecognition of the word. In addition, some of the new locations, persons and organizations will not be captured. As a solution for this issue, a methodology can be proposed to use TF-IDF statistical method along with NER to find the candidate keywords. TF-IDF is based on the frequency of occurrence of a word in the corpus. Since the corpus contains more than 100,000 tweets related to news from different domains, this approach will be a suitable way to reach the goal. The following figure depicts the flow of candidate keyword extraction from a given text.

After extracting the candidate keywords from the given tweet text, it is essential to find synonyms for the words that are not identified by NER tagging to expand the search for associate news posted on twitter. Different users will post the same news in different manners. Wordnet lexical database which has been used by major search engines can be used to retrieve similar words and the value for the similarity of two words is done by using word2vec using Gensim [25]. The words having more similarity can be taken as candidate keywords. To find words with more similarity a threshold value can be defined. to define a threshold value a statistical experiment can be conducted.

To generate key phrases using the above extracted keywords, n-gram can be used and it will give a large number of word combinations if a limitation for the number of keywords to be extracted is not given. Therefore, a threshold value must be decided by testing, to reach the expected output. this will make generating key phrases more effective and easier to handle. The keyword extraction from the given tweet is carried out to extract relevant news articles that have been posted on twitter for fake news detection; to make the relevant news collection more accurate and effective, in other words, to collect news which is truly related to the claimed news.

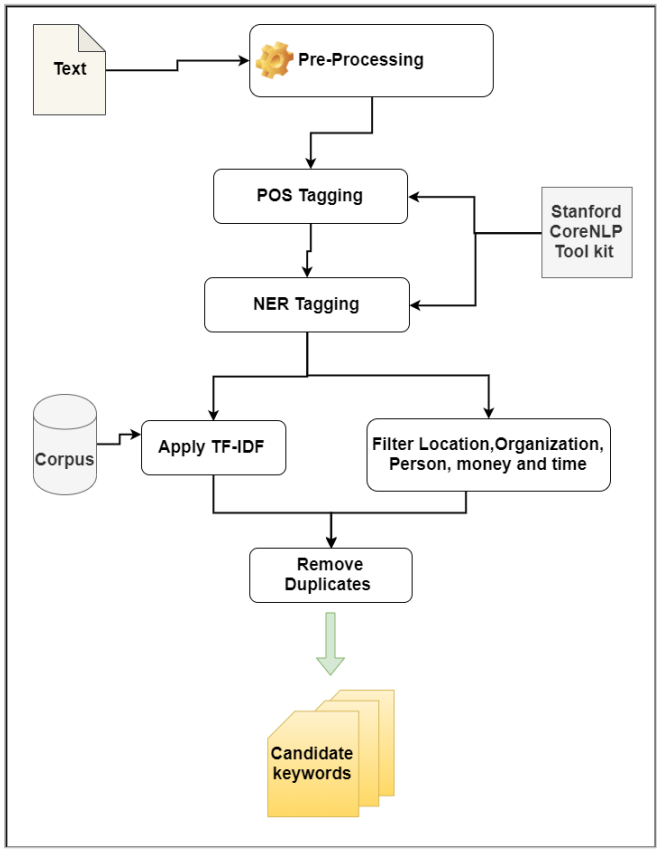

Fig. 4. The flow of candidate keyword extraction

After extracting key phrases, they can be used as parameters for the Twitter streaming API. Key phrases can be assigned to the track variable provided in the Twitter API streaming method. Extracted keywords are saved into an array and assigned to the track variable in the twitter API. Then it will retrieve all the tweets which contain those key phrases. In fact, it will receive tweets that contain at least one of the key phrases or there could be a combination of keywords in those tweets. At least ten thousand records will be collected from twitter API if it is possible. This will not guarantee that each and every tweet received is relevant to the claimed tweet. Therefore, further processing must be conducted to filter the news articles to find those that are more relevant to the claimed tweet.

This process focuses on the most important factors related to the claimed tweet such as person names, locations, organizations, countries and specifically mentioned dates or the received date of the tweet. Before starting the filtering process, it is essential to remove duplicate records. Therefore, tweets that contain the same important factors mentioned above, contain most of the extracted keywords in the text and has a tweet date close to the date on which the claim is made, are filtered from the extracted set of tweets.

IV. Evaluation

After selecting the proper threshold value to determine and choose the words with more similarity, an evaluation was conducted by comparing the threshold value and the accuracy of the output. Table I depicts the selected values and accuracies.

TABLE I. Comparing threshold values with accuracy

Threshold value	Accuracy of the final output	Assumed Reason
0.5	51.4%	Selecting more synonyms which does not have much similarity
0.69	67.6%	Moderate and provide the most relevant synonyms
0.75	49.2%	Selecting less synonyms even though the similarity is high

According to the statistical experiment carried out, the similarity threshold value was set to 0.69.

Turing test [3] will be used to evaluate the extracted keywords from the given tweet text because there is no other evaluation method that is better than the evaluation carried out by human intervention.

As mentioned above, the Turing test is carried out with human intervention. The intervention of humans to the test is important and the output of the evaluation depends upon the humans who are going to be used for the testing. The humans who will participate in the Turing test will be experts in the language. Since the system is about news posted on twitter and the system is a generic model that can be applied to any domain, people who write English articles can be used. Social media platforms are commonly used by the younger generation of the country. Therefore, undergraduates involved in the area of language will be suitable for the Turing test.

Table II depicts the participants of the Turing test; human keyword generators used to build the test dataset.

TABLE II. Participants of the Turing test

Type of Participants	Justification	Minimum Qualification
English article writers	They have a vast knowledge on language manipulation for article writing on any area.	Active journalists
Graduates	Highly experienced professionals on using English in academics and professional fields	Completed a bachelor's degree (related to language)
Undergraduates	Knowledge and experience on using English language for academics	Reading for a bachelor's degree (related to language)
General public	Uses of Social media frequently	Daily twitter users

A test data set was created by manually annotating 500 tweets with keywords and the output for each tweet was taken through the proposed approach. Then a comparison was carried out and the accuracy of the proposed approach was determined.

When calculating the accuracy of the system there are two main factors to be considered.

1. Number of keywords
2. Extracted keywords including synonyms

Table III depicts the types of records used with the assumptions.

TABLE III. ASSUMPTION AND RECORD TYPES

Record type	Number of keywords	Number of matches	Status
1	Equal	All are matched	Identical
2	Equal	Less than 80%	Negative
3	Greater than the human generated keyword set	Greater than 80%	Positive
4	Greater than the human generated keyword set	Less than 80%	Negative
5	Less than the human generated keyword set	-	Negative

The accuracy was calculated using the following equation (1).

$$\text{Successful percentage} = \frac{X + Y}{N} * 100\,\% \quad (1)$$

X = Identical records Y = Positive records

N = total number of records (tweet set)

According to the data set and the evaluation conducted there were 107 identical records and 231 positive records and the total percentage of accuracy was 67.6%. Since this keyword extraction consists of words that have similarities, the accuracy is dependent on the intelligence, knowledge and vocabulary of the people who were selected for the Turing test.

Moreover, this evaluation can be conducted separately for each type of participants, then the accuracy can be gained according to each category separately.

Evaluation by comparing existing methods to extract keywords from unstructured text such as using Stanford core NLP [3], automatic keyword extraction using brown clustering [14], Rapid Automatic Keyword extraction algorithm (RAKE) and existing methods to extract keywords from structured text such as Keyword Extraction Algorithm KEA cannot be conducted because those models do not extract keywords with the suitable synonyms. If those approaches are evaluated against the proposed approach a low accuracy will be given

The tweets that are collected using the extracted keywords and twitter API are filtered to extract the most relevant set of tweets for the claimed tweet.

V. CONCLUSION

It takes some time for a person to read the whole text and get an idea about its content, but, keywords have made this task easy. A keyword is a word that represents the whole idea of the text. Therefore, attention has been paid on keyword extraction methods from text using NLP tools. It is easy to extract keywords from structured documents by considering their linguistic features. But for unstructured text such as tweets, it has become difficult because linguistic features cannot be found due to character limitations. In fact, the language pattern is different from those informal documents.

This paper proposes a methodology to extract keywords from a given tweet text for the purpose of retrieving relevant news that has been posted on twitter to collect data for fake news detection. The proposed method uses Stanford core NLP, POS tagging, NER as well as TF-IDF statistical method for keyword extraction. In Addition, Wordnet lexical database has been used to find synonyms and Ginsim can be used along with word2vector to find synonyms for the words and the amount of similarity of words. Then the bi-gram technique is used to generate key phrases to increase the accuracy and efficiency for retrieving relevant news. Extracted keywords are used to gather the most relevant news tweets for the claimed tweet. In fact, the set of tweets retrieved were filtered and duplicates were removed to get a clean set of tweets to help detect fake news. The Evaluation can be done using the Turing test and more attention should be paid to the standards of the participants of the test.

A dedicated corpus has been implemented with more than 100000 tweet news from different domains such as sports, politics, etc. Stanford core NLP toolkit was selected because it's also built upon using a set of tweets. In the proposed method, essential keywords such as the name of a person, location, organization, date and time will not be missed in the array of candidate keywords even though they are not captured by the TF-IDF method. Most importantly, this method is a generic method which does not depend on a specific domain.

As future work, the method which is used to find synonyms and similarity of two words can be modified to give a more precise output which will lead to an increment of performance.

ACKNOWLEDGMENT

The author of this paper would like to express gratitude for the supervisor for her immense support and guidance to make this paper a success.

REFERENCES

[1] S. Siddiqi and A. Sharan, "Keyword and Keyphrase Extraction Techniques: A Literature Review", *International Journal of Computer Applications*, vol. 109, no. 2, pp. 18-23, 2015. Available: https://www.researchgate.net/publication/272372039_Keyword_and_Keyphrase_Extraction_Techniques_A_Literature_Review. [Accessed 3 June 2019].

[2] S. K. Bharti and K. S. Babu "Automatic Keyword Extraction for Text Summarization: A Survey", *CoRR*, vol. 170403242, 2017. Available: http://arxiv.org/abs/1704.03242. [Accessed 5 June 2019]

[3] T. Weerasooriya, N. Perera and S. Liyanage, "A method to extract essential keywords from a tweet using NLP tools", *2016 Sixteenth International Conference on Advances in ICT for Emerging Regions*, pp. 29-34, 2016. [Accessed 30 May 2019].

[4] C. Manning, M. Surdeanu, J. Bauer, J. Finkel, S. Bethard and D. McClosky, "The Stanford CoreNLP Natural Language Processing Toolkit", *Proc. 52nd Annu. Meet. Assoc. Comput. Linguist. Syst. Demonstr.*, pp. 55–60, 2019. Available: https://nlp.stanford.edu/pubs/StanfordCoreNlp2014.pdf. [Accessed 7 October 2019].

[5] R. Levy and G. Andrew, "Tregex and Tsurgeon: tools for querying and manipulating tree data structures", *5th Int. Conf. Lang. Resour. Eval. (LREC 2006)*, pp. 2231–2234, 2006. Available: http://www.mit.edu/~rplevy/papers/levy-andrew-2006.pdf. [Accessed 7 December 2019].

[6] [*Levenshtein distance*, [Online] Available: https://rosettacode.org/wiki/Levenshtein_distance. [Accessed: 07- Oct- 2019].

[7] A. Joulin, E. Grave, P. Bojanowski and T. Mikolov, "Bag of Tricks for Efficient Text Classification", *arXiv:1607.01759v3 [cs.CL]*, 2016. Available: https://arxiv.org/pdf/1607.01759.pdf. [Accessed 7 December 2019].

[8] [Online]. Available: http://www.ims.uni-stuttgart.de/projekte/corplex/TreeTagger/DecisionTreeTag%ger.html. [Accessed: 07- Oct- 2019].

[9] Amit Kumar Mondal and Dipak Kumar Maji "Improved Algorithms For Keyword Extraction and Headline Generation From Unstructured Text", p. 14. Available: https://pdfs.semanticscholar.org/d177/faa6f8c92c19e9b54ab1eaa94b510482425a.pdf. [Accessed 5 June 2019]

[10] Zhao, Wayne Xin and Jiang, Jing and He, Jing and Song, Yang and Achananuparp, Palakorn and Lim, Ee-Peng and Li, Xiaoming, "Topical Keyphrase Extraction from Twitter", *Association for Computational Linguistics*, pp. 379-388, 2011. Available: {http://dl.acm.org/citation.cfm?id=2002472.2002521. [Accessed 5 June 2019].

[11] Y. Liu, K. Young, Y. Curtis, B. Aragona and Z. Wang, "Social Bonding Decreases the Rewarding Properties of Amphetamine through a Dopamine D1 Receptor-Mediated Mechanism", *J Neurosci. 2011*, pp. 7960–7966., 2011. Available: https://www.ncbi.nlm.nih.gov/pmc/articles/PMC3114880/pdf/zns7960.pdf. [Accessed 7 October 2019].

[12] Ian H. Witten, Gordon W. Paynter, Eibe Frank, Carl Gutwin and Craig G. Nevill-Manning" KEA: Practical Automatic Keyphrase Extraction", vol. 23, 2015. Available: https://www.cs.waikato.ac.nz/~ml/publications/2005/chap_Witten-et-al_Windows.pdf. [Accessed 28 August 2019].

[13] D. Blei, A. Ng and M. Jordan, "Latent Dirichlet Allocation", *Journal of Machine Learning Research 3 (2003)*, pp. 993-1022, 2003. Available: http://www.jmlr.org/papers/volume3/blei03a/blei03a.pdf. [Accessed 8 October 2019].

[14] Marujo, Luís & Ling, Wang & Trancoso, Isabel & Dyer, Chris & Black, Alan & Gershman, Anatole & Martins de Matos, David & Neto, João & Carbonell, Jaime. (2015). Automatic Keyword Extraction on Twitter. 2. 637-643. 10.3115/v1/P15-2105. Available: https://www.researchgate.net/publication/283816235_Automatic_Keyword_Extraction_on_Twitter. [Accessed 5 June 2019]

[15] Z. Zhu, M. Li, L. Chen, Z. Yang and S. Chen, "Combination of Unsupervised Keyphrase Extraction Algorithms", *2013 International Conference on Asian Language Processing, Urumqi, 2013, pp. 33-36.*, 2019. Available: http://ieeexplore.ieee.org/stamp/stamp.jsp?tp=&arnumber=6645997&isnumber=6645979. [Accessed 5 June 2019].

[16] M. Abulaish and T. Anwar "A SUPERVISED LEARNING APPROACH FOR AUTOMATIC KEYPHRASE EXTRACTION", *International Journal of Innovative Computing, Information and Control*, vol. 8, 2012. Available: https://pdfs.semanticscholar.org/e36e/2955cd30917d62091ac5ef15bf71eed84616.pdf. [Accessed 26 August 2019].

[17] Turney, Peter "Learning Algorithms for Keyphrase Extraction", *Inf. Retr.*, vol. 2, pp. 303-336, 2000.Available: https://www.researchgate.net/publication/220479857_Learning_Algorithms_for_Keyphrase_Extraction/citation/download. [Accessed 5 July 2019].

[18]] I. Witten, G. Paynter, E. Frank, C. Gutwin and C. Nevill-Manning, "KEA: Practical Automatic Keyphrase Extraction", *Proceedings of the Fourth ACM conference on Digital Libraries, August 11-14, 1999, Berkeley, CA, USA*, pp. 254-255, 1999. Available: https://www.cs.waikato.ac.nz/~ml/publications/2005/chap_Witten-et-al_Windows.pdf. [Accessed 4 May 2019].

[19] O. Medelyan,& I. Witten (2006). Thesaurus based automatic keyphrase indexing. 296 - 297. 10.1145/1141753.1141819. Available: https://www.researchgate.net/publication/224061561_Thesaurus_based_automatic_keyphrase_indexing [Accessed 4 May 2019].

[20] Taemin Jo, Jee-Hyong Lee "Latent Keyphrase Extraction Using Deep Belief Networks", *International Journal of Fuzzy Logic and Intelligent Systems*, vol. 15, pp. 153-158, 2015. Available: https://www.e-sciencecentral.org/articles/SC000013961. [Accessed 28 August 2019].

[21] J. P. Tensuan, A. Azcarraga "NEURAL NETWORK BASED KEYWORD EXTRACTION USING WORD FREQUENCY, POSITION, USAGE AND FORMAT FEATURES", *Research Congress 2012 De La Salle University*, 2013. Available: https://pdfs.semanticscholar.org/9302/03b85f789972107df4bc08da77632e477b84.pdf. [Accessed 5 June 2019].

[22] Wang, Yang & Gong, Yeyun & Huang, Xuanjing. (2016). Keyphrase Extraction Using Deep Recurrent Neural Networks on Twitter. 836-845. 10.18653/v1/D16-1080. Available: https://www.researchgate.net/publication/311990419_Keyphrase_Extraction_Using_Deep_Recurrent_Neural_Networks_on_Twitter. [Accessed 10June 2019].

[23] Manning, Christoper & Surdeanu, Mihai & Bauer, John & Finkel, Jenny & Bethard, Steven & McClosky, David. (2014). The Stanford CoreNLP Natural Language Processing Toolkit. Proceedings of 52Nd Annual Meeting of the Association for Computational Linguistics: System Demonstrations. 10.3115/v1/P14-5010. Available: https://www.researchgate.net/publication/272091377_The_Stanford_CoreNLP_Natural_Language_Processing_Toolkit [Accessed 10 June 2019]

[24] T.Weerasooriya, N. Perera, S.R. Liyanage, (2017). KeyXtract Twitter Model - An Essential Keywords Extraction Model for Twitter Designed using NLP Tools. Available: https://www.researchgate.net/publication/319035921_KeyXtract_Twitter_Model_An_Essential_Keywords_Extraction_Model_for_Twitter_Designed_using_NLP_Tools. [Accessed 5 June 2019].

[25] [Online]. Available: https://www.guru99.com/word-embedding-word2vec.html. [Accessed: 20- Nov- 2019]