TELKOMNIKA, Vol.16, No.4, August 2018, pp. 1704~1711
ISSN: 1693-6930, accredited First Grade by Kemenristekdikti, Decree No: 21/E/KPT/2018
DOI: 10.12928/TELKOMNIKA.v16i4.9062

News Reliability Evaluation using Latent Semantic Analysis

Guo Xiaoning*[1], Tan De Zhern[2], Soo Wooi King[3], Tan Yi Fei[4], Lam Hai Shuan[5]
[1,2,4,5]Engineering Big Data Lab, Faculty of Engineering, Multimedia University, Persiaran Multimedia, 63100, Cyberjaya, Selangor, Malaysia, 1-300-80-0668
[3]Faculty of Computing and Informatics, Multimedia University, Persiaran Multimedia, 63100, Cyberjaya, Selangor, Malaysia, 1-300-80-0668
*Corresponding author, e-mail: guo.xiaoning@mmu.edu.my[1], darentan94@gmail.com[2], wksoo@mmu.edu.my[3], yftan@mmu.edu.my[4], hslam@mmu.edu.my[5]

Abstract

The rapid rise and widespread of 'Fake News' has severe implications in the society today. Much efforts have been directed towards the development of methods to verify news reliability on the Internet in recent years. In this paper, an automated news reliability evaluation system was proposed. The system utilizes term several Natural Language Processing (NLP) techniques such as Term Frequency-Inverse Document Frequency (TF-IDF), Phrase Detection and Cosine Similarity in tandem with Latent Semantic Analysis (LSA). A collection of 9203 labelled articles from both reliable and unreliable sources were collected. This dataset was then applied random test-train split to create the training dataset and testing dataset. The final results obtained shows 81.87% for precision and 86.95% for recall with the accuracy being 73.33%.

Keywords*: Fake news detection, Natural language processing, Latent semantic analysis, Cosine similarity, Tf-idf*

1. Introduction

The news industry had changed greatly in the last decade from newspapers and TV broadcasts to internet news sites to blogs and social media. There are wide ranges of news sources available free online. The convenience of social media paired with smart devices, gave means to record and share news online within minutes of their occurrence. This heightened level of accessibility in turn produces news content much faster and at a greater quantity that may be relayed and shared with no significant third party filtering, fact-checking, or editorial judgment [1].

While the internet has enabled the rapid sharing of knowledge and information, it had also made it incredibly easy to spread false news information ('Fake News') with altered facts or deliberately made up stories. The term 'Fake News' was made prevalent during the 2016 United States presidential election. With fake news stories being churned out at an increasingly fast pace every day on multiple channels such as Facebook, Twitter, and other media on the Internet, it is becoming an increasingly serious problem for our society.

Internet sites such as Snopes.com and Politifact.com mainly employing manual efforts in conducting fact checks to combact fake news. Google Chrome also produced extensions with varying results, some aimed at automating fact checking for social media feeds (e.g. FiB) and others provide alerts when the user visits a webpage known for spreading fake news or contains links to unreliable sources (e.g.Fake News Alert and B.S. Detector). Due to the rapidly production of news articles, there is a definite need for an automated method to assist human fact checkers to verify the veracity of news articles.

Driven by this motivation, many researchers have explored using automated methods with the aim of identifying fake news articles on the Internet. Jin and colleagues developed a method [2] to automatically verify the credibility of news being shared on Sina Weibo. The method used included of a 3-layer credibility network which represents events on different scales to evaluate its credibility. It was found that by grouping similar messages together, it is possible to represent significant parts of an event better by reducing the effects from noisy data.

Received Mar 28, 2018; Revised June 8, 2018; Accepted June 29, 2018

This group later examined the possibility of verifying news by utilizing conflicting viewpoints [3], this is accomplished by creating a credibility network based on supporting and opposing tweets using an unsupervised topic model.

Conroy, Rubin and Chen [4, 5] made use of linguistic patterns [6] to identify click baits and fake news articles. It was found that linguistic patterns, such as the use of suspenseful language, unresolved pronouns, a reversal narrative style, forward referencing, image placement, reader's behaviour and several other cues can be used to detect click baits.

In this paper, a method is proposed to determine whether an article is 'reliable' or 'unreliable' based on its similarity with articles collected from both reliable and unreliable sources. The proposed method utilizes several Natural Language Processing (NLP) techniques such as phrase detection, Term Frequency-Inverse Term Frequency (TF-IDF), cosine similarity and Latent Semantic Analysis (LSA) to identify the reliability of the news articles. A collection of 9203 labelled articles from both reliable and unreliable sources were used. This dataset is then applied an 80/20 random test-train split to create the training dataset and testing dataset. Details on experimental setup as well as data collection method please refer to section 4 of this paper. The final results obtained shows 81.87% for precision and 86.95% for recall with the accuracy being 73.33%.

This paper is organized as follows: section 2 describes the proposed method and its overall, the techniques used in this project are outlined in section 3. In section 4, the experimental setup and data collection method is presented. The results and discussion are given in section 5 and finally, the conclusion and suggestions for future work is in section 6.

2. The Proposed Method/Algorithm

In this section, we describe the propose method to evaluate if query articles are reliable or unreliable. Figures 1 and 2 presents the overall flow of both the training and test phases. In the training phase as shown in Figure 1, labelled news articles from the testing dataset are read from the database into the program. After applying several text pre-processing methods such as stopword removal, tokenization and removing words that appears once, the text data is then converted into a corpus using term frequency-inverse term frequency (TF-IDF) representation. The TF-IDF representation is then put into an LSA model and a news article LSA model is created that will return two lists of similar articles and their topic numbers. These two lists are for reliable and unreliable articles respectively.

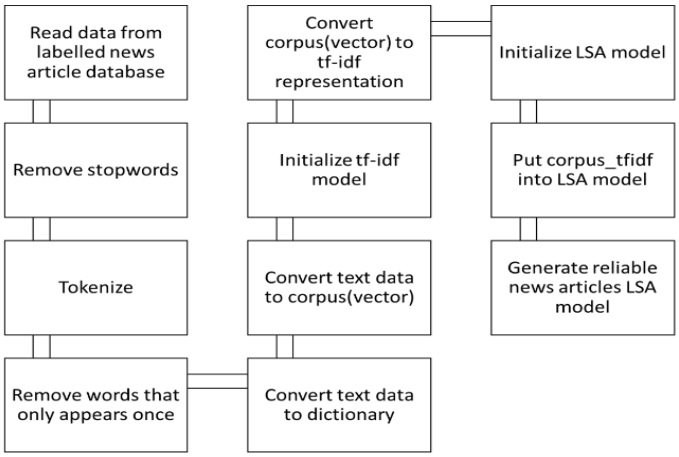
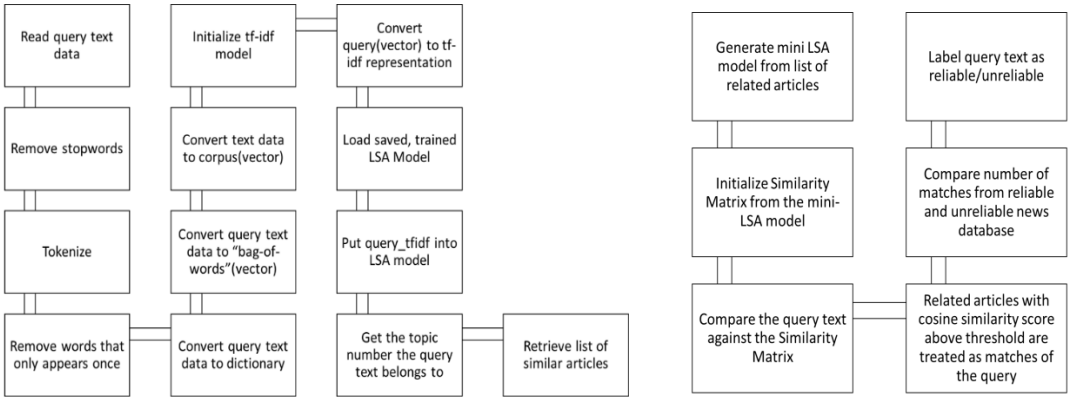


Figure 1. Training Stage

The testing phase as shown in Figure 2 consists of two parts. In the first part, the same text pre-processing method in Training stage is applied to the query text. The query text is then converted into a TF-IDF representation and into the LSA model that was trained in the training phase. This will return the query articles' corresponding topic numbers to that obtained from the

training phase. This means that the articles with the same topic number will most likely be similar in content. In the second part, to improve the final results, the articles under each topic number, from both the labelled training set as well as the supposingly unlabeled testing set will need to be further compared using cosine similarity matrix. This is done by first generating a mini LSA model for that topic. Then the query is compared against the similarity matrix and the number of matches from the model is returned. The solution labels the query text as reliable or unreliable depending on the weightage obtained respectively.



Part 1: Generate list of similar articles and topic number Part 2: Evaluate cosine similarity

Figure 2. Testing Stage: Part 1 and 2

3. Research Method

3.1 Term Frequency-inverse Document Frequency (TF-IDF)

In order to represent an article with limited number of its keywords so further comparison of articles can be made possible, TF-IDF is adopted. TF-IDF is used for text feature representation in this solution.

After the news articles are pre-processed, they are converted into TF-IDF vectors which assigns weightages to each of the words in the articles depending on 'important' the word is. TF-IDF contains a matrix of the words and their respective weightages which serves as a representation of news articles in n-dimensional vector space. The corpus (collection of news articles) is represented as a set of term frequencies (TF) where weight for each term is dependent on inverse document frequency (IDF). Semantically similar or related news articles will be positioned closer in the vector space depending on the training method depending on whether words or documents being used.

$$w_{i,j} = tf_{i,j} \times \log \frac{N}{df_i} \quad (1)$$

As shown in equation 1, the weightage of a word *i,* $w_{i,j}$ is obtained by multiplying the number of times word *i* occurred in *j,* $tf_{i,j}$ with the log of the total number of documents, *N* divided by the number of documents that contains word *i*, df_i. A word that appears less frequent in the whole document, will have a high weightage indicating that it is an important word. In a study done in 2017, it was found that term weighting method such as tf-idf performed well for feature selection in cases of high dimension feature space [7].

3.2 Phrase Detection (Bigram)

Phrase detection [8] is applied to our collection of news articles to detect and generate phrases from the tokenized words. In comparison to word only detection, this would provide a better representation of the news article. It is a technique that automatically detects common phrases from sentences based on how frequent two words occur together otherwise known as the principle of collocation. It consists of two steps. Firstly, a detector must be trained using a

collection of documents. Then, the trained detector is used to initialize Phrase object which is then used to detect bigrams (2-worded phrases) in any sentences. This method can also be extended to detect trigrams (3-word phrases).

$$score(w_i, w_j) = \frac{count(w_i w_j) - \delta}{count(w_i) \times count(w_j)} \quad (2)$$

The formula in equation 2 above is used to find the bigram score of two words, w_i and w_j. The number of times the two words appear beside each other in a sentence, count($w_i w_j$) is deducted by the discounting coefficient, δ divided by the multiplication of the number of occurrences of $w_i$and w_j. The discounting coefficient, δ is to prevent excessive phrases containing infrequent words from forming.

3.3 Latent Semantic Analysis

In this paper, Latent Semantic Analysis (LSA) is used to generate a list of similar articles. Latent Semantic Analysis (LSA) is essentially a dimension reduction technique to find underlying (latent) meaning/concept in documents. It has been shown to be an effective computational text processing method [9]. It builds semantic space/representation from a corpus of text where two conceptually similar words/documents will be mapped close together. Semantic space/representation can be then used to find similarity between different media (words, sentences, paragraphs, documents) using cosine similarity.

LSA works by performing dimensionality reduction on a set of documents to extract a spatial representation of word meanings. The reduction uses a well-known matrix factorization technique, Singular Value Decomposition (SVD) that decomposes a matrix into three matrices namely, matrix U, D and V^T to find a lower rank approximation of the term-document matrix.

$$X = UDV^T \quad (3)$$

A known advantage of a knowledge-independent approach such as LSA is that it takes up less resource as compared to knowledge-dependent approach such as Wordnet[10]. LSA is often compared with another topic modelling technique known as Latent Dirichlet Allocation (LDA). In the study done by [11], it is found that LSA outperforms LDA on both small-scale and large-scale test cases in terms of performance when processing Patent data collection. LSA makes high dimensional word embeddings such as TF-IDF matrix more manageable for future operations such as clustering/classifications. Choosing the right dimension for k is important as a k-value which is too low would not give a proper representation of relationships between the terms and documents and conversely, a k-value which is too high would create too much noise. Past research [12] have shown no further gains beyond 300 dimensions. Hence, a dimension of 300 is chosen for our experiment.

3.4 Cosine Similarity

In the context of this project, Cosine similarity is used to evaluate the reliability of a query article. It is used to measure the similarity between the query news article text and list of related articles from a reliable and unrealiable news article datasets [13]. Related articles with cosine similarity score above a selected threshold are considered as a match to the query news article. The cosine similarity between two vectors (or two documents on a vector space in this news case) is a measure that calculates the cosine of the angle between them. It measures the orientation of the vectors but not the magnitude (distance).

The cosine of the angles serves as a comparison between the documents on a normalized vector space that shows how related two documents are based on the angle between them. Cosine similarity is found to be well suited to measure similarity of word vectors in high-dimensional space [14]. Good performance found in the experimental evaluation involving 15 different datasets done by [15] of performing k-mean clustering with cosine similarity. Some important applications of this technique are in the domains of unsupervised document organization, automatic topic extraction and fast information retrieval or filtering.

$$\vec{x} \cdot \vec{y} = \|\vec{x}\|\|\vec{y}\| \cos \emptyset \quad (4)$$

$$\cos \emptyset = \frac{\vec{x} \cdot \vec{y}}{\|\vec{x}\|\|\vec{y}\|} \quad (5)$$

Equation 5 is derived from the equation 4 which is the Euclidean dot product formula. equation 5 gives the cosine similarity score of two vectors, x and y which is obtained by dividing the dot product of vectors $\vec{x}$ and $\vec{y}$ with product of the magnitude of vector x, $\|\vec{x}\|$ and y, $\|\vec{y}\|$. Similar article vectors will be in the same direction with small angle deviations between them, producing a high cosine similarity score. Unrelated article vectors will have a nearly orthogonal angle, producing a cosine similarity score approaching 0. Lastly, opposite article vectors will have an angle approaching 180°, producing a negative similarity score.

4. Experimental Setup

For the experiment, the solution was ran on a Cloudera Cluster consisting of 1 Master node running on 4CPU Cores and 14GB RAM and 6 Slave nodes running on 1 CPU Core and 4GB RAM each. The host machine is equipped with a Intel® Core™ i7-4790 CPU @ 3.60GHz Octa-Core CPU and 32GB RAM and running on CentOS Linux 7. The programming language used to code the solution were Python 3.1.2 along with the Gensim 2.3.1 and NLTK 3.2.4.

4.1 Data Collection

4.1.1 Corpus of News Articles

A dataset of news articles published by Signal Media in conjunction with the conference called Recent Trends in News Information Retrieval 2016 to help research dealing with news articles was obtained. The dataset is made up of 1 million news articles that are mainly English and also other non-English and multilingual articles from the time period of 1st-30th September 2015. It consists of 265,512 Blog articles and 734,488 News articles compiled from 93,000 unique sources. In order to label these articles as to whether they are from reliable or unreliable sources, a dataset of known reliable and unreliable news sources is required.

4.1.2 Labelled News Sources

Two datasets were obtained from Opensource.co and FakeNewsWatch.com to represent reliable and unreliable news sources. From Opensource.co, 841 labelled news sources were obtained whilst FakeNewsWatch.com provides 92 labelled news sources. The final news source dataset consists of 933 sources of which 912 are considered unreliable and 21 are reliable.

4.1.3 The Labelled News Articles Dataset

The final dataset of labelled news articles is obtained by performing an SQL JOIN of the news sources with the corpus of news articles. The dataset has 5 columns which are News Id, News Title, News Article, Publish Date and Label and has a total of 9203 news articles consisting of 7634 articles from reliable sources and 1569 articles from unreliable sources. This dataset is then applied an 80/20 random test-train split to create the training dataset and testing dataset. The training set consists of 7355 rows while the testing set consists of 1848 rows.

5. Results and Discussion

5.1. Phrase Detection

An experiment was conducted to determine the effectiveness of bigram phrase detection on the result of topic detection technique, LSA using a sample of 100 articles consisting of 25 topics. As seen in the results below, the keywords of topic detection with phrase detection contains phrases that gives a better representation of the topic that is detected. Figures 3 and 4 shows the Top 10 keywords along with their weightages for the first two topics that were detected. Topic 1 was about “Malaysia’s *bomoh* charged in court” while Topic 2 was about “Beauty and the Beast banned in Malaysia” Positive weightages indicate the word contributes to the topic positively whereas negative weightages indicate the word contributes to

the topic negatively. This shows that the inclusion of the phrase detection technique improves the accuracy of LSA in the fake news detection system.

```
[(0,
  u'0.397*"ibrahim" + 0.199*"federal" + 0.199*"jawi" + 0.180*"malaysia" + 0.171*"islam" + 0.162*"territories" + 0.158*"also" + 0.154*"said" + 0.149*"police" + 0.136*"north"'),
 (1,
  u'-0.357*"film" + -0.340*"gay" + -0.282*"beast" + -0.266*"disney" + -0.263*"beauty" + -0.198*"moment" + -0.198*"malaysia" + -0.185*"scene" + -0.184*"cut" + -0.149*"movie"')]
```

Figure 3. Excerpt of results without bigram phrase detection

```
[(0,
  u'0.370*"ibrahim" + 0.205*"beauty_beast" + 0.193*"said" + 0.189*"malaysia" + 0.165*"also" + 0.154*"police" + 0.144*"film" + 0.134*"federal_territories" + 0.127*"antics" + 0.127*"raja_bomoh"'),
 (1,
  u'0.378*"beauty_beast" + 0.260*"film" + 0.215*"malaysia" + -0.204*"ibrahim" + 0.199*"gay_moment" + 0.192*"disney" + 0.140*"movie" + 0.134*"country" + 0.132*"scene" + 0.107*"abdul_hamid"')]
```

Figure 4. Excerpt of results with bigram phrase detection

5.2. Cosine Similarity Threshold

This next experiment was conducted to determine the best cosine similarity threshold for the fake news detection system for a news article to be considered as a match. The threshold was chosen by comparing the accuracy measures obtained at different thresholds on a sample of 500 articles. It was found that the threshold of 0.7 as shown in Table 1 gave the best overall accuracy compared to others. The reason is because setting higher thresholds (stricter matches), there is a gradual decrease across all accuracy measures and also a significant increase in articles being discarded. There were 45 articles without matches at threshold 0.7 compared to 220 articles as shown in Table 2 without matches at threshold 0.9 in this sample dataset of 500 rows which makes up 44%, almost half of the size of sample dataset.

Table 1. Accuracy measures of Fake News Detection system at different similarity thresholds

Threshold	Precision	Recall	Accuracy
0.6	80.99%	85.91%	72.01%
0.7	**81.79%**	**88.01%**	**74.10%**
0.8	81.16%	85.58%	71.84%
0.9	78.64%	75.35%	63.94%

Table 2. Numbers of articles classified at different similarity thresholds

Threshold	Number of articles classified	Undetermined	No Matches
0.6	443	28	29
0.7	**417**	**38**	**45**
0.8	381	30	89
0.9	269	11	220

5.3 News Reliability Evaluation Framework Result

As it can be seen from the Tables 3 and 4 the solution showed a high precision and recall values of 81.87% and 86.95% respectively. The high precision value indicates that the system classified the news articles correctly of all the articles that is recalled. The high recall value indicates that the system recalled correctly out of all the events that it deems to be correct. Both of these measures shows that the system has good performance in evaluating the reliability of news articles and identifying potential 'Fake News'.

Table 3. Confusion Matrix of the results (article count and percentages)

		Actual Value: Positives	Actual Value: Negatives
Predicted Value	Positive	1106 (71.77%)	245 (15.90%)
	Negative	166 (10.77%)	24 (1.56%)

Table 4. Precision, Recall and Accuracy

Measures	Precision	Recall	Accuracy
Values	0.81865285	0.869496855	0.733290071
Percentage	81.87%	86.95%	73.33%

The accuracy of the system is at 73.33% and also gave a low true negative of 1.56% and a high false negative of 10.77%. One probable reason for this could due to the dataset is not being well balanced in terms of reliable and unreliable article count. There are 7634 reliable articles while only 1569 unreliable articles in the final dataset. This difference in the size of the dataset might have affected the ability of the system to identify the features of the unreliable articles. The results could be improved if the number of articles and unreliable articles used were more balanced.

6. Conclusion

In this project, a news reliablility evalutation system consisting of several Natural Language Processing techniques and a topic detection method, Latent Semantic Analysis was implemented in order to explore its performance in helping users to automatically identify fake news articles on the Internet. The results have displayed good accuracy and also showed great potential as these techniques are not restricted to the domain of news articles. With some modification, it may be extended to other domains such as fraud detection, data anomaly checking, sentiment analysis and so forth.

For future direction, it would be interesting to perform experiments to compare performance of Latent Semantic Analysis (LSA) with Latent Dirichlet Allocation (LDA) and Non-negative Matrix Factorization (NMF) on in the domain of fake news detection. By performing the said experiment, the effectiveness of each technique as well as find out the strengths and weaknesses of each technique can be identified. Another technique to explore would be the “semantically similar” methods such as Word Mover’s Distance (WMD) algorithm which can take into account of negation in sentences. This would greatly improve the accuracy of the system.

Acknowledgements

This work was funded by the TM Research & Development (TM R&D) Grant.

References

[1] Allcott H, Gentzkow M. Social Media and Fake News in the 2016 Election. *Journal of Economic Perspectives.* 2017; 31(2): 211-36.
[2] Jin Z, Cao J, Jiang YG, Zhang Y. *News credibility evaluation on microblog with a hierarchical propagation model.* InData Mining (ICDM), 2014 IEEE International Conference. 2014 Dec 14: 230-239.
[3] Jin Z, Cao J, Zhang Y, Luo J. *News Verification by Exploiting Conflicting Social Viewpoints in Microblogs.* InAAAI 2016: 2972-2978.
[4] Conroy NJ, Rubin VL, Chen Y. *Automatic deception detection: Methods for finding fake news.* Proceedings of the Association for Information Science and Technology. 2015; 52(1): 1-4.

[5] Chen Y, Conroy NJ, Rubin VL. *News in an online world: The need for an automatic crap detector.* Proceedings of the Association for Information Science and Technology. 2015; 52(1): 1-4.
[6] Chen Y, Conroy NJ, Rubin VL. *Misleading online content: Recognizing clickbait as false news.* In Proceedings of the 2015 ACM on Workshop on Multimodal Deception Detection. 2015: 15-19).
[7] Fauzi MA, Arifin AZ, Yuniarti A. Arabic Book Retrieval using Class and Book Index Based Term Weighting. *International Journal of Electrical and Computer Engineering (IJECE).* 2017; 7(6): 3705-10.
[8] Mikolov T, Sutskever I, Chen K, Corrado GS, Dean J. Distributed representations of words and phrases and their compositionality. *In Advances in neural information processing systems*. 2013: 3111-3119.
[9] Rosario B. Latent semantic indexing: An overview. Techn. rep. INFOSYS. 2000; 240: 1-6.
[10] Yang W, Lin L. Process Improvement of LSA for Semantic Relatedness Computing. *TELKOMNIKA (Telecommunication Computing Electronics and Control)*. 2014; 12(4): 1045-52.
[11] Cvitanic T, Lee B, Song HI, Fu K, Rosen D. *LDA v. LSA: A Comparison of Two Computational Text Analysis Tools for the Functional Categorization of Patents.* InICCBR Workshops. 2016; 41-50.
[12] Bradford RB. *An empirical study of required dimensionality for large-scale latent semantic indexing applications.* In Proceedings of the 17th ACM conference on Information and knowledge management 2008; 153-162.
[13] Buchta C, Kober M, Feinerer I, Hornik K. Spherical k-means clustering. *Journal of Statistical Software.* 2012; 50(10): 1-22.
[14] Basha MJ, Kaliyamurthie KP. An Improved Similarity Matching based Clustering Framework for Short and Sentence Level Text. *International Journal of Electrical and Computer Engineering (IJECE).* 2017; 7(1): 551.
[15] Zhao Y, Karypis G. Empirical and theoretical comparisons of selected criterion functions for document clustering. *Machine Learning.* 2004; 55(3): 311-31.