IJCSE International Journal of Computer Sciences and Engineering Open Access
Research Paper Vol.-7, Issue-1, Jan 2019 E-ISSN: 2347-2693

Document Categorization for Probabilistic Redundant Documents

S. Singh[1*], K. Jain[2]

[1,2]Dept. of Computer Science and Engineering College of Technology and Engineering, MPUAT Udaipur, India

[*]*Corresponding Author: shipika.cse@gmail.com*

Available online at: www.ijcseonline.org

Accepted: 23/Jan/2019, Published: 31/Jan/2019

***Abstract*—** Text categorization is an active research area in information retrieval and machine learning. The major issue regarding preprocessing the document for this categorization is redundancy. The redundant documents slow down the learning steps of classification and also affect its efficiency and scalability. To resolve this issue it is preferred, first identify the duplicates and then perform the classification. This paper proposes to apply the Similarity Measure for duplicate detection and Random forest for classification. The results are evaluated using '20 newsgroups' data sets with generated duplicate documents. Accuracy and time parameters show better results in the proposed method than that in the existing text categorization model.

Keywords— Duplicate-detection, text categorization, information retrieval, similarity measure

I. INTRODUCTION

Text document has turned into the most well-known sort of data storage facility, particularly with the expanded ubiquity of the web and the internet. This comes with more demands like feasible representation model, multiple duplicates, high dimensionality, etc. Thus to manage the memory legitimately, it is essential to have the capacity to recognize whether a specific document has just been gone into the system or not, and how efficiently information can be recovered from the system. Having the capacity to identify the duplicate document, can enhance the execution of the search engine and use to recover documents from the particular corpus.

Classification techniques are used to analyze data and predict labels that describe important properties of data. Many classification techniques have been developed from a single label [1],[2] to multi-label [3],[4],[5] with the increase in dimensionality, thus increased the challenge for replicas. Thus for detecting duplicates, there are some existing methods like image matching, Simhash, a signature based method, checksum, feature selection, Shingling and using salient terms/phrases. Here, we propose a similarity measure as duplicate detection technique.

In Section II the related work of classification and duplicate detection are discussed. Section III describes the architecture and technology of proposed work. Section IV shows the results of the experiments performed over the 20 Newsgroup dataset and discussion related to the effectiveness of the proposed model and Section VI concludes research work with future directions.

II. RELATED WORK

A method of text categorization using Random Forest algorithm [1] is proposed to categorize documents using decision trees. First, a Vector Space Model is built by taking features as its axis and plotting documents and categories in it. Decision trees are built using the features of category. If the testing document gets more positive votes as compared to negative for a category, then the testing document must be of this category. Random Forest is a Famous Integrated Learning algorithm for categorization. The Decision Tree gives exact dataset. But it does not deal with the duplicate documents each time a document is given to it. Therefore duplicate document is again passed to the thousands of decision trees. This reduces the system consistency of information retrieval.

To reduce the dimensionality problem with multiple labels, hybrid model of random forest and rocchio algorithm [3] is proposed. For multi-label, vector space model is used where all features are axis and categories and documents are plotted in the model. For new document, categories lying above the threshold are passed to random forest classifier. The selected category is finally allocated to the new document and this document is updated into the training dataset using rocchio vector updater.

Adaptive duplicate detection used learnable string similarity measures [8] which handles the calculated text distance

© 2019, IJCSE All Rights Reserved

functions for every database parameter and presented that these measures are applied for adjusting the analyzed idea which fits in the field's domain. Two text similarity measures are demonstrated: an extended variation of learnable string edit distance, and a novel vector-space based measure that takes help of Support Vector Machine (SVM) for while training the system. The accuracy shows that this model works fine with string comparison duplicate detection.

A web crawler expands its quality [9], by identifying whether a recently crawled web page is a near-duplicate of previously crawled web pages or not. Charikar's fingerprinting technique and simhash are used for differentiating the already present f-bit fingerprints with the provided fingerprint in at most k bit-positions, where k is very small. Simhash is used because it works with small-sized fingerprints. But when it comes with large datasets it fails to work efficiently.

Simhash [6], a signature-based technique which differentiate the near duplicate documents. Simhash's accuracy is improved by taking external metadata and checking its feature selection approach. This extra metadata available to the documents is included here, excluding all the numbers and dates (in addition to stop words), making a threshold under which weights did not influence the final log(x) simhash, and skewing the weights by arbitrary functions, x and cos(x), to demonstrate if the algorithm could exploit knowledge about the structure of the documents.

Eldhose [10] proposed a five-stage algorithm to detect the near duplicate web pages, this includes preprocessing, minimum weighting, filtering and verification and classification stages of the web page. Jacquard threshold value t is used in similarity verification, for 0<t<1. If t.t=1, it will return all the records which contains at least a single word or similar content to the user input record i. If t.t=1, returns only the exact record with the domain of web page exists The execution of the proposed TDW (Term Document Weight) Matrix with minimum weighting helps in reducing the number of comparisons, but don't work to detect exact duplicates.

So by combining the work of classification and duplicate detection, similarity measure is proposed to detect the duplicate and binding it with random forest for classification for unique documents.

Similarity Measure

Similarity is a method which builds an index for a given set of documents. The result is a vector of numbers as large as the size of the initial set of documents, that is, one float for each index document. With these index values similarity between query document and given set of documents can be computed.

The set of documents D_{Train}={d1, d2, d3…….dn}, where n is the total number documents, are converted into vector form.

For all the documents, tf-idf (term frequency-inverse document frequency) is generated with only relevant terms. Relevant terms are those terms which gives some value to the context, irrelevant term are eliminated in this.

The generated index is in vector form

$$X = \{x_1, x_2, x_3, \ldots\ldots\ldots x_n\} \quad (1)$$

where, x_i = tf-idf [d_i]

for $1 \le i \le n$

Same way the testing document D_{test} is also converted into its vector form 'y'.

$$y = tf - idf[D_{test}] \quad (2)$$

Compare y against all the values of X, store the result in sims vector,

$$Sims = \{c_1, c_2, c_3, \ldots\ldots\ldots, c_n\} \quad (3)$$

if y==x_i , for $1 \le i \le n$

c_i=1

By checking the sims vector, duplicates can be found,

If sims[i]==1, for $1 \le i \le n$

Duplicate of D_{test} is present in D_{Train} at index i.

label [D_{test}] = label [d_i]

Else

Categorize the D_{test}.

This comparison is made because two duplicate documents will be containing the same terms with the same tf-idf value. These duplicate documents will not be processed further for the classification (as its classification is done at the time of first original document arrived)

III. METHODOLOGY

In this paper, we propose similarity measure for eliminating the duplicate documents from the corpus. The system architecture of the proposed system is given in Figure 1 . A text document D_{Test} and training data sets D_{train} are given as an input to it. Then features are extracted from all the documents ($D_{test,}$[f] , D_{train}[f]). Using stop word remover, irrelevant words are deleted from both the training data set and testing document vector. After this, features are changed into its root form using the Porter stemmer algorithm so to avoid the high dimensionality problem. Resulted features are stored along with their frequency value like a key-value pair. 'Key' denotes the term and 'value' as its frequency value. For both training dataset and testing document this term frequency is generated. Following a Tf-idf matrix is built using Tf-idf generator which is placed before classifier to get the tf-idf value of every term in a document.

Tf-idf(term frequency-inverse document frequency) tells the relevance of a term with the document. Inverse document frequency is the value, which shows in how many documents this term is coming. More document frequency means term is very common and has less importance. With the help of this, tf-idf matrix is formed for both training dataset and testing

© 2019, IJCSE All Rights Reserved

document. If any one of the training matrix matches with the testing matrix, the label attached to that training document is given as an output of the machine. Otherwise, the testing document is further sent for classification." These tf-idf values are given as an input to the Random Forest Classifier.
Random forest is based on decision tree that uses tf-idf for the classification. Nodes of the decision trees are selected randomly from the given set of terms. Testing document is then passed by all the decision trees, to get validated. From this, the number of positive and negative votes is obtained. Document having higher positive votes than the negative votes, are termed as relevant to that category and that category is allocated to the testing document. The vector of the training data set is created only once and used for all upcoming testing documents.

IV. RESULTS AND DISCUSSION

The effectiveness of the proposed method is determined by performing three experiments, taking accuracy measure, time parameter and F_1 score as the performance measure. All the experiments are performed on 20 Newsgroups dataset. In the first experiment, accuracy of the proposed model is being compared with accuracy of the existing model. In the second experiment, comparison is done between proposed model and other duplicate detection techniques using time as a parameter and in the third experiment, F_1 scores are compared with four classification techniques Bernoulli NB, Rocchio, SVC and KNN.

Dataset: 20 Newsgroups

20 Newsgroups dataset is taken from Usenet articles Ken Lang collected from 20 different newsgroups. Four categories Computer.graphics, Science.space, Talk.religion and Alt.athesim are chosen. This dataset is unique, contains 3387 number of documents. To check the efficiency of proposed method, different size of data sets is formed with varied percentage of duplicate documents (generated). The size of training data set and testing data set along with the percentage of randomly generated duplicates of the document are given in Table 1 . 3% duplicated documents mean 100 duplicate documents are added to the dataset, total documents are now 3487 (i.e. 3% of 3387 is 100). Similarly, documents for 30%, 100%, 150% and 200% are generated.

A. Experiment to compare accuracy with existing model.

The metric accuracy is used to evaluate the performance of the system. The formula for accuracy is given in Eqn. (4)

Table 1 Duplicated documents (in percentage) with total number of training and testing documents.

Duplicated documents (in %)	Number of documents duplicated	Total number of Training documents	Total number of Testing documents
3%	100	2789	698
30%	1000	3509	878
100%	3,387	5419	1355
150%	5080	6773	1694
200%	6774	8128	2033

$$Accuracy = \frac{TP}{TP + TN} \quad (4)$$

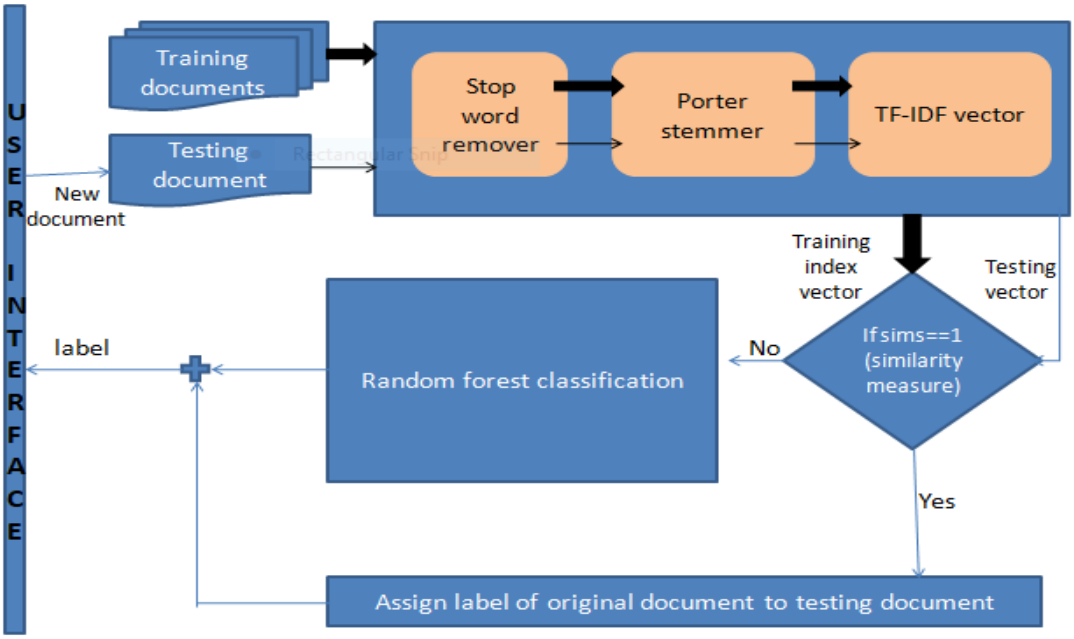

Figure 1 Proposed Similarity based Classification architecture

© 2019, IJCSE All Rights Reserved

Where TP presents True Positive, FP presents False Positive. Accuracy is calculated for different sets of duplicate document (in percentage). The accuracies for existing text categorization [3] model and test categorization with similarity measure for duplicate detection method (proposed method) are shown in Figure 2 .

From the Figure 2 , it is understood that the accuracy of proposed method increases with the increase in number of duplicates, because this model don't check for classification in case of duplicate documents. The accuracy of proposed and existing method reaches 1 when the number of duplicate documents increases, this is because model learns the duplicate documents and detects easily. By taking only the relevant features and a finite number of categories, the error rate has been reduced. And thus improves the accuracy.

B. Experiment to calculate and compare the time parameter.

To compare the efficiency of proposed method with other duplicate detection method, Simhash and feature selection, we adopt time measures (in seconds). The time parameter is calculated with 3%, 30% and 100% duplicates. Time parameter is the time taken by the model to classify the testing document. The values obtained by using different methods for same dataset are shown in Table 2.

Table 2 Time comparison for Simhash, Proposed method and Feature selection

Duplicate documents (in %)	Simhash (in sec.)	Proposed method (in sec.)	Feature selection (in sec.)
3%	32	4	12
30%	44	7	31
100%	75	8	42

From Table 2 , Simhash and Feature Selection method's 'time' factor increases rapidly with the increase in duplicate documents. Although accuracy of Simhash [6] is very high as compared to Feature selection and proposed model, because Simhash calculates hash for every document. And Feature selection consumes time in comparison of features for checking the duplicates.

C. Experiment with other classification technique

20 Newsgroups datasets with 100% duplicate, the experiment was conducted. Here the comparison is done against four different classification methods, BernoulliNB, Rocchio, SVC and KNN. We compare the effectiveness of each method, we adopt the performance measure $F_{\beta(\beta=1)}$. It considers both precision and recall to compute the score. The formula for calculating F_1 value is given in Eqn (5).

$$F_1 = 2 * \frac{P * R}{P + R} \quad (5)$$

The value of F_1 is considered for comparison. P represents precision and R represents recall. The F_1 score of proposed method and other classification technique is showed in Table 3. Result shows that among these classification techniques, F_1 score is high for proposed method.

Table 3 F1 score for proposed method and other classification technique with 100% duplicated documents.

	Bernoulli NB	SVC	Rocchio	KNN	Proposed
F1-score	92.4	95	95.5	89.8	96

V. CONCLUSION AND FUTURE SCOPE

This research paper proposes a new duplicate detection model for text categorization using the similarity measure. This model overcomes the disadvantage of other text categorization that doesn't check for duplicates and re-process the same document over multiple times.

By using the similarity measure model, duplicates can be checked and wiped out from further processing. We can find the duplicates by comparing the tf-idf value of the testing document with tf-idf values of the training documents, where testing document can be the duplicate of any existing document in the training dataset. For new documents,

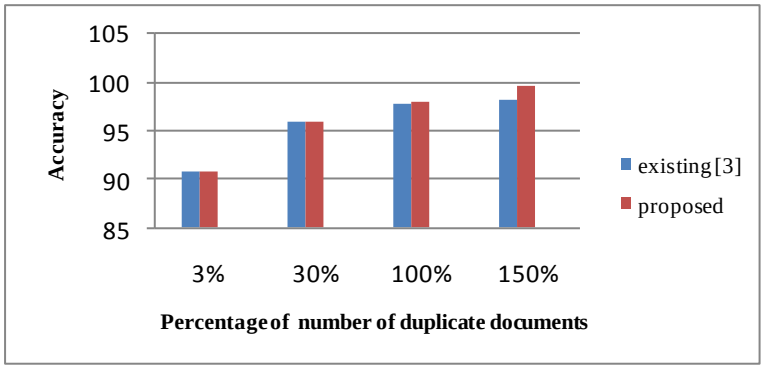

Figure 2 Accuracy (in %) of existing [3] and proposed method with different sets of duplicate documents (in %)

© 2019, IJCSE All Rights Reserved

random forest algorithm is used to classify the document. Experiments on 20 Newsgroup dataset with a varied number of generated duplicates is done to check the accuracy of proposed text categorization model. The results shows that the accuracy of proposed model is better than the existing model.

In future, more efficient duplicate detection techniques can be introduce which can reduce the execution time of the system. Also, a modified Simhash can be proposed which takes less time to create the hash table.

REFERENCES

[1] D. Xue, F. Li, "Research of Text Categorization Model based on Random Forests," IEEE International Conference on Computational Intelligence & Communication Technology, pp. **173-176**, **2015**.

[2] G. Gao, S. Guan, "Text Categorization Based on Improved Rocchio Algorithm," International Conference on Systems and Informatics, pp. **2247-2250**, **2012**.

[3] Thamarai, S.S., Kartikeyan, P., Vincent, A., Abinaya, V., Neeraja, G. and Deepika, R. 2016.Text Categorization using Rocchio Algorithm and Random Forest Algorithm. In the IEEE 2016 Eighth International Conference on Advanced Computing (ICoAC) held at Chennai, India, pp. **7-12**, **2017.**

[4] J.Y. Jiang, S.C. Tsai, S.J. Lee, "FSKNN: Multi-label text categorization based on fuzzy similarity and k nearest neighbors," Expert Systems with Applications, Vol. **39**, Issue. **3**, pp. 2813-2821, **2012**.

[5] M.L. Zhang, Z.H. Zhou, "A lazy learning approach to mullti-label learning," National Laboratory for Novel Software Technology, Vol. **40**, Issue. **7**, pp. 2038-2048, **2007**.

[6] S. Seshasai," Efficient near duplicate document detection for specialized corpora", Massachusetts Institute of Technology, **2008**.

[7] W. Zong, F. Wu, L.K. Chu, D. Schulli, "A discriminative and semantic feature selection method for text Categorization," School of Management, Xian Jiatoong University, China, IntJ.Production Economics, Vol.**165**, pp. 215-222, **2015**.

[8] M. Bilenko, R.J. Mooney," Adaptive Duplicate Detection Using Learnable String Similarity Measures", Proceedings of the Ninth ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, pp. **39-48**, **2003**.

[9] G.S. Manku, A.D. Sarma, A. Jain," Detecting Near Duplicates for Web Crawling", International World Wide Web Conference Committee (IW3C2), pp **141-149**, **2007**.

[10] E.P. Sim," Classification & Detection of Near Duplicate Web Pages using Five Stage Algorithm",Online International Conference on Green Engineering and Technologies (IC-GET), **2015**.

© 2019, IJCSE All Rights Reserved