International Journal of Computer Applications Technology and Research
Volume 7–Issue 08, 292-296, 2018, ISSN:-2319–8656

Measure the Similarity of Complaint Document Using Cosine Similarity Based on Class-Based Indexing

Syahroni Wahyu Irianandа
Electrical Engineering Department,
Faculty of Engineering,
Brawijaya University.
Malang, East Java, Indonesia

Muhammad Aziz Muslim
Electrical Engineering Department,
Faculty of Engineering,
Brawijaya University
Malang, East Java, Indonesia

Harry Soekotjo Dachlan
Electrical Engineering Department,
Faculty of Engineering,
Brawijaya University
Malang, East Java, Indonesia

Abstract: Report handling on "LAPOR!" (Laporan, Aspirasi dan Pengaduan Online Rakyat) system depending on the system administrator who manually reads every incoming report [3]. Read manually can lead to errors in handling complaints [4] if the data flow is huge and grows rapidly, it needs at least three days to prepare a confirmation and it sensitive to inconsistencies [3]. In this study, the authors propose a model that can measure the identities of the Query (Incoming) with Document (Archive). The authors employed Class-Based Indexing term weighting scheme, and Cosine Similarities to analyse document similarities. CoSimTFIDF, CoSimTFICF and CoSimTFIDFICF values used in classification as feature for K-Nearest Neighbour (K-NN) classifier. The optimum result evaluation is pre-processing employ 75% of training data ratio and 25% of test data with CoSimTFIDF feature. It deliver a high accuracy 84%. The k = 5 value obtain high accuracy 84.12%

Keywords: Complaints Document, Text Similarity, Class-Based Indexing, Cosine Similarity, K-Nearest Neighbour, LAPOR!

1. INTRODUCTION

The amount of incoming complaints and public opinion data on "LAPOR!" system (Online Peoples Complaint Service and Aspirations) can serve as a source of information to measure the performance of government service [1]. It processes an average of 900 reports every day, only 13 % - 14% of the reports, while about 86% remain subject to unknown and archived. The most used channel is via SMS s around 80 % - 90% report [2]. The report handling depends on the system administrator who reads every incoming report [3] . This can lead to errors in handling complaints [4], and if the data flow is very large it can take at least three days, this is sensitive to inconsistencies [3]. Limited administrators and high complaint report rates are a major cause of the lack of quality of service responsiveness characteristics [2]. A solution to that problem of complaints analysis is needed. It could help the "LAPOR!" Administrator in determining the category, so big data analysis becomes very important [2].

In this study the authors propose a model or approach that can measure and identify the similarity of document reports conducted in computerized that can identify the similarity between the Query (Q) with Document (D) collections,. This research employs Class-Based Indexing term weighting scheme, then compare with other term weighting schemes like TFIDF and TFICF. The weight values of TFIDF, TFICF, and TFIDFICF then converted into Cartesian coordinates and calculated similarities using the Cosine Similarity function to analyze the resemblance of text documents by obtaining similarity by measuring it in vector space model. Cosine Similarity value from those weighting scheme (CoSimTFIDF, CoSimTFICF, CoSimTFIDFICF) to be setup as a set of features for the classification process. Next is the process of classifying the text using the K-Nearest Neighbor (K-NN) method for document classification and predicting new document categories based on those features. This study aims to identify and evaluate text similarity using TFIDFICF (Class Indexing Based) method and Cosine Similarity.

Relevan research conducted [6] by utilizing TF.IDF.ICF for e-complaint classification of students using Centroid Based Classifier, combined with TF.IDF.ICF, Cosine Similarity, and Class Feature Centroid. [7] Categorize creative ideas on a company using K-NN and TF.IDF.ICF algorithms. [8] Classifying SambatOnline complaint of Malang City using K-NN algorithm, Cosine Similarity and Chi-Square than TFIDF. [9] Using the K-NN algorithm and TFIDF feature selection, and Categorical Proportional Difference (CPD). The same dataset is used [10] by employing the NW-K-NN algorithm, the term weighting TFIDF filter N-Gram, and Unigram on preprocessing. The experimental results [11] show that the classification of text can be/is used to evaluate the quality of service with the data text of handling a customer complaint (complaint). This method can solve the automatic evaluation problem in customer complaint handling management. [11]

2. METHOD

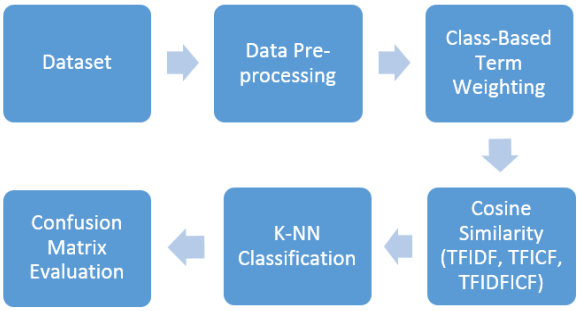

Figure 1 Document Classification Framework

Generally the problem-solving framework can be seen in Figure 1, which consists of Data Collecting (Dataset), Text Preprocessing, Text Representation, Feature Selection (Feature Selection) includes common term weighting scheme (TFIDF) and Class-Based Indexing (ICF), K-NN classification, and Confusion Matrix evaluation.

This study uses three weighting schemes for comparison and evaluation to obtain weightings that have the most optimal results. Tests conducted are experiments on the process of Pre-processing that is through the sub-process Stemming and not using Stemming. Experiments with different term weightings using TF-IDF, TF-ICF, and TF-IDF-ICF along with Cosine Similarity variation experiments based on each term weighting. Experiments with variations in the amount of data, and variations in ratio of training data and data testing. Then on the final result will be evaluated the effect of both performances.

The method used to analyze the similarity between the newly reported incoming reports (Query) and the report that the administrator has processed (Document) is Cosine Similarity. The term-weighting results with TF-IDF, TF-ICF and TF-IDF-ICF are then converted into Cartesian coordinates and calculated using the Cosine Similarity function to obtain the angle of similarity and measure the vector distance. The textual classification process is based on the Cosine Similarity feature of TF-IDF (CoSimTFIDF), TF-ICF (CoSimTFICF), TF-IDF-ICF (CoSimTFIDFICF) using different weighting schemes. The greater the value of the three cosine similarity features that are close to the value of 1 (one), then the more like a Query (q) with Document collection (d). Method K-Nearest Neighbor (KNN) chosen to classify and predict the category of the Query.

2.1 Class-Based Indexing

A category-based term weighting scheme is proposed [12]. This research introduces Frequency Category Reverse *(Inverse Category Frequency)* in the term weighting scheme for text classification tasks. Two concepts are defined as *Class Frequency (CF)* is the number of categories in which the term *(t)* appeared and *Inverse Class Frequency (ICF)* whose formula is similar to IDF [12]. The next *Class-Based Indexing (ICF)* concept was developed by [13]. *Inverse Class Frequency (ICF)* pay attention to the occurrence of terms in the category/class set. Term rarely appears in many classes is a term that is valuable for classification. The less the occurrence of the term, the value will be greater or closer to the value of 1 (one), and conversely the more the occurrence of the term is the value of smaller or close to the value of 0 (zero). The importance of each term is assumed to have a proportion that is in contrast to the number of classes containing term. Accurate indexing also depends on the term importance of the class or the scarcity of terms in the whole class *(rare term)*. So we need class-based term weighting called inverse class frequency (ICF). However, ICF only takes into account the terminology of the class regardless of the number of terms in the document into the class. In this research we use traditional TF-IDF-ICF [13] The following formula of ICF is calculated by the formula:

$$ICFLog_i = Log_2\left(\frac{|c|}{cf_i}\right) \quad (1)$$

Where C is the number of uh classes/categories in the collection (cf_i) is the number of classes/categories containing terms.

In classical VSM, which relies on document indexing, the digital representation criteria of the text in the document vector are the product of local parameters (frequency terms) and global parameters (IDF), ie TF.IDF. In the category of tasks correspond to the class frequency, term weighting scheme, the ICF (categorical global parameter) is multiplied by TF.IDF, generating TF.IDF.ICF. And the formula of traditional TF.IDF.ICF shown on equation (2).

$$W_{TF.IDF.ICF}(t_i, d_j, c_k) = tf_{(t_i,d_j)} \times \left(1 + \log\frac{D}{d(t_i)}\right) \times \left(1 + \log\frac{C}{c(t_i)}\right) \quad (2)$$

Where C denotes the number of categories defined in the collection, $c(t_i)$ is the number of categories in the collection where it occurs at least once, $c(t_i)/C$ is known as CF, and $C/c(t_i)$ is the ICF of term t_i.

2.2 Cosine Similarity Measure

$$\cos = \frac{\boldsymbol{Q} \bullet \boldsymbol{D}}{|\boldsymbol{Q}||\boldsymbol{D}|} = \frac{\sum_{i=1}^{n} \boldsymbol{Q}_i \; X \; \boldsymbol{D}_i}{\sqrt{\sum_{i=1}^{n}(\boldsymbol{Q}_i)^2} \; X \; \sqrt{\sum_{i=1}^{n}(\boldsymbol{D}_i)^2}} \quad (3)$$

Where Q denote the vector of documents, D is the query vector. Q • D is the multiplication of dot vectors Q and vector D it's obtain inner product. |Q| is the length of vector Q (Magnitude of Q) while |D| is the length of vector D (Magnitude of D) then |Q||D| is the cross product between |Q| and |D|. The weight of each term in a document is non-negative. As a result the cosine similarity is non-negative and bounded between 0 and 1. Cos (Q_iD_i) = 1 means the documents are exactly similar and the similarity decreases as the value decreases to 0. An important property of the cosine similarity is its independence of document length. Thus cosine similarity has become popular as a similarity measure in the vector space model [14]

2.3 Preparation and Data Processing

In this study, main dataset using published "LAPOR!" complaint stream data that published on public data sharing portal http://data.go.id. This data can be freely downloaded at the open government data sharing *(Indonesia Open Government)*. This study uses several experimental scenarios, one of which is the dataset variation shown in Table 3. This scenario aims to investigate the effect of the number of rows of data on related processes.

Table 3 Partition Table Data Document (D) and Query (Q)

Dataset Series	90% (D)	10% (Q)	Amount of Data
Dataset100	90	10	100
Dataset200	180	20	200
Dataset300	270	30	300
Dataset400	360	40	400

In experiment, this research is done *Data Partition* or data partition. This is done by dividing the total number of data rows in the dataset on table 3 into two parts: **1) Dataset *Documents (D)*** employ 90%, **2) Dataset *Query (Q)*** use 10% as in table 3. After the process of Preprocessing with stemming and without stemming the dataset is further divided into two parts, ie 90% for the document dataset (D) and 10% used for the query dataset (Q). Data sharing is also done randomly *(random sampling)* thus obtained members dataset in table 3

3. RESULT AND DISCUSSION

3.1 Comparison of Cosine Similarity Based on Term Weighting

After a series of manual cosine similarity between TFIDF, TFICF, and TFIDFICF we can see the result of cosine similarity compared to figure 2. And in cosine simiality

calculation it is found that document recommendation result based on Cosine Similarity value with TFIDF weighting scheme, TFICF, and TFIDFICF is document D5 with the largest cosine value is 0.705 or 70.5% based on TFICF weighting .

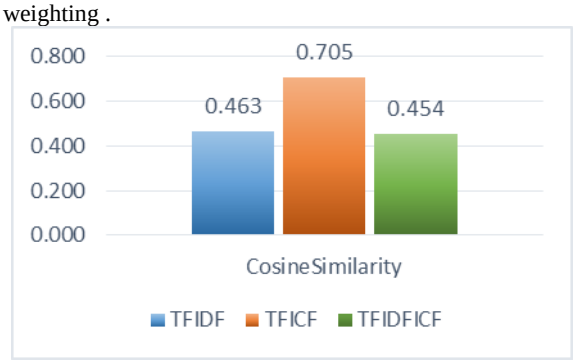

Figure 2 Chart of Cosine Similarity Based on the Term Weighting Scheme

2) Experimental Results Variation Preprocessing

Table 5 Table of Preprocessing Variations Testing Scenarios

Scenario	*Number of D ata*		*Evaluation Results (%)*			
Term Weighting	***Train***	***Test***	***A***	***P***	***R***	***F1***
TFIDF	***75***	***25***	**84.00**	**30.30**	**33.30**	**31.73**
TFICF	***75***	***25***	80.00	24.20	32.40	27.71
TFIDFICF	***75***	***25***	80.00	26.00	32.40	28.85
TFIDF	***90***	***31***	46.15	17.71	19.82	18.71
TFICF	***90***	***31***	**58.06**	**31.88**	**21.13**	**25.42**
TFICFICF	***90***	***31***	45.16	18.14	17.21	17.66

In this experiment aims to evaluate the performance of term weighting Class Indexing Based (TFIDFCF) compared to TFIDF term weighting performance, and TFICF. The dataset used is Dataset200, with 75% training data comparison ratio and 25% data testing. Using six categories and then labeled *(class)* as another name *(alias) is* shown in the following table:

Here are experimental results based on testing with variations of preprocessing stemming and without stemming. The evaluation used is Accuracy (A), Precision (P), Recall (R) and F1-Measure (F1) using macro average model, it is used considering multi-class classification .

Figure 5 Graph of evaluation testing by stemming process (in percent)

3) Variation Testing Weight Feature

Table 6 Test Results Weight TFIDF Feature

Evaluation	**Number of Datasets**			
	100	**200**	**300**	**400**
A	69.23	**84.00**	60.53	63.33
P	18.89	**30.30**	17.36	21.22
R	18.89	**33.33**	18.89	21.15
F	18.89	**31.70**	6 PM	18.47

Based on the results listed in table 6 , the best accuracy for CoSimTFIDF is at Dataset200 which is 84% with F-Measure 31.70%, then Dataset100 with 69.23% accuracy but F-Measure value is quite low at 18.89%, while the highest F-Measure values obtained from Dataset200 obtained the best K-NN classification results for the classification of complaint reports with CoSimTFIDF features. The precision and recall values in Dataset200 also show results with the highest values among other datasets.

Table 7 Test Results Weight TFICF Feature

Evaluation	**Number of Datasets**			
	100	**200**	**300**	**400**
A	76.92	**80.00**	71.05	61.67
P	12.82	24.17	**28.24**	13.57
R	16.67	**32.41**	20.37	19.96
F	14.49	**27.68**	19.72	16:00

Based on the results listed in table 7 , the best accuracy value for CoSimTFICF is on Dataset200 which is 80% with F-Measure 27.68%, then Dataset100 with 76.92% accuracy but F-Measure value is quite low ie 14.49%, Dataset200 obtained the best K-NN classification results for the classification of complaint reports with CoSimTFICF features. The precision also shows good results with 24.17% and the recall value of Dataset200 also shows the highest value of the other datasets of 32.41%.

Table 8 Test Results Weight TFIDFICF Feature

Evaluation	**Number of Datasets**			
	100	**200**	**300**	**400**
A	69.23	**80.00**	60.53	63.33
P	12.50	**25.99**	22.80	19.52
R	3pm	**32.41**	19.63	21.15
F	13.64	**28.7**	18.86	18.86

Based on the results listed in table 8 , the best accuracy for CoSimTFIDFICF is on Dataset200 which is 80% with F-Measure 28.7%, then Dataset100 with 69.23% accuracy but F-Measure value is low ie 13.64%, Dataset200 obtained the best K-NN classification results for the classification of complaint reports with CoSimTFICF features. The precision also shows good results with 25.99% and the recall value in Dataset200 also shows the highest value among other datasets of 32.41%.

4) Accuracy of Classification Process Based on Value k

The experiment uses Dataset200 with a preprocessing process using Stemming and CoSimTFIDF features.

Table 9 Accuracy (%) KNN Based on Value k

Value k Ratio	***1***	***2***	***3***	***4***	***5***
25/75	75.71	80.00	78.57	80.00	**80.00**
75/25	**83.33**	**83.33**	**83.33**	**83.33**	**83.33**
40/60	67.86	80.36	82.14	80.36	**82.14**
60/40	76.32	**84.21**	**84.21**	**84.21**	**84.21**

From the test, the result of K-NN algorithm accuracy with the test of 60% training ratio and 40% test data has a high accuracy level seen from the result of k = 1 is quite low, but increased 8% when testing k = 2 value until k-5 with a stable and equal value of 84.21%, while in this test found that the ratio of 75% training data and 25% test data resulted in an accuracy of 83.33% lower 6.7% of experiments with variations of preprocessing at Table 5.39 is 84% . This is very possible because sampling of trainer data and test data used is

random sampling method. In this pen can be seen that with the value k = 5 all variations of the ratio of training data and test data has maximum value than other k values. Thus it can be concluded based on the test that has been done that the value k = 5 is the optimal value

5) Results Comparison of KNN Accuracy Based on Features and Dataset

Figure 6 KNN Accuracy Based on Features and Dataset

The experiment was done by determining the dataset used ie Dataset100 and Dataset200. With a comparison ratio of 75% training data and 25% (25/25) of data testing. The k value used is k = 5. In the first test feature used only Cosine Similarity feature based on TFIDF term weighting, then in the next test used Cosine Similarity feature based on TFICF, next CoSimTFIDFICF. The best accuracy result has been achieved using TFIDF-based Cosine Similarity (CoSimTFIDF) feature that 84% in Dataset200 increased 4% from both Cosine Similarity TFICF and TFIDFICF features. While on Dataset100 obtained the best accuracy value using CoSimTFICF feature that is 76,92% increase about 6% from both other features

6) Accuracy Results With Variations of Data Ratios

In this experiment using Dataset200 with variations of preprocessing process using stemming and without stemming. As has been found in Table 5.43 where the value of K that has optimal results is k = 5, then set in this test the classification of K-NN using the value k = 5. Comparative ratio of trainee data and test data for various results. The following is the result of accuracy testing based on the ratio of data and features in table 5.45

Table 10 Accuracy On Term Weighting variations

Pre processing	Ratio (%)	Accuracy (%)		
		CoSim TFIDF	CoSim TFICF	CoSim TFIDFICF
With Stemming	25:27	66.67	66.67	66.67
	75:25	**84.00**	**80.00**	**80.00**
	40:60	66.67	78.33	71.67
	60:40	60.00	75.00	70.00
No Stemming	25:27	54.50	64.86	60.36
	75:25	54.05	55.41	55.41
	40:60	52.81	60.67	60.67
	60:40	55.46	57.98	57.98

Based on these results it was found that the optimum accuracy result with preprocessing Stemming and best result of all features is 75% training data ratio and 25% test data on TFIDF feature-based Cosine Similarity that is 84%. Then CoSimTFICF feature with 40% training data ratio and 60% test data

4. CONCLUSIONS

A. Conclusion

In the test results that have been carried out, it was found that **1)** S kem *term* terming TFIDF has a significant influence on the accuracy of classification. **2)** Tests with variations of stemming process using TFIDF-based *Cosine Similarity* feature (CoSimTFIDF) by employing 75 training data and 25 test data resulted in the best K-NN algorithm accuracy of 84%, with 30.3% precision, 33.3% recall, and f-measure 31.73%. This result is better 35% than a preprocessing without stemming is about 58%. **3)** The test to investigate the values of k = 1,2,3,4 and 5 with 100 data of train and test with variation of training data ratio and different test data is value k = 5. The best accuracy value obtained is the ratio of 60:40 that is 84.21%.

B. Suggestion

Some things that can be developed for further research in the same scope include: **1)** We recommend that the addition of variations of preprocessing ie stopword list different languages eg Sundanese, Basaha Java, Slang / Slang and so forth. **2)** *Cross Validation* techniques should also be used to obtain the ratio of training data and proportional K-NN test data .

5. REFERENCES

[1] A. Sofyan And S. Santosa, “Text Mining Untuk Klasifikasi Pengaduan Pada Sistem Lapor Menggunakan Metode C4.5 Berbasis Forward Selection,” *Cyberku J.*, Vol. 12, No. 1, Pp. 8–8, 2016.

[2] I. Surjandari, “Application of Text Mining for Classification of Textual Reports: A Study of Indonesia’s National Complaint Handling System,” in *6th International Conference on Industrial Engineering and Operations Management (IEOM 2016)*, Kuala Lumpur, Malaysia.

[3] A. Fauzan and M. L. Khodra, “Automatic multilabel categorization using learning to rank framework for complaint text on Bandung government,” in *2014 International Conference of Advanced Informatics: Concept, Theory and Application (ICAICTA)*, 2014, pp. 28–33.

[4] S. Tjandra, A. A. P. Warsito, and J. P. Sugiono, “Determining citizen complaints to the appropriate government departments using KNN algorithm,” in *2015 13th International Conference on ICT and Knowledge Engineering (ICT Knowledge Engineering 2015)*, 2015, pp. 1–4.

[5] W. H. Gomaa and A. A. Fahmy, “A Survey of Text Similarity Approaches,” *Int. J. Comput. Appl.*, vol. 68, no. 13, pp. 13–18, 2013.

[6] M. A. Rosid, G. Gunawan, and E. Pramana, “Centroid Based Classifier With TF – IDF – ICF for Classfication of Student’s Complaint at Appliation E-Complaint in Muhammadiyah University of Sidoarjo,” *J. Electr. Electron. Eng.-UMSIDA*, vol. 1, no. 1, pp. 17–24, Feb. 2016.

[7] R. R. M. Putri, R. Y. Herlambang, and R. C. Wihandika, “Implementasi Metode K-Nearest Neighbour Dengan Pembobotan TF.IDF.ICF Untuk Kategorisasi Ide Kreatif Pada Perusahaan,” *J. Teknol. Inf. Dan Ilmu Komput.*, vol. 4, no. 2, pp. 97–103, May 2017.

[8] C. F. Suharno, M. A. Fauzi, and R. S. Perdana, "Klasifikasi Teks Bahasa Indonesia Pada Dokumen Pengaduan Sambat Online Menggunakan Metode K-Nearest Neighbors (K-NN) dan Chi-Square," *J. Pengemb. Teknol. Inf. Dan Ilmu Komput. Vol 1 No 10 2017*, Jul. 2017.

[9] N. H. A. Sari, M. A. Fauzi, and P. P. Adikara, "Klasifikasi Dokumen Sambat Online Menggunakan Metode K-Nearest Neighbor dan Features Selection Berbasis Categorical Proportional Difference," *J. Pengemb. Teknol. Inf. Dan Ilmu Komput. Vol 2 No 8 2018*, Oct. 2017.

[10] A. A. Prasanti, M. A. Fauzi, and M. T. Furqon, "Klasifikasi Teks Pengaduan Pada Sambat Online Menggunakan Metode N-Gram dan Neighbor Weighted K-Nearest Neighbor (NW-KNN)," *J. Pengemb. Teknol. Inf. Dan Ilmu Komput. Vol 2 No 2 2018*, Aug. 2017.

[11] S. Dong and Z. Wang, "Evaluating service quality in insurance customer complaint handling throught text categorization," in *2015 International Conference on Logistics, Informatics and Service Sciences (LISS)*, 2015, pp. 1–5.

[12] D. Wang and H. Zhang, "Inverse-Category-Frequency based supervised term weighting scheme for text categorization," *J. Inf. Sci. Eng.*, vol. 29, no. 2, pp. 209–225, Dec. 2010.

[13] F. Ren and M. G. Sohrab, "Class-indexing-based term weighting for automatic text classification," *Inf. Sci.*, vol. 236, pp. 109–125, Jul. 2013.

[14] A. Huang, "Similarity Measures for Text Document Clustering," in *Proceedings of the New Zealand Computer Science Research Student Conference*, Christchurch, New Zealand, 2008, pp. 49–56.