Combination of Cosine Similarity Method and Conditional Probability for Plagiarism Detection in the Thesis Documents Vector Space Model

Ristu Saptono, Heri Prasetyo, and Ade Irawan
Department of Informatics,Universitas Sebelas Maret, Surakarta, Indonesia.
ristu.saptono@staff.uns.ac.id

***Abstract*—Plagiarism is one of negative impact derived from the internet growth. It can take place in various place, one of the examples is higher education environment. Plagiarism can cause many disadvantageous to other parties. So, there must be a detection system to avoid this kind of bad thing. In this proposed research, there will be made a plagiarism detection system by implementing Vector Space Model (VSM). Cosine Similarity used to make the rank of the paragraphs based on the formed angle from query vector and collection vector. The number of the taken words from the query paragraph will be derived from the calculation of the conditional probability value. After testing phase has been finished, there will be a conclusion that VSM can be implemented in the system. There are 10 testing paragraphs that compared with the collection paragraphs. The best result shows from threshold 0.3 for the conditional probability and 0.2 for cosine similarity with 54.28% for the average precision and 100% for the average recall.**

***Index Terms*—Conditional Probability, Cosine Similarity, Plagiarism, Vector Space Model.**

I. INTRODUCTION

Information and communication technology especially internet, growth significantly year to year. The information access from one place to another one become very quickly and easily. This can bring positive or negative impact. One example of the negative impact is the plagiarism. Plagiarism defined as the act of plagiarizing or copying the others works such as ideas, writing ideas, then claim it as a result of his own work without including reference of the original source [1].

Plagiarism can occur in various places for example in high education environment. Undergraduate students are required to make a thesis as the degree acquisition requirement. The big amount of information related to the thesis material makes the students get the material easily without changing it by using copy and paste facilities. It can lead to the rampant plagiarism. It can also cause many disadvantageous to other parties [2]. Therefore, there must be a system that can detect plagiarism.

One method that can be embedded to the plagiarism system is Vector Space Model (VSM) that will represent the document to be vector in the vector space. The vectors then measured as the proximity value. One method to measure those values is Cosine Similarity [3]. Inside the research by [4] VSM can be utilized well and give the better result than the previous research. Stemming by using Nazief-Adriani algorithm in preprocessing step done before doing VSM. It gives the better result than the system without stemming [5].

Based on the previous research, in this proposed research will be implemented VSM for detecting plagiarism. Paragraphs that derived from parsed document will be turned into the query and it will be compared with the available paragraph in database. Nazief-Adriani algorithm is used in stemming process inside preprocessing phase. The weighting of TF-IDF used for giving the term weight. The term weight of the query paragraph and the collection paragraph then represented into vectors and then the proximity will be measured by using the Cosine Similarity method. The result then ranked in descending order. The last step is counting the taken words percentages from query against the collection paragraphs by using Conditional Probability.

II. LITERATURE REVIEW

A. Plagiarism

Plagiarism originated from Latin “plagiarus” which means kidnapping. The definition of plagiarism according to Big Indonesian Dictionary is "plagiarism that infringes copyright". Meanwhile, according to [1] plagiarism is the act of copying or stealing the others works such as ideas, writing ideas, then claim it as a result of his own work without including reference of the original source.

According to Parvati in [6], the type of plagiarism is divided into 4 such as [1]:

a) Word-for-word
   Each word is copied exactly without any changes.
b) Plagiarism of authorship
   The name of the author is changed to his own name and then acknowledges the work to be his work.
c) Plagiarism of Ideas.
   Ideas from others are recognized as his ideas.
d) Plagiarism of Sources
   The source is not written on the work using the quotation.

According to Sastroasmoro in [7], plagiarism based on the percentage of words taken or traced is divided into 3 categories, such as:

a) Light Plagiarism: < 30%.
b) Medium Plagiarism: 30% - 70%.
c) Heavy Plagiarism: >70%.

B. Text Preprocessing

Text Preprocessing is the way of transforming the existing data form in this research into smaller one so that the existing data is ready to be processed to the next stage [8]. There are some steps of text preprocessing:

a) Document Parsing
Document breaking is the stage where the existing document is broken down into paragraphs.
b) Case Folding
Case Folding is the stage done to change all the words in the text into lowercase.
c) Tokenizing
Tokenizing is a step undertaken to separate every word in text. Each word is then referred to as a token.
d) Filtering
Filtering is the stage where each token of the previous process is filtered so that only relevant words or tokens are obtained. In the meantime, irrelevant tokens are omitted.
e) Stemming
Stemming is a process done to get the root word of every word. The process is done by removing the prefixes and affixes contained in a word.
f) Indexing
Indexing is a process done to build an index database of document collections.

C. Nazief Adriani Algorithm

Inside [9] was explained that Nazief Adriani algorithm is a stemming algorithm that often used in the information retrieval for Indonesian language documents. In the stemming process, the algorithm used will be different from one to the others depending on the language used. This is because the structure and form of words in the language used is not the same. In Indonesian text documents, the process will be more difficult because of the removal of various types of affixes to obtain the root word in the document.

D. TF-IDF Weighting

Weighting TF-IDF is the process of calculating the weight value of a word that indicates the importance of the word on the document in a collection [10]. The calculation result is obtained by multiplying the value of TF with IDF value according to the following Equation [11].

$$W_{t,d} = TF_{t,d} \times ln\left(\frac{N}{df_t}\right) + 1 \quad (1)$$

where $W_{t,d}$ is the value of the weight of the word t in document d. The value of $TF_{t,d}$ is the frequency of the word t in document d. N is the total document and df_t is a lot of documents containing the word t.

E. Vector Space Model

Vector Space Model (VSM) is a model developed by Gerald Salton on the information retrieval (IR) system. In this model each document that belongs to the collection and query document will be represented in a vector in the vector space [12]. The vector consists of the word index (term index). Where weight will be given to those words [13].

The illustration of VSM can be seen in Figure 1.

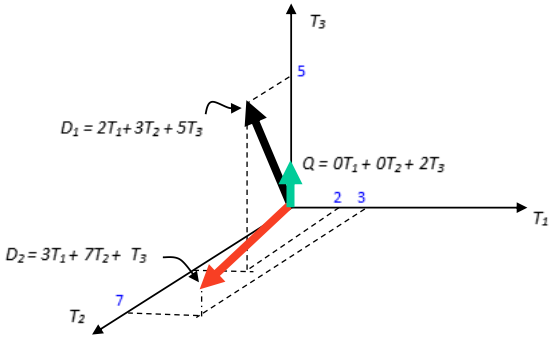

Figure 1: Document Representation and Vectors in the Vector Space [14]

In the Vector Space Model a collection of documents can also be represented in the matrix. Representation of the matrix can be seen as follows:

$$\begin{array}{c} \\ D_1 \\ D_2 \\ \vdots \\ \vdots \\ D_m \end{array} \begin{array}{c} \begin{array}{cccc} T_1 & T_2 & .... & T_n \end{array} \\ \begin{pmatrix} w_{11} & w_{21} & \dots & w_{n1} \\ w_{12} & w_{22} & \dots & w_{n2} \\ \vdots & \vdots & & \vdots \\ \vdots & \vdots & & \vdots \\ w_{1m} & w_{2m} & \dots & w_{nm} \end{pmatrix} \end{array}$$

Figure 2: Term-Document Matrices [14]

The word or term is denoted by T where n is the number of words so that T = (T_1, T_2, ..., T_n). Document denoted by D with m is the number of documents so D = (D_1, D_2, ..., D_m). As for w_{nm} is the weight of the word in the document m [14].

F. Cosine Similarity

Cosine Similarity is a method for measuring the level of similarity between two vectors. Calculations in this method are done by calculating the Cosine value between two vectors [13]. Here is the Cosine Similarity formula:

$$Sim(Q, D_i) = \frac{\sum_{j=1}^{V} w_{Q,j} \times w_{i,j}}{\sqrt{\sum_{j=1}^{V} w_{Q,j}^2 \times \sum_{j=1}^{V} w_{ij}^2}} \quad (2)$$

where Q is Query, D_i is document i, $W_{Q,j}$ is the weight of j term in Q query, and the weight of j term in i^{th} document.

If the calculated value derived from Cosine Similarity method is bigger and close to 1, so it can be said that two vectors have high similarity.On the other hand, if the calculated value is smaller and close to 0, so it can be said that two vectors have low similarity. The calculation value range starts from 0 until 1. Value 0 if the two vectors on the calculation are not at all the same. While the value of 1 if both vectors are the same [15].

G. Conditional Probability

Conditional Probability is the probability value of occurrence A occurs on condition B has occurred [16]. Conditional Probability is formulated as follows:

$$P(A|B) = \frac{P(A \cap B)}{P(B)} \quad (3)$$

where P (A∩B) is the intersection of opportunity A and B and (B) represents Opportunity B.

III. RESEARCH METHODOLOGY

A. Data collection

In this study, the data used is the thesis documents of Informatics undergraduate student Sebelas Maret University Surakarta. Data obtained through UNS central library repository that located at http://digilib.uns.ac.id.

B. Preprocessing dan Indexing Library

Preprocessing is the initialization process of input data processing in this research. Input data in the form of collection documents and query documents from the user will be preprocessed to convert the text document data into smaller data form. Preprocessing in the document in this study consists of several steps such as paragraph breaking, case folding, tokenizing, filtering, and stemming. After preprocessing has been done, the next process is term indexing, term weighting, and weighted indexing term. Weights are calculated using TF-IDF weighting. The TF-IDF weighting will consider the frequent occurrence of term in the document and in the corpus.

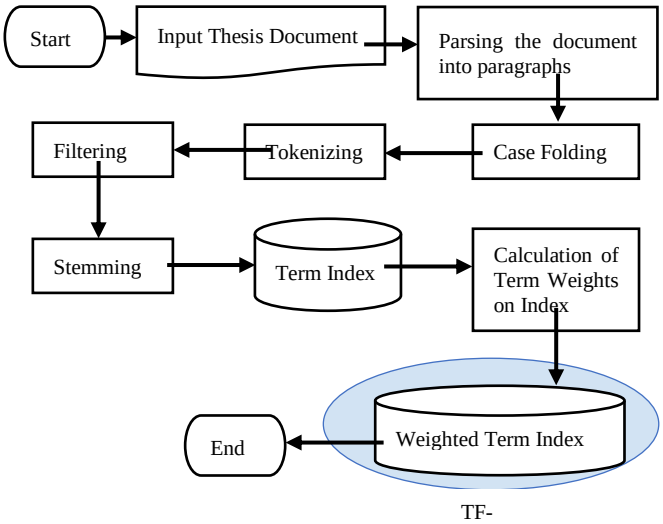

Figure 3: Preprocessing and Indexing Library Phase

The preprocessing and indexing stage of the document is shown in Figure 3.

C. Vector Space Model Implementation

In this research the implementation phase of the Vector Space Model is used to measure the similarity between the paragraph belongs to the library document with the query paragraphs entered using the value from the angle formed by the vector paragraph library with the vector of the query paragraph. Vector Space Model was chosen because in the previous research by [4] VSM gave the best result. The first step is doing preprocessing of the query document. The preprocessing steps of the query document are similar to preprocessing against the collection document. Furthermore, preprocessing result then calculated by using TF-IDF weighting. The weight of the words query then calculated by multiplying the TF value of the query by the weighted IDF value in the indexing of the collection document. The next process is representing the weight of words in the query paragraph and the weight of words in the collection paragraph into a vector. The vectors similarity degree then calculated by counting the formed angle by using Cosine Similarity method.

The results of Cosine Similarity calculations then ranked by sorting the results of descending calculations. The last stage is to count the many words taken by the query paragraph against the paragraphs in the library. The calculation is done by using Conditional Probability theory.

The implementation steps of the Vector Space Model are illustrated in Figure 4.

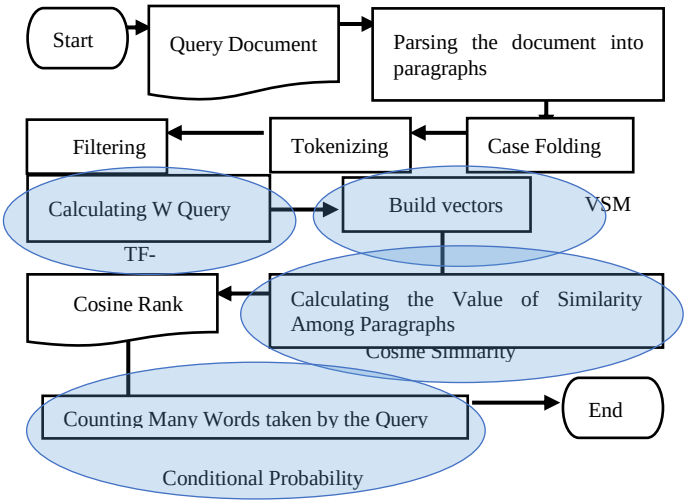

Figure 4: VSM Implementation Phase

D. System Implementation

The system in the research will be implemented by using Java programming language. The database used is MySQL.

E. Testing and Analysis of Results

In this study, the purpose of testing is to know whether good or not the method used in this study. Measurements are made by calculating the recall and precision values.

a) Recall

Recall is a comparison between the number of relevant documents found by the number of all relevant documents in the collection [17].

The formula of Recall is written as follows:

$$Recall = \frac{|\{relevant\ documents\}| \cap |\{retrived\ documents\}|}{|\{relevant\ documents\}|} \quad (4)$$

b) Precision

Precision is a comparison between the number of relevant documents found by the total number of documents found [17].

Precision is mathematically written as follows:

$$Precision = \frac{|\{relevant\ documents\}| \cap |\{retrieved\ documents\}|}{|\{retrieved\ documents\}|} \quad (5)$$

IV. RESULTS AND DISCUSSION

In this study, the data will be collected in the database taken from digilib.uns.ac.id. Total data obtained is 160 pdf documents. The data is a thesis document from Informatics undergraduate students UNS from the year 2007 - 2012 in Chapter II. Details of data obtained are in Table 1.

Table 1
Details of data

No	Year	Amount
1	2007	19
2	2008	42
3	2009	46
4	2010	25
5	2011	19
6	2012	9
Total Amount		160

Before entering the VSM step, there will be done data preprocessing text. The results of text processing consist of paragraph parsing, Case Folding, Tokenizing, Filtering, Stemming, and Indexing resulted in 10131 paragraphs with index term built as much as 230932 terms.

The VSM implementation process will calculate the angle values formed between the vectors query and the collection vectors by using Cosine Similarity. The amount of the taken paragraphs used for the test are 10 paragraphs that derived from collection documents.

After the cosine value has been calculated, there will be calculating process for the percentage of words taken from the collection paragraph using conditional probability. Based on the calculation results then the precision value and recall value will be calculated. Before doing the testing process, the Threshold value is given to Cosine Similarity and Conditional Probability.

This is the calculation result from one of the testing paragraphs. Based on the calculation results from Table 2, there are two paragraphs with high similarity, they are p7328 and p3947. The other paragraphs, some of them are irrelevant with the query paragraph but they are retrieved, for example paragraphs p3259 and p3263. Both of them have high conditional probability values, but they are not included in the relevant paragraph. It is caused by the parsing error from the document to be paragraph. This is because there are few mistakes when parsing process from a document into paragraphs.

Table 2
Cosine and conditional calculation results

No	Paragraph	Cosine	Conditional	Document
1	p7238	1	1	d123
2	p3947	0.756	0.7	d63
3	p3259	0.108	1	d52
4	p3263	0.108	1	d52
5	p9299	0.108	1	d157
6	p3260	0.108	1	d52
7	p3262	0.108	1	d52
8	p9309	0.108	1	d157
9	p9313	0.108	1	d157
10	p926	0.106	0.304	d14
11	p3943	0.101	0.313	d63

There are four testing scenarios. The conditional threshold is set at 0.3 for all scenarios. While there are 4 values of the cosine threshold such as 0.25 on scenario 4, 0.2 on scenario 3, 0.15 on scenario 2 and 0.1 on scenario 1. Recall and precision calculation results for all scenarios are shown in Table 3.

Table 3
Calculation results from all of the scenarios

Par	Scenario 1		Scenario 2		Scenario 3		Scenario 4	
	Prec (%)	Rec (%)	Prec (%)	Rec (%)	Prec (%)	Rec (%)	Prec (%)	Rec (%)
p1630	18.6	100	30.8	100	40.0	100	53.3	100
p2507	13.3	100	25.0	100	33.3	100	80.0	100
p3701	36.1	100	43.3	100	46.4	100	56.5	100
p5542	70.0	100	70.0	100	77.8	100	87.5	100
p9324	31.8	100	53.8	100	100.0	100	100.0	85.7
p5486	16.9	100	36.7	100	40.7	100	52.4	100
p6974	23.1	100	30.0	100	37.5	100	39.1	100
p7238	18.2	100	100.0	100	100.0	100	100.0	100
p7620	14.8	100	21.1	100	28.6	100	40.0	100
p5586	18.5	100	26.3	100	38.5	100	41.7	100

Based on the calculation of precision and recall shown in Table 3, the threshold value that will be used is the value in scenario 3, that is 0.3 for Conditional Probability and 0.2 for Cosine Similarity. At the threshold the average precision value is 54.28% and the average recall is 100%. This threshold is chosen because the averages of precision and recall are higher than the averages of them in scenario 1and 2. The threshold in scenario 4 is not used because the average of recall has decreased, less than 100%, although the average of precision is high.

V. CONCLUSION

Based on the research that has been done then it can be concluded that Vector Space Model with Cosine Similarity and Conditional Probability can be implemented in plagiarism detection system. The test yields a threshold value of 0.3 for Conditional Probability and 0.2 for Cosine Similarity. The average precision and recall obtained with the threshold is 54.37% and 100%.

REFERENCES

[1] S. Dewanto, Indriati and I. Cholissodin, "Deteksi Plagiarisme Dokumen Teks menggunakan Algoritma Rabin-Karp dengan Synonym Recognition".

[2] D. Purwitasari, P. Y. Kusmawan and U. L. Yuhana, "Deteksi Keberadaan Kalimat Sama sebagai Indikasi Penjiplakan dengan Algoritma Hashing Berbasis N-Gram," *Jurnal Ilmiah KURSOR*, vol. VI, no. 1, pp. 37-44, 2011.

[3] L. Alkawero, "Pemanfaatan Metadata dalam Menilai Kesamaan Proposal Penelitian," 2013.

[4] Jovita, Linda, A. Hartawan and D. Suhartono, "Using Vector Space Model in Question Answering System," in *International Conference on Computer Science and Computational Intelligence (ICCSCI 2015)*, 2015, pp. 305-311.

[5] T. Mardiana, T. B. Aji and I. Hidayah, "Stemming Influence on Similarity Detection of Abstract Written in Indonesia," *TELKOMNIKA*, vol. XIV, no. 1, pp. 219-227, 2016.

[6] I. W. S. Priantara, D. Purwitasari and U. L. Yuhana, "Implementasi Deteksi Penjiplakan dengan Algoritma Winnowing pada Dokumen Terkelompok," in *Jurusan Teknik Informatika, Fakultas Teknologi Informasi, Institut Teknologi Sepuluh November Surabaya*, pp. 1-9, 2011.

[7] Herqutanto, "Plagiarisme, Runtuhnya Tembok Kejujuran Akademik," *eJurnal Kedokteran Indonesia* , vol. I, no. 1, pp. 1-3, 2013.

[8] W. E. Waliprana and M. L. Khodra, "Update Summarization Untuk Kumpulan Dokumen Berbahasa Indonesia," *Jurnal Cybermatika*, vol. I, no. 2, pp. 6-10, 2013.

[9] L. Agusta, "Perbandingan Algoritma Stemming Porter dengan Algoritma Nazief & Adriani untuk Stemming Dokumen Teks Bahasa Indonesia," in *Konferensi Nasional Sistem dan Informatika*, 2009, pp. 196-201.

[10] A. Mishra and S. Vishwakarma, "Analysis of TF-IDF Model and its Variant for Document Retrieval," in *International Conference on Computational Intelligence and Communication Networks*, 2015, pp. 772-776,.

[11] T. B. Adji, Z. Abidin and H. A. Nugroho, "System of Negative Indonesian Website Detection Using TF-IDF and Vector Space Model," in *International Conference on Electrical Engineering and Computer Science*, 2014, pp. 174-178.

[12] P. D. Turney and P. Pantel, "From Frequency to Meaning: Vector Space Models of Semantics," *Journal of Artificial Intelligence Research*, pp. 141-148, 2010.

[13] F. Mohammadi, "A New Approach To Focused Crawling: Combination of Text summarizing With Neural Networks and Vector Space Model," *ACSIJ Advances in Computer Science: an International Journal*, Vol. 2, Issue 3, No. 4, pp. 31-36, 2013.

[14] R. Mandala and H. Setiawan, "Peningkatan Performansi Sistem Temu-Kembali Informasi dengan Perluasan Query Secara Otomatis," 2002.

[15] T. M. Isa and T. F. Abidin, "Mengukur Tingkat Kesamaan Paragraf Menggunakan Vector Space Model untuk Mendeteksi Plagiarisme," in *Seminar Nasional dan ExpoTeknik Elektro*, 2013, pp. 229-234.

[16] M. Baron. *Probability and Statistics for Computer Scientists, 2nd ed.*. Richardson: CRC Press, 2014.

[17] A. Indranandita, B. Susanto and A. R. C, "Sistem Klasifikasi dan Pencarian Jurnal dengan Menggunakan Metode Naive Bayes dan Vector Space Model," *Jurnal Informatika*, 2008.