International Journal of Scientific Research in Computer Science, Engineering and Information Technology
© 2020 IJSRCSEIT | Volume 6 | Issue 1 | ISSN : 2456-3307
DOI : https://doi.org/10.32628/CSEIT195672

Analysis of Matric Product Matching Between Cosine Similarity with Term Frequency-Inverse Document Frequency (TF-IDF) and Word2Vec in PT. Pricebook Digital Indonesia

Harni Kusniyati[1], Arie Aditya Nugraha[2]

Faculty of Computer Science, Mercu Buana University, Jakarta Barat, Indonesia

ABSTRACT

Consumers today have the option to purchase products from thousands of e-commerce. However, the completeness of the product specifications and taxonomies used to organize products differently in different electronic shop differently. To improve the consumer experience, Pricebook approach for integration of the product through the website to find the cheapest price from various platforms. In our writing, we do approach by using a model of neural language such as TF-IDF (term frequency-inverse document frequency) as well as Word2vec by using the method of cosine similarity. TF-IDF is a way to give the relationship a word weighting (term) against the document. Semantic vector or word embedding is one way to represent the structure of a sentence will be in align with manipulating sentences into vector shapes with Word2Vec. Cosine similarity method is a method to calculate the similarity between two objects that is expressed in two vectors by using keywords (keywords) of a document as the size so that it leads to more products matching good performance and categorization. In addition, we compare the results of the representation of the TF-IDF with Word2vec against a number of the data.

Keywords : Product Matching, Cosine Similarity, Tf-Idf, Word2vec.

I. INTRODUCTION

In recent years, with new anyway that is from internet services and e-commerce, the number of products sold through e-commerce has been growing rapidly. This new research estimates that total retail e-commerce sales for the year 2017 in Indonesia only 7.059 billion [1]. However, there is one major problem in the process of launching products and purchases that must be faced. The same product can be found on many e-commerce fruit, but information about product offers a very wide range of fruit in e-commerce. In addition to that, there is no global identifier for the product, and offer most often not mutually related to each other.

Price comparison site Pricebook provides provider information about the price range based on product specifications, as well as the lowest price will help online shoppers as well as site visitors are doing price comparisons before they buy these products at the store offline. E-Commerce like Lazada, Blibli and C2C e-commerce such as Tokopedia and Bukalapak illiquid market with hundreds of millions of products and thousands of sellers [2]. Not all prices on e-commerce and the market can get, considering many products

must be in mismatch with the database product. Thus the company tagline determines whether the product on the market or the e-commerce s matches the product that is present in the database.

The fundamental challenges and problems with the matching product or Product Matching is a bit once the correct source. The same products can be found in many different electronics store, but information about the product is very different in the various different electronics store [8]. There are billions of products and there is no absolute structure that determines how the product must be identified in the entire store. UPC (Universal product code) may be able to assist in ease of classification and determination of the product, but most. e-commerce sites don't expose it in the page display their products. This becomes a challenge next i.e. the depth of product information throughout the entire store. Not all stores have a great product naming title, UPC or GTIN, MPN (manufacturer's part number) and other attributes to aid in matching a product. The other main challenge is scale. The company has more than 8 million records of products and to identify the suitability of a given document, (note the product) needs to be compared in all stores.

For example, like the following example: while the products sold the same but naming a smartphone Apple iPhone X Lazada, Tokopedia, Bukalapak, and Blibli.

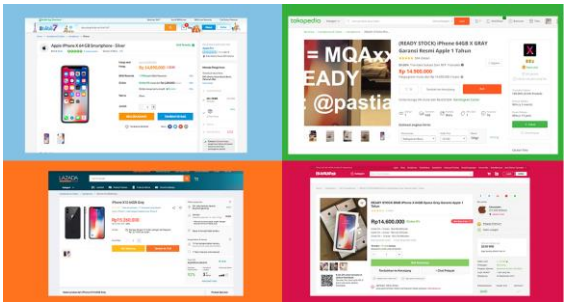

Fig. 1 The difference in the name of iPhone X in each e-commerce

Cosine Similarity method is the method used to calculate the level of similarity (similarity) between two documents. Clustering is defined as the effort data grouped into cluster such that data within the same cluster have more in common than with data on a different cluster. In addition TF-IDF algorithm that can be used to search for the product as a measure of the level of similarities between databases with keywords obtained from the extraction of the text in the name of goods in e-commerce.

In the meantime, to perform the function or similarity matching used Vector Space Model. On the algorithm of vector space model used the formula to find the value of the cosines of angles between two vectors of each document (WD) weights and weights of keywords (WK). The method of calculating the value of proximity between two documents or the suitability of this method of Cosine Similarity.

Notice any problems in doing matching products e-commerce products against products in the database, then the researchers interested in conducting research and developing Product Process Matching with the title "Analytics Metric Products Matching Between the Cosine Similarity with the Term Frequency-Inverse Document Frequency (TF-IDF) and Word2Vec in PT. Pricebook Digital Indonesia ".

II. METHODS AND MATERIAL

The purpose of this study is to implement and analyze the results of Cosine Similarity using TF-IDF and Word2Vec to create vector representations of words with relatively fast time and with a large enough dataset.

Through this research, it is expected to provide benefits:

- For the company, it can facilitate product matching to be faster and more efficient with a high level of similarity.
- For researchers, this application can be a reference for developing applications and research related to text mining.
- For the wider community, this research can be a source of information relating to product matching.

A. *Research Methodology*

Method of Document Frequency-Reverse Terms Frequency is a method of weighting a word (term) on a document. In its use, this method is the concept of Term Term (TF) and Document Frequency (DF). The frequency of the term is a weighting method that calculates the number of occurrences of a word (term) in a document. The Frequency of Inverse Documents is a weighting method that calculates the number of occurrences of a word (term) in each document. The weight of a document that can be generated through the results of the TF and IDF calculations is formulated in the calculation below.

$$W_{dt} = tf_{dt} * IDF_t$$

Where the formula above consists of :

- *d*: document d
- *t*: t-word from the keyword
- *W*: document d weight on t-word
- *tf*: the number of words searched for in a document
- *IDF*: Inversed Document Frequency IDF value is obtained from
- *IDF*: log2 (D / df)
- Where
- D: total document
- *df*: many documents that contain the word searched

After the weight (W) of each document is known, the sorting process is carried out where the greater the W value, the greater the level of similarity of the document to the keyword, and vice versa.

The vector space model is a model used to measure the similarities between documents and requests. In this model, queries and documents are considered vectors in n-dimensional space, where n is the sum of all terms in the lexicon. The Lexicon is a list of all terms in the index. One way to overcome this in the vector space model is to take a vector. The process can be done in a query vector, document vector, or in the second vector. In the vector space model algorithm is used a formula to find the cosine value of an angle between two vectors of each document weight (WD) and the weight of the keyword (WK). The formula used in the space model vector is as follows:

$$Cosinus \rightarrow sim(d_j, q) = \frac{d_j \cdot q}{|d_j| \cdot |q|} = \frac{\sum_i W_{ij} \cdot W_q}{\sqrt{\sum_i W_{ij}^2} \cdot \sqrt{\sum_i W_q^2}}$$

B. *Dataset*

Information :

- Wij: The weight of the word i in the document j
- Wq: Query weight

Calculation of the cosine angle value between these two vectors is known as the Cosine Similarity method. The cosine angle value between two vectors determines the similarity of two objects compared where the smallest value is 0 and the largest value is 1.

The study uses a dataset consisting of product data in 1 months and 3 months of the year 2017. Product data 1 month as much as 27,128 record sorta data for 3 months as much as 88,033 and each record dataset has only 1 product name feature.

C. *Proposed Method*

The proposed method in this study was the implementation method of Cosine Similarity using TF-IDF and Word2Vec. This research was built to generate the value of the similarity between data feeds to the new product names and product data with the matched. The order of the classification method shown in Fig. 2.

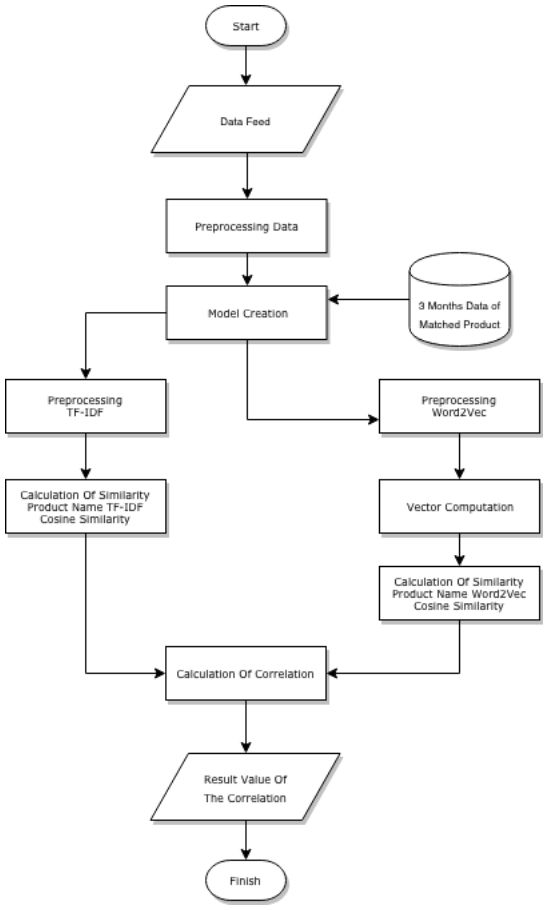

Fig. 2 Block Diagram of the Proposed Method

From Figure 2 above, the General description of the stages of research are as follows:

- The system reads the data input in the form of product data feeds from a variety of e-commerce.
- The system performs preprocessing of data i.e. data cleaning like eliminating unused character.
- Perform System Modelling phases with the help of Python 3.7. This model is used to find out the value of a vector on each product name.
- The system performs a preprocessing stage that is by doing the weighting on each word using the method Word2Vec.
- The system performs vector computing, namely changing the word vector based on models that have been built.
- After the Word2Vec method is successful, the system performs a preprocessing stage again by doing the weighting on each word using TF-IDF.
- The system performs vector computing. The value of the similarity of the name of the product stored in the form of a list and export into a .xls format.
- The system performs the calculation of the correlation of similarities with the previous product.
- In the preparation stage, the dataset is divided into two parts: 80% for training and 20% for testing.

D. *Evaluation*

This research do some evaluation to show the performance of each method. Evaluation of performance including accuracy, precision, as well as the Area under the curve (AUC).

The value of accuracy indicates proximity of measured the results of the actual values. This value describes how accurate the results of the classification process. Precision measures the level to which the result approaching one another, i.e. when measurement of crowding together. Therefore, the higher the precision level, the smaller the variation between measurements.

Area under the Curve (AUC) is a measure of the suitability of the method. AUC represents the value of the sensitivity and specificity with boundary value 0-1. In addition, the results of this evaluation are categorized

Based on the values obtained from each measurement. [21] Gorunescu categorize classification results based on AUC values as follows:

- 0.90 – 1.00 = excellent classification;
- 0.80 – 0.90 = good classification;
- 0.70 – 0.80 = fair classification;
- 0.70 – 0.60 = poor classification;
- 0.60 – 0.50 = failure.

III. RESULTS AND DISCUSSION

In this study, the proposed method is cosine similarity combined with TF-IDF that would be compared to cosine similarity with Word2Vec. The first experiment is to evaluate the accuracy and precision of the TF-IDF and the second experiment is to evaluate the performance method of Word2vec. Each experiment uses two datasets, 1 month and 3 months of product name dataset.

For 1 month data is consist of:
- 27.128 usable data
- After cleaning : 20.659 data
- Train data (80%) : 16.527
- Test data (20%) : 4.132
- TF-IDF : 8.395, Word2Vec : 100

For 3 months data is consist of:
- 88.033 usable data
- After cleaning : 67.091 data
- Train data (80%) : 53.672
- Test data (20%) : 13.419
- TF-IDF : 17.527, Word2Vec : 100

This experiment uses Python 3.7. This experiment uses hardware having the following specifications: Intel Core i5, with 8 GB of RAM.

A. *Similarity Result Using Cosine Similarity + TF-IDF for 1 Month Dataset*

This experiment measures the similarity between two non-zero vectors of inner product space that measures the cosine of the angle between them using product data uses the term frequency-inverse document frequency (TF-IDF) method. The results of the dataset can be seen in Table I and II.

TABLE I. 1 MONTH TF-IDF COSINE DISTRIBUTION

TR: 0.5		**TRUE**	
		+	-
Predicted	+	3356	46
	-	691	39

Table I present the analysis the result of similarity using Cosine Similarity and TF-IDF method. The above values consist of True true (TT), True false (TF), False True (FT), and False False (FF). That the level of similarity for the value of 0.5 down has an accuracy of 0.82 or 82% and for the precision, the level is 0.98 or 98%.

TABLE II. 1 MONTH TF-IDF COSINE DISTRIBUTION

TR: 0.6		**TRUE**	
		+	-
Predicted	+	3242	160
	-	613	117

While the level of similarity for the value of 0.6 down has an accuracy of 0.81 or 81% and for the precision level is 0.95 or 95% are presented in Table II.

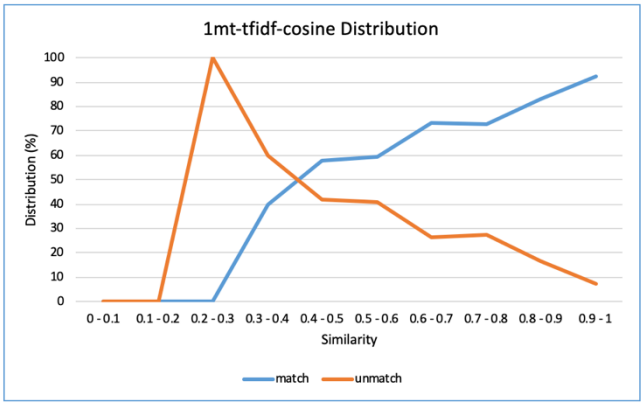

Fig. 3 One month TF-IDF Cosine Distribution

B. *Similarity Result Using Cosine Similarity + Word2Vec for 1 Month Dataset*

By using same method to measures the similarity between two non-zero vectors of inner product space that measures the cosine of the angle between them using product data uses Word2Vec method.

The results of the dataset can be seen in Table III and IV.

TABLE III. 1 MONTH WORD2VEC COSINE DISTRIBUTION

TR: 0.5		TRUE	
		+	-
Predicted	+	2444	2
	-	1669	17

In word2vec the similarity level for the value of 0.5 down has an accuracy of 0.59 or 59% with a precision level of 0.99 or 99%.

TABLE IV. 1 MONTH WORD2VEC COSINE DISTRIBUTION

TR: 0.8		TRUE	
		+	-
Predicted	+	1889	547
	-	1058	628

While the level of similarity for the value 0.8 on average in word2vec has an accuracy of 0.61 or 61% with a precision level of 0.77 or 77%.

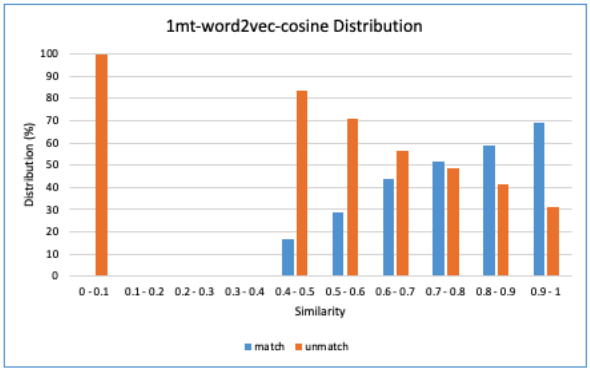

Fig. 4 One month Word2Vec Cosine Distribution

C. *Similarity Result Using Cosine Similarity + TF-IDF for 3 Month Dataset*

The results of the dataset can be seen in Table V and VI.

TABLE V. 3 MONTH WORD2VEC COSINE DISTRIBUTION

TR: 0.5		TRUE	
		+	-
Predicted	+	10872	84
	-	2349	114

The level of similarity for the value of 0.5 down has an accuracy of 0.81 or 81% and for the precision, the level is 0.99 or 99%.

TABLE II. 1 MONTH TF-IDF COSINE DISTRIBUTION

TR: 0.6		TRUE	
		+	-
Predicted	+	10577	379
	-	2114	349

While the level of similarity for the value of 0.6 down has an accuracy of 0.81 or 81% and for the precision level is 0.96 or 96%.

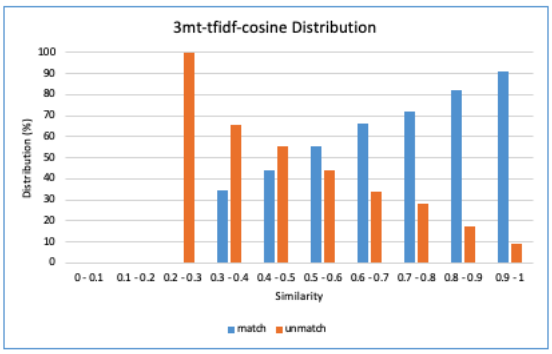

Fig. 5 Three month TF-IDF Cosine Distribution

D. *Similarity Result Using Cosine Similarity + Word2Vec for 3 Month Dataset*

The results of the dataset can be seen in Table VII and VIII.

TABLE VII. 1 MONTH WORD2VEC COSINE DISTRIBUTION

TR: 0.5		TRUE	
		+	-
Predicted	+	8522	6
	-	4853	38

In word2vec the similarity level for the value of 0.5 down has an accuracy of 0.63 or 63% with a precision level of 0.99 or 99%.

TABLE VIII. 1 MONTH WORD2VEC COSINE DISTRIBUTION

TR: 0.8		TRUE	
		+	-
Predicted	+	7070	1458
	-	3124	1767

While the level of similarity for the value 0.8 on average in word2vec has an accuracy of 0.65 or 65% with a precision level of 0.82 or 82%.

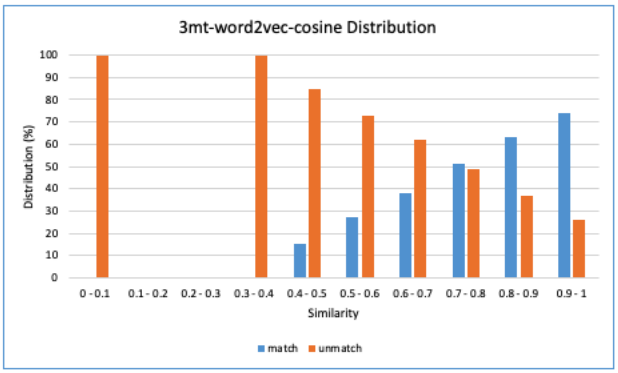

Fig. 6 Three month Word2Vec Cosine Distribution

IV. CONCLUSION

Based on the above research it can be concluded that if the data is still small, the TF-DF method is better than Word2Vec. We have seen that TF-IDF is an efficient and simple algorithm for matching words in a query to documents that are relevant to that query. From the data collected, we see that TF-IDF returns documents that are highly relevant to a particular query.

Word2Vec requires big data for the best results. Finally, in the process Word2Vec requires more time in processing data than using TF-IDF.

V. REFERENCES

[1]. P. Ristoski, P. Petrovski, P Mika. and H. Paulheim, "A Machine Learning Approach for Product Matching and Categorization" Univ. of Mannheim, B6, 26, 68159 Mannheim, 2016

[2]. C. F. Arias, J. Zuniga, G. Sidorov, I. Batyrshin, and A. Gelbukh, "A tweets classifier based on cosine similarity," Instituto Politecnico Nacionalm Mexico City, Mexico, 2017.

[3]. M. Pradeepa and V. Mohanraj, (2016) "Achieving effective keyword ranked search by using TF-IDF and cosine similarity," International Research Journal of Engineering and Technology (IRJET). 124-130

[4]. S. G. Ikwad, P Swarnalatha, and R. Agarwal, "Content Parsing Using Data Mining TF-IDF Algorithm Implementation", Univ. of Vellore, Tamil Nadu, 2016.

[5]. Haxia Liu, "Sentiment Analysis of Citations Using Word2vec," Univ. of Nottingham Malaysia, 43500 Semenyih, Selangor Darul Ehsan, 2017.

[6]. S. Qaiser and R. Ali, "Text Mining: use of TF-IDF to Examine the Relevance of Words to Documents ," International Journal of Computer Application (0975-8887) volume 181-No1 July 2018, pg 25-29.

[7]. S. Brindha, K. Prabha, and S. Sukumaran, "The Comparison of Term Based Methods Using Text Mining," Internatoinal of Computer Science and Mobile Computing, Vol 5 Issue 9, September-2016, pg 112-116.

[8]. M. Long and Z. Yanqing, "Using Word2Vec to Process Big Text Data," 2015 IEEE International Conference on Big Data (Big Data). DOI:10.1109/BigData.2015.7364114

[9]. K. Maher and M. S. Joshi, "Effectiveness of Different Similarity Measures for Text Classification and Clustering," (IJCSIT) International Journal of Compter Science and Information Technologies, Vol 7(4), 2016, 1715-1720.

[10]. D. Gunawan, C. A. Sembiring, and M. A. Budiman, " The Implementation of Cosine Similarity to Calculate Text Relevance between Two Documents," 2nd International Conference on Computing and Applied Informatics 2017, DOI:10.1088/1742-6596/978/1/012120.

[11]. Cherid Anis, "Asymmetric And Symmetric Cryptography To Secure Social Network Media Communication: The Case Of Android-Based E-Learning Software," International Research Journal of Computer Science (IRJCS) Issue 01, Volume 5 (January 2018), pg 1-8

[12]. S. Mujiono, and SK. Purwanto, "The Implementation of E-learning System Governance to Deal with User Need, Institution Objective, and Regulation Compliance". TELKOMNIKA, Vol 16 No 3, June 2018, pp. 1332~1344 DOI:10.12928/TELKOMNIKA.v16i3.8699

[13]. ECOMMERCEIQ. (2017) Top E-Commerce Sites Indonesia. [Online]. Available: https://ecommerceiq.asia/top-ecommerce-sites-indonesia.

[14]. (2016) The APJII Website, [Online] Penetration and behavior of Indonesian Internet Users. Available: https://apjii.or.id/downfile/file/surveipenetrasiinternet2016.pdf

Cite this article as :

Harni Kusniyati, Arie Aditya Nugraha, "Analysis of Matric Product Matching Between Cosine Similarity with Term Frequency-Inverse Document Frequency (TF-IDF) and Word2Vec in PT. Pricebook Digital Indonesia ", International Journal of Scientific Research in Computer Science, Engineering and Information Technology (IJSRCSEIT), ISSN : 2456-3307, Volume 6 Issue 1, pp. 105-112, January-February 2020. Available at doi : https://doi.org/10.32628/CSEIT195672
Journal URL : http://ijsrcseit.com/CSEIT195672