UTILIZING LEXICAL RELATIONSHIP IN TERM-BASED SIMILARITY MEASURE TO IMPROVE INDONESIAN SHORT TEXT CLASSIFICATION

Husni Thamrin[1] and Atiqa Sabardila[2]
[1]Department of Informatics, Universitas Muhammdiyah Surakarta, Jl A Yani, Pabelan, Sukoharjo, Indonesia
[2]Department of Education of Bahasa Indonesia, Universitas Muhammadiyah Surakarta, Indonesia
E-Mail: husni.thamrin@ums.ac.id

ABSTRACT

This paper compares the performance of text similarity algorithms that use pure cosine function and two others that use Dice function and considers word relatedness. Relatedness of two words is determined in a case by looking at lexical relationship, and in another case by looking at the co-occurrences of two words in a corpus. Text similarity score is used in classification of Indonesian short texts using k-nearest neighbour. The study employed more than 150 short texts, of which 112 were used in learning and 43 were used for testing. The short texts were sentences or phrases from a SWOT (strength, weakness, opportunity and threat) analysis of an organization. Manual classification of the SWOT issues was conducted by the organization and the result was treated as classification target. Our research shows that the factor of word relatedness in semantic vectors increase the level of sentence similarity score and it enhances the performance of text classification. Without word relatedness, the F-Measure of k-nearest neighbour classification algorithm is 0.39. Inclusion of word relatedness using lexical relationship in a classification algorithm improve F-Measure as high as 0.595, while word relatedness based on co-occurrences increases F-Measure to a level of 0.4.

Keywords: word relatedness, dictionary, semantic similarity, text classification, bahasa Indonesia.

INTRODUCTION

Classification is a process of identifying to which category a new observation belongs among a set of known categories, where those categories have already had data as the training set. Classification may be divided into two types, that is, supervised and unsupervised classification. Supervised classification is different from unsupervised one in that the first has a special step called the learning phase, prior to getting through implementation phase and testing phase. Unsupervised classification is a method to group objects without the initial learning step (Sentosa, 2007).

Classification may be applied to text objects or documents. Text classification find various implementations, such as to identify whether an incoming email message is a spam, to classify a news article as a sport article or politics, and to decide whether public opinions are in positive or negative mood (Manning *et al*., 2009).

A lot of methods have been proposed to conduct text and document classification. Among the popular methods are decision tree, rule-based classifier, support vector machine (SVM), artificial neural network, k-nearest neighbour, and Bayesian algorithm (Aggarwal and Zhai, 2012).

A text classifier processes and investigates words that build a text to determine the document class. Different algorithms treat words that build a text in various ways, either by checking the existence of words (binary methods), by counting the number or frequency of words, or by considering their meanings. Methods that consider word existence or word frequency can have problems because two same words may have different meaning in different contexts and two different words may have similar meanings.

To overcome the problems, word relatedness may be utilized to calculate text similarity in classification process. Research by (Yazdani and Popescu-Belis, 2012) calculates word relatedness based on the contents and links in an encyclopaedia. The authors concluded that the use of word relatedness gives a very good result although it is still lower than the best one. When calculating the similarity of two documents, they obtained a correlation score of 0.6, which is close to the score for LSA (latent semantic analysis) method.

Forms of word relatedness have been tested in term-based similarity calculation and gives positive results. The study by (Liu and Wang, 2013) utilized WordNet to calculate word relatedness. They tried to improve the performance of cosine similarity by putting the relative distance of words in word net as the entries of semantic vectors. By setting a parameter to an optimum value, they successfully obtain a better performance compared to other methods such as Lesk (Lesk, 1986), Leacock-Chodorow (Leacock and Chodorow, 1998) and Wu-Palmer (Wu and Palmer, 1994).

On the other hand, Islam et al. (Islam *et al*., 2012) have used Google tri-gram to get the level of word relatedness. Google n-gram is a database containing words or phrases and the frequency they were used in queries by users of Google search website. The background idea is that words that are used together more often are likely to have more relatedness than words that are used less together. They claimed that similarity calculation using Dice function that utilizes Google n-gram produces correlation as high as 0.9 with expert judgements.

Studies on classification of Indonesian text have been rare. We got a single publication on Indonesian short text by Laksana and Purwarianti (2014) which studied classification of tweets to official Twitter account of Bandung city government. They found that Support Vector Machine and Label Power Set are two methods that perform best in term of accuracy during tweet classification, outperforming several other methods including decision tree and Naive Bayes algorithm.

This paper describes the use of word relatedness as the entries of semantic vectors in calculating similarity of two short texts in Bahasa Indonesia. The text similarity score is applied for classification process using k-nearest neighbour. Our experiments show that the use of word relatedness increases text similarity and improves the performance of classification of short text in Bahasa Indonesia.

METHOD

Our study was conducted on more than 150 short texts, of which 112 were used in learning and 43 were used for testing. The short texts were sentences or phrases from a SWOT (strength, weakness, opportunity and threat) analysis of an organization. A special team in the organization has grouped the SWOT statements into 4 categories, i.e. human resources, student activities, teaching and research activities, and others. Those categories will be denoted as class 1 - 4 hereafter.

During the SWOT analysis session, the organization divided participants into 4 groups. Each group was asked to express their ideas on strengths and weaknesses of the organization, and opportunities and threats that they face. In such a session, people rarely produce complex sentences and give more often short sentences or phrases. Different groups produced different statements about various things, but they also produced similar expressions for the same idea, or expressed similar ideas with different wording. For example, a group created the statement: “Manajemen keuangan belum distandarkan dan belum tersosialisasi dengan baik” (Financial management has not been standardized and it is poorly communicated) while another group produced the statement: “*Skema penghonoran masih belum standar*” (Compensation scheme does not yet have a standard). At a later step, the special team collected group expressions and sorted the statements into the four categories; hence the team has conducted manual clustering.

For the purpose of this research, we have separated statements of one group from those of the others. Texts from three groups are treated as the learning objects of the classification algorithm and the rest are considered as the testing objects.

The research conducted was to categorize a number of short texts using k-nearest neighbour algorithm. Distance of two text objects is calculated by three different similarity functions, i.e. pure cosine function, Dice function considering word-to-word similarity based on word co-occurrence in a local corpus, and Dice function considering word-to-word similarity based on lexical relationship. To decide the group or category for a text, we went through the following steps:

1. Pre-processing,
2. Calculating the similarity measure of the text to all other texts,
3. Determining the k-nearest neighbours and deciding a group to which the text belongs.

Pre-processing was conducted by eliminating unwanted symbolic characters such as question marks and brackets but preserving alphanumeric characters, points, hyphens, and spaces. This process is known as case folding. Thereafter, phrases or sentences were parsed to become a list of words. Common words, or stop words are ignored and omitted from the list. In this research, we did not do stemming, i.e. finding the roots of derived words.

Similarity between the processed text and all other text is calculated using three similarity measure. The first measure is the pure cosine similarity function which calculates similarity based on the count of exact matching words in two sentences. The second measure and third measure considers word to word relatedness in calculating sentence similarity and are based on Dice similarity function.

The next step is to select the k neighbouring texts (we choose k = 10) that have the highest similarity score (i.e. least distance). A class is decided for the text when the nearest neighbours mostly belong to the class (method K1). We consider an alternative where a text is decided to be in a class if the accumulative weighted distance of the nearest neighbours is the least for that class (method K2) as described in (Voulgaris and Magoulas, 2008).

Cosine function for text similarity calculation has the following mathematical expression:

$$\mathrm{Sim}(D_1, D_2) = \frac{V_1 \bullet V_2}{|V_1||V_2|} \tag{1}$$

where D_1 and D_2 are the two text documents, V_1 and V_2 are semantic vectors of D_1 and D_2, respectively. The two semantic vectors have the same length and it is equal to the total number of terms in the two documents. An element in a semantic vector has a one to one correspondence with a term that may occur in the documents. The value of an element is set one if the corresponding term exists in a document, otherwise it is set zero.

Dice similarity measure is shown in equation (2) with the meaning of symbols similar to those used in equation (1) and sum$\{V\}$ is the sum of vector elements.

$$\mathrm{Sim}(D_1, D_2) = \frac{2\,V_1 \bullet V_2}{\mathrm{sum}\{V_1\} + \mathrm{sum}\{V_2\}} \tag{2}$$

Semantic vectors in Cosine and Dice function usually contains elements in natural numbers representing the number of respective terms that exist in a document, so element values are positive whole numbers starting from 0. In our case, however, element values range from 0 to 1 depending on word-to-word relatedness.

There are two kinds of word relatedness in our study: word co-occurrences in local corpus and word lexical relationship. Local corpus is built from sentences or phrases in the learning set. Words that occur in the same sentence has larger relatedness than words that occure in the same class set. Lexical relationship is obtained from modified version of Kamus Besar Bahasa Indonesia (or big Indonesian dictionary) and the relatedness of two words depends on their relationship whether they are synonymous, hypernymous or holonymous. The values for lexical relationship are available in (Thamrin and Wantoro, 2014).

The results of classification are stored into a confusion matrix (see Table-1) for performance evaluation. A value of *M₁₂* in the matrix, for instance, means that the SWOT team has decided that as many as *M₁₂* texts belong to class 1 but the algorithm identified the texts as members of class 2.

Table-1. Confusion matrix.

Decided by team	Identified by algorithm				Total decided
	Class 1	Class 2	...	Class n	
Class 1	M_{11}	M_{12}		M_{1n}	
Class 2	M_{21}	M_{22}		M_{2n}	
...					
Class n	M_{n1}	M_{n2}		M_{nn}	
Total identified					

Performance of the algorithms is measured by three parameters: Recall (*R*), Precision (*P*) and F-measure (*F*). Values for those parameters range from 0 to 1 that represents the worst and the best performance measure. In multiple group classification, recall of class j designates the number of texts identified to be in class j compared to the total texts that actually in class j. Precision of class j designates the number of texts correctly identified as class j compared to the total number of texts identified as class j (Sokolova and Lapalme, 2009). Precision and recall for multigroup classification are calculated using equation 3 and 4. F-measure combines the other two parameters and may be adjusted to represent significance of either of P or R and is calculated using equation 5.

$$R_j = \frac{M_{jj}}{\sum M_{ji}} \tag{3}$$

$$P_j = \frac{M_{jj}}{\sum M_{kj}} \tag{4}$$

$$F_j = 2P_jR_j/\left(P_j + R_j\right)$$

$$F = \frac{F_j \sum M_{ji}}{n} \tag{5}$$

RESULTS AND DISCUSSIONS

After conducting classification using k-nearest neighbour, confusion matrix was constructed and performance measure was calculated using equation 3 - 5. We have gone through two alternatives in the classification process. Firstly, a text is grouped to a class if most out of 10 neighbours belong to the class (method K1), and secondly, a text is grouped to a class if the accumulative similarity of its neighbours for the class is maximum (method K2).

Classification using method K1 results in performance measures as shown in Table-2. The table confirms that calculation using Dice function with word lexical relationship has the largest value of recall (R) and precision (P) for all classes. As an example, calculation using cosine function and Dice function with word co-occurrence result in zero value of R and P for classification to class 1, while calculation using Dice function with word lexical relationship results in R = 0.5 and P = 1.

Table-2. Recall and precision of classification with k-nearest neighbour (method K1).

Calculation	Perf. Meas.	Class			
		1	2	3	4
Cosine	R	0	1	0	0.714
	P	0	0.625	0	0.962
Dice w/ word co-occurrence	R	0	1	0	0.629
	P	0	0.714	0	0.957
Dice w/ word lex. relationship	R	0.5	1	0	0.943
	P	1	0.714	0	0.971

Classification using method K2 gives recall and precision values presented in Table-3. The result is similar to that of method K1 described in the previous paragraph, in terms that best performance measures are attained when classification uses Dice similarity function with word lexical relationship.

F-measures of classification using the two method (K1 and K2) and the three calculation functions have been computed and depicted in Figure-1. The figure shows that calculation using Dice function with word lexical relationship gives the highest value of F-measure compared to the other calculations. In average F-measure for calculation using cosine function, Dice with word co-occurrence, and Dice with word lexical relationship are 0.39, 0.4 and 0.595, respectively. The fact suggests that the inclusion of word lexical relationship in semantic vectors improves the performance of Dice function in classifying short texts using k-nearest neighbour.

Table-3. Recall and precision for classification with weighted k-nearest neighbour (method K2)

Calculation	Perf. Meas.	Class			
		1	2	3	4
Cosine	R	0	1	0	0.714
	P	0	0.556	0	0.962
Dice w/ word co-occurence	R	0	1	0	0.743
	P	0	0.625	0	0.963
Dice w/ word lex. relationship	R	0.5	0.6	0	0.943
	P	1	0.6	0	0.917

Recalling results by (Voulgaris and Magoulas, 2008) and (Hamzah *et al.,* 2008), similarity calculation using cosine function gives better results than calculation using Dice function. Our results suggests that similarity calculation using Cosine function does not give different result compared to calculation using Dice function provided that the semantic vector include word co-occurrence similarity score. Figure-1 verifies that classification using Dice function with word co-occurrence has comparable performance with classifiction using Cosine function.

Our results as described in the above paragraphs agree with previous findings. Many investigators have demonstrated that sentence similarity score may be improved by including word similarity score in semantic vectors of term based similarity algorithms (Islam *et al*., 2012; Liu and Wang, 2013; Thamrin and Sabardila, 2014).

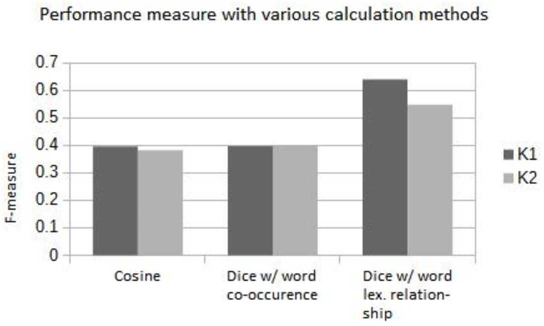

Figure-1. Performance measure for classification using various functions.

If accuracy is used as a performance measure, we would get the following figures: 0.843, 0.861, and 0.926 for Cosine, Dice with word co-occurence and Dice with word lexical relationship methods, respectively. The values are in contrast to results obtained by Wulandini and Nugroho (2009) where classification of Indonesian text using k-nearest neighbour had an accuracy of 0.49, which was claimed to be the worst method that they studied.

So far, most researches on Indonesian text classification are conducted on medium sized to long documents. Articles from news portals is most often employed because they are easily obtained, for example studies by (Februariyanti, 2012; Kurniawan *et al*., 2012; Samodra *et al*., 2009). Several studies employs scientific articles, either abstract or full paper, e.g. (Hamzah, 2012; Wijaya and Tjiharjadi, 2010). Results of the studies are various but most claimed that accuracy bigger than 60-80% were obtained.

Our current study has focused on similarity score of short texts in bahasa Indonesia. Many similarity algorithms do not depend on language and perform well for many applications. However, as pointed out by (Song *et al*., 2014), there are problems when the algorithms are applied to classification of short text because of the information sparseness. Sparseness may be reduced by getting more information from knowledge base such as dictionary, thesaurus or synset. The use of knowledge base is language dependent and the success depends on the richness of the observed language.

This study has used lexical relationship in the form of synonymy and hypernymy obtained in Indonesian dictionary. The result suggests that dictionary of bahasa Indonesia can be used to reduce information sparseness during text similarity calculation, and such calculation can produce better similarity score and improve the performance of classification of short text in bahasa Indonesia.

CONCLUSIONS

The study investigates the utilization of lexical relationship in form of synonymy and hypernymy to increase sentence similarity score and applies the method to do classification of short texts in Bahasa Indonesia. Similarity score of two texts is increased when the term-based calculation considers word relatedness. Consequently, the performance of text classification using k-nearest neighbour has improved.

Without word relatedness, the F-Measure of k-nearest neighbour classification is 0.39. Word relatedness using lexical relationship improves the performance of classification algorithm where F-Measure as high as 0.595 is attained. Word relatedness based on co-occurrences slightly increases the F-Measure to a level of 0.4.

The use of word relatedness reduces information sparseness of short texts. Therefore, similarity score of two sentences can be increased by including word-to-word relationship in sentence similarity calculation.

REFERENCES

Aggarwal, C.C., Zhai, C. 2012. A survey of Text Classification Algorithms. In: Mining Text Data. Springer US, p. 533.

Februariyanti, H. 2012. Klasifikasi Dokumen Berita Teks Bahasa Indonesia menggunakan Ontologi. J. Teknol. Inf. Din. 17, 14-23.

Hamzah, A. 2012. Klasifikasi teks dengan naïve bayes classifier (nbc) untuk pengelompokan teks berita dan abstract akademis. In: Prosiding Seminar Nasional Aplikasi Sains & Teknologi (SNAST) Periode III. pp. 269-277.

Hamzah, A., Soesianto, F., Susanto, A., Istiyanto, J.E. 2008. Studi Kinerja Fungsi-fungsi Jarak dan Similaritas dalam Clustering Dokumen Teks Berbahasa Indonesia. In: Seminar Nasional Informatika.

Islam, I., Milios, E., Keselj, V. 2012. Text Similarity Using Google Tri-Grams. In: 25th Canadian Conference on Advances in Artificial Intelligence. pp. 312–317.

Kurniawan, B., Effendi, S., Sitompul, O.S. 2012. Klasifikasi Konten Berita Dengan Metode Text Mining. Dunia Teknol. Informasi-Jurnal Online 1, 14-19.

Laksana, J., Purwarianti, A. 2014. ge. Indonesian Twitter text authority classification for government in Bandung. In: Proc. International Conference of Advanced Informatics: Concept, Theory and Application (ICAICTA). Bandung, pp. 129-134.

Leacock, C., Chodorow, M. 1998. Combining local context and WordNet similarity for word sense identification. In: WordNet, an Electronic Lexical Database. The MIT Press.

Lesk, M.E. 1986. Automatic Sense Disambiguation Using Machine Readable Dictionaries: How to Tell a Pine Cone from an Ice Cream Cone. In: Proceedings of SIGDOC Conference. Toronto.

Liu, H., Wang, P. 2013. Assessing Sentence Similarity Using WordNet based Word Similarity. J. Softw. 8, 1451–1458.

Manning, C.D., Raghavan, P., Schultze, H. 2009. Introduction to Information Retrieval. Cambridge University Press.

Samodra, J., Sumpeno, S., Hariadi, M. 2009. Klasifikasi Dokumen Teks Berbahasa Indonesia dengan Menggunakan Naïve Bayes. In: Seminar Nasional Electrical, Informatics, Dan IT's Education. pp. 1-4.

Sentosa, B. 2007. Data Mining Teknik Pemanfaatan Data untuk Keperluan Bisnis. Graha Ilmu, Surabaya.

Sokolova, M., Lapalme, G. 2009. A Systematic Analysis of Performance Measures for Classification Tasks. Inf. Process. Manag. 45, 427-437.

Song, G., Ye, Y., Du, X., Huang, X., Bie, S., 2014. Short Text Classification: A Survey. J. Multimed. 9, 635–643.

Thamrin, H., Sabardila, A. 2014. Using Dictionary as a Knowledge Base for Clustering Short Texts in Bahasa Indonesia. In: International Conference on Data and Software Engineering. ITB Bandung, Bandung.

Thamrin, H., Wantoro, J. 2014. An Attempt to Create an Automatic Scoring Tool of Short Text Answer in Bahasa Indonesia. In: Proceeding of International Conference on Electrical Engineering, Computer Science and Informatics (EECSI 2014). IAES, Yogyakarta.

Voulgaris, Z., Magoulas, G.D. 2008. Extensions of the k Nearest Neighbour Methods for Classification Problems. In: Proceedings of the 26th IASTED International Conference on Artificial Intelligence and Applications.

Wijaya, M.C., Tjiharjadi, S. 2010. Aplikasi Klasifikasi Dokumen Menggunakan Metoda Naïve Baysian. In: Seminar Nasional Informatika 2010. pp. 56-59.

Wu, Z., Palmer, M. 1994. Verb Semantics and Lexical Selection. In: Proceedings of the 32nd Annual Meeting of the Association for Computational Linguistics. La Cruces, New Mexico.

Wulandini, F., Nugroho, A.S. 2009. Text Classification Using Support Vector Machine for Webmining Based Spatio Temporal Analysis of the Spread of Tropical Diseases. In: Proc. International Conference on Rural Information and Communication Technology.

Yazdani, M., Popescu-Belis, A. 2012. Computing Text Semantic Relatedness using the Contents and Links of a Hypertext Encyclopedia. Artif. Intell.