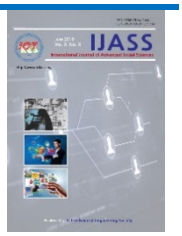
ISSN: 2586-761x (Print) Vol. 01, No. 01, June 2018

International Journal of Advanced Social Sciences

Source: http://ictaes.org

Manuscript received : May 02, 2018 ; Revised : May 22, 2018 ; Accepted : May 27, 2018



Comparing Graph based K Nearest Neighbor with Traditional Version in Word Categorization in NewsPage.com

Taeho Jo

School of Game, Hongik University

tjo018@hongik.ac.kr

Abstract

In this research, we propose the KNN version where words are encoded into graphs, instead of numerical vectors, as the approach to the word categorization. It is able to represent a word into a graph where vertices are text identifier and edges are their similarities, and we try to make the mutual reinforcement between word and text categorization by connection them with each other. In this research, words are encoded into graphs each of which is represented into a list of edges, the semantic similarity metric between two graphs is defined, and the KNN algorithm is modified by adopting the proposed similarity metric. This research shows better results of the proposed KNN version which receives a graph as input data than the traditional version, in categorizing words which are from news articles. In future, we connect mutually the word categorization with the text categorization for reinforcing them at same time.

Keywords: *Word Categorization, Graph, Graph Similarity, Graph based K Nearest Neighbor*

1. INTRODUCTION

The word categorization is referred to the process of assigning one or some among the predefined categories to each word. We may consider the two tasks as word categorizations: the lexical classification which is the process of classifying words by their spellings, and the POS (Part of Speech) tagging which is the process of classifying them by their grammatical functions. The scope is restricted to the semantic classification where words are classified based on their topics or meaning. We mention the K Nearest Neighbor as the approach and modify it into the version which receives a graph as its input data. This section covers the motivation, the idea, and validation of this research.

Let us consider the motivations for doing this research. The fact that we may expect the synergy effect between the word categorization and the text categorization motivates for setting the former as the task of this research. The KNN algorithm is a simple approach to the data classification for starting to modify machine learning algorithms. A graph is more graphical representation than a numerical vector for visualizing a word. This research is intended to avoid the poor discrimination from the sparse distribution in each numerical

Corresponding author : Taeho Jo
Author's affiliation : Hongik University
Email: tjo018@hongik.ac.kr

Copyright © ICT-AES

vector, by encoding them into alternative ones.

In this research, we propose the modified KNN version as the approach to the word categorization. The words are encoded into graphs where nodes are text identifiers and edges are similarities between texts, and the similarity metric between them is defined. By adopting the proposed similarity metric, for computing the similarity between the test example and each training example, the KNN is modified. The modified version is applied to the word categorization which is covered in this research. The discriminations among numerical vectors are improved by encoding words into alternative representations to numerical vectors.

In this research, we will validate empirically the proposed approach to the word categorization as the better version than the traditional KNN version. We extract words which are classified their own topics from the news collection: 20NewsGroups. The traditional KNN version and the proposed version are compared with each other. We observe the better results of the proposed KNN version in classifying words. In addition, it is possible to represent words or texts more graphically from encoding them into graphs.

Let us mention the organization of this research. In Section 2, we explore the previous works which are relevant to this research. In Section 3, we describe in detail what we propose in this research. In Section 4, we validate empirically the proposed approach by comparing it with the traditional one. In Section 5, we mention the significances of this research and the remaining tasks as the conclusion.

2. PREVIOUS WORK

Let us survey the previous cases of encoding texts into structured forms for using the machine learning algorithms to text mining tasks. The three main problems, huge dimensionality, sparse distribution, and poor transparency, have existed inherently in encoding them into numerical vectors. In previous works, various schemes of preprocessing texts have been proposed, in order to solve the problems. In this survey, we focus on the process of encoding texts into alternative structured forms to numerical vectors. In other words, this section is intended to explore previous works on solutions to the problems.

Let us mention the popularity of encoding texts into numerical vectors, and the proposal and the application of string kernels as the solution to the above problems. In 2002, Sebastiani presented the numerical vectors are the standard representations of texts in applying the machine learning algorithms to the text classifications [4]. In 2002, Lodhi et al. proposed the string kernel as a kernel function of raw texts in using the SVM (Support Vector Machine) to the text classification [5]. In 2004, Lesile et al. used the version of SVM which proposed by Lodhi et al. to the protein classification [6]. In 2004, Kate and Mooney used also the SVM version for classifying sentences by their meanings [7].

It was proposed that texts are encoded into tables instead of numerical vectors, as the solutions to the above problems. In 2008, Jo and Cho proposed the table matching algorithm as the approach to text classification [8]. In 2008, Jo applied also his proposed approach to the text clustering, as well as the text categorization [12]. In 2011, Jo described as the technique of automatic text classification in his patent document [10]. In 2015, Jo improved the table matching algorithm into its more stable version [11].

Previously, it was proposed that texts should be encoded into string vectors as other structured forms. In 2008, Jo modified the k means algorithm into the version which processes string vectors as the approach to the text clustering [12]. In 2010, Jo modified the two supervised learning algorithms, the KNN and the SVM, into the version as the improved approaches to the text classification [13]. In 2010, Jo proposed the unsupervised neural networks, called Neural Text Self Organizer, which receives the string vector as its input data [14]. In 2010, Jo applied the supervised neural networks, called Neural Text Categorizer, which gets a string vector as

its input, as the approach to the text classification [15].

The above previous works proposed the string kernel as the kernel function of raw texts in the SVM, and tables and string vectors as representations of texts, in order to solve the problems. Because the string kernel takes very much computation time for computing their values, it was used for processing short strings or sentences rather than texts. In the previous works on encoding texts into tables, only table matching algorithm was proposed; there is no attempt to modify the machine algorithms into their table based version. In the previous works on encoding texts into string vectors, only frequency was considered for defining features of string vectors. In this research, we propose that words should be encoded into graphs, and modify the KNN into the version which processes graphs instead of numerical vectors, as the approach to the keyword extraction.

3. PROPOSED WORK

This section is concerned with what we propose in this research. Words are encoded into graphs and the graph similarity is defined. The AHC algorithm is modified into the version which receives a group of graphs as clustering targets and computes the proposed similarity metrics among data items. The modified version is applied to the topic based word clustering. In this section, we describe what is proposed in this research.

Let us explain the process of encoding a word into a graph. The word is given as the encoding target and texts which include the word are retrieved from a corpus as graph vertices. The similarities among retrieved texts are computed as edge candidates, and some among them are selected by their similarities as edges. In this process, a weighted undirected graph which consists of weighted edges is generated as the word representation. In the graph, the text identifiers are given as vertices and the text similarities are given as edges.

The two graphs are expressed as two edge sets, $G_1 = \{e_{11}, e_{12}, \ldots, e_{1n}\}$ and $G_2 = \{e_{21}, e_{22}, \ldots, e_{2n}\}$, assuming the identical number of edges in both of them, and an individual edge is expressed into an entry: $e_{ki} = (v_{ki1}, v_{ki2}, w_{ki})$. In computing the similarity between two edges, we consider the three cases: $sim(e_{1i}, e_{2j}) = \frac{1}{2}(w_{1i} + w_{2j})$ in case of both identical vertices, $sim(e_{1i}, e_{2j}) = w_{1i} w_{2j}$ in case of either identical vertices, and $sim(e_{1i}, e_{2j}) = 0$ in case of no identical vertex. The similarity between an individual edge and a graph is computed by Equation (1),

$$sim(e_{1i}, G_2) = \max_{j=1}^{n} sim(e_{1i}, e_{2j}) \quad (1).$$

The similarity between two graphs is computed by Equation (2),

$$sim(G_1, G_2) = \frac{1}{n} \sum_{i=1}^{n} sim(e_{1i}, G_2) \quad (2)$$

The weight which is associated with each edge is always given as a normalized value between zero and one, so the product of two weights results in a reduced value, when either of two vertices in an edge is same to those of another edge.

Figure 1 presents the proposed KNN algorithm. In advance, the training words are encoded into graphs. A novice word is encoded into a graph, its similarities with ones which represent the training ones by Equation (2), and the most k similar training words are selected its nearest neighbors. The label of the novice word is decided by voting ones of the nearest neighbors. We may consider the KNN variants which are derived from this version by discriminating the similarities and the attributes.

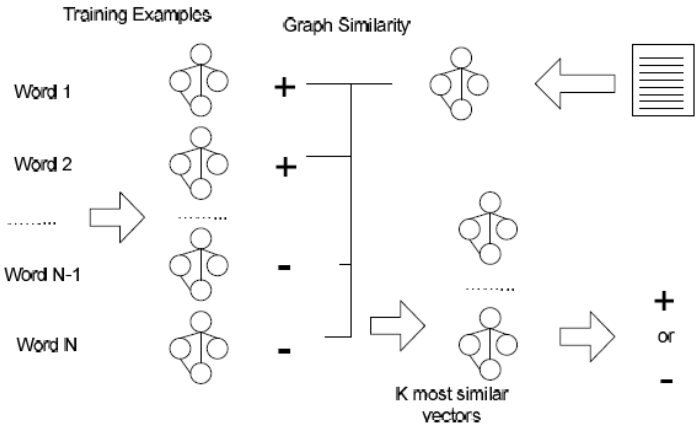


Figure 1. The Proposed Version of KNN

Let us make some remarks on what is proposed in this research. Even if the KNN is a very simple machine learning algorithm, it is useful for implementing a light version of classification system. Even if it takes much tome for computing the proposed similarity metric, it tackled against the poor discriminations from the sparse distribution of numerical vectors. We may use words which are called contexts as vertices as well as texts for representing words into graphs. The proposed KNN is described in more detail in [16].

4. EXPERIMENTAL RESULTS

This section is concerned with the experiments for validating the better performance of the proposed version on the collection: NewsPage.com. The four categories are predefined in this collection and from the collection, NewsPage.com, we gathered the words category by category as the labeled ones. Each word is allowed to be classified into only one of the four categories. In this set of experiments, we apply the traditional and proposed version of KNN to the classification task, without decompose it into binary classifications, and use the accuracy as the evaluation measure. In this section, we observe the performance of the both versions of KNN, by changing the input size.

Table 1. The Number of Texts and Words in NewsPage.com

Category	#Texts	#Training Words	#Test Words
Business	500	300	75
Health	500	300	75
Internet	500	300	75
Sports	500	300	75
Total	2,000	1,200	300

In Table 1, we specify NewsPage.com, which is the text collection as the source for extracting classified words in this set of experiments. The text collection was used in the previous works for evaluating approaches to text categorization [11]. In each category, we extract 375 important words for building the collection of labeled words for evaluating the approaches to word categorization. In each category, the set of 375 classified words is partitioned into the 300 words as training examples and the 75 words as test examples, as shown in Table 1. We select words by their frequencies concentrated in the given category combined with subjectivities in building the word collection.

Let us mention the empirical process for validating the proposed approach to the task of word categorization. We extract the important words from each category in the above text collection, and encode them into numerical vectors. For each text example, the KNN compute its similarities with the 1200 training examples by the cosine similarity, and select the three most similar training examples as its nearest neighbors. Each of the 300 test examples is classified into one of the four categories: Business, Sports, Internet, and Health, by voting the labels of its nearest neighbors. The classification accuracy is computed by dividing the number of correctly classified test examples by total number of test examples, for evaluating the both versions of KNN.

Figure 2 illustrates the experimental results from categorizing the words using the both versions of KNN algorithm. The y-axis indicates the accuracy which is the rate of the correctly classified examples in the test set. Each group in the x-axis is the input size as the dimension of numerical vectors and the number of edges of graphs. In each group, the gray and black bar indicate the performance of the traditional and proposed version of KNN algorithm, respectively. The most right group indicates the average over accuracies of the left four cases.

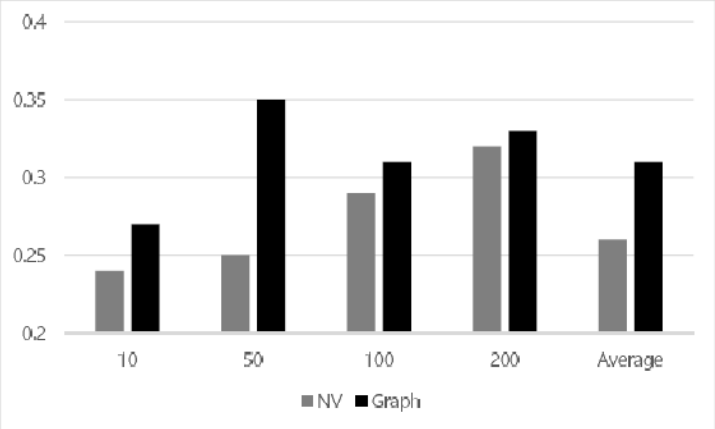


Figure 2. Experimental Results

Let us make discussions on the results from doing the word categorization, using the both versions of KNN algorithm, as shown in Figure 2. The accuracy which are the performance measure of this classification task is in range between 0.24 and 0.35. The proposed version of KNN algorithm works better in the all input sizes. The performance difference between them is maximal in the input size 50 but small in the large input sizes. In this set of experiments, we conclude that the proposed version works outstandingly better than the traditional one, in averaging over the four cases.

5. CONCLUSION

This work was Let us mention some remaining tasks for doing the further research. We need to validate more the proposed approach in categorizing words in specific domains such as medicine, engineering, and economics, and customize it correspondingly. We need to consider other schemes of encoding words into graphs and other similarity measures between graphs. We modify other machine learning algorithms into their graph based versions where a graph is given by itself as the input data. We implement a word categorization system by adopting the proposed approach.

ACKNOWLEDGEMENT

This work was supported by Hongik University in 2017.

REFERENCE

[1] Jo, T. (2006). The Implementation of Dynamic Document Organization using Text Categorization and Text Clustering}, PhD Dissertation of University of Ottawa, 2006.

[2] Noy, N.F. & Hafner, C. D. (1997), State of the Art in Ontology Design, AI Magazine, 18(3), 53-74

[3] Allemang, D. and Hendler, J., (2011), Semantic Web for the Working Ontologies, Morgan Kaufmann.

[4] Sebastiani, F., (2002), Machine Learning in Automated Text Categorization, ACM Computing Survey, 34(1), 1-47.

[5] Lodhi, H., Saunders, C., Shawe-Taylor, J., Cristianini, N., and Watkins, C., (2002), Text Classification with String Kernels, Journal of Machine Learning Research, 2(2), 419-444.

[6] Leslie, C.S., Eskin, E., Cohen, A., Weston, J., and Noble, W.S., (2004), Mismatch String Kernels for Discriminative Protein Classification, pp467-476, Bioinformatics, 20(4), 467-476.

[7] Kate, R. J. and Mooney, R.J., (2006), Using String Kernels for Learning Semantic Parsers, Proceedings of the 21st International Conference on Computational Linguistics and the 44th annual meeting of the Association for Computational Linguistics, (pp. 913-916), ACM.

[8] Jo, T. and Cho, D., (2008), Index based Approach for Text Categorization, International Journal of Mathematics and Computers in Simulation, 2(1), 127-132.

[9] Jo, T., (2008), Single Pass Algorithm for Text Clustering by Encoding Documents into Tables, Journal of Korea Multimedia Society, 11(12), 1749-1757.

[10] Jo, T., (2011). KR. Patent No. 10-2009-0041272. Korea, Device and Method for Categorizing Electronic Document Automatically.

[11] Jo, T., (2015), Normalized Table Matching Algorithm as Approach to Text Categorization, Soft Computing, 19(4), 839-849.

[12] Jo, T., (2008), Inverted Index based Modified Version of K-Means Algorithm for Text Clustering, Journal of Information Processing Systems, 4(2), 67-76.

[13] Jo, T., (2010), Representation of Texts into String Vectors for Text Categorization", Journal of Computing Science and Engineering, 4(2), 110-127.

[14] Jo, T., (2010), NTSO (Neural Text Self Organizer): A New Neural Network for Text Clustering, Journal of Network Technology, 1(1), 31-43.

[15] Jo, T., (2010), NTC (Neural Text Categorizer): Neural Network for Text Categorization", International Journal of Information Studies, 2(2), 83-96.

[16] Jo, T., (2016), Graph based KNN for Content based Word Classification", The Proceedings of 12th International Conference on Multimedia Information Technology and Applications, (pp. 24-29).