Automatic Labeling to Classify News Articles Based on Paragraph Vector

TAISHI SAITO† OSAMU UCHIDA‡
† Graduate School of Engineering, Tokai University
‡ Dept. of Human and Information Science, Tokai University
4-1-1 Kitakaname, Hiratsuka, Kanagawa 259-1292
JAPAN
† 6beim018@mail.u-tokai.ac.jp, ‡ o-uchida@tokai.ac.jp

Abstract: - Getting useful information from the Internet plays an important role. A news site is one of the Internet services often used for obtaining information on the Internet. The news site has advantages such that information update is fast and there are abundant kinds of information, and in recent years there are sites that collaborate with multiple newspaper companies and post bulk content. However, as there are a lot of articles, there are problems that it is difficult to find the articles we would like to read. Therefore, how to classify and present articles is an important issue. In this study, we consider the category classification of documents using a distributed representation of sentences. Specifically, we propose a method to classify articles by extracting words with similar meanings from sentence vectors of each category and assigning them as labels.

Key-Words: - Distributed representation, paragraph vector, neural network, automatic labeling, text classification, category classification

1 Introduction

Getting useful information from the Internet plays an important role. Though it was common to obtain the latest information of society from newspapers, TV news, etc. conventionally, due to the popularization of the Internet, especially spread of smartphones, it became easier to obtain the latest information at any time easily. However, along with that, the amount of information has increased enormously, and choosing information is very important in obtaining useful information. A news site is one of the Internet services often used for obtaining information on the Internet. The news site has advantages such that information update is fast and there are abundant kinds of information. In recent years, there are sites that collaborate with multiple newspaper companies and post bulk content. However, as there are a lot of articles, there are problems that it is difficult to find the articles we would like to read [1]. Therefore, how to classify and present articles is an important issue nowadays.

Labeling is one of the quite effective ways as a solution to the above issue. Labeling helps to make it easier to grasp and search the content of the article. Therefore, labeling is used not only for news articles but also for classification of SNS [2] and images [3]. Currently, laboring is usually done manually. Therefore, there is a problem that fluctuation occurs in the label to be given, and the same information cannot be gathered well. Also, it costs much to add labels to a large amount of information.

In this study, we consider the category classification of documents using a distributed representation of sentences [4]. Specifically, we propose a method to classify articles by extracting words with similar meanings from sentence vectors of each category and assigning them as labels. By this method, similar words are predicted from the text of the sentences and given automatically, it is possible to eliminate the labor of attaching labels with human beings, and it is possible to prevent human subjectivity from affecting labeling.

We have already applied this method to Japanese news article data [5]. We could get several tags which related articles. However, these tags include article specific words, that is, some tags cannot be used to classify. The Japanese language has a special structure of sentences as a cause. Also, the Japanese language is often used different words in the same sense. We think that accuracy improvement of this method can be performed by using English as an intermediate language. Then, in this study, we verify to what extent tags useful for classification can be acquired for English sentences.

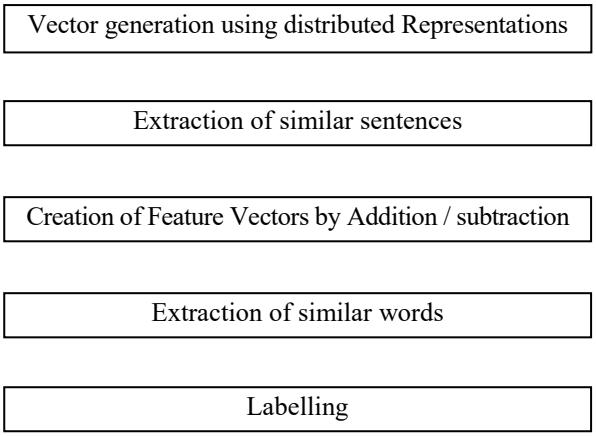


Fig. 1 Flow of the proposed method

2 Related Work

Several studies aiming at automatically assigning labels to text data to which labels are not assigned have been conducted.

For example, Shiotsu et al. proposed a method of visualization of news articles by multiple tagging using TF-IDF [6]. Their study aimed at facilitating article search. In their method, morphological analysis is first performed on headings and texts, and keywords are extracted. Then, the important tag is determined from the TF-IDF value of the extracted keyword. However, the composition of sentences differs depending on the genre of articles, and there is a possibility that the place of emergence of important words may change, and problems such as adding synonym tags to individual tags are not considered.

Keruma et al. conducted study to automatically assign labels to topics generated by topic models to deal with the problem that the interpretation of the topic depends on the subjectivity of the person and that the topic cannot be properly interpreted when the knowledge is insufficient with respect to the group of sentences which extracted the topic [7]. In their method, phrases are extracted from the target document by n-gram, and a phrase whose mutual information exceeds the threshold is a label candidate. Evaluation of labels is done by the method called LSA-weighted frequency (LSAF).

3 Proposed Method

3.1 Brief overview

In this study, we convert sentences and words into vectors using distributed expressions. Using the result of the conversion, these sentences are categorized and labeled. Figure 1 shows the flow of the proposed method.

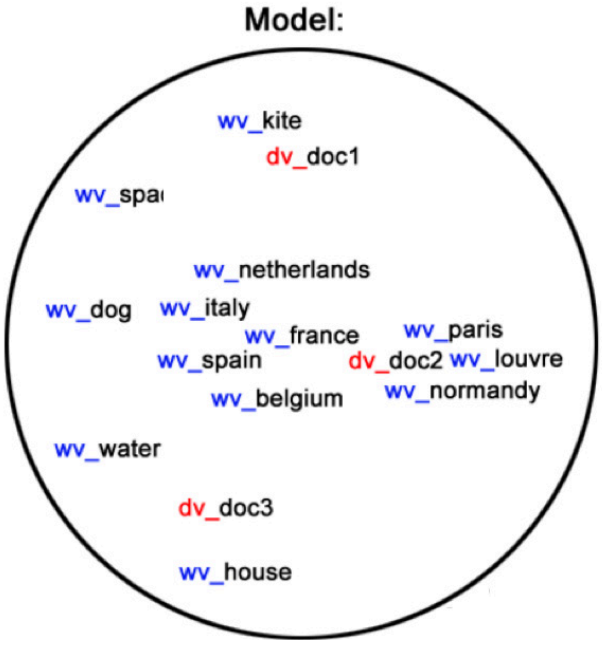


Fig. 2 Example of word and sentence vectors

First, we convert the news articles to sentence vectors using distributed Representations. At the same time, words are also converted to word vectors using distributed representation, and sentence vectors and word vectors are represented in the same vector space. Figure 2 shows an example of an expression, where wv represents a word vector, and dv represents a sentence vector.

Next, a sentence vector and a word vector like the sentence vector of the news article text to be labeled are extracted. After that, vector operation is performed to create a new vector. When a label of the same genre is given to a document with high vector similarity, vectors of the target sentence and similar sentences are added. Also, if a label of another genre is given, subtract it from a vector of the target sentence.

Finally, a word vector like the created new vector is extracted and given as a label.

3.2 Distributed representation

Distributed representation is a technique of expressing a word as a high-dimensional real vector which is constructed by the two-layer neural network. It can be learned to include linguistic properties of words and phrases and similar words to have similar vectors. To create a distributed representation of a word, there is a method of creating a co-occurrence frequency vector based on

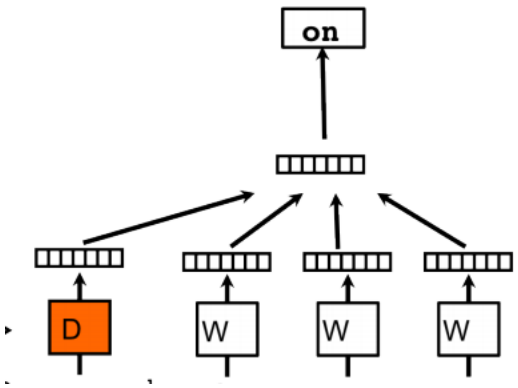


Fig. 3 Structure of PV-DM model

a distribution hypothesis that words of the same meaning appear in the same context, and a method of obtaining from learning by a neural network.

In this study, we use PV-DM (Paragraph Vector with Distributed Memory) model obtained from learning by a neural network. By learning the probability of the surrounding word W(t+n) of the word of interest W(t) and paragraph id, a distributed representation of the word is created. Figure 3 shows the structure of the PV-DM model. The weight of the hidden layer is a word vector, and the weight of the hidden layer is adjusted to have a high probability of the surrounding words for the target word.

Since the feature quantities of vectors contain linguistic properties, word vectors obtained by this method can perform the semantic computation of words. For example, "queen" can be obtained as the word vector most like the calculation result of "king - man + woman".

3.3 Vectorization of sentences

PV (Paragraph Vector) is a distributed expression for sentences. In the proposed method, we convert all the body of the news article into a vector by PV. PV is generated by the co-occurrence of words in sentences, and a vector indicating the nature of a sentence is generated from the sequence of words in the sentence. By this method, it is possible to make a variable length sentence into a fixed length vector, and it can be applied to machine learning which requires a constant number of input features. Also from this nature, sentence vectors and word vectors are expressed in fixed length in the same space. We use this vector to perform category classification and word extraction.

3.4 Extraction of Similar Words and Sentences

A sentence having a vector like the vector of the inputted sentence is extracted. For a target input sentence, extract words and sentence vectors of closest vectors from the created vector space. The similarity between two articles is calculated using the cosine similarity between each vector.

$$\cos\theta = \frac{(x, y)}{|x| \cdot |y|} \quad (1)$$

3.5 Creation of feature vectors by addition and subtraction

A vector of the target sentence and a vector of the extracted similar sentences are used to create a new feature vector. A calculation example is shown below.

"technology news" (target sentence)
+ "technology news" (similar sentence)

By performing such calculation, it is easy to extract words close to the classified genre at the time of similar word extraction by strengthening the feature amount of the specific region concerning the target article and removing the features of other genres.

3.6 Labelling

A word vector like the created feature vector is extracted, and only nouns are extracted from the extracted word group. Then, five words with the highest similarity are assigned as labels.

4 Verification Experiment

To verify the usefulness of the proposed method, a verification experiment was conducted. We experimented whether labels could classify by using news articles with categories are "technology news," "business news," "sports news," "politics news," or "health news," in about 50 thousand English articles of Reuters news [9]. These articles were acquired by using a crawler program. Each category has about 10 thousand articles.

This time, words were extracted for articles related "technology news," "business news," "sports news," "politics news," or "health news," of articles. As an example, to compare how many related words can be extracted, words similar to the vector of the target article related news article (case 1), and words similar to the vector created by the calculation (case 2) were extracted. Tables 1 and 2 show examples of the extracted words. We got the top 20 words

similar to articles. Then, how much other words are associated with other similar articles is shown in Figs.4-7.

We describe how much the same label is given to relevant articles. If there is only one article with a label, it is difficult to search similar articles. Conversely, when the same label is attached to multiple articles, the label can be said to be a useful label for searching for similar articles.

5 Result and Discussion

First, we consider the result of targeting technical articles. This targeted article is an article about a certain tech company. Among the acquired labels, many words such as company name, person name, product or service name related to the article were acquired. As shown in Fig.4, about 60% of labels are given to similar articles. The 40% labels are not attached as one article, it is a related word, but it cannot be used for searching for similar articles. Next, as shown in Fig.5, about 90% of labels are given to similar articles. This is the result of the label acquired with the vector generated by the calculation. By adding vectors of related articles and vectors of target articles, acquisition of words that can be given to multiple articles has increased significantly. However, words that can express the content of the article are excluded, and it seems that it became somewhat difficult to use in summary of the article.

Next time, we consider the results of targeting business articles. This targeted article is also an article about a certain tech company. As shown in Fig.6, about 70% of labels are given to similar articles. As shown in Fig.7, about 80% of labels are given to similar articles. Comparing Fig.6 with Fig.7, the label obtained by the calculation process was able to give more articles. However, the label of the business article has less change in the rate of giving the label than the technology article. It is thought that this is because the genre of the news article most similar to the vector of the target article obtained in the calculation process was a technology article. In this method, it is only the calculation process to consider the genre in the label assignment. In the process of creating a vector, only the features of the text of the article are acquired. Therefore, depending on the target article, there is a possibility that labels given by calculation may have features of articles of different genres. The business article used for the experiment this time was mentioned about the tech company. Therefore, even if vector calculation is performed on the same genre, the feature possessed by the article is considered to be weak. From the viewpoint of searching for articles, this is considered to be useful in that the articles can be related regardless of the genre. However, from the viewpoint of abstracting the article, there is a possibility that the feature of the genre of the news is lost by the article. In this case, the label given to the business article was given technical terms in addition to the company name and person related to the company. Therefore, in the summary of the sentence, judgment of the result is considered to change significantly depending on the subjectivity of the person.

6 Conclusion

In this study, we proposed an automatic labeling method which extracts words with similar meanings from sentence vectors of each category using distributed representation and assigns them as labels. Distributed representation is constructed with two layers of neural networks. It includes the linguistic nature of words and phrases, and similar words can be learned to have similar vectors. Thus, when words similar to given sentences are estimated, it is possible to eliminate human subjectivity.

Moreover, since words are acquired from the created vector space rather than directly using the words in the sentence, even if the latest article is input, the nearest word vector and the similar article can be acquired from the learned models. By calculation processing, it is possible to acquire only those words with the required strong features. Therefore, very useful labels can be obtained from the viewpoint of elimination of subjectivity and retrieval of similar articles.

As a future work, when creating vectors with distributed representation, we are considering processing by changing the structure of sentences. To put in concrete, unify all sentences' verbs into the present form. Also, we are thinking to change multiple words to singular type. Distributed representation is made by considering words around the word making up the vector. For that reason, past tense and plural words may influence, and even words of the same meaning may have different vectors created as words of different meanings.

Moreover, we would like to create more appropriate feature vectors by improvement of calculation. In this time, we calculated the same genres and created a new feature vector. However, in searching for similar articles, similar articles exist

even if the genre of articles is different. To search for similar articles, it is important to be able to extract such articles.

References:

[1] Fujitsu Laboratory, http://www.fujitsu.com/jp/group/fri/report/cyber/research/4/title07.html

[2] T. Kobayashi, H. Suzuki, A. Hattori and H. Haruno, A life log system that performs automatic tagging of Twitter's tweet, *The Special Interest Group Technical Reports of IPSJ*, 2013-GN-87, Vol.6, 2013, pp.1-5. (in Japanese)

[3] M. Tsukada, M. Iwamura and K. Kise, Distorted Character Recognition and Automatic Labeling, *Technical Report of IEICE*, Vol. 111, No. 317, 2011, pp. 93-98. (in Japanese)

[4] Q. V. Le and T. Mikolov, Distributed Representations of Sentences and Documents, *Proc. of 31st International Conference on Machine Learning*, 2014.

[5] T. Saito and O. Uchida, Automatic Labeling for News Article Classification Based on Paragraph Vector, *Proc. 9th International Conference on Information Technology and Electrical Engineering*, 2017.

[6] K. Shiotsu and S. Iwashita, A Method for Automatic Tagging for Classification and Retrieval of News Contents, *Proc. 18th Annual Conference of the Association for Natural Language Processing*, 2012, pp.529-530. (in Japanese)

[7] R. Keruma, N. Toma, Y. Akamine, K. Yamada and S. Endo, A Basic Study about Automatic Label Generation on Topic Model, *Proc. 76th Annual Conference of the Information Processing Society of Japan*, 6C-4, 2014. (in Japanese)

[8] T. Mikolov, I. Sutskever, K. Chen, G. S. Corrado and J. Dean, Distributed Representations of Words and Phrases and Their Compositionality," *Proc. 26th International Conference on Neural Information Processing Systems*, 2013, pp.3111-3119.

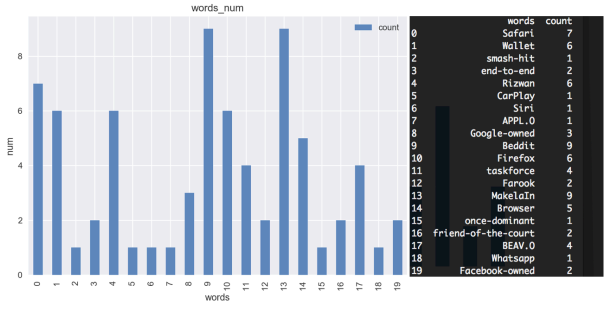
[9] Reuter News, https://www.reuters,com/

Table 1 Extracted top 20 words (Technology related articles)

rank	Case 1	Case 2
1	Safari	Safari
2	Wallet	Wallet
3	smash-hit	Beddit
4	end-to-end	Facebook-owned
5	Rizwan	Rizwan
6	Carplay	Brower
7	Siri	LNKD.N
8	APPL.O	Firefox
9	Google-owned	County-owned
10	Beddit	Location-based
11	Firefox	Embattled
12	taskforce	smash-hit
13	Farook	Farook
14	MakelaIn	BEAV.O
15	Brower	MakelaIn
16	Once-dominant	Carplay
17	Friend-of-the-court	Innotek
18	BEAV.O	Google-owned
19	Whatsapp	Apphabet's
20	Facebook-owned	iCloud

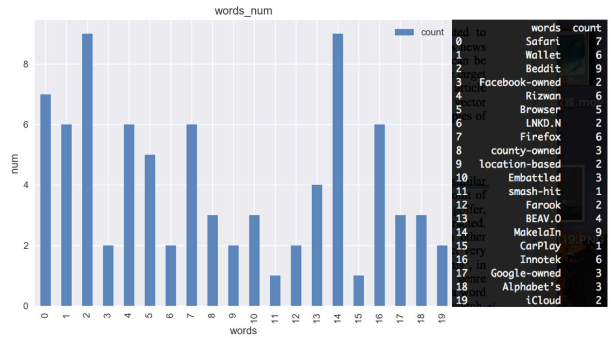
Table 2 Extracted top 20 words (Politics related articles)

rank	Case 1	Case 2
1	BEAV.O	Firefox
2	Watches	Beddit
3	eye-tracking	Rizwan
4	Vivint	Safari
5	briefs	MakelaIn
6	Lagardere	geo-fencing
7	Lionsgate	HexaTier
8	Firefox	Wallet
9	Allo	Vringo
10	HexaTier	laser-based
11	heavyweights	EMC.N
12	Peloton	end-to-end
13	BATS.Z	ad-blocking
14	Wallet	Maps
15	Logic	Osaka-based
16	Rizwan	eminent
17	MakeleIn	BEAV.O
18	Motions	nitride
19	Beddit	county-owned
20	taskforced	incomparable



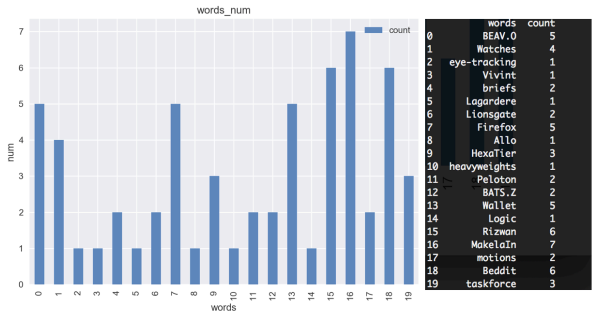
	words	count
0	Safari	7
1	Wallet	6
2	smash-hit	1
3	end-to-end	2
4	Rizwan	6
5	CarPlay	1
6	Siri	1
7	APPL.O	1
8	Google-owned	3
9	Beddit	9
10	Firefox	6
11	taskforce	4
12	Farook	2
13	MakelaIn	9
14	Browser	5
15	once-dominant	1
16	friend-of-the-court	2
17	BEAV.O	4
18	Whatsapp	1
19	Facebook-owned	2

Fig. 4 The result of Technology (Case1)



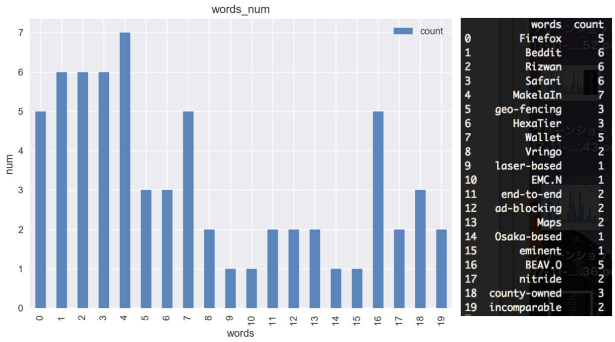
	words	count
0	Safari	7
1	Wallet	6
2	Beddit	9
3	Facebook-owned	2
4	Rizwan	6
5	Browser	5
6	LNKD.N	2
7	Firefox	6
8	county-owned	3
9	location-based	2
10	Embattled	3
11	smash-hit	1
12	Farook	2
13	BEAV.O	4
14	MakelaIn	9
15	CarPlay	1
16	Innotek	6
17	Google-owned	3
18	Alphabet's	3
19	iCloud	2

Fig. 5 The result of Technology (Case2)



	words	count
0	BEAV.O	5
1	Watches	4
2	eye-tracking	1
3	Vivint	1
4	briefs	2
5	Lagardere	1
6	Lionsgate	2
7	Firefox	5
8	Allo	1
9	HexaTier	3
10	heavyweights	1
11	Peloton	2
12	BATS.Z	2
13	Wallet	5
14	Logic	1
15	Rizwan	6
16	MakelaIn	7
17	motions	2
18	Beddit	6
19	taskforce	3

Fig. 6 The result of Politics (Case1)



	words	count
0	Firefox	5
1	Beddit	6
2	Rizwan	6
3	Safari	6
4	MakelaIn	7
5	geo-fencing	3
6	HexaTier	3
7	Wallet	5
8	Vringo	2
9	laser-based	1
10	EMC.N	1
11	end-to-end	2
12	ad-blocking	2
13	Waps	2
14	Osaka-based	1
15	eminent	1
16	BEAV.O	5
17	nitride	2
18	county-owned	3
19	incomparable	2

Fig. 7 The result of Politics (Case2)