Text Categorization Methods Application for Natural Language Call Routing

Roman Sergienko[1], Tatiana Gasanova[1], Eugene Semenkin[2] and Wolfgang Minker[1]

[1]*Institute of Communications Engineering, Ulm University, Albert Einstein-Allee 43, 89081 Ulm, Germany*

[2]*Department of System Analysis and Operation Research, Siberian State Aerospace University,*
Krasnoyarskiy Rabochiy Avenue 31, 660014 Krasnoyarsk, Russian Federation

Keywords: Text Classification, Call Routing, Text Preprocessing, Natural Language Processing, Features Selection.

Abstract: Natural language call routing can be treated as an instance of topic categorization of documents after speech recognition of calls. This categorization consists of two important parts. The first one is text preprocessing for numerical data extraction and the second one is classification with machine learning methods. This paper focuses on different text preprocessing methods applied for call routing. Different machine learning algorithms with several text representations have been applied for this problem. A novel text preprocessing technique has been applied and investigated. Numerical experiments have shown computational and classification effectiveness of the proposed method in comparison with standard techniques. Also a novel features selection method was proposed. The novel features selection method has demonstrated some advantages in comparison with standard techniques.

1 INTRODUCTION

Natural language call routing remains a complex and challenging research area in machine intelligence and language understanding. This problem is important and topical for modern automatic call service design. A number of works have recently been published on natural language call routing: (Chu-Carroll and Carpenter, 1999), (Kuo and Lee, 2003), (Witt, 2011), (Jan and Kingsbury, 2010), (Sarikaya et al., 2011).

Generally natural language call routing can be considered as two different problems. The first one is speech recognition of calls (transforming speech to the text) and the second one is call categorization for further routing. This paper focuses on text categorization methods applied for call routing.

Text classification can be considered to be a part of natural language understanding, where there is a set of predefined categories and the task is to automatically assign new documents to one of these categories. The method of text preprocessing and text representation influences the results that are obtained even with the same classification algorithms.

The most popular model for text classification is vector space model. In this case text categorization may be considered as a machine learning problem. Complexity of text categorization with vector space model is compounded by the need to extract the numerical data from text information before applying machine learning methods. Therefore text categorization consists of two parts: text preprocessing and classification using obtained numerical data.

There exist more advanced approaches for text preprocessing such as TF-IDF (Salton and Buckley, 1988) and Confident Weights(Soucy and Mineau, 2005). A novel text preprocessing method (Gasanova et al., 2013) is also considered, which has some similarities with ConfWeight method, but has improved computational effectiveness. It is important to notice that we use no morphological or stop-word filtering before text preprocessing. It means that text preprocessing can be performed without expert or linguistic knowledge and text preprocessing is language-independent. Term weighting can be also applied as features selection method because we can ignore terms with low weight values. It provides dimensionality reduction.

In this paper we have used k-nearest neighbours, Bayes classifier, fast large margin based on support vector machine (SVM) (Fan et al., 2008), and neural network as classification methods. RapidMiner (Shafait et al., 2010) has been used as implementation software.

As a call routing problem we use a database from

Sergienko R., Gasanova T., Semenkin E. and Minker W..
Text Categorization Methods Application for Natural Language Call Routing.
DOI: 10.5220/0005139708270831
In *Proceedings of the 11th International Conference on Informatics in Control, Automation and Robotics* (ASAAHMI-2014), pages 827-831
ISBN: 978-989-758-040-6

the company Speech Cycle. Calls are represented as a text after speech recognition. The utterances were manually transcribed and classified into 20 classes (call reasons), such as appointments, operator, bill, internet, phone or video. Calls that cannot be routed certainly to one reason of the list are classified to class TE-NOMATCH.

We have investigated text categorization for call routing with different text preprocessing and machine learning methods. The main aim of investigation is to evaluate the competitiveness of the novel text preprocessing method (Gasanova et al., 2013) and the novel features selection method in comparison with state-of-the-art techniques. The criteria are classification efficacy (macro F-measure) and computational time.

This paper is organized as follows: In Section 2, we describe the problem and the database. Section 3 describes text preprocessing methods. The classification algorithms and features selection methods are presented in Section 4. Section 5 reports on the experimental results. Finally, we provide concluding remarks in Section 6.

2 CORPORA DESCRIPTION

The data for testing and evaluation consists of about 300.000 user utterances recorded in English language from caller interactions with commercial automated agents. Utterances from this database are manually labelled by experts and divided into 20 classes (TE-NOMATCH, appointments, operator, bill, internet, phone etc.) Class TE-NOMATCH includes utterances that cannot be put into another class or can be put into more than one class. The database is also unbalanced, some classes include much more utterances than others (the largest class TE-NOMATCH includes 27.85% utterances and the smallest one consists of only 0.16% utterances). A lot of calls contain only one or two words.

Utterance duplicates were removed. The preprocessed database consisting of 24458 utterances was divided into a training (22020 utterances, 90.03%) and test set (2438 utterances, 9.97%) such that the percentage of classes remained the same in both sets. The size of the dictionary of the whole database is 3464 words, 3294 words appear in the training set, 1124 words appear in the test set, 170 words which appear only in the test set and do not appear in the training set (unknown words), 33 utterances consisted of only unknown words, and 160 utterances included at least one unknown word.

3 TEXT PREPROCESSING METHODS

3.1 Binary Preprocessing

The simplest approach is to take each word of the document as a binary coordinate and the size of the feature space will be the size of our vocabulary.

3.2 TF-IDF

TF-IDF (term frequency - inverse document frequency) is a well-known approach for text preprocessing based on multiplication of term frequency tf_{ij} (ratio between the number of times i^{th} word occurs in j^{th} document and the document size) and inverse document frequency idf_i.

$$tf_{ij} = \frac{t_{ij}}{T_j}, \tag{1}$$

where t_{ij} is the number of times the i^{th} word occurs in the j^{th} document. T_j is the document size (number of the words in the j^{th} document).

There are different ways to calculate the weight of each word. In this paper we run classification algorithms with the following variant.

$$idf_i = \log \frac{D}{n_i}, \tag{2}$$

where D is the number of documents in the training set and n_i is the number of documents that have the i^{th} word.

3.3 ConfWeight

Maximum Strength (Maxstr) is an alternative method to find the word weights. This approach has been proposed in (Soucy and Mineau, 2005). It implicitly does feature selection since all frequent words have zero weights. The main idea of the method is that the feature f has a non-zero weight in the class c only if the f frequency in documents of the c class is greater than the f frequency in all other classes. The ConfWeight method uses Maxstr as an analogy of IDF:

$$CW_{ij} = \log(tf_{ij} + 1) \cdot Maxstr(i). \tag{3}$$

Numerical experiments (Soucy and Mineau, 2005) have shown that the ConfWeight method is more effective than TF-IDF with SVM and k-NN as classification methods. The main drawback of the ConfWeight method is computational complexity. This method is more computationally demanding than

TF-IDF method because the ConfWeight method requires time-consuming statistical calculations such as Student distribution calculation and confidence interval definition for each word.

3.4 Novel Term Relevance Estimation (TRE)

The main idea of the method (Gasanova et al., 2013) is similar to ConfWeight but it is not so time-consuming. The idea is that every word that appears in the article has to contribute some value to the certain class and the class with the biggest value we define as a winner for this article.

For each term we assign a real number term relevance that depends on the frequency in utterances. Term weight is calculated using a modified formula of fuzzy rules relevance estimation for fuzzy classifiers (Ishibuchi et al., 1999). Membership function has been replaced by word frequency in the current class. The details of the procedure are the following:

Let L be the number of classes; n_i is the number of articles which belong to the i^{th} class; N_{ij} is the number of there j^{th} word occurrence in all articles from the i^{th} class; $T_{ji} = N_{ji}/n_i$ is the relative frequency of the j^{th} word occurrence in the i^{th} class.

$R_j = max_i(T_{ji})$, $S_j = arg(max_i(T_{ji}))$is the number of class which we assign to the j^{th} word;

The term relevance, C_j, is given by

$$C_j = \frac{1}{\sum_{i=1}^{L} T_{ji}} \cdot \left(R_j - \frac{1}{L-1} \cdot \sum_{i=1, i \neq S_j}^{L} T_{ji} \right). \qquad (4)$$

C_j is higher if the word occurs more often in one class than if it appears in many classes. We use novel TW as an analogy of IDF for text preprocessing.

The learning phase consists of counting the C values for each term; it means that this algorithm uses the statistical information obtained from the training set.

4 CLASSIFICATION ALGORITHMS AND FEATURES SELECTION

We have considered 4 different text preprocessing methods (binary representation, TF-IDF, ConfWeight and novel TRE method) and compared them using different classification algorithms. The methods have been implemented using RapidMiner (Shafait et al., 2010). The classification methods are:

-k-nearest neighbours algorithm with weighted vote (we have varied k from 1 to 15);

-kernel Bayes classifier with Laplace correction;

-neural network with error back propagation (standard setting in RapidMiner);

-fast large margin based on support vector machine (FLM) (standard setting in RapidMiner).

We use macro F-score as a criterion of classification effectiveness. Precision for each class i is calculated as the number of correctly classified articles for class i divided by the number of all articles which algorithm assigned for this class. Recall is the number of correctly classified articles for class i divided by the number of articles that should have been in this class. Overall precision and recall are calculated as the arithmetic mean of the precisions and recalls for all classes (macro-average). F-score is calculated as the harmonic mean of precision and recall.

Term weighting provides also a simple features selection. We can ignore terms with low weight values (idf, Maxstr, or novel TW). In our paper we propose a novel feature selection method using term weighting. This method can be applied only for text classification problems. At first we calculate relative frequency of each word in each class with the training sample. After that we choose the class with the maximum value of the relative frequency for each word. Therefore, each word in the vocabulary has the corresponding class. When we get a new text for classification we calculate for each class the sum of the weights of the words which belong to the this class. After this procedure we have number of attributes equals to number of classes. Therefore, the method provides very small number of attributes. Also the method can be applied for binary preprocessing (standard features selection is impossible for binary preprocessing).

5 RESULTS OF NUMERICAL EXPERIMENTS

We have implemented 4 different text preprocessing methods (binary method, TF-IDF, ConfWeight and the novel TRE method). At first we have measured computational effectiveness of each text preprocessing technique. We have tested each method 20 times with the same computer (Intel Core i7 2.90 GHz, 8 GB RAM). Figure 1 compares computational times for different preprocessing methods.

We can see in Figure 1 that binary preprocessing is the fastest one. TF-IDF and the novel TRE are approximately one and a half times slower than binary preprocessing and they have almost the same computational efficiency. The most time-consuming method

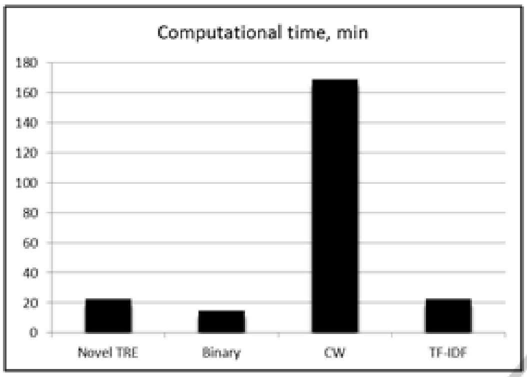

Figure 1: Computational effectiveness of the text preprocessing methods.

is ConfWeight (CW). It requires approximately eight times more time than TF-IDF and the novel TRE.

After that we implemented different classification algorithms for all text preprocessing techniques using RapidMiner. Bayes classifier and *k*-nearest neighbours algorithm can be applied without features selection. In this situation number of classes equals to size of the vocabulary. We applied these algorithms also with ignoring 50% of the words with the lowest values of the weights (for novel TW method also ignoring 90% of the words). Also we have applied the novel features selection method that allows to use number of attributes equals to number of classes.

For artificial neural networks and support vector machine the dimensionality is more critical than for Bayes classifier and *k*-nearest neighbours algorithm. Therefore, these classification algorithms were applied only with the novel feature selection method.

Table 1 present the F-scores obtained on the test corpora for different text preprocessing methods and different classification algorithms with different features selection techniques. The best value is shown in bold. Results of *k*-nearest neighbours algorithm are presented with the best value of *k*. The novel features selection method is identified as ”sums”.

We can see in Table 1 that the best classification accuracy is provided with novel TRE approach as text preprocessing and *k*-NN as classification algorithm (*k*=4). It is close to the results with ConfWeight and *k*-NN but the novel TRE approach is more efficient for computation. We can also conclude that the *k*-NN algorithm is the best one for all text preprocessing methods. Binary preprocessing is more effective than TF-IDF with all classification methods. This can be explained by the fact that the database contains very short calls (often only one word) and repeatability of words in one call is close to zero. TF-IDF is more appropriate for large documents with a large number of repetitive words.

Table 1: The results of the numerical experiments

Method	***k*-NN**	**Bayes**	**Neural network**	**FLM**
Binary (all words)	0.663	0.667	-	-
Binary (sums)	0.534	0.522	0.524	0.478
TF-IDF (all words)	0.625	0.573	-	-
TF-IDF (50% words)	0.212	0.229	-	-
TF-IDF (sums)	0.537	0.537	0.488	0.440
CW (all words)	0.720	0.404	-	-
CW (50% words)	0.392	0.193	-	-
CW (sums)	0.716	0.691	0.680	0.594
Novel TRE (all words)	**0.721**	0.662	-	-
Novel TRE (50% words)	0.716	0.667	-	-
Novel TRE (10% words)	0.623	0.555	-	-
Novel TRE (sums)	0.670	0.627	0.593	0.561

The numerical results have shown advantages of the novel features selection methods. The standard features selection method with term weighting does not allow to get appropriate results with the ConfWeight and the TF-IDF preprocessing methods, it is possible only with the novel TRE (it is an additional advantage of the novel TRE). The novel features selection method works appropriately with all text preprocessing techniques and allows to use very small number of attributes.

6 CONCLUSIONS

This paper reported on call classification experiments on large corpora using different text preprocessing methods and classification methods. We have tested binary representation, ConfWeight, TF-IDF and the novel term relevance estimation approach as preprocessing techniques. We have used *k*-NN, Bayes approach, neural network, and fast large margin based on support vector machine as classification algorithms.

Numerical experiments have shown that ConfWeight method is more effective for classification as TF-IDF and binary preprocessing but this method is more time-consuming. The novel term relevance estimation method allows reaching better classification effectiveness than ConfWeight. Computational effectiveness of the novel TRE is the same as computational efficiency of TF-IDF.

Also the novel features selection method was proposed. The method allow to use very small number of attributes (equals to number of classes) and allow to get appropriate classification results for all text preprocessing methods.

REFERENCES

Chu-Carroll, J. and Carpenter, B. (1999). Vector-based natural language call routing. *Computational linguistics*, 25(3):361–388.

Fan, R.-E., Chang, K.-W., Hsieh, C.-J., Wang, X.-R., and Lin, C.-J. (2008). Liblinear: A library for large linear classification. *The Journal of Machine Learning Research*, 9:1871–1874.

Gasanova, T., Sergienko, R., Semenkin, E., Minker, W., and Zhukov, E. (2013). A semi-supervised approach for natural language call routing. *Proceedings of the SIGDIAL 2013 Conference*, pages 344–348.

Ishibuchi, H., Nakashima, T., and Murata, T. (1999). Performance evaluation of fuzzy classifier systems for multidimensional pattern classification problems. *Systems, Man, and Cybernetics, Part B: Cybernetics, IEEE Transactions on*, 29(5):601–618.

Jan, E.-E. and Kingsbury, B. (2010). Rapid and inexpensive development of speech action classifiers for natural language call routing systems. In *Spoken Language Technology Workshop (SLT), 2010 IEEE*, pages 348–353. IEEE.

Kuo, H.-K. J. and Lee, C.-H. (2003). Discriminative training of natural language call routers. *Speech and Audio Processing, IEEE Transactions on*, 11(1):24–35.

Salton, G. and Buckley, C. (1988). Term-weighting approaches in automatic text retrieval. *Information processing & management*, 24(5):513–523.

Sarikaya, R., Hinton, G. E., and Ramabhadran, B. (2011). Deep belief nets for natural language call-routing. In *Acoustics, Speech and Signal Processing (ICASSP), 2011 IEEE International Conference on*, pages 5680–5683. IEEE.

Shafait, F., Reif, M., Kofler, C., and Breuel, T. M. (2010). Pattern recognition engineering. In *RapidMiner Community Meeting and Conference*, volume 9.

Soucy, P. and Mineau, G. W. (2005). Beyond tfidf weighting for text categorization in the vector space model. In *IJCAI*, volume 5, pages 1130–1135.

Witt, S. M. (2011). Semi-automated classifier adaptation for natural language call routing. In *INTERSPEECH*, pages 1341–1344.