A Semi-supervised Approach for Natural Language Call Routing

Tatiana Gasanova
Institute of Communications Engineering, Ulm University, Germany
tatiana.gasanova@uni-ulm.de

Roman Sergienko
Institute of Computer Science and Telecommunications, Siberian State Aerospace University, Russia
romaserg@list.ru

Wolfgang Minker
Institute of Communications Engineering, Ulm University, Germany
wolfgang.minker@uni-ulm.de

Eugene Zhukov
Institute of Computer Science and Telecommunications, Siberian State Aerospace University, Russia
zhukov.krsk@gmail.com

Eugene Semenkin
Institute of Computer Science and Telecommunications, Siberian State Aerospace University, Russia
eugenesemenkin@yandex.com

Abstract

Natural Language call routing remains a complex and challenging research area in machine intelligence and language understanding. This paper is in the area of classifying user utterances into different categories. The focus is on design of algorithm that combines supervised and unsupervised learning models in order to improve classification quality. We have shown that the proposed approach is able to outperform existing methods on a large dataset and do not require morphological and stop-word filtering. In this paper we present a new formula for term relevance estimation, which is a modification of fuzzy rules relevance estimation for fuzzy classifier. Using this formula and only 300 frequent words for each class, we achieve an accuracy rate of 85.55% on the database excluding the “garbage” class (it includes utterances that cannot be assigned to any useful class or that can be assigned to more than one class). Dividing the “garbage” class into the set of subclasses by agglomerative hierarchical clustering we achieve about 9% improvement of accuracy rate on the whole database.

1 Introduction

Natural language call routing can be treated as an instance of topic categorization of documents (where the collection of labeled documents is used for training and the problem is to classify the remaining set of unlabeled test documents) but it also has some differences. For instance, in document classification there are much more terms in one object than in single utterance from call routing task, where even one-word utterances are common.

A number of works have recently been published on natural language call classification. B. Carpenter, J. Chu-Carroll, C.-H. Lee and H.-K. Kuo proposed approaches using a vector-based information retrieval technique, the algorithms designed by A. L. Gorin, G. Riccardi, and J. H. Wright use a probabilistic model with salient phrases. R. E. Schapire and Y. Singer focused on a boosting-based system for text categorization.

The most similar work has been done by A. Albalate, D. Suendermann, R. Pieraccini, A. Suchindranath, S. Rhinow, J. Liscombe, K. Dayanidhi, and W. Minker. They have worked on the data with the same structure: the focus was on the problem of big part of non-labeled data and only few labeled utterances for each class, methods of matching the obtained clusters and the given classes have also been considered; they provided the comparison of several classification methods that are able to perform on the large scale data.

The information retrieval approach for call routing is based on the training of the routing matrix, which is formed by statistics of appearances of

Proceedings of the SIGDIAL 2013 Conference, pages 344–348,
Metz, France, 22-24 August 2013. ©2013 Association for Computational Linguistics

words and phrases in a training set (usually after morphological and stop-word filtering). The new caller request is represented as a feature vector and is routed to the most similar destination vector. The most commonly used similarity criterion is the cosine similarity. The performance of systems, based on this approach, often depends on the quality of the destination vectors.

In this paper we propose a new term relevance estimation approach based on fuzzy rules relevance for fuzzy classifier (H. Ishibuchi, T. Nakashima, and T. Murata., 1999) to improve routing accuracy. We have also used a decision rule different from the cosine similarity. We assign relevancies to every destination (class), calculate the sums of relevancies of words from the current utterance and choose the destination with the highest sum.

The database for training and performance evaluation consists of about 300.000 user utterances recorded from caller interactions with commercial automated agents. 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.*

A significant part of the database (about 27%) consists of utterances from the “garbage” class (*_TE_NOMATCH*). Our proposed approach decomposes the routing task into two steps. On the first step we divide the “garbage” class into the set of subclasses by one of the clustering algorithms and on the second step we define the call reason considering the “garbage” subclasses as separate classes. We apply genetic algorithms with the whole numbers alphabet, vector quantization network and hierarchical agglomerative clustering in order to divide “garbage” class into subclasses. The reason to perform such a clustering is due to simplify the detection of the class with non-uniform structure.

Our approach uses the concept of salient phrases: for each call reason (class) only 300 words with the highest term relevancies are chosen. It allows us to eliminate the need for the stop and ignore word filtering. The algorithms are implemented in C++.

As a baseline for results comparison we have tested some popular classifiers from RapidMiner, which we have applied to the whole database and the database with decomposition.

This paper is organized as follows: In Section II, we describe the problem and how we perform the preprocessing. Section III describes in detail the way of the term relevance calculating and the possible rules of choosing the call class. In Section IV we present the clustering algorithms which we apply to simplify the “garbage” class detection. Section V reports on the experimental results. Finally, we provide concluding remarks in Section VI.

2 Problem Description and Data Preprocessing

The data for testing and evaluation consists of about 300.000 user utterances recorded from caller interactions with commercial automated agents. Utterances from this database are manually labeled 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 6790 utterances and the smallest one consists of only 48 utterances).

The initial database has been preprocessed to be a binary matrix with rows representing utterances and columns representing the words from the vocabulary. An element from this binary matrix, aij, equals to 1 if in utterance i the word j appears and equals to 0 if it does not appear.

Utterance duplicates were removed. The preprocessed database consisting of 24458 utterances was divided into train (22020 utterances, 90,032%) and test set (2438 utterances, 9,968%) 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 training set, 1124 words appear in test set, 170 words which appear only in test set and do not appear in training set (unknown words), 33 utterances consisted of only unknown words, and 160 utterances included at least one unknown word.

3 Term Relevance Estimation

For each term we assign a real number term relevance that depends on the frequency in utterances. Term relevance is calculated using a modified formula of fuzzy rules relevance estimation for fuzzy classifier. Membership function has been replaced by word frequency in the current class. The details of the procedure are:

Let L be the number of classes; n_i is the number of utterances of the $_i$th class; N_{ij} is the number of *j*th word occurrence in all utterances of the *i*th class; $T_{ji}=N_{ji}/n_i$ is the relative frequency of *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 *j*th word;

The term relevance, C_j, is given by

$$C_j = \frac{1}{\sum_{i=1}^{L} T_{ji}} (R_j - \frac{1}{L-1} \sum_{\substack{i=1 \\ i \neq S_j}}^{L} T_{ji}).$$

C_j is higher if the word occurs often in few classes than if it appears in many classes.

The learning phase consists of counting the C values for each term, it means that this algorithm uses the statistical information obtained from train set. We have tested several different decision rules defined in Table 1.

	Decision rules	
RC	$A_i = \sum_{j:S_j=i} R_j C_j$	For each class *i* we calculate A_i Then we find the number of class which achieves maximum of A_i $winner = \arg(\max_i A_i)$
RC max	$A_i = \sum_{j:S_j=i} \max R_j C_j$	
C	$A_i = \sum_{j:S_j=i} C_j$	
C with limit	$A_i = \sum_{\substack{j:S_j=i \\ C_j>const}} C_j$	
R	$A_i = \sum_{j:S_j=i} R_j$	

Table 1. Decision Rules

The best obtained accuracies is achieved with the decision rule C, where the destination is chosen that has the highest sum of word relevancies from the current utterance. In Table 2 we show the obtained results on the whole database and database without "garbage" class.

	Train	Test
With class "garbage"	0,614	0,551
Without class "garbage"	0,887	0,855

Table 2. Performance of the new TRE approach

4 Clustering methods

After the analysis of the performances of standard classification algorithms on the given database, we can conclude that there exists one specific class (class *_TE_NOMATCH*) where all standard techniques perform worse. Due to the non-uniform structure of the "garbage" class it is difficult to detect the whole class by the proposed procedure. If we apply this procedure directly we achieve only 55% of accuracy rate on the test data (61% on the train data). We suggest to divide the "garbage" class into the set of subclasses using one of the clustering methods and then recount the values of C_j taking into account that there are 19 well defined classes and that the set of the "garbage" subclasses can be consider as separate classes.

In this paper the following clustering methods are used: a genetic algorithm with integers, vector quantization networks trained by a genetic algorithm, hierarchical agglomerative clustering with different metrics.

4.1 Genetic Algorithm

The train set accuracy is used as a fitness function. Each individual is the sequence of nonnegative integer numbers (each number corresponds to the number of "garbage" subclass). The length of this sequence is the number of utterances from train set which belong to the "garbage" class.

We apply this genetic algorithm to find directly the optimal clustering using different numbers of clusters and we can conclude that with increasing the clusters number (in the "garbage" class) we get better classification accuracy on the whole database. We have used the following parameters of GA: population size = 50, number of generation = 50, weak mutation, tournament selection, uniform crossover, averaged by 50 runs. Applying this method we achieve about 7% improvement of accuracy rate on train data and about 5% on test data.

4.2 Vector Quantization Network

We have also implemented vector quantization network. For a given number of subclasses we search for the set of code vectors (the number of code vectors is equal to the number of subclasses). These code vectors are optimized using genetic algorithm where as a fitness function we use the classification quality on the train set. Each code vector corresponds to a certain "garbage" subclass. The object belongs to the subclass if the distance between it and the corresponding code vector is smaller than the distances between the object and all other code vectors. Applying this algorithm to the given database we obtain results similar to the results of the genetic algorithm.

4.3 Hierarchical Agglomerative Clustering

In this work we consider hierarchical agglomerative binary clustering where we set each utterance to one subclass and then we consequently group classes into pairs until there is only one class containing all utterances or until we achieve a certain number of classes. The performance of hierarchical clustering algorithms depends on the metric (the way to calculate the distance between objects) and the criterion for clusters union. In this work we use Hamming metric and Ward criterion (J. Ward. 1963).

5 Experimental results

The approach described above has been applied on the preprocessed corpus which has been provided by Speech Cycle company. We propose that only terms with highest value of RC (product of R and C) are contributed to the total sum. We have investigated the dependence of the new TRE approach on the frequent words number (Figure 1). The best accuracy rate was obtained with more than 300 frequent words. By using only limited set of words we eliminated the need of stop and ignore words filtering. This also shows that the method works better if utterance includes terms with high C values. This approach requires informative well-defined classes and enough data for statistical model.

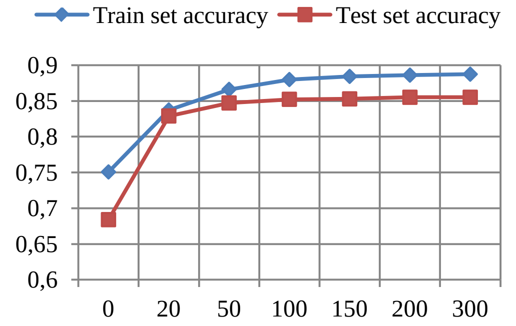

Figure 1. New TRE approach with different numbers of frequent words (x-axis: number of frequent words; y-axis: accuracy)

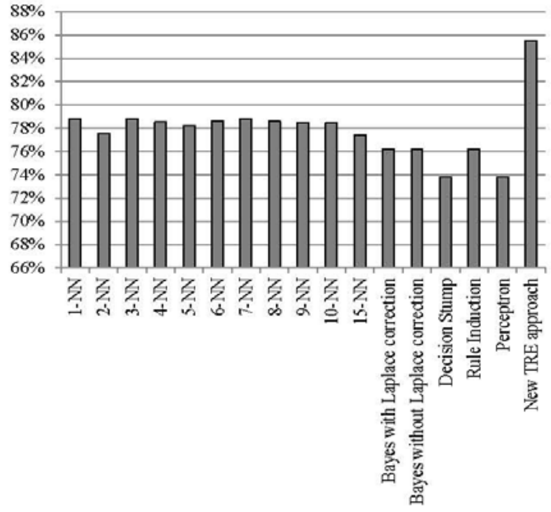

Figure 2. Overall accuracy

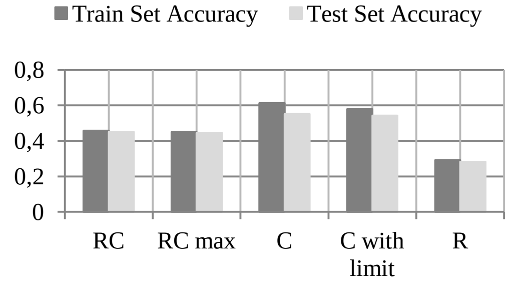

Figure 3. Comparison of decision rules (x-axis: decision rule; y-axis: accuracy)

We have tested standard classification algorithms (k-nearest neighbors algorithms, Bayes classifiers, Decision Stump, Rule Induction, perceptron) and the proposed approach on the database with “garbage” class and on the database without it (Figure 2). The proposed algorithm outperforms all other methods with has an accuracy rate of 85.55%. Figure 3 provides accuracies of different decision rules. Applying the proposed formula to the whole database we obtain 61% and 55% of classification quality on train and test data. We should also mention that the common tf.idf approach gives us on the given data 45% and 38% of accuracy rate on the train and test data. The proposed approach performs significantly better on this kind of data.

Using the agglomerative hierarchical clustering we achieve about 9% improvement. The best classification quality is obtained with 35 subclasses on the train data (68.7%) and 45 subclasses on the test data (63.9%). Clustering into 35 subclasses gives 63.7% of accuracy rate on the test data.

6 Conclusion

This paper reported on call classification experiments on large corpora using a new term relevance estimation approach. We propose to split the classification task into two steps: 1) clustering of the “garbage” class in order to simplify its detection; 2) further classification into meaningful classes and the set of “garbage” subclasses. The performance of the proposed algorithm is compared to several standard classification algorithms on the database without the “garbage” class and found to outperform them with the accuracy rate of 85.55%.

Dividing the “garbage” class into the set of subclasses by genetic algorithm and vector quantization network we obtain about 5% improvement of accuracy rate and by agglomerative hierarchical clustering we achieve about 9% improvement of accuracy rate on the whole database.

References

A. Albalate, D. Suendermann, R. Pieraccini, and W. Minker. 2009. *Mathematical Analysis of Evolution, Information, and Complexity*, Wiley, Hoboken, USA.

A. Albalate, D. Suendermann D., and W. Minker. 2011. *International Journal on Artificial Intelligence Tools*, 20(5).

A. Albalate, A. Suchindranath, D. Suendermann, and W. Minker. 2010. *Proc. of the Interspeech 2010, 11th Annual Conference of the International Speech Communication Association*, Makuhari, Japan.

A. Albalate, S. Rhinow, and D. Suendermann. 2010. *Proc. of the ICAART 2010, 2nd International Conference on Agents and Artificial Intelligence*, Valencia, Spain.

A.L. Gorin, G. Riccardi, and J. H. Wright. 1997. *Speech Commun*., vol. 23, pp. 113–127.

B. Carpenter and J. Chu-Carroll. 1998. *Proc. ICSLP*-98, pp. 2059–2062.

C.-H. Lee, B. Carpenter, W. Chou, J. Chu-Carroll, W. Reichl, A. Saad, and Q. Zhou. 2000. *Speech Commun*., vol. 31, no. 4, pp. 309–320.

D. Suendermann, J. Liscombe, K. Dayanidhi, and R. Pieraccini. 2009. *Proc. of the SIGDIAL 2009*, London, UK.

H. Ishibuchi, T. Nakashima, and T. Murata. 1999. *Trans. on Systems, Man, and Cybernetics*, vol. 29, pp. 601-618.

H.-K. Kuo and C.-H. Lee. 2000. *Proc. of ICSLP'00*.

J. Chu-Carroll and B. Carpenter. 1999. *Comput. Linguist*., vol. 25, no. 3, pp. 361- 388.

J. Ward. 1963. *Journal of the American Statistical Association*, 58 (301): 236-244.

J. H. Wright, A. L. Gorin, and G. Riccardi. 1997. *Proc. Eurospeech-97*, pp. 1419–1422.

K. Evanini, D. Suendermann, and R. Pieraccini. 2007. *Proc. of the ASRU 2007*, Kyoto, Japan.

R. E. Schapire and Y. Singer. 2000. *Mach. Learn*., vol. 39, no. 2/3, pp. 135–168.