A Oversampling Method for Label Imbalanced Text

Yong Wang[1, a], Jingyan Lin[1, b] and Ying Wang[1, c]

[1]School of computers, Guangdong University of Technology, Guangzhou 510006, China.

[a]wangyong@gdut.edu.cn, [b]609633396@qq.com, c13610143113@126.com

Abstract

Text classification is the most common task in the field of natural language processing. In the text classification task, label imbalanced datasets are often encountered. If the traditional text sampling method is adopted, the categories with few samples are often ignored and are easily misclassfied, which greatly affect the result of the classification task. For this reason, we propose a new oversampling method for the minority class samples to alleviate the problem of too few samples in some categories. For a text of a minority class, first we find out the synonym of a word according to the pre-trained Word2Vec model. By means of replacing the word with its synonym, we can obtain new samples. Then we develop a CNN text classifier for this task. The result of the experiment show that compare with the tradictional oversampling method, the new oversampling method can improve the accuracy and F1-Score of the classification, effectively avoiding overfitting.

Keywords

Label imbalanced, Oversampling, Word2vec, CNN.

1. Introduction

Text classification is a base task in the field of natural language processing and has always been a hot topic of research.When a training set with an imbalanced category is encountered, the features extracted by major classes are richer than those of a few. A sample tend to be classified as majority class and minority classes are misclassified[1]. Improving the classification accuracy and recall rate of minority classes has become an urgent problem in text categorization tasks. At data level, there are mainly oversampling, undersampling and feature selection[2]. At model level, the general methods are cost-sensitive learning and model integration[3]. Cost-sensitive learning improves the accuracy of minority classification by increasing the cost of incorrect classification of minority samples, while model integration combines multiple models to get the final prediction results, such as Boosting[4].

Inspired by SMOTE algorithm[5], this paper proposes a new oversampling method for class imbalanced text classification. According to the pre-trained Word2vec model, similar words of the keywords are found as synonyms and replace the keywords, new samples are obtained.

2. Related Work

He et al.[6] proposed a oversampling method called ADASYN, which adaptively adjusts the oversampling ration of the minority classes samples according to the density distribution. The features of the majority classes are retained as much as possible while the number of the samples are reduced. Sheng et al.[7] Proposed IDP-SMOTE oversampling method combined with density peaks clustering and SMOTE algorithm.

For feature selection, M Wasikowski et al.[8] compared three feature selection methods in labels imbalanced text datasets, and found that S2N and FAST are better. Zhang et al.[9] proposed a feature selection method to distinguish the majority classes from the minority classes using the minimum probability difference of the document as the scoring criterion. Wang et al.[10] proposed a new two-side fisher(TSF) feature selection method. TSF can control combination of positive features and negative features explicitly and tackle the imbalanced problem.

3. CNN Text Classifier

The CNN text classifier[11] is mainly composed of input layer, embedding layer, convolution layer, pooling layer, dropout layer[12], fully connected layer and softmax layer. The structure of CNN text classifier is shown in Fig. 1.

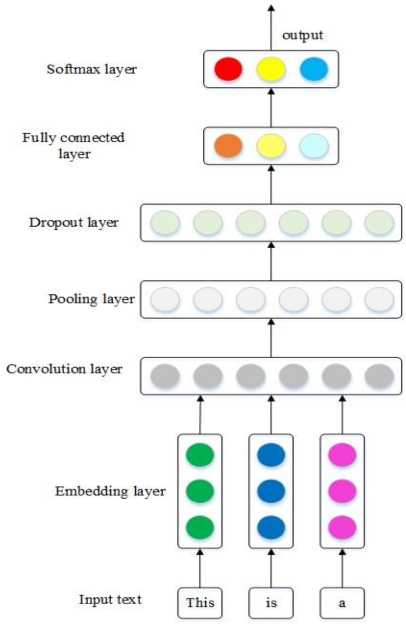

Fig. 1 Structure of the CNN text classifier

Let x_i be the k-dimension word vector which represent the i-th word in the sentence. A sentence of length n can be represent as:

$$S_{i:n} = x_1 \oplus x_2 \oplus \ldots \oplus x_n \tag{1}$$

Then we will apply a convolution operation.In text categorization task, convolution operation can be used to extract ngrams features of text. A filter w is applied to a window of h words to extract feature which can represent as $S_{i:i+h-1} \in \mathbb{R}^{h\times k}$. A feature generated from a window of words by:

$$c_i = f(w * S_{i:i+h-1} + b) \tag{2}$$

f is non-linear function and generally is tanh or relu. b is a bias term. The convolution filter is used in each window of words in the sentence to get new feature map:

$$c = [c_1, c_2, \ldots, c_{n-h+1}] \tag{3}$$

We then apply a max pooling operation over the feature map and get the value $\hat{c} = \max(c)$ which is corresponding to the feature capture by this filter. Each convolution filter can get a type of feature. In practice, in order to extract more features, we will use multiple filters.

Finally, y is input into the softmax layer to calculate the probability that the input sample belongs to each class.The parameters of the network are updated by back propagation algorithm, and the cross entropy loss function is chosen as the loss function. Let y_i be the label of the i-th sample and p_i be the prediction of the i-th sample, the loss function can represent by:

$$loss = -\sum_{i=1}^{n} y_i \log p_i \tag{4}$$

4. Oversampling Based on Word2vec

4.1 Introduction of Word2vec

Word2vec is a word embedding tool proposed by Mikolov et al.[13] in 2013. It can map words into low-dimensional space. By calculating the similarity between word vectors and measuring the relationship between them.

4.2 Oversampling Process

Generally, when the the number of majority class samples is 4 times greater than the minority class samples, we will consider that this is a label imbalanced dataset. The oversampling number of minority samples can be determined according to this ratio. If the ratio of the number of majority sample to the minority sample is less than four to one, do not oversample, otherwise oversample the minority sample, so that the ratio of the majority sample to the minority sample is four to one. For each class, the specific oversampling process is as follows:

Step 1: choose the classes with the largest number of samples as the majority class and the others as the minority class.

Step 2: determine the oversampling number m of a minority class. Let the number of samples of majority class is a and the number of samples of a minority class is b , then m can be caculated by:

$$m=\begin{cases} 0\,, & \frac{a}{b}\le 4 \\ \frac{a}{4}-b, & \frac{a}{b}>4 \end{cases} \tag{5}$$

Step 3: set k and t. k is the number of words to be replaced and t means that each keyword is randomly selected and replaced from its previous t synonyms according to Word2vec model.

Step 4: combine the new generated minority samples with the original minority samples.

5. Experiment

5.1 Dataset

We test our model on WenYing industry text classification dataset and the dataset is a brief introduction of each company. There are 11 types of data, 4774 samples in training set and 1301 samples in test set. The average number of words per sample is 81. There is a serious class imbalance problem in training data. The number of classes with the largest number of samples (class 4) is 23 times that of classes with the least number of samples (class 1). The number of different classes are shown in Table 1.

Table 1 The number of different classes

class	1	2	3	4	5	6	7	8	9	10	11
number	54	98	1271	1268	227	810	303	206	163	278	96

5.2 Experiment Setup and Hyperparameter

We train the Word2vec model on the training set, test set and Sougou News. The dimension of the word vector is 150. The convolutional neural network text classifier is built with Google open source deep learning framework Tensorflow. The width of the convolution filter is 3, 4, 5 and the number of each size is 128. The L2 regularization is set to 0.01. The dropout keep probability is 0.5 during training and 1 during testing.

The batch size is set to 64. Adam is used as the optimizer when updating parameters by back propagation. The training set is divided 10% as verification set when training, which is convenient for observing the training result and adjusting parameters.

In out experiment, accuracy and F1-Score were used to evaluate the performance of the oversampling method. Among them, F1-Score uses Micro F1-Score, and its calculation is as follows:

$$Mirco\ F1-Score = \frac{2 * p_{mic} * r_{mic}}{p_{mic} + r_{mic}} \tag{6}$$

p_{mic} is the micro precision and r_{mic} is the micro recall. Let TP be true positive and FP be false positive and FN be false negative. p_{mic} and r_{mic} can calculate respectively by:

$$p_{mic} = \frac{\sum_{i=1}^{n} TP_i}{\sum_{i=1}^{n} (TP_i + FP_i)} \tag{7}$$

$$r_{mic} = \frac{\sum_{i=1}^{n} TP_i}{\sum_{i=1}^{n} (TP_i + FN_i)} \tag{8}$$

Micro F1-Score takes into account the number of categories, so it can better evaluate the classification result in label imbalanced text classification.

5.3 Experiment Result and Analysis

In order to better compare the over-sampling method fused TFIDF and Word2vec with other oversampling methods, the following four oversampling methods are designed. We set k=6, t=4 in this experiment and m is the number of oversampling.

Method 1: do not oversample minority class samples.

Method 2: m samples are taken directly from a minority class samples.

Method 3: m samples were selected from a minority samples. k words are selected as words to be replaced. According to Word2vec model, one of the first t synonyms was randomly selected for replacement.

The evaluation indexes are accuracy and Micro F1-Score.We will shuffle the training set before training. The result of the three oversampling methods are as Table 2.

Table 2 Result of the four oversampling method.

Method	Accuracy	F1-Score
Without oversampling	0.8812	0.8692
Direct oversampling	0.8841	0.8753
Replace k keywords with its synonyms	0.8895	0.8868

The oversampling method propose by this paper performs best among these three oversampling methods, with the highest accuracy and F1-Score. Compared with without oversampling, the accuracy is improved by 0.8% and that of F1-Score by 1.7%. Compared with the oversampling method of direct oversampling, F1-Score improves by 1.1%, which further illustrate the importance of synonyms and the effectiveness of the oversampling method proposed in this paper. Compared with other oversampling methods, the oversampling method which replaces k words with their synonyms can effectively avoid overfitting and improve the accuracy and F1-Score.

6. Conclusion

In this paper, we propose a text over-sampling method which use Word2vec model to alleviate the misclassification of a minority class in imbalanced datasets. For a minority class samples, the synonyms of the words are found by pre-trained Word2vec model, and the words are replaced by its synonyms to generate new samples. Experiments on text categorization of imbalanced datasets show that compared with other oversampling methods, the proposed oversampling method has improved both in accuracy and F1-Score.

For future, we would like to improve the self-adaptability of the algorithm, which can automatically determine the values of m, k and t. Moreover, we will try to apply a text-based denoising algorithm to improve the quality of generated samples.

Acknowledgements

The research work is supported by Innovation Carrier Construction Project in Guangdong Province under grant No.2013B09030008.

References

[1] Ping G, Yang Y. Oversampling Algorithm Oriented to Subdivision of Minority Class in Imbalan ced Data Set. Computer Engineering, 43(02):241-247, 2017.

[2] Haixiang G, Yijing L, Shang J , et al. Learning from class-imbalanced data: Review of methods and applications. Expert Systems with Applications, 73:220-239, 2017.

[3] Nan Z, Xiaofang Z, Lijun Z. Overview of Imbalanced Data Classification. Computer S-cience, 4 5(S1):22-27+57, 2018.

[4] Qiuje L, Yaobing M, Zhiquan W. Research on Boosting-based Imbalanced Data Classifi-cation. Computer Science, 38(12):224-228, 2011.

[5] Chawla N V, Bowyer K W, Hall L O, et al. SMOTE: Synthetic Minority OverSampling Techiqu e. Journal of Artificial Intelligence Research, 16(1):321-357, 2002.

[6] He H, Bai Y, Garcia E , et al. ADASYN: Adaptive synthetic sampling approach for i-mbalanced learning. IEEE International Joint Conference on Neural Networks. 1322-1328, 2008.

[7] Kai S, Zhong L, Dechao Z, et.al. IDP-SMOTE resampling algorithm for imbalanced cl-assificati on. Application Research of Computers, (01):1-6, 2019.

[8] Wasikowski M, Chen X W. Combating the Small Sample Class Imbalance Problem Usi-ng Feat ure Selection. IEEE Transactions on Knowledge & Data Engineering, 22(10):1388-1400, 2010.

[9] Yanxiang Z, Haixia P. A Feature Selection Method Based on Discriminative Ability for Multicla ss Text Categorization on Imbalanced Data. Journal of Chinese Information Processi-ng, 29(04): 111-119, 2015.

[10] Jie W, Deyu, L, Suge W. TSF Feature Selection Method for Imbalanced Text Sentime-nt Class ification. Computer Science, 43(10):206-210, 2016.

[11] Yoon Kim. Convolutional Neural Networks for Sentence Classification. Conference on Empeir ical Methods in Natural Language Processing, 1746-1751, 2014.

[12] Srivastava N, Hinton G, Krizhevsky A, et al. Dropout: a simple way to prevent neural networks from overfitting. Journal of Machine Learning Research, 15(1):1929-1958, 2014.

[13] Mikolov, Tomas, et al. Efficient Estimation of Word Representations in Vector Space. Comput er Science, 2013.