COMP90049 Knowledge Technologies
Project 2: Geolocation of Tweets with Machine Learning

Anonymous

1 Introduction

As a increasing number of people are using social application, such as Facebook and Twitter. It is very valuable to analyse the information on these social applications appropriately and legally. Analysing Geolocation of tweets based on the content is one of most importance aspect. It can bring enormous value to social security and advertising industry.

This project is going to use different Machine Learning methods to classify those tweets into three different Geolocations based on the contents of those tweets. Besides, the project will analyse the results of these methods and make appropriate improvements.

2 Related Work

The analysis of text content through machine learning is a significant research direction, so there are many papers on internet which are helpful for me to better understand this project.

In this paper *You Are Where You Tweet: A Content-Based Approach to Geo-locating Twitter Users*, the author Cheng, Caverlee, and Lee (2010) analysed the local words in tweets, choosing some words which have strongly local information and use several machine learning models, such as Naive Bayes, SVM, AdaBoost, to calculate the results and analyze the results.

3 Data Sets

The data sets in this project are provided by Eisenstein, O'Connor, Smith, and Xing (2010). The format of these data is arff, which is a special document format used by Weka machine learning workbench (Holmes, Donkin, and Witten, 1994). There are three kinds of data sets for different purposes. The training data sets will be used for building a model. And this system will give one of three class to each instance of testing data set. The development data sets is used for evaluating whether a model is good enough.

Furthermore, the data sets were divided into two different types. The first type is according to the frequency of words showed in twitter, selecting the first N words that appear most frequently as attributes in this data sets, these data sets called mostXX. Another type is using Mutual Information and Chi-Square values to choose the best N terms as attributes for each class in this bestXX data sets, and remove the duplicate words.

4 Baseline

Baseline is an important reference object. If the performance of a algorithm is worse than the baseline, it is unacceptable. In order to evaluate the performance of this system, this project adopts R-0 as baseline. We use training data sets to train the model and apply the model to development data sets to get the results.

As the table show below.

Data sets	Percentage of correct classification	Data sets	Percentage of correct classification
Most10	64.4146 %	Best10	64.4146 %
Most20	64.4146 %	Best20	64.4146 %
Most50	64.4146 %	Best50	64.4146 %
Most200	64.4146 %	Best200	64.4146 %

Table 1: The result of baseline

As the results given by Weka, because most of the label in the class are NewYork, so all the classes predicted in the development data sets is NewYork, besides the instances of these files are the same. The results of baseline in these files are the same.

5 Evaluation Metrics

In this program, we will use three kinds of evaluation metrics to analyse the results of different machine learning methods. Also, we can see the performance of different machine learning methods for this task by using these metrics.

1. Accuracy: It is the proportion of instances whose label were correctly predicted. In Weka, the accuracy was displayed behind Correctly Classified Instances.

$$Accuracy = \frac{TP + TN}{TP + TN + FP + FN}$$

2. Precision: This is the proportion of correct classified instances when these instances were classified as positive one. Here is the formula of precision.

$$Precision = \frac{TP}{TP + FP}$$

3. Recall: This term is the proportion of predicted actual positive instances in all the actual positive instances.

$$Recall = \frac{TP}{TP + FN}$$

6 Methods

6.1 Naive Bayes

In this project, We use Naive Bayes classifier to classify the tweets instances into three different geographical locations based on these attributes abstracted from tweets instances. Naive Bayes classifier is a model of supervised learning and it is based on probability theory(McCallum, Nigam, et al., 1998).

6.2 The results of Naive Bayes

In the data sets, because the attribute tweet-id is the unique one, each instance has the different tweet-id, it is meaningless, we remove this attribute from files. The user-id is a important information to predict the geolocation, when a user has many tweets, we know the locations of these tweets are likely to be the same. But this attribute should be delete, it cannot generalize very well, when new users come in, they have different user-id. The rest of the attributes did not change.

As we can see, in table 2, if we use the Naive Bayes classifier to the original data sets, the accuracy is lower than the accuracy of baseline.

Data sets	Accuracy	Data sets	Accuracy
Most10	63.1333 %	Best10	63.1509 %
Most20	61.0938 %	Best20	62.0783 %
Most50	57.6114 %	Best50	60.0946 %

Table 2: The Accuracy of Naive Bayes

As the number of attributes increases, the accuracy decreases.

This is because most classes in training data sets are NewYork and it is also the same situation in development data sets. This results the accuracy of baseline is high. Besides some attributes in this data sets is not real independent, and many words do not have any geographical sense. These words are noise. It will reduce accuracy of results.

6.3 Decision Trees

Decision Tree is also a supervised learning method. It is a flow-chart-like tree structure. Its internal nodes represents the attributes of the test. The branches of decision trees are the attribute value of the test. The leaf nodes of the tree is the class labels(Quinlan, 1986).

6.4 The results of Decision Trees

Data sets	Percentage of correct classification	Data sets	Percentage of correct classification
Most10	64.4146 %	Best10	65.1405 %
Most20	64.0179 %	Best20	65.3227 %
Most50	62.3281 %	Best50	65.1493 %

Table 3: The Accuracy of Decision Trees

In the table above, we know that the accuracy of best data sets are better than most data sets and the accuracy of Zero-R, because the attributes in the best data are preprocessed by Mutual Information and Chi-Square. These attributes are more closely related to class than those in most data sets.

7 Features Engineering

As shown in the previous results, both in most data sets and best data sets, it is not the more attributes a model has, the more accurate it is.

So we will choose the Best10 data sets to do feature engineering.

Data sets	Attributes
Most10	a, i, im, lol, me, my, rt, the, to, u
Best10	and, are, atl, atlanta, bomb, childplease, da, famu, gsu, gw, haha, hahaha, hella, ii, inhighschool, la, lmaoo, lmaooo, lml, parody, rt, thatisall, the, wet

Table 4: The result of Decision Trees

From the table above, we can see that in Most10 data sets, these words do not have geographical sense. However, in data sets Best10, there are some words having geographical sense, such as 'atlanta', 'la', 'gsu' , 'la' is probably the abbreviation for Los Angeles, and 'gsu' is abbreviation for Georgia State University.

Through the analysis above, we can delete some useless words, in order to reducing the impact of these meaningless words to the accuracy of this model. We will use Decision Tree as a prediction model.

Removed Words	Accuracy (%)	Removed Words	Accuracy (%)
and	65.1258	are	65.1405
atl	65.0946	atlanta	65.0817
bomb	65.1317	child-please	65.1405
da	65.1405	famu	65.1052
gsu	65.1405	gw	65.1375
haha	64.8907	hahaha	64.9495
hella	65.0552	ii	65.1405
inhigh-school	65.1434	la	65.1111
lmaoo	65.1405	lmaooo	65.1405
lml	65.1404	parody	65.1405
rt	65.1464	thatisall	65.1258
the	65.1317	wet	65.1405

Table 5: The Accuracy of the model after removing the word

The table above is the accuracy after removing those words in Best10 data sets. Using original data sets, the accuracy of Decision Tree is 65.1405 %. However, when some words were removed, the accuracy of the decision tree were increased or did not change. These words are meaningless for this model or they are noises terms for this model, so we remove those words. The remaining words are list in the table below.

and	atl	atlanta	bomb
famu	gw	haha	hahaha
hella	la	thatisall	the

Table 6: The remaining terms in Best10 after processing

8 Results and Analysis

After processing Best10 data sets, we will use Decision Tree(J48 in weka) and Naive Bayes to train and test these processed data sets. The results are displayed below.

Methods	Accuracy	Precision (Weighted)	Recall (Weighted)
Naive Bayes	63.1774 %	49.2 %	63.2 %
Decision Tree	65.1464 %	65.8 %	65.1 %

Table 7: The results of two methods using Best10 after processing

The performance of the system is improved a little, but it is not obvious. Many reasons result in this situation.

Firstly, many instances of the data sets do not contain any feature attributes. That is all the attributes in these instances are '0', but the classes for these instances are different. For example, the tweet-id 914 and 1303, they do not contain these attributes, but one of them is NewYork, another is California. The data sparsity results the accuracy cannot be very high.

Secondly, although we have left some words closely related to geographical information, the number of these words in instances are too small to have a significant impact on the accuracy.

Thirdly, even some terms has strong geographical information, but it still mislead the model.

For example, the tweets-id 210983 in train-best10.arff, the instance has 'atlanta' this term, but location of this tweet is California.

Besides, the Accuracy, Precision(Weighted), and Recall(Weighted) of Decision Tree are higher than Naive Bayes, this phenomenon is because some words in tweets are not independent. Naive Bayes need each of the attribute to be independent. But in real tweets, some words must be put together in one sentence.

9 Conclusions

This project use Decision Tree (J48 in Weka) and Naive Bayes to classify tweets instances in data sets. At First, we use each method to calculate 6 different kinds of original data sets respectively. Then we choose one data sets doing feature engineering, applying these two same methods to the processed data sets. At last, the results was analysed.

We use the Zero-Rule as baseline and get the accuracy of this model in every data sets. However, the accuracy of Naive Bayes model is lower than baseline, this is because the data sets have a lot of meaningless words, these words will reduce the accuracy of Naive Bayes. Because the feature of the Zero-Rule, these words will not influence the behavior of Zero-Rule. The Accuracy of Decision Tree is higher than Naive Bayes is because these words are not real independent.

In processing the data sets, because we find that the number of attributes have little influence to the performance of this system, we choose Best10 as the data sets. After Processing the data sets, the accuracy of these two methods are all improved, but it is not very obvious. This is because this data set is too sparse, the attributes has little impact to accuracy.

If we can increase the weight of words related to geographic information, and find an appropriate way to solve the problem of data sparsity, then the classification accuracy will be further improved.

References

Cheng, Z., Caverlee, J., Lee, K. (2010). You are where you tweet: a content-based approach to geo-locating twitter users. In *Proceedings of the 19th acm international conference on information and knowledge management* (pp. 759–768).

Eisenstein, J., O'Connor, B., Smith, N. A., Xing, E. P. (2010). A latent variable model for geographic lexical variation. In *Proceedings of the 2010 conference on empirical methods in natural language processing* (pp. 1277–1287).

Holmes, G., Donkin, A., Witten, I. H. (1994). Weka: A machine learning workbench.

McCallum, A., Nigam, K., et al. (1998). A comparison of event models for naive bayes text classification. In *Aaai-98 workshop on learning for text categorization* (Vol. 752, pp. 41–48).

Quinlan, J. R. (1986). Induction of decision trees. *Machine learning*, *1*(1), 81–106.