COMP30027 Report

1. Introduction

Online data is growing rapidly these days, and it comes in many forms, with text tweets playing an important part. By analyzing a user's tweets, we can get different kinds of information about that user. In this project, with the absence of metadata, we tried to predict the location of the user when he/she posted the tweet by only analyzing the tweet text and training predict models.

2. Related literature

A considerable amount of work has been devoted to making geolocation prediction based on twits [1, 2]. For desktop machine geolocation, methods that map IP addresses to physical locations is quite popular [3]. But this kind of method cannot be applied in predicting mobile users' location. Text data tends to contain salient geospatial expressions that are particular to specific regions. Attempts to using this data directly have been based on analysis of gazetted expressions [4] or the identification of geographical entities [5].

3. Preprocess

The project's available data contains the top 10, top 50, and top 100 words selected based on mutual information and chi square values. However, through the experiment on development raw data, it was found that neither the top ten, the top one hundred, nor the top one thousand achieved good accuracy (based on the performance of Gaussian Naïve Bayes classifier, Multinomial Naïve Bayes classifier, Decision Tree classifier and Logistic Regression classifier). Open the data set and observe directly, one can find that the data appears in the form of one hot. A considerable part of the data values are all zeros, which means they do not contain any words in the feature set. This situation prompted the following steps: pre-filtering, feature selection, and tune hyper-parameters.

3.1 pre-filtering

Previous studies [6] based on twitter text usually included the step of pre-filtering. In previous studies, words that are not composed of alphabetic characters, those less than three characters in length, and those that occur no more than ten times are generally removed. However, other recent studies [7] have shown that short characters such as VIC, TAS (state codes), and non-alphabetic characters such as 3000, 3051, (post codes) are also of great value. Therefore, in this study, only words appearing less than ten times are removed, and non-alphabetic characters as well as short words are all kept.

3.2 feature weighting and selection

In the process of selecting features, the most critical part is to determine the score of this feature. Common options include using chi-square and mutual information to measure the value of a feature. Through experiments on the development data set, the features of one data set were composed of the first 1,000 words selected by chi-square, and the features of the other data set were composed of the first 1,000 words selected by mutual information. The accuracy evaluation of Gaussian Naïve Bayes, Multinomial Naïve Bayes, Decision Tree and Linear Regression shows that the accuracies of all models are higher when mutual information is used as feature evaluation baseline. Therefore, mutual information is used to pick the top 1,000 features.

3.3 tune hyper-parameters

3.3.1 choose the number of features

The more the number of feature is, the more words it covers will be. To some extent,

it can help the model to obtain more information. Therefore, this experiment improves the number of features. Due to time and hardware constraints, more features also mean longer running time. Therefore, this experiment only adds one order of magnitude, increasing the number of features from 100 to 1000.

3.3.2 meta-classifier for stacking

When selecting stacking's meta-classifier, the LinearSVC with the best performance is selected.

4. Evaluation and Analysis

Classifiers tried on train data are Gaussian Naïve Bayes, Multinomial Naïve Bayes, Decision Tree, linearSVC, Logistic Regression, Zero-R, One-R.

Firstly, train data with top 1000 mutual information value features which comes from train-raw.csv was used to train all the models, and among all classifiers, LinearSVC and Logistic Regression performed the best. Zero-R is used as a baseline. The accuracy of Zero-R is 0.25. The accuracy of Gaussian Naïve Bayes is 0.363. The accuracy of Multinomial Naïve Bayes is also 0.363, and slightly better than Gaussian Naïve Bayes' accuracy. The reason for the poor performance of Gaussian Naïve Bayes may be that the distribution of training data does not conform to the normal distribution then cannot fit the operation principle of Gaussian Naïve Bayes. The reason for the poor performance of Multinomial Naïve Bayes may be that the attributes of training data are not independent of each other, which makes the big premise of Naïve Bayes formula invalid. However, the accuracy of Bayes classifiers is still higher than Zero-R, suggesting that Naive Bayes classifiers still find some rules in the data. Decision Tree accuracy was 0.371. This accuracy is pretty high. However, since the decision tree has a higher possibility of overfitting, the higher accuracy may not be an objective manifestation of its true accuracy. The two classifiers with the highest accuracy are linearSVC classifier and Logistic Regression, both around 0.378. This phenomenon indicates that the rule of data set may conform to the linear model to some extent.

Based on the unoptimistic performance of these models, stacking method is added to the experiment to try to improve the accuracy. The meta-classifier used by stacking is the linearSVC classifier which has the best performance mentioned above. When stacking is used, the accuracy evaluated based on development data is lower than that of LinearSVC or logistic Regression. The accuracy of the latter two is 0.378, while that of stacking method is only 0.344. This phenomenon indicates that the previous classifier is suspected of overfitting.

In addition to the model itself, there are also reasons from the data itself. First of all, the problem brought by using mutual information value to select feature is that the selected feature may appear relatively frequently, but it is not necessarily the most valuable one, which leads to high bias. Finally, in the top1000 words selected in this experiment, duplicated words are not removed, and some of the same words appear in different forms and are judged to have different features, which is also not conducive to the training of the model.

6. Conclusion

Real life problem is sophisticated and the accuracy of prediction in this experiment is not that satisfying. Through feature engineering and applying stacking, the accuracy has been approved a little bit, from 0.30 (using "train-top100.csv" and "dev-top100.csv" to evaluate) to 0.33. But there are still more room to improve on feature selection part to gain data with higher usability.

References

[1] Zhiyuan Cheng, James Caverlee, and Kyumin Lee. 2010. You are where you tweet: a content-based approach to geo-locating twitter users. In Proc. of CIKM, pages 759–768, Toronto, Canada.

[2] Jacob Eisenstein, Brendan O'Connor, Noah A. Smith, and Eric P. Xing. 2010. A latent variable model for geographic lexical variation. In Proc. of EMNLP, pages 1277–1287, Cambridge, MA, USA.

[3] Orkut Buyukokkten, Junghoo Cho, Hector GarciaMolina, Luis Gravano, and Narayana Shivakumar. 1999. Exploiting geographical location information of web pages. In ACM SIGMOD Workshop on The Web and Databases, pages 91–96, Philadelphia, USA.
[4] Jochen L. Leidner and Michael D. Lieberman. 2011. Detecting geographical references in the form of place names and associated spatial natural language. SIGSPATIAL Special, 3(2):5–11.

[5] Gianluca Quercini, Hanan Samet, Jagan Sankaranarayanan, and Michael D. Lieberman. 2010. Determining the spatial reader scopes of news sources using local lexicons. In Proc. of the 18th International Conference on Advances in Geographic Information Systems, pages 43–52, San Jose, USA.

[6] B. Han, P. Cook, T. Baldwin, Text-based twitter user geolocation prediction, J. Artif. Int. Res. 49 (1) (2014) 451-500

[7] S. Roller, M. Speriosu, S. Rallapalli, B. Wing, J. Baldridge, Supervised text-based geolocation using language models on an adaptive grid, in: Proceeding of the 2012 Joint Conference on Empirical Methods in Natural Language Processing and Computational Natural Language Learning, Association for Computational Linguistics, 2012, pp.1500-1510