COMP90049 Report on Classifying the Geolocation of Tweets

Anonymous

1 Introduction

In location-based social network, managers advertise users with a physical presence and gain profit through the geolocation information provided by users (Wang et al., 2013). However, it is limited to obtain geolocation information through the user check-in data. As a result, predicting a user's geolocation through the text information released by the user has gradually attracted the attention of researchers (Cheng et al., 2010).

Based on the Twitter dataset, this report applies five machine learning algorithms (0-R, Naive Bayes, K-Nearest Neighbor, Logistic Regression and Random Forest) for text information of tweets to predict a user's geolocation, and proposes two methods (Majority Voting and Max Average Probability) for optimizing the performance of prediction based on the user's id. The appropriate model is selected as the classifier by evaluating the performance of different models. Finally, the report discusses the shortcomings of the model, optimization ideas and ethical issues.

2 Literature review

Some previous works mainly focused on the feature set selection. For example, Han evaluated lots of feature selection methods to obtain location indicative words feature and used NB and LR classifier on geolocation prediction. (Han et al., 2014), Chi used multinomial Naive Bayes classifier on five different feature sets (such as #Hashtags and @Mentions) to predict the geolocation. (Chi et al., 2016)

Some tried to use some optimized models to improve the performance of geolocation prediction. For example, Rahimi proposed a transductive multiview geolocation model, GCN, using Graph Convolutional Networks to predict semi-supervised user geolocation (Rahimi et al., 2018), Thomas presented a LSTM neural network architecture for the prediction of city labels and geo-coordinates for tweets (Thomas et al., 2017).

Eisenstein offered the resource of tweets raw data set (Eisenstein et al., 2010).

3 Data set

This report uses two type of data sets obtained from processing raw data to train and evaluate models.

3.1 Bags of word

In this model, raw text data can be split into plenty of words, frequent and very infrequent words are removed. Each word is mapped to a unique number. A text is represented as a list of sets containing the unique number and the count of occurrences of the word. Table 1 shows that the distributions of training data and validation data are similar.

Corpus	Features	Labels
Training data	133795 instances	NORTHEAST: 52582 SOUTH: 49901 WEST: 16228 MIDWEST: 15084
Validation data	11475 instances	NORTHEAST: 4295 SOUTH: 4266 WEST: 1484 MIDWEST: 1430
Testing data	12018 instances	None

Table 1- Number of instances and distribution of training data, validation data and testing data

3.2 TF-IDF

TF-IDF is the abbreviation of Term Frequency-Inverse Document Frequency, which is a statistical method used to evaluate the importance of a term to a document. It is the product of term frequency (TF) and inverse document frequency (IDF). TF is the weight of a term that occurs in a document, which is equal to the number of times a term appears in the document divided by the total number of terms in the document. IDF is a measure of the universal importance of a term, which is equal to the logarithmically scaled value of the total number of documents divided by the number of documents containing the term (Schütze et al., 2008).

The number of instances and distribution of training data, validation data and testing data are the same as Bags of Word, showing in Table 1.

4 Method

4.1 Data preprocessing

For bag of words data set, I performed one-hot encoding for each feature. If the instance contains a certain word, the feature value is 1, otherwise it is 0. For tf-idf data set, if the instance contains a certain word, that feature value is the tf-idf value, otherwise it is 0. After processing, both of them are transformed into high-dimensional sparse matrices, where each instance contains 2038 features.

4.2 Classification methods

I use the processed bag of words data (one-hot) to train KNN model and NB model. The reasons why I choose the methods are:

- KNN does not need assumptions about data, which is easy to explain and interpret.
- NB has a good performance with high-dimensional data such as text classification.

I use the tf-idf data to train LR model and RF model. The reasons why I choose the methods are:

- The implementation of LR is simple and it is very efficient for large data set.
- Random forest can handle high-dimensional feature, which does not need to perform feature selection.

4.2.1 0-R

The Zero-Rule classifier is a baseline classifier, which simply uses the category of the majority class in training data as the predicted value. The feature values have no contribution to the model, so 0-R classifier is usually used as a benchmark for other classifiers.

4.2.2 Naive Bayes

Naive Bayes is a supervised machine learning algorithm based on the Bayes theorem and the assumption that features are independent. NB classifier estimates the prior probability for each class based on the training data set, then estimates the conditional probability of each feature given each class using maximum likelihood estimation. The predicted result can be obtained from the following formula:

$$\hat{\boldsymbol{y}} = \arg\max_{c\in Y} \boldsymbol{P(c)} \prod_{i=1}^{n} \boldsymbol{P(x_i|c)}$$

where $\hat{\boldsymbol{y}}$ is the predicted class, $\boldsymbol{P(c)}$ is the prior probability for class $\boldsymbol{c}$, $\boldsymbol{P(x_i|c)}$ is the conditional probability of the feature $\boldsymbol{x_i}$ given class $\boldsymbol{c}$, $\mathbf{Y}$ is the classes set and $\boldsymbol{n}$ is the number of features.

In this task, Bernoulli Naive Bayes is used due to one-hot features.

4.2.3 K-Nearest Neighbor

The main idea of KNN is that if an instance has the highest similarity to the k instances in the training set, then the class of this instance is the majority of the k instances' classes. Similarity is measured by distance.

In this task, I use Euclidean distance to measure the similarity. I evaluate the accuracy of validation data by setting k from 5 to 500 to obtain the appropriate k value.

The formula of Euclidean distance is as follows:

$$\mathbf{dist(X,Y)} = \sqrt{\sum_{i=1}^{n}(\mathbf{x_i} - \mathbf{y_i})^2}$$

where $\mathbf{X}$ and $\mathbf{Y}$ are two instances, $\mathbf{x_i}$ and $\mathbf{y_i}$ are the feature values and $\mathbf{n}$ is the number of features.

4.2.4 Logistic Regression

Logistic regression is a linear algorithm to solve the binary classification problems. Because of optimizing posterior probability directly, LR is a probabilistic discrimiNaive model which does not need feature independence assumption. The formula of the probability of a class is as follows:

$$\boldsymbol{P(y|x;\theta)} = \frac{\mathbf{1}}{\mathbf{1} + \boldsymbol{e^{-z}}}$$

$$\boldsymbol{z} = \boldsymbol{\theta_0} + \sum_{i=1}^{n} \boldsymbol{\theta_i x_i}$$

where $\boldsymbol{\theta}$ is a weight vector for features, $\boldsymbol{n}$ is the number of features, y is a class, x is an instance and $\boldsymbol{x_i}$ is a feature. The linear value z can be mapped into a value between 0 and 1 by using logistic function. Then we define a decision boundary to obtain the predicted class. LR optimizes the weights by using stochastic average gradient descent to minimize the cross-entropy loss.

In this project, I use one-vs-rest method to make LR solve multi-classification problems, which means that many models are trained, each model is trained for only one class predicted whether an instance is the class or not, I get the final prediction through the results (probability) of all models.

4.2.5 Random Forest

Random Forest is an ensemble of random

tree, which can minimize overall variance without introducing model bias. Random Tree is a type of Decision Tree where only some of the features will be considered in each node. Each Random Tree is trained by using different bagged data set.

In this task, there are 100 Random Trees in the RF model where each RT will only consider the square root number of features. I evaluate the accuracy of validation data by setting max depth of each Random Tree from 5 to 50 to obtain the appropriate max depth value.

4.3 Optimization methods

Through the observation of the data set, I found that the same user may have multiple tweets. In other words, multiple instances of the same user should be predicted to be the same category. Therefore, I propose two methods to improve the final accuracy of the model. I apply the two methods to NB model, LR model and RF model, and evaluate their accuracy to judge whether the methods are optimizing the performance of models.

4.3.1 Majority Voting

After obtaining the predictions through the classification model, I will use majority of the prediction classes of multiple instances of a user as the new prediction classes for all instances of this user.

4.3.2 Max Average Probability

After obtaining the probability of each class through the classification model, I will use the class with the largest average probability of each class of multiple instances of the same user as the new predictions for all instances of this user.

4.4 Evaluation methods

Accuracy is the percentage of correct predictions among all predictions. Accuracy assumes that it is equally important to correctly predict different classes, this task meets the assumption, so I choose accuracy the evaluation metric.

5 Results

5.1 Tuning for KNN model

For KNN, I set k from 5 to 500 to calculate the accuracy of validation data. The figure 1 implies that when k is 330, accuracy is the highest, which is about 0.43. Accuracy rises fast when k is small, I think it is because there are some noise instances that affect the results. when k becomes larger, the accuracy does not change much.

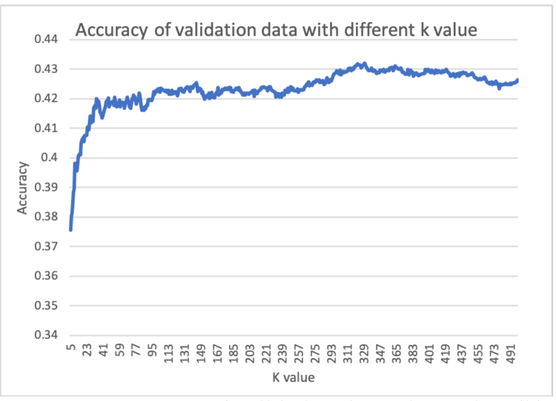

Figure 1- Accuracy of validation data when using different k value

5.2 Tuning for RF model

For RF, I set max depth of each Random Tree from 5 to 50 to calculate the accuracy of validation data. The figure 2 shows that when max depth equals to 40, accuracy is the highest, which is about 0.45. When max depth is small, the accuracy rises quickly, but as the max depth is larger, the score does not change much.

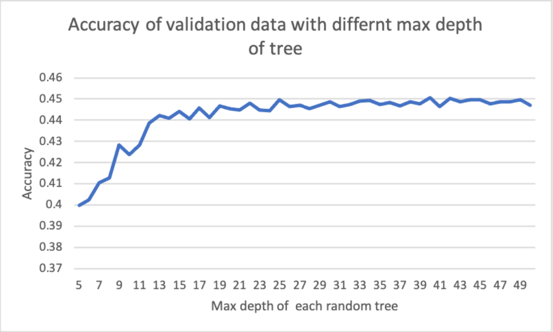

Figure 2- Accuracy of validation data when using different max depth of each random tree.

5.3 Overall

For NB and KNN, the data set is the processed bag of words data. For LR and RF, the data set is the processed tf-idf data.

Model	Accuracy
0-R baseline	0.37429
KNN	0.43102
NB	0.45647
NB + Majority Voting	0.55486
NB + Max Average Probability	**0.57682**

Table 2- Accuracy of validation data when using different model and bag of words data.

According to Table 2, for the bag-of-words data set, the accuracy of all models is higher than the 0-R benchmark. The accuracy of NB (0.45647) is higher than that of KNN (0.43102). Both optimization methods can significantly improve the accuracy of the NB model and Max Average Probability has a better accuracy improvement effect than Majority Voting. The model with highest accuracy is NB + Max Average Probability, whose accuracy is 0.57682.

Model	Accuracy
0-R baseline	0.37429
LR	0.46022
LR + Majority Voting	0.57534
LR + Max Average Probability	**0.59529**
RF	0.44845
RF + Majority Voting	0.48148
RF + Max Average Probability	0.52845

Table 3- Accuracy of validation data when using different model and tf-idf data.

According to Table 3, for the tf-idf data set, the accuracy of all models is also higher than the 0-R benchmark. The accuracy of LR (0.46022) is higher than that of RF (0.44845). Both optimization methods can significantly improve the accuracy of models and Max Average Probability has a better accuracy improvement effect than Majority Voting. The highest accuracy among models is LR + Max Average Probability, whose accuracy is 0.59529.

Overall, LR + Max Average Probability performs best in this task.

6 Discussion

6.1 Models analysis

When the instances of different classes are unbalanced, KNN may not make a good performance. For example, if the instances size of one class is too large, when you try to predict for a new instance, you'll find most of its neighbors are the majority class ones. In this task, the sizes of class NORTHEAST and SOUTH are almost 4 times those of WEST and MIDWEST, large size class takes the lead.

The performance of RF is not good as NB and LR, because RF is not suitable for sparse matrices. In this task, both bag of words data and tf-idf data are sparse matrices.

NB need the assumption that each feature is independent while LR do not. In this task, I think there is a relationship between some words. For example, words with opposite emotions do not appear in the same text normally. As a result, LR performs better than NB in this task.

There some ideas to improve the accuracy of the models:

- Using word2dev data set. In the bag of words and tf-idf data set, they only represent the importance of words in the text, but they ignore the context which may bring more information to models to get a better performance.
- Performing feature set selection, different types of features will also improve the performance of the models, such #Hashtags and @Mentions.
- Using more complex models. such as CNN and RNN, which can obtain more complex information from the training set to get a better performance in accuracy.

6.2 Ethical issues

Originally, software administrators need user authorization to obtain user's geolocation information. However, relying on machine learning models, software administrators can obtain user's geolocation information without the user's knowledge. I think this is an infringement of privacy.

In addition, models may make wrong predictions due to the imbalance of data classes, which may cause some wrong information being pushed to the user by applications. I think it has a bad influence on both application managers and users.

7 Conclusions

For the task of classifying the geolocation of tweets, NB + Max Average Probability has the highest accuracy (0.57682) by using the processed bag of words data (one-hot). Logistic Regression with Max Average Probability has the optimal performance by using the tf-idf data, whose accuracy of validation data set is 0.59529.

Overall, Logistic Regression with Max Average Probability is the best model for this task. In this project, I found that choosing appropriate model with appropriate features and tuning the parameters have a crucial influence on the final accuracy.

References

Wang, H., Terrovitis, M. , & Mamoulis, N. . (2013). Location recommendation in location-based social networks using user check-in data. ACM.

Cheng, Z., Caverlee, J. , & Lee, K. . (2010). You are where you tweet: a content-based approach to geo-locating twitter users. Acm Conference on Information & Knowledge Management. ACM.

Paul, Cook, Timothy, Baldwin, Bo, & Han. (2014). Text-based twitter user geolocation prediction. Journal of Artificial Intelligence Research.

Chi, L., Lim, K., & Alam, N., & Butler, C.(2016). Geolocation Prediction in Twitter Using Location Indicative Words and Textual Features.

Rahimi, A. , Cohn, T. , & Baldwin, T. . (2018). Semi-supervised User Geolocation via Graph Convolutional Networks.

Thomas, P. , & Hennig, L. . (2017). Twitter Geolocation Prediction Using Neural Networks. International Conference of the German Society for Computational Linguistics and Language Technology. Springer, Cham.

Eisenstein, J., O’Connor, B., Smith, N. A., & Xing, E. (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).

Schütze, H., Manning, C. D., and Raghavan, P. (2008). Introduction to information retrieval, volume 39. Cambridge University Press Cambridge.