Project2: Geolocation of Tweets with Machine Learning

1 Introduction

Twitter is one of most popular social networks that produces over 200 billion tweets per year . This report is for the purpose of building a classifier of geolocation for the Twitter users, according to the content of their Tweets they have sent. The locations of users are classified into five groups (five United Stated cities), which are Boston (B), Houston (H), San Diego (SD), Seattle (Se) and Washington DC (w). Weka system is used to analyze these huge amount American Tweets, which provides various Machine Learning algorithms and feature engineering methods.

2 Feature Engineering

The provided files development, training and testing data are used to improve the performance of the classifying. The first preprocessing step is building new development, training and testing vector instances, and some reducing of Tweets are necessary.

2.1 Word Stemming

Reducing the inflected words of their original form is called word stemming (Popovic, 1992). The algorithm that was used in this report is Porter Stemmer for preprocessing the Tweet files.

Porter Stemmer algorithm is a commonly used English stemmer, in this report, the Porter Stemmer reduced the sample space of the problem and partially corrected the miss-spelled location words.

The stemmed forms of some words are showing below.

word	**stem**
consist	consist
consisted	consist
knee	knee
kneel	knee

A sample differences in Tweets are showing below.

	Tweet text	
Original Tweet	finished a puzzle game	finish a puzzle game
Stemmed Tweet	switched to …	switch to ..

2.2 Stops Words and Excess White Space Removing

There are a large number of Tweets contain exceedingly common English words or just one character. These Tweets carried very little entropy in determining features for locating the user's geolocation. So that, it is necessary to remove these words or characters.

Additionally, the removing of stops words has no significant influence on locating the geolocation of user. The NLPK library for python was used to remove the stop words for Tweets. Some examples are showing below.

Removed Feature Token	Tweets
'you' 'with' 'me' 'any' 'but' 'if'	But, if you need any help with any design related stuff, let me know.
'it's' 'so' 'and' 'too' 'much'	OMG! It's so much fun and free too.

3 Geolocation Classification

In this report, Supervised Rule Based and Bayesian Machine Learning algorithm provided by Weka GUI framework are explored. The large number of Twitter vector instances are used to analyze the performance of the Machine Learning algorithms. Moreover, the performance of each algorithm was measured using weighted average F1 metrics and classification. The results will display in following sections.

3.1 Zero-R Decision Rule

The rule of Zero-R decision is classifying all instance according to the most common class in the training data. It is the most commonly used baseline in machine learning.
Boston, B, class occurred in 29.7% of instances, is the most frequent class in the training data. In addition, according to the Weka Zero-R model each user of Twitter is classified to B class, the performance accuracy is 26.8% and F1 weighted average of 0.134.

3.2 One-R Decision Rule

The method of One-R rule is creating one rule for each attribute in the training data, then selects the rule with the smallest error rate as its one rule. This algorithm is in nature and usually performs very well on many datasets (Robert, 1993).

The Weka One-R Machine Learning algorithm has reasonably well

performance than 0-R algorithm with 30.1% accuracy of classifying training data. Moreover, it also performed better weighted F1-score than Zero-R did, which was 0.178.

3.3 Naive Bayes

The one of supervised learning algorithms used in the repot is Naive Bayes method, which is based on applying Bayes' theorem with the assumption of independence between every pair of features. Assumed that a development feature vector is x_1 through $x_{n,}$ and a class variable is y, Bayes' theorem has following relationship:

$$P(y \mid x_1, \ldots, x_n) = \frac{P(y)P(x_1, \ldots, x_n \mid y)}{P(x_1, \ldots, x_n)}$$

Based on the naive independence assumption that

$$P(x_i \mid y, x_1, \ldots, x_{i-1}, x_{i+1}, \ldots, x_n) = P(x_i \mid y),$$

For all i, this relationship is simplified to

$$P(y \mid x_1, \ldots, x_n) = \frac{P(y) \prod_{i=1}^{n} P(x_i \mid y)}{P(x_1, \ldots, x_n)}$$

Since $P(x_1, \ldots, x_n)$ is constant given the input, the following classification rule can be used:

$$P(y \mid x_1, \ldots, x_n) \propto P(y) \prod_{i=1}^{n} P(x_i \mid y)$$

$$\Downarrow$$

$$\hat{y} = \arg\max P(y) \prod_{i=1}^{n} P(x_i \mid y),$$

The different naïve Byes classifiers differ mainly by the assumptions they make regarding the distribution of $P(x_i \mid y)$. Gaussian and Multinomial algorithm will be discussed below.

3.31 Gaussian Naive Bayes

Gaussian Naive Bayes implements the Naive Bayes algorithm for classification. The likelihood of the features is assumed to be Gaussian:

$$P(x_i \mid y) = \frac{1}{\sqrt{2\pi\sigma^2 y}} \exp\left(-\frac{(x_i - \mu_y)^2}{2\pi\sigma^2 y}\right)$$

The performance of Weka Gaussian Naive Bayes algorithm is little worse than Zero-R and One-R algorithm. The accuracy of classification is 25.8%. Moreover, comparing with One-R and Zero-R it achieved a better weighted F1 score, which is 0.251. It seems that Gaussian algorithm is not an efficient model for the Twitter attribute values.

3.32 Multinomial Naive Bayes

Multinomial Naive Bayes is one of two classic Naive Bayes text classification algorithm and it uses a multinomial distribution for each of the features (George, 1995). This approach is suitable for classification with discrete features (f_{xc} represents the total number of word x in all Tweets vectors belonging to class c) :

$$P(x_i|\ y) = \frac{1+f_{ic}}{N+\sum_{x=1}^{N} f_{xc}}$$

The Multinomial Naive Bayes algorithms performed best among other algorithms that I have discussed above. The correct classifying is about 43.8% and the F1-Score is 0.429.

The following tables demonstrate the building times of and classification performance of each algorithm implemented in the Weka framework.

	0-R	1-R	Gaussian NB	Multinomial NB
Time(seconds)	0.03	1.30	1.44	0.30

	0-R	1-R	Gaussian NB	Multinomial NB
Accuracy	26.8%	30.1%	25.8%	43.8%
F1-Sore	0.134	0.178	0.251	0.436

4 Conclusion

All algorithms were run in the test step after preprocessing, and Multinomial Naive Bayes is the most suitable algorithm for Tweets classification with the best overall results.

For future work, it would include using different approaches like Decision Tree based algorithms for data mining on different representations of Twitter data.

References

About twitter. https://about.twitter.com/company, Nov. 2014.

Robert C Holte. *Very simple classification rules perform well on most commonly used datasets. Machine learning,* 11(1):63–90, 1993.

George H John and Pat Langley. *Estimating continuous distributions in bayesian classifiers. In Proceedings of the Eleventh conference on Uncertainty in artificial intelligence,* pages 338–345. Morgan Kaufmann Publishers Inc., 1995.

Mirko Popovic and Peter Willett. *The effectiveness of stemming for natural-language access to slovene textual data.* Journal of the American Society for Information Science, 43(5):384 – 390, 1992.