Name: Cedric Ansley Chin Shen Wang
Matric Number: U096996N
IEM2201D

Detecting Linguistic Differences Between Singaporean and Malaysian Tweets with Naïve Bayesian Classification

Introduction

In recent years, the large volume of Twitter activity has made it a site of linguistic investigation – e.g.: (Eisenstein, O'Connor, Smith, & Xing, 2010), (González-Ibáñez, Muresan, & Wacholder, 2011). Such investigations require the creation of tweet corpora as a data source for linguistic research.

A serious effort towards creating a Singaporean tweet corpus will have to grapple with the problem of identifying Singapore-only tweets. A solution to this is to use the Twitter API[1] to grab tweets geo-tagged with Singaporean coordinates. This approach is problematic, however, as Twitter's geo-tagging feature lacks accuracy: tweets tagged in Singapore are often labelled as coming from Johor Bahru instead.

A potential solution to this problem is to use a machine learning algorithm, called a classifier, to detect distinguishing linguistic features of Singaporean and Malaysian tweets captured from this API. Such linguistic features may include non-English lexical items such as Malay words from the Malaysian tweets, and slang words such as *sia* or *lah* that are widely used in Singaporean English. These linguistic features may then be used to differentiate tweets from the two countries.

In this project I demonstrate the use of a Naïve Bayes Classifier to identify a set of distinguishing linguistic features, drawn from a corpus of 4138 Singaporean and 1487 Malaysian tweets. This project limits itself to studying English tweets, or tweets of English mixed with non-English lexical items (such as slang words). The tweets were collected over a 3 week period in March 2012. The study presented here:

1. describes a classification methodology
2. analyzes several interesting linguistic features picked by the classifier
3. and lastly, discusses the accuracy of my results and drawbacks of my approach

Background

Twitter is a microblogging service that consists of rapid, 140-character updates called tweets from millions of users around the world. Twitter's large volume, as

[1] An API, short for Application Programming Interface, is an interface for programmatically accessing data, usually from a software service.

well as its relative openness (i.e.: availability of tweet data through APIs and an open website) makes it a compelling medium with which to explore linguistic research questions.

Twitter has been the source of several geolocation and sociolinguistic studies. Eisenstein, O'Connor, Smith, and Xing presented a method to predict "an author's geographical location from raw text" in 2010 (Eisenstein, O'Connor, Smith, & Xing, 2010). At MIT, Benson, Haghighi and Barzilay have demonstrated a method "to accurately induce event records from Twitter messages, evaluated against events from a local city guide." (Benson, Haghighi, & Barzilay, 2011). In both cases, data could be extracted by examining the linguistic features of tweets. Both papers also relied on machine learning models trained on a collection of tweets.

Methodology

Tweet Collection

Tweets were collected by a custom program for a period of 3 weeks, from the Twitter API. The tweets collected were restricted to Johor Bahru and Singapore.

Each tweet was saved to a database with the following information:

- tweet
- location
- username

In total, 32,905 tweets were collected in this 3 week period.

Data Preprocessing

Tweets captured directly from the streaming API may not be used immediately for classification. These tweets may include non-English tweets, as well as other non-linguistic items such as URLs. As such features may pollute our results, some processing must first be done before passing the tweet data to the naïve Bayes classifier.

I performed the following processing steps:

i) Language detection and filtering

A portion of tweets retrieved from the streaming API was not in English. I used the Chromium Compact Language Detector[2] to detect and filter out non-English tweets.

Tweets detected as an 'Unknown' language were labeled so because they contained a mixture of English and Malay. These tweets fall within the scope of the study presented here, and so were left in the corpus.

[2] http://code.google.com/p/chromium-compact-language-detector/

ii) Filtering of Twitter-specific features

Tweets may contain a number of Twitter-specific lexical items. For example, an '@ reply' is used to denote replies to another user.[3] These lexical items are unwanted, as they are not linguistic features. I programmatically removed the following items from the tweet content, leaving the rest of the words in the tweet body intact:

1. 'RT' – denoting retweets
2. '@username' – denoting replies to another user
3. '#hashtag' – denoting trends or topics that the tweet is associated with

iii) Link removal

As with (Eisenstein, O'Connor, Smith, & Xing, 2010), all tweets with links were removed from the corpus to prevent the risk of accidentally adding spam tweets or non-linguistic items such as URLs to the corpus.

iv) Stop words removal

Stop words are the set of the most commonly used words in a language (e.g. *the, a, an* and *want* in English). These words are of little value to the classification algorithm and are usually removed prior to classification. NLTK's[4] stop words corpus was used to remove all English stop words from my tweet corpus. I consider Malay to be non-English lexical items of interest, and so did not remove any Malay stop words.

Naïve Bayes Classification

After preprocessing, tweet data may finally be passed through the classifier. The study presented here uses NLTK's implementation of the Naïve Bayes Classifier. I shall briefly cover the concept of a Naïve Bayes Classifier before explaining two steps involved in applying the classifier to my data.

Naïve Bayes Classification: a brief overview

The Naïve Bayes Classifier is a probabilistic classifier based on applying Bayes' theorem to a set of tagged data, using a strong independence assumption. (Rish, 2001) In essence, a Naïve Bayes Classifier takes two tagged data sets (e.g. tweets tagged 'from Singapore' and tweets tagged 'from Malaysia') and calculates the probability of features[5] that are present in each of these two sets. The feature we use for this study is the presence of words in the tweets. For example, the presence of *lah* in a tweet is considered a feature. The classifier will then calculate the probability of the word *lah* appearing in the Singaporean tweet data set compared to the Malaysia set.

After the probabilities of all the words present in the Singapore tweet set and Malaysia tweet set have been calculated, the classifier is then tested on a new set

---

[3] e.g.: "@bob212 shall we go for lunch?" denotes a tweet publicly addressed to Twitter user bob212.

[4] Natural Language Toolkit, also known as NLTK, is a collection of libraries and programs for natural language processing in Python.

[5] A feature is any trait that may be useful in identifying the data set. In this study, the feature we choose to use are the words present in the tweets; however other features (e.g. language use, name of location, time) are valid alternatives

of tagged tweets. This set of tweets is called the test set. The classifier makes guesses as to whether the tweets in the test set are from Singapore or from Malaysia based on multiplying the probabilities of the words present in the tweet. These probabilities are either in favour of Singapore or Malaysia. After making its guess, the classifier compares its guess to the tag of the tweets in the test set (either Malaysia or Singapore) to check if it is correct. Finally, it outputs an accuracy score of all the guesses it has made in the test set, as well as a list of most informative features.

The Naïve Bayes Classifier thus requires two things:

1. a set of features drawn from the tweet data I have collected
2. two tagged data sets, one of Singaporean tweets, and one of Johor Bahru tweets

These are dealt with as follows:

1) Feature selection
I chose to extract both unigrams and bigrams as features from the tweets I have collected. Unigrams are single words present in a tweet (e.g. *lah*, *love*). A bigram is a pair of words present in a tweet. START and END tags in a bigram denote the start and end of a sentence, respectively.

2) Manual tagging of tweets
Tweets are tagged as either being from Malaysia or from Singapore. This tagging process was done manually, by visiting the profiles of Twitter users to determine if they were Singaporean or Malaysian.

Selection of test and training sets
For each run of the classifier, 80% of tweets were randomly selected from a total of 4138 Singaporean and 1487 Malaysian tweets, to be used as a training set. The remaining 20% of tweets were used as a test set.

Results and Analysis

Presented below are the 32 most informative unigram features. (Rows shaded in grey are interesting features.)

Most Informative Features	Label	Ratio
think	SG : MY	5.9 : 1.0
get	SG : MY	4.9 : 1.0
say	SG : MY	4.7 : 1.0
omg	SG : MY	4.5 : 1.0
got	SG : MY	4.1 : 1.0
home	SG : MY	4.0 : 1.0
also	SG : MY	4.0 : 1.0
never	SG : MY	4.0 : 1.0
want	SG : MY	3.7 : 1.0
uh	SG : MY	3.5 : 1.0
good	SG : MY	3.5 : 1.0
sleep	SG : MY	3.5 : 1.0
sia	SG : MY	3.5 : 1.0

Table 1: Unigram features indicative of Singapore, sorted in order of usefulness

Most Informative Features	Label	Ratio
dia	MY : SG	8.0 : 1.0
thanks	MY : SG	5.9 : 1.0
ni	MY : SG	5.9 : 1.0
tak	MY : SG	5.9 : 1.0
♥	MY : SG	5.5 : 1.0
ke	MY : SG	5.2 : 1.0
nk	MY : SG	5.2 : 1.0
tu	MY : SG	4.8 : 1.0
pon	MY : SG	4.2 : 1.0
kan	MY : SG	4.2 : 1.0
5	MY : SG	4.2 : 1.0
apa	MY : SG	4.2 : 1.0
g	MY : SG	4.2 : 1.0
tonight	MY : SG	4.2 : 1.0
left	MY : SG	3.3 : 1.0
mane	MY : SG	3.3 : 1.0
every	MY : SG	3.3 : 1.0
update	MY : SG	3.3 : 1.0
mana	MY : SG	3.3 : 1.0

Table 2: Unigram features indicative of Malaysia, sorted in order of usefulness

Accuracy[6] **for the above set:** 0.840909090909

[6] The maximum accuracy is 1, and the minimum is 0.5

A tweet containing 'dia'[7] is 8 times more likely to be Malaysian than Singaporean. A tweet containing 'sia'[8] is 3.5 times more likely to be Singaporean than Malaysian. The majority of unigrams indicative of Malaysia are Malay words.

The words in the unshaded rows are unlikely to be meaningful, because idiosyncracies specific to a corpus are likely to crop up when the corpus size is small. For instance, it is possible that *think*, *get*, and *say* appear in Table 1 as positive Singaporean features simply because they are more prevalent amongst the Singaporean tweets present in my corpus. These common words should not be interpreted as representative features, unless tested against a bigger corpus.

Presented below are the 32 most informative bigram features.

Most Informative Features	Label	Ratio
lah, END	SG : MY	3.8 : 1.0
dont, like	SG : MY	2.2 : 1.0
START, im	SG : MY	1.8 : 1.0

Table 3: Bigram features indicative of Singapore, arranged in order of usefulness

Most Informative Features	Label	Ratio
ke, END	MY : SG	2.7 : 1.0
d, END	MY : SG	2.7 : 1.0
START, keep	MY : SG	2.7 : 1.0
good, morning	MY : SG	2.7 : 1.0
mine, END	MY : SG	2.7 : 1.0
START, haha	MY : SG	2.6 : 1.0
day, END	MY : SG	1.6 : 1.0
START, good	MY : SG	1.6 : 1.0
ya, allah	MY : SG	1.6 : 1.0
ni, hahaha	MY : SG	1.6 : 1.0
die, END	MY : SG	1.6 : 1.0
START, nothing	MY : SG	1.6 : 1.0
im, END	MY : SG	1.6 : 1.0
mcm, nk	MY : SG	1.6 : 1.0
damn, END	MY : SG	1.6 : 1.0
babe, END	MY : SG	1.6 : 1.0
bro, END	MY : SG	1.6 : 1.0
end, END	MY : SG	1.6 : 1.0
back, home	MY : SG	1.6 : 1.0
im, fine	MY : SG	1.6 : 1.0
hahahaha, END	MY : SG	1.6 : 1.0
START, kalau	MY : SG	1.6 : 1.0
think, youre	MY : SG	1.6 : 1.0
tak, END	MY : SG	1.6 : 1.0
remember, everything	MY : SG	1.6 : 1.0

[7] The Malay word for him/her

[8] A Singaporean slang word, used as an emphatic particle or exclamation.

ill, go	MY : SG	1.6 : 1.0
START, d	MY : SG	1.6 : 1.0
START, sex	MY : SG	1.6 : 1.0

Table 4: Bigram features indicative of Malaysia, arranged in order of usefulness

Accuracy for the above set: 0.870652173913

Surprisingly, the Singaporean expression *lah* at the end of a sentence is 3.8 times more likely to be Singaporean than Malaysian, despite the *lah* particle being present in both Malaysian and Singaporean English. More predictably, *Ya Allah* is 1.6 times more likely to be Malaysian than Singaporean, while *omg*, (Table 1) is 4.5 times more likely to be Singaporean than Malaysian. *Ya Allah* is a Muslim exclamation, while *omg* (short for 'Oh My God') is not.

These results seem consistent with what we know of the two countries. Malaysian tweets consist of primarily Malay lingusitic features, and the two religious exclamations detected as features are indicative of each country's respective relgious makeups. A linguistic-classifier approach to differentiating Singaporean and Malaysian tweets appears to have some credibility.

The accuracy scores for both unigram and bigram classifiers are fairly high, at 0.84 and 0.87 respectively. It is tempting to claim that the features presented above may be used effectively for the auto-detection of Singapore versus Malaysian tweets. However, this accuracy may not be meaningful without testing against a larger corpus of tweet data. As discussed earlier, the classifier used in this study had only a small corpus to work with, which is not likely to be representative of all tweets from Singapore and Malaysia. I suspect that many of the features presented above are idiosyncracies specific to the corpus of tweets collected.

The performance of the classifier could be improved using a larger corpus.

Conclusion

The results from this research project show that there is some promise in the classification of linguistic features as a method of Twitter corpus-building. A more rigorous approach, with better classification algorithms may be undertaken in the near future. Such an approach will likely require the collection of a larger, more comprehensive Twitter corpus. It is my hope that such an approach may be used to automatically classify tweets by location, easing the difficulty of data gathering for future Twitter-based corpus linguistic research.

References

Eisenstein, J., O'Connor, B., Smith, N. A., & Xing, E. P. (2010). A Latent Variable Model for Geographic Lexical Variation. *EMNLP 2010 - Conference on Empirical*

Methods in Natural Language Processing, Proceedings of the Conference. , 1277-1287.

Benson, E., Haghighi, A., & Barzilay, R. (2011). Event Discovery in Social Media Feeds. *Proceedings of the 49th Annual Meeting of the Association for Computational Linguistics* , 389-398.

González-Ibáñez, R., Muresan, S., & Wacholder, N. (2011, June 19-24). Identifying Sarcasm in Twitter: A Closer Look. *Proceedings of the 49th Annual Meeting of the Association for Computational Linguistics: shortpapers* , 581-586.

Rish, I. (2001). An empirical study of the naive Bayes classifier. *IJCAI 2001 Workshop on Empirical Methods in Artificial Intelligence.* (pp. 192-198). Seattle, USA: IBM Research Division.