Scott Bickel, Brian Golden, Steve Styer (Baker’s Kilobyte)

CMPSC 448

Project Milestone 2

2/25/14

# **Datasets**

Two datasets of tweets will be explored and used as learning data, one of 600,000 data points and another of 2 million data points. Datasets were collected by searching for tweets matching various “hashtags” over 5-12 weeks. Each of the data sets has the following schema/attributes for data points:

* Tweet ID – a unique identifier for the tweet
* Source User ID – the user that posted the tweet
* Retweet User ID – the user that posted the original tweet being retweeted
* Website – website referenced in the tweet, if any
* Tweet Time – timestamp for the tweet’s posting time
* Hashtag – the hashtag that the search found a match against, i.e. why the tweet is in the dataset
* Week number – the week number from the start of data collection
* Day number – the day number from the start of data collection

# **Learning Tasks**

Based on our data, we will attempt to develop an effective model for determining the informative value of a tweet, i.e. does it have informative worth or is it a conversational device?

# **Feature Generation**

To make a prediction of a tweet’s informative value, the feature vector will be generated accordingly:

1. Check if a website is the text of the tweet
2. User retweet count (i.e. higher retweet counts may indicate a more informative tweet)
3. Check if tweet is a direct reply to another user (i.e. a tweet is a direct reply to another user if its text begins by mentioning that user)
4. Aspects mentioned in tweet, generated/identified using the Stanford Named Entity Recognizer (<https://github.com/dat/stanford-ner>). Such aspects included:
   1. location mentioned
   2. person or people mentioned
   3. organization mentioned

# **Label Generation**

Labels for the data were generated manually, in as consistent a manner as possible. This was done by viewing individual tweets and judging whether or not they were informative or conversational.

# **Models**

To make our predictions we used a Support Vector Machine, as it is a binary decision. Specifically, we made use of the SVC class in the [sklearn](http://scikit-learn.org/stable/) machine learning library for python. This was fed our feature vectors and manual labels for the training data.

# **Initial Diagnostics**

We initially used the Perceptron algorithm to fit our data and we found that the mean score of the predictions was approximately 52%. To see the effectiveness of another machine learning model, we used a Support Vector Machine class from the sklearn library. This was trained using our labeled data, and found the mean score of these predictions was approximately 62%. This improvement in accuracy score by 10% makes sense, as the Perceptron algorithm works best when the data used in training the model is linearly separable.

Below is a plot of the initial fit of the data using a Support Vector Machine. Currently the data is very clustered; adding new features and modifying current ones will be the priority as we try to improve the overall accuracy of predictions made by the model.

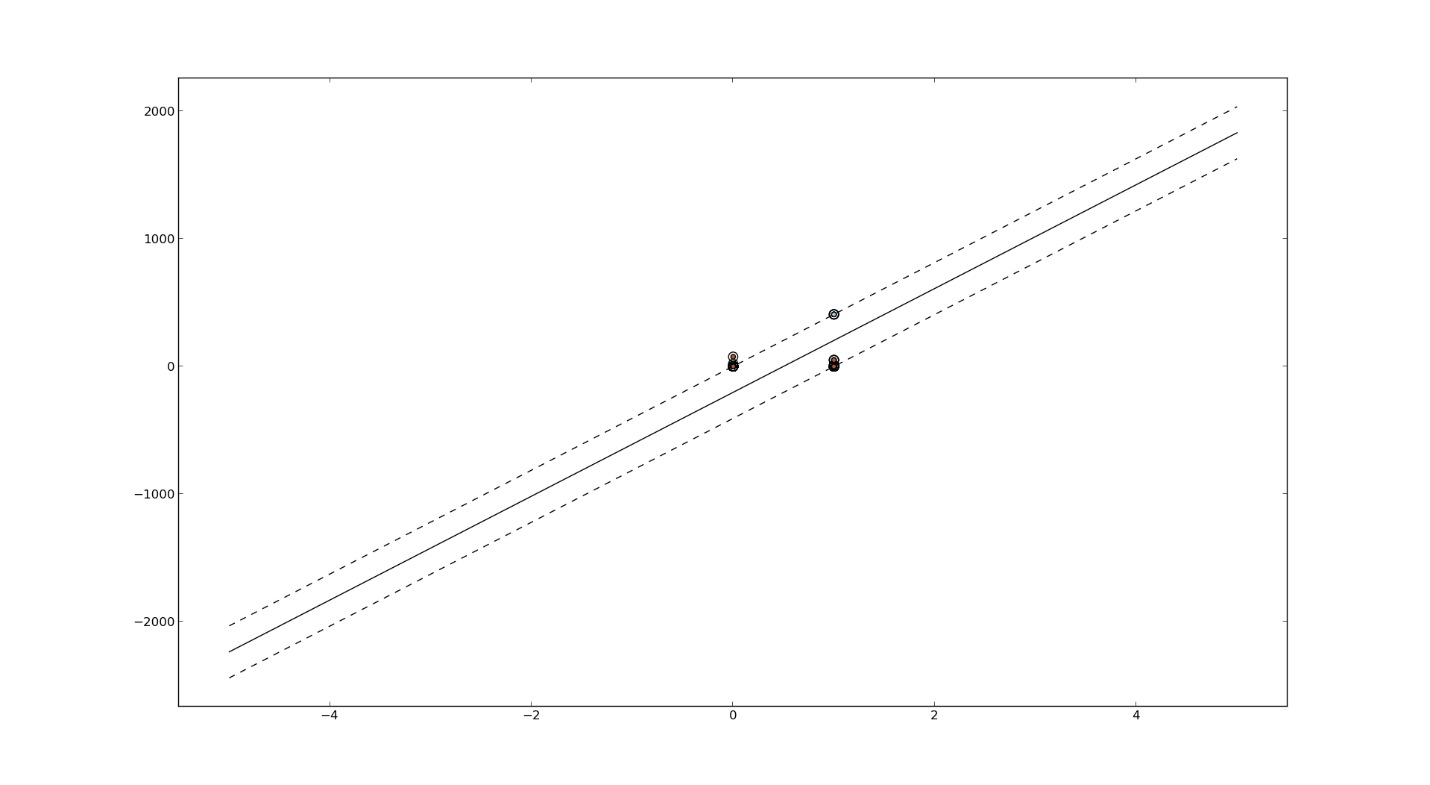


Figure : Plot of Tweet Predictions Made by Support Vector Machine

# **Estimated Timeline**

* 2/26/14 – 3/7/14: Improve feature generation, by evaluating the usefulness of current features and potentially adding new ones.
* 3/17/14 -3/25/14: Run additional diagnostics to determine the effect of feature refinement.