Twender:
Gender Classification from Tweets

Matthew Stanley

Friday 4th December, 2015

1 Abstract

Twitter is a popular social media platform that produces an enormous amount of user created information and associated meta-data. However, useful pieces of demographic information such as gender, age, ethnicity, political orientation, are not present as part of the definition of a user. Thus a particularly interesting challenge presents itself; is it possible to automatically determine these properties. In terms of this paper only the attribute gender will be studied. The goal of this research is three-fold: to create a simple, efficient and accurate framework for collecting and labeling training data, to create an accurate classifier for predicting the gender of Twitter users, and finally to produce a simple interactive web application for showcasing the result of the latter two goals.

2 Introduction

2.1 Motivation

Due to the ever-growing popularity of rich interactive web applications, including but not limited to those related to social media, and the ever-improving technologies and techniques for storing, distributing and analyzing large amounts of data, the web has become a producer of massive amounts of user generated content and associated meta-data. While in many cases a challenging task the outcome of harnessing such a massive amount of information is almost always extremely beneficial.

Analysis and visualization of said data can lead to massive improvements in areas such as user experience and application performance. Most of this analysis is targeted at better understanding users themselves and how they interact with the application. This is especially true of user demographics, such as age, gender, ethnicity, etc. User demographics are important two-fold, on an application level they can be collectively used to profile the user base, on a per user scale they can be use for targeted advertisement (and other content) and search prediction. Overall at the user level this information can be used to tailor a much more personal experience.

2.2 Data

As of September 30, 2015, Twitter's estimated monthly active users was 320 million with millions of tweets posted per day. With 79% of user accounts outside of the United States and support for 35+ languages Twitter also boasts a global user base. Besides having a large group of diverse users, Twitter is built on top of an impressive infrastructure which provides developers

with well documented access to well maintained data.

In terms of data, Twitter provides a large amount of meta-data in addition to the text of a particular tweet. For example Twitter provides user location data, which could be a specific geographic coordinate at the time of posting a tweet or the users area of residence. Twitter also provides relationship data such as followers, follower count, user mentions, in reply to links, etc. Twitter also allows access to user profile information such as the user's bio, posted media and links to external websites or blogs curated by the user.

The actual message content of the tweet contains some interesting properties. To begin with the user is limited to 140 characters per tweet, which results in differing linguistic structures than unrestricted prose (this will be discussed later on in greater detail). Tweets can also contain other entities not seen in regular text such as user mentions (user specific links that begin with @), hashtags (a string of words without whitespace proceeded by #), which serve to facilitate searching and grouping of tweets but have also been used to summarize the content of the tweet by the user. Tweets can also contain URLs and media including videos, images and gifs. This research utilizes user location, user name and tweet text body only.

2.3 Related Work

Automatically determining certain demographic attributes of authors based on samplings of their texts is a well researched topic and falls into the more general category of authorship classification. However most of this research has been conducted on polished written texts including books, e-mails, blogs, etc.[2], it was not until the research of Rao et al.[3] that authorship classification of tweets was explored in depth.

Rao et al. explore the problem of determining the following 4 latent attributes of a twitter user: gender, age, regional origin, and political orientation. The authors also explore the effectiveness of different features, most relevant to this work, they determined that between three different sets of features, network structure (relationships between followers and following users), communication behavior (frequency of response, retweet and tweets) and sociolinguistic features, the final feature set was the most effective. Research by Pennacchiotti and Popescu[2] also supports the determination that a rich sociolinguistics based feature set from a tweet's text works well. Rao et al. achieve a result of 72.33% accuracy for classifying gender using a stacked classifier model.[3]

In terms of corpus generation, i.e. labeled training data, most researchers use complex or expensive methods. Rao et al. use a crawl of twitter to produce a set of users and then manually label these users.[3] Burger et al. utilize external links to other blog platforms or social media applications on which users have provided a more detailed bio or for which gender is a defined attribute of the user.[1] For this research I leverage the user names as an inexpensive means of creating a labeled corpus of tweets.

3 Experimental Design

3.1 Tools

I chose to implement almost the entirety of this project in the programming language python. Which means that the project is uniform and consistent throughout in terms of language and style. Python has excellent facilities for working with and manipulating text based data in the language proper. Python also contains a multi-

tude of libraries for natural language processing and machine learning.

For this project I utilized the python libraries tweepy, scikit-learn and flask. The tweepy library provides a python wrapper for interacting with the Twitter API, it handles Oauth authentication and provides an interface to both static data queries and streaming data. Scikit-learn is a large machine learning library which I use for feature extraction and construction of a classifier. As for the final library, flask is a micro-framework for writing web-based applications in python. I also make use of the javascript library d3 for data visualization as part of the final application.

3.2 Overview

In order to produce a reusable and modular framework I separated the functionality into three sections. The first section contains functionality for collecting and labeling tweets for use as a training corpus. The second section contains the functionality for building and training a classifier as well as a simple interface for predicting a user's gender based on the classification of their tweets, i.e. gender prediction. The third and final section contains the foundation for a simple interactive web application for demonstrating the resulting classifier.

3.2.1 Tweet Collection

The main challenge of collecting tweets is to devise a method for labeling them with gender so as to produce a valid set of training tweets. For this project I wanted to create a method to accomplish this that was lightweight, simple and efficient. I decided to use the user name to determine a gender label for the training dataset.

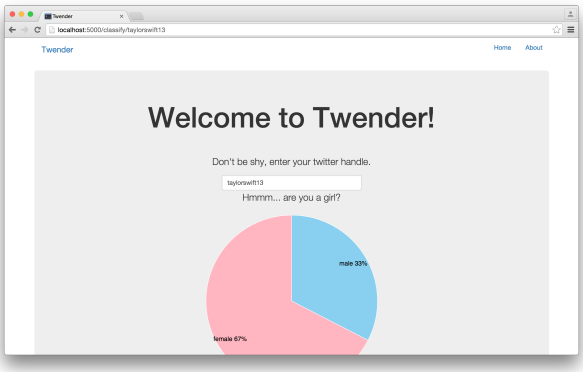

Figure 1: Twender web application in action

In order to create the necessary (name, gender) mapping I chose to use a popular baby names dataset from the social security administration in the United States. The dataset is constructed from names specified on social security card applications for U.S. births since 1879. The dataset contains 1,825,433 name entries for which 93,889 are unique names and is broken down by year.

Each name entry in the dataset contains a gender (which was submitted as part of the social security application) and the occurrence of that name and gender pair. In order to resolve times when a name corresponds to both genders, I calculate the total number of occurrences of the name for each gender, if the percentage of male occurrences is above 75% and vice versa then the name is used otherwise the name is discarded for having an ambiguous gender and thus being an unreliable determiner of gender.

In terms of actual collection I use Twitter's streaming API to collect a semi-random sampling of tweets in real-time. As the tweets are streamed they are inspected for a first name (the first word of their user name, where word is defined by whitespace) that is present in the valid names determined by the process above. If the

name is present it is labeled and stored otherwise the tweet is discarded.

In an effort to collect tweets in an unbiased fashion I stream tweets based on location where the bounding box specified is anywhere on Earth. This is to avoid streaming by options such as key words which could potentially make the resulting dataset biased in some way. However I have chosen to only include tweets that are in English, which means that the resulting classifier is highly language dependent and will prove inaccurate for any other language. In addition the dataset is composed of equal number of tweets labeled female as male to avoid bias of one particular label.

3.2.2 Tweet Analysis

This section is concerned mostly with feature extraction from the tweets in the training dataset as well as classifier construction and training. For this project I chose to employ a naive Bayes classifier. This is one of the simplest classifiers, but can easily be replaced by another. For example, support vector machines are widely used in text classification problems.

Feature extraction occurs both upon training or fitting the classifier to the training dataset and in the actual classifying or prediction step. Feature extraction is broken into a two step process, first the text is tokenized and then counted, this results in a matrix with shape (number of samples, number of features). I then use TF-IDF to convert the matrix of raw counts to real-valued numbers.

TF-IDF or term frequency-inverse document frequency is a feature extraction technique which is superior to simple frequency counts. Term frequency refers to the number of times for which a term appears in a document. Inverse document frequency is a measure of how much or how little information a term conveys. IDF is the number of documents in the dataset divided by the number of documents containing the term in question.

Intuitively IDF attributes higher importance to terms which are rare in the documents and lower importance to those that are common. As a result common stop words and terms with little meaning are filtered out. A feature will be given a higher TF-IDF weighting per document if the term appears extensively in that particular document but is rare across all documents.

3.2.3 Tweet Application

The final section is the web application which I decided to create in order to showcase the result of the previous two sections. The web application is a simple lightweight application built using the flask micro-framework. Upon loading the website the user is prompted to enter their twitter handle (the part following the @ symbol) The application tries to fetch 200 of the user's most recent status or tweet posts. It is important to note that retweets are not utilized in this classification process. Each of these tweets is classified individually after which the user is given an overall classification based on whether a higher percentage of the tweets were classified male or female. The application also displays the tweets used in the classification process which have been colored to indicate gender. In addition the user presented with a visualization which displays the percentage of the tweets which were classified by each label.

4 Results and Analysis

The following section will detail the experimental results. In order to correctly test the end result classifier I collected another set of tweets to use for testing. I collected these tweets in the same manner described in section 3.2.1. The testing dataset includes 5600 labeled tweets, and is equally divided by both genders. It should be noted that there exist two major deficiencies with this method of testing. First since these tweets are collected in the same manner as the training dataset the labeling is not guaranteed to be correct. Second I test the classifier on a per tweet basis not a per user basis.

For testing the classifier is fit to the entirety of the training dataset and then tested on the entirety of the testing dataset, I note only the percent of tweets correctly identified. Using this means of testing, and training on a dataset of size 22,700 tweets I determined the classifier to be 62.29% accurate.

I attempted several strategies to improve the accuracy of the classifier, these attempts are documented as follows. As it stands I use the default tokenizer provided by scikit-learn, and use an n-gram size of one word. This means that each word in the tweet text is considered an individual token and therefore a potential feature. As a result the extracted features contain some that are seemingly useless.

The main sources of these features are from user mentions, hashtags and website URLs. The worst of these would appear to be URLs as this would result features such as "https", and "bq27wrzijp". In order to remedy these issues I tried removing URLs, user mentions and hashtags separately before training the classifier and recorded the result. In addition I tried replacing the naive Bayes classifier with an SVM. The results for these changes are shown in table 1.

Refinement	% correct
Remove user mentions	60.68%
Remove hashtags	61.88%
Remove URLs	61.20%
Using SVM	60.59%

Table 1: Classifier refinements and their outcomes

From table 1 we can see that these refinements in fact lower the accuracy of the resulting classifier. Intuitively these changes would increase the accuracy of the classifier by removing random data from the features, this however is not the case.

As well as recording the result of these refinements I have also recorded the change in accuracy that result from increases in training data. These result are shown in figure 2. The sample training datasets to produce this graph start with a training set size of 1,000 tweets comprised of exactly half male and half female labeled tweets. For each successive data point an additional 1,000 tweets is added to the original training set. The percent correctly classified is then computed for each training set size.

The graph in figure 2 shows that the accuracy of the classifier still has potential to increase with an increase in training set size. However, it appears that the accuracy that could be gained from an increase in training dataset size is approaching its limit.

5 Conclusion

In conclusion there exist many difficulties in authorship classification of tweets. We can see that the unique structure and restrictions placed on tweets give rise to novel sociolinguistic features

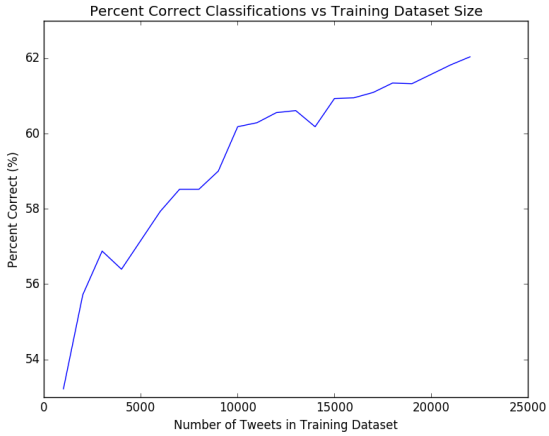

Figure 2: Change in classifier accuracy with increasing size of training set

not found within regular text documents. For example, the restriction of a tweet to 140 characters causes some users to leverage more abbreviations, slang, and intentional (and unintentional) misspellings. We also find novel linguistic entities such as hashtags, user mentions and URLs.

There also exists debate over the nature of gender and the impact it has on language identity. It is clear from the research of Rao et al. and others that there does exist differences in the language use of female and male authors, as they are able to build a fairly accurate classifier on tweet text alone. However, the extent of the influence of gender on language use requires further research.

It is also questionable whether the problem of gender classification can be framed as a binary decision. The argument has been made that sex refers to physical biological characteristics and that gender refers to a social construct which is much better suited for a one-dimensional continuum instead of binary classification. Nguyen et al. discuss this topic in their research, they give an example of tweets authored by a 16-year old biological male which exhibit linguistic characteristics often associated with female writing.

With these things in mind I present the framework described in this paper as a starting point for further research into the topic of gender classification of Twitter users from their tweets.

References

[1] John D Burger, John Henderson, George Kim, and Guido Zarrella. Discriminating gender on twitter. In *Proceedings of the Conference on Empirical Methods in Natural Language Processing*, pages 1301–1309. Association for Computational Linguistics, 2011.

[2] Marco Pennacchiotti and Ana-Maria Popescu. A machine learning approach to twitter user classification. *ICWSM*, 11:281–288, 2011.

[3] Delip Rao, David Yarowsky, Abhishek Shreevats, and Manaswi Gupta. Classifying latent user attributes in twitter. In *Proceedings of the 2nd international workshop on Search and mining user-generated contents*, pages 37–44. ACM, 2010.