CIS 520 Final Project (2015)

Team: **Tiresias**

Team members: Xiao Hu, Zilu Zhou and Di Zhang

Abstract

Vast amount of data generated by various social network platforms provides us with intriguing opportunities to gain insight into people’s thoughts and behaviors. Such insight also facilitates prediction of properties of users that are important for both business strategic planning and academic research. One example is to predict one’s gender based on his/her language use and profile picture in Twitter. Here, given a train set consisting of word count of the 5,000 most frequent words in Twitter, profile images and image features of approximately 4,998 users whose gender is known, we present a trained ensemble model that includes random forests, logistic regression and linear support vector classification that achieved 89.65% accuracy on a validation set with a size of 4,997. In this report, we discuss the methods we attempted, their caveats, and potential improvements of our methods that may increase the accuracy of our prediction further. In addition, we explore some bias in word usage between male and female and reflect upon how this bias reflects differences in interests between genders.

1 Introduction

Accurate predication of demographic attributes, such as gender in our case here, is important for marketing, personalization, recommendation system, and research in social sciences. Some previous studies have attempted gender prediction using different data types from social networks. For example, Miller et al used stream algorithms that extract continuous N-gram features from tweets[i]. Burger et al discovered that gender prediction using full names alone can reach 89% accuracy, which increased to 92% when combined with tweet texts, screen names, descriptions[ii]. You et al used multiple posted pictures from induvial users on Flickr and achieved ~70% accuracy, using a scale-invariant feature transform algorithm to extract unchanged features of various images of a certain object[iii]. In our final project here, we face a distinct set of challenges from those in aforementioned studies, constrained by available types of data sets. First, we are given word count for each twitter user, instead of full tweets, which are more likely to contain contextual information that provides rationale for an N-gram approach. Second, profile images are inevitably diluted by many noisy features, which sometimes make it even difficult for gender identification by direct visualization. In the section below, we report methods we experimented and their test results.

2 Methods and Results

The first decision we had to make was whether to incorporate words and images (or their dimensionality-reduced form) separately or together into our prediction model. Our quick initial logistic regression tests using PCA-ed images (PCA on the combined train and test images, as in HW6) alone only achieved <60% accuracy. This led us to think that images, or its PCA-ed form, alone might be a

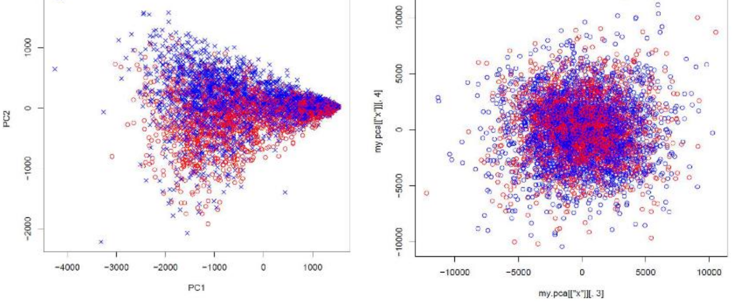

Fig. 1: Top 2 PCs of words (left) and top 2 PCs of images (right). Red circles denote females, whereas blue crosses denote males.

poor predictor of gender. In fact, whereas the first 2 PCs of words already show relatively significant partition between genders, the first 2 PCs of images fail to show a pattern of separation (Fig. 1). Therefore, we decided to combine both train and test observations and include word counts, PCs of images and image features. This results in a matrix with no. of rows=9995 (observations); and no. of columns=5000 (word count) + # of Image PCs + 7 (Image features).

Next, we implemented a series of methods, including generative methods such as Naïve Bayes (NB) and k-means clustering; discriminative methods such as logistic regression, decision trees (specifically random forests (RF)), and support vector machine (SVM); instance-based method such as k-nearest neighbors (KNN). Those with >69% accuracy are presented here in Fig. 2. It is worth noting that we selected from #5 to #30 and from #20 to #30 PCs of image data, because we were suspecting that some of the top-ranked PCs might not explain the variance in gender, a hypothesis that was later abandoned with cross validation when we train our final model. As seen in Fig. 2, the worst performing method was KNN, followed by NB, which was marginally better. On the other hand, discriminative methods such as logistic regression, RF and SVM classification worked much better, reaching ~83%, ~88% and ~83-87% accuracies, respectively (Note here that svm.SVC and svm.linearSVC are two slightly different implementation of SVC approach. The former[iv] is part of the Matlab-based libsvm package, whereas the latter is from python-based scikit[v] package. linearSVC uses a different decision hyperplane to address multiclass classification using SVM[vi], which, worked better in our hands when used for binary classification).

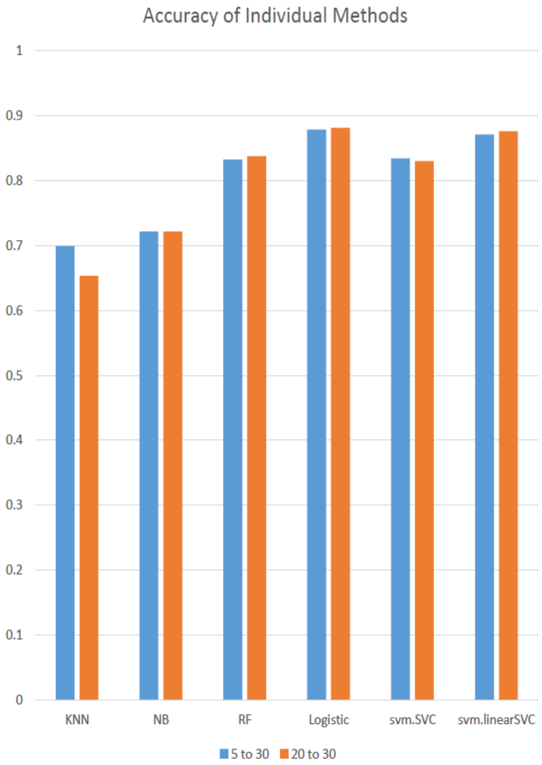

Fig. 2: Comparison of accuracy among individual type of methods. Blue bars represent results obtained by using 5-30 PCs of images as train data, whereas orange bars represent those with 20-30 PCs of images as train data. svm.SVC: from Matlab-based libsvm package. svm.linearSVC: from python-based scikit pckage

To improve our prediction accuracy, we attempted to use an ensemble model combining and weighing them according to their accuracy: 1st ensemble model:

$$KNN + NB + 2RF + 2LinearSVC + 3Logisic$$

This model reached 87% accuracy.

Suspecting that dropping the two lower-performing models might help further improve accuracy, we then came up with our 2nd ensemble model:

$$RF + Logistic + LinearSVC$$

Our 2nd ensemble model achieved 89.65% accuracy.

The final challenge for us was to reproduce this python-based, best model so far in Matlab, something that was made difficult by the lack of linearSVC in Matlab. We resolved this issue by substituting linearSVC with SVM in the libsvm package using a histogram intersection kernel function. We therefore arrived at our final ensemble model:

$$Logistic\ regression + RF + SVM$$

Here, with extensive cross-validation, we determined that logistic regression (from liblinear package) is given L2 penalty. RF is assigned a tree number of 250. SVM here (from libsvm) uses a histogram intersection kernel. Again, input features consist of all words, 1-30 PCs of images, which account for 75% variance, together with image features.

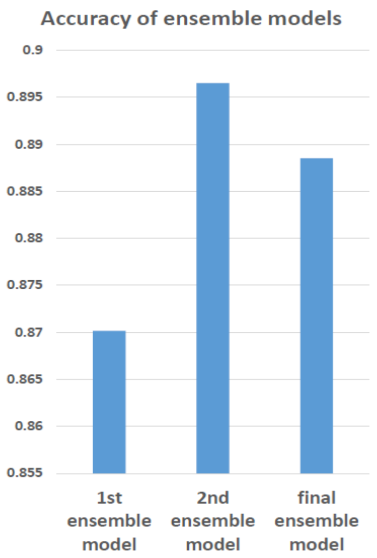

Fig. 3: a comparison of accuracy of three ensemble models

Our final model scored 88.85% with the validation set. A comparison among our ensemble models can be seen in Fig. 3. Furthermore, a complete table listing accuracies for most of methods we tested can be seen in table 1.

Individual Methods implemented in Python		
	5 to 30 PC	20 to 30 PC
KNN	0.699	0.653
NB	0.721	0.721
RF	0.832	0.837
Logistic	0.878	0.881
svm.SVC	0.834	0.830
svm.linearSVC	0.871	0.876
Methods implemented in matlab		
	1-30 PC	
adaboosting	82%	
kmeans	56%	
decision tree	72%	
NB	65%	
Emsemble methods		
	Accuracy of ensemble models	
1st ensemble model	0.870	
2nd ensemble model	0.897	
final ensemble model	0.889	

Table 1: accuracy table of methods we tested

3 Analysis and Discussion

To understand why some methods performed better than others, it is essential to consider the data type we are dealing with and the underlying principle of each method. With a >5000-dimension feature space, it was not surprising that methods such as KNN or K-means clustering did not work so well, due to the curse of dimensionality and the sparsity of data. It is plausible that some dimensionality reduction method that reduces dimension of words might help improve the performance of KNN or K-means. However, benefits of dimensionality reduction of information-rich word data has to be carefully weighed against the risk of worse performance by the ensemble method.

As for NB, it only scored marginally better than KNN. A probably explanation for it is that NB, which optimizes joint probability, relies upon an important assumption that features are independent from each other. Unfortunately, as we will see in next section, this is not true with words, a considerable number of which fall into certain topics or follow certain grammar rules, and therefore often appearing together. By comparison, logistic regression does not rely on such assumption; it models conditional probability of y given x instead. SVM, on the other hand, does not have any probabilistic assumptions-it only tries to minimize classification error. This explains why logistic regression and SVM worked much better than NB. It is also important to point out that selection of kernel function is critical to the success of SVM classification. Proven previously in homework and here in the project, intersection kernel works very well with training based on words.

It is interesting that a decision tree-based RF method worked well for us, an observation also reported by other teams. A possible explanation is that there is a fair number of words that provide enough information for gender identification so that when these word features are randomly selected for making decisions, they collectively are capable of contributing to a fairly accurate gender prediction. In fact, it is likely that the number of decision trees we chose, 250, based upon cross validation in which we saw accuracy does not decrease further between 300 and 500, was not enough, because the winning team reported using a tree number of 1000.

In addition to trying various methods to include in our final model, we also attempted several data processing tricks, hoping to add new informative features while reducing non-informative ones. For example, we obtained the average of each pixel for male and female and then calculated the sum of L2 norm of each image pixel, charactering how each given image deviates away from the averaged male or female image. In addition, we used a Matlab built-in face detection algorithm to extract faces and to reduce background. Finally, we converted all the images to grayscale, reducing 2/3 of image columns. None of these techniques were proved to be superior, and was thus not used in our final submission,

Things we would definitely love to investigate further to help improve our model include: 1) to optimize our data normalization in order to achieve better accuracy with SVM, which is not scale-invariant; 2) to separate words and images, subjecting them to different training methods, instead of always combining and training them together; 3) to test HMM-based as well as other algorithms for image recognition and feature extraction; 4) to carefully examine data points for which there is no consensus prediction, taking them to additional rounds of training; 5) to more extensively explore the model parameter space to find optimize our parameters.

4 Visualization and a Story

We set out to address the question that which image pixels and which image feature is most predictive of gender. Fig. 4 shows the negative log false discovery rate of each indexed pixel. It turns out that image pixels with indexes: 4196, 4596, 4696, 4797, 4198, 5067 and 5068 (colored in red) are the most predictive of gender. In terms of image features, Fig. 5 shows that feature #1 (age) , #2 (smiling) and #7 (face size) are the most predictive of gender.

A more thorough examination of words enable us to gain some interesting insights into different topics that interest male and female. For example, after we plot the ratio of normalized word frequencies, shown in Fig. 6, we are able to see that words that are most strongly associated with female users including those that fall into categories such as shopping: “@etsy”, “giveaway” and “#win”; jewelry: “earrings” and “necklace”; relationships “hubby” and “bestie”; and emoticons: “:o)” and “T_T”. By comparison, words that are most strongly associated with male users belong to topics such as finance: “trading” “#job” “usd”; technology: “nokia” “motorola” “microsoft” “android” and “developer”; and sports: “bulls.”

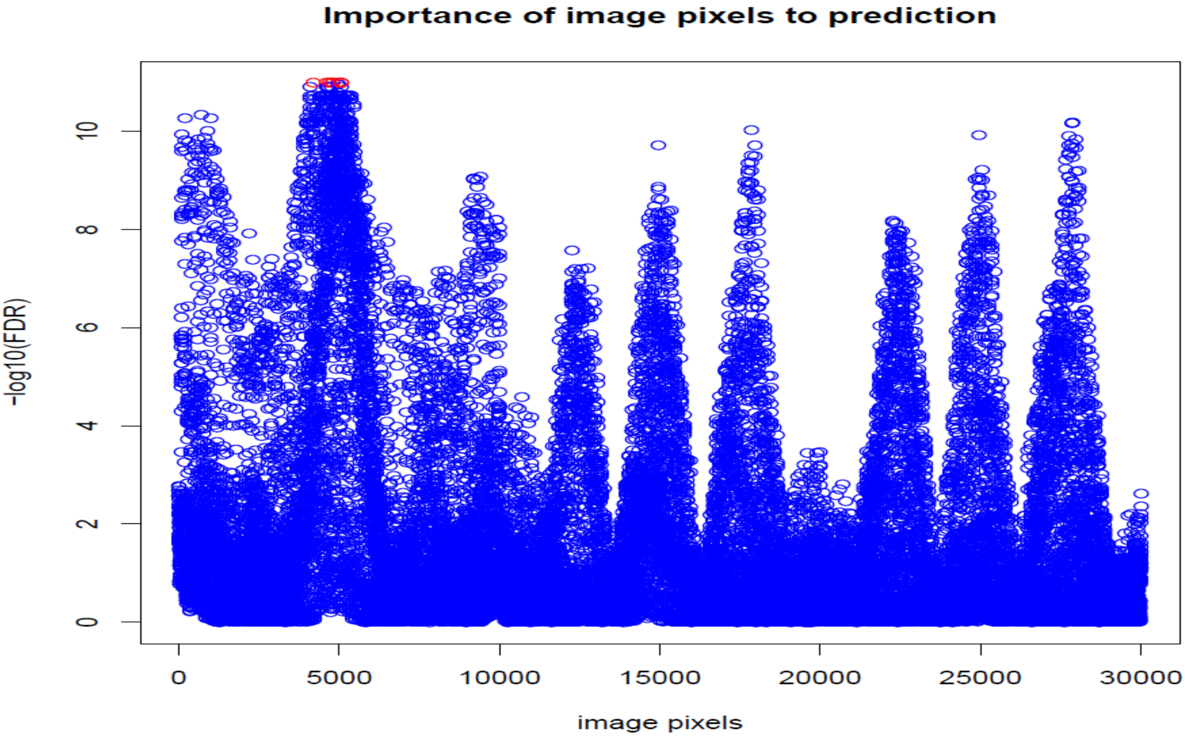

Fig. 4: -logFDR of each image pixel. Red data points denote pixels that are most predictive of gender.

Such findings make sense. Twitter’s distinguishing feature is that it is designed for disseminating condensed messages in real time. Therefore, topics that require rapid attention are more often tweeted by users who are interested, either due to work (journalism or trading) or due to hobby.

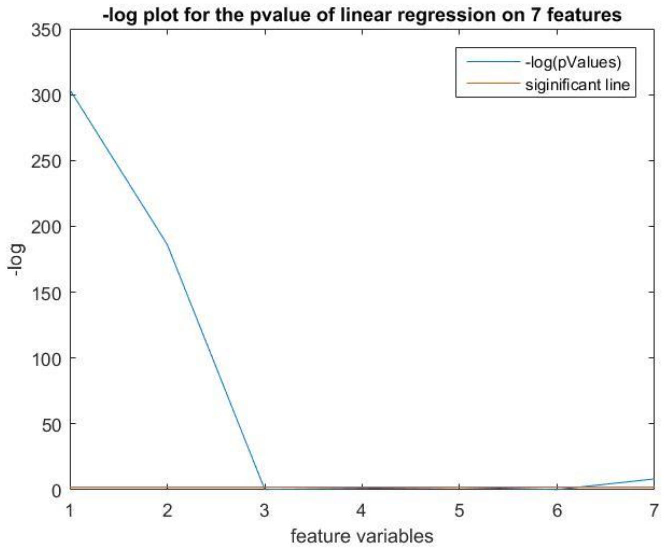

Fig. 5: p-value of each image feature in gender prediction: image feature 1, 2 and 7 are the most predictive

One fascinating observation here is that male Twitter users are more likely to mention "nokia" and "microsoft." Such observation, together with the fact that male users seem to use much more finance-related words, raises the possibility that during the period when this data set was collected, there might be some major corporate finance events involving Microsoft and Nokia. Indeed, a quick google search reveals that Microsoft purchased Nokia's devices business in April, 2014[vii]. We hypothesize that this event contributed to the fact that male users used "microsoft" and "nokia" much more frequently than female users. Furthermore, such an observation allows us to infer the dating of the collected data.

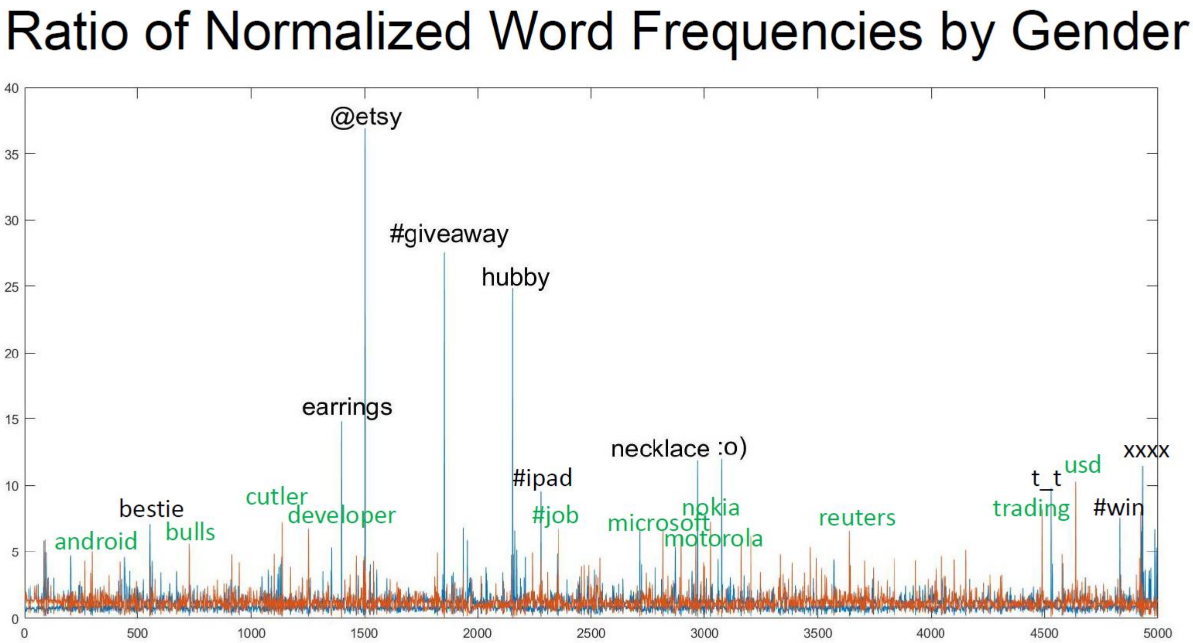

Fig. 6: Ratio of normalized word frequencies suggesting words more likely used by male or female. X-axis represents 5000 words; y-axis represents ratio of normalized word frequency. Signal for female is colored blue, while signal for male is colored in orange. Top words most indicative of genders is written in black for females and in cyan for males.

References

[i] Z. Miller, B. Dickinson and W. Hu, "Gender Prediction on Twitter Using Stream Algorithms with N-Gram Character Features," International Journal of Intelligence Science, Vol. 2 No. 4A, 2012, pp. 143-148. doi: 10.4236/ijis.2012.224019.

[ii] J. Burger, J. Henderson, G. Kim, and G. Zarrella. Discriminating Gender on Twitter. In Proceedings of the Conference on Empirical Methods in Natural Language Processing, EMNLP ’11, pages 1301–1309, 2011.

[iii] Quanzeng You, Sumit Bhatia, and Jiebo Luo. The eyes of the beholder: Gender prediction using images posted in online social networks. In Social Multimedia Data Mining, 2014.

[iv] Chih-Chung Chang and Chih-Jen Lin, LIBSVM : a library for support vector machines. ACM Transactions on Intelligent Systems and Technology, 2:27:1--27:27, 2011. Software available at http://www.csie.ntu.edu.tw/~cjlin/libsvm

[v] Scikit-learn: Machine Learning in Python, Pedregosa et al., JMLR 12, pp. 2825-2830, 2011.

[vi] Crammer and Y. Singer. On the algorithmic implementation of multi-class kernel-based vector machines. Machine Learning Research, 2:265–292, 2001.

[vii] Microsoft Closes Its $7.2 Billion Purchase Of Nokia, Business Insider, http://www.businessinsider.com/microsoft-closes-nokia-acquisition-2014-4