ML1819 Research Assignment 2

Team 17

Task 107 - How well can the gender of Twitter users be predicted?

Shubham Khanna
Department of Computer Science and Statistics
Trinity College Dublin
khannas@tcd.ie

Neeraj Athalye
Department of Computer Science and Statistics
Trinity College Dublin
athalyen@tcd.ie

Aditya Misra
Department of Computer Science and Statistics
Trinity College Dublin
misraa@tcd.ie

Shubham: Worked on SVM Classifier, data processing and the report

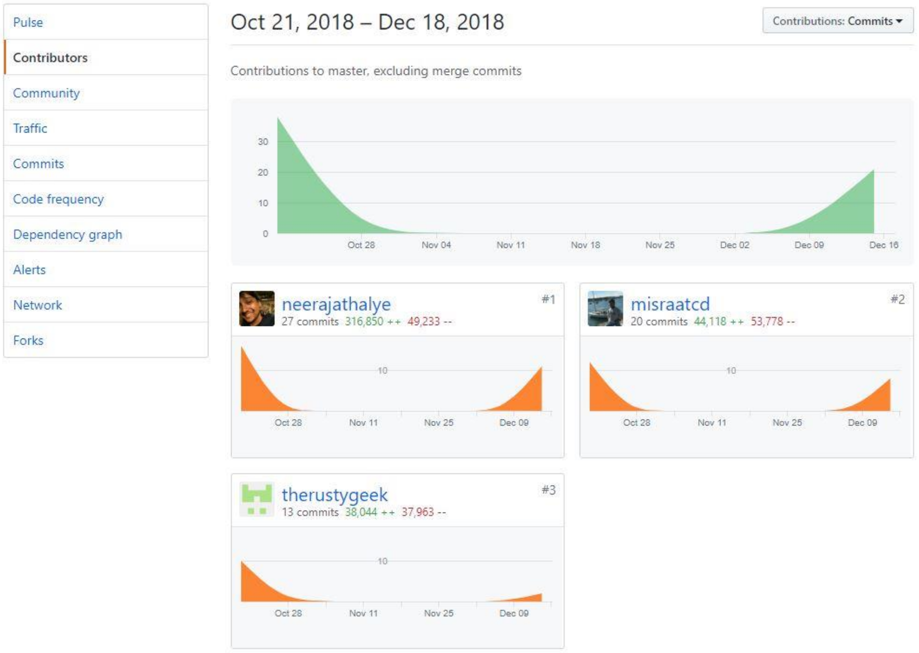
Neeraj: Worked on Logistic Regression, data processing and data visualization

Aditya: Worked on Naïve Bayes, data processing and data visualization

Word Count: 1497

GitHub Repository: https://github.com/neerajathalye/ML1819--task-107--team-17

GitHub Repository Activity: https://github.com/neerajathalye/ML1819--task-107--team-17



Oct 21, 2018 – Dec 18, 2018

Contributions to master, excluding merge commits

neerajathalye #1
27 commits 316,850 ++ 49,233 --

misraatcd #2
20 commits 44,118 ++ 53,778 --

therustygeek #3
13 commits 38,044 ++ 37,963 --

Twitter Gender Classification Based on User Profile

Shubham Khanna
Department of Computer Science and Statistics
Trinity College Dublin
khannas@tcd.ie

Neeraj Athalye
Department of Computer Science and Statistics
Trinity College Dublin
athalyen@tcd.ie

Aditya Misra
Department of Computer Science and Statistics
Trinity College Dublin
misraa@tcd.ie

ABSTRACT

In today's world, Machine Learning and Artificial Intelligence are increasingly becoming ubiquitous. Hence, it's natural to search for its applications in our daily life. Social media marketing plays a key role in augmenting revenues for a company. Therefore, it is essential to know about the target audience and their habits, gender plays a vital role governing such habits. In our paper, we try to demonstrate how ML can be used to determine the gender of a Twitter user. We have used three techniques namely, Logistic Regression, SVM and Naïve Bayes and have done a comparative analysis by varying the number of features. The highest accuracy achieved was in the case of SVM with an accuracy of 68.8% using text and tweet description as features.

Keywords- Gender classification, Twitter, machine learning, features, hyperparameters, Support Vector Machine, Logistic Regression, Naïve Bayes.

1. INTRODUCTION

Machine Learning classification can be used to divide the customer segments based on gender for better delivery of goods and services. To do the same, we have used a twitter dataset from Kaggle [1] which was used for the CrowdFlower AI predictor. The algorithms deployed for classification of gender are:

- Logistic Regression
- Naïve Bayes
- Support Vector Machine(SVM)

The remainder of this paper is structured as follows, in section 2, we discuss the related work, in section 3 we comprehensively discuss our methodology in which we discuss the implementation with various visualizations done for feature selection, section 4 describes the results and the findings of the study including the metrics used for the comparison analysis between the three algorithms with respect to, section 5 throws light on some of the limitations and the future scope of our study.

2. RELATED WORK

Fernández et. al [2], throw light on recent researches in the field of gender classification using machine learning techniques and how selection of features affects the. They used the Face ++ API for extracting the gender from profile pictures. The highest accuracy they achieved (89.53%) was with Neural Networks. Geng et. al [3], focused on soft biometrics which is the study of the relationship between gender and study of their virtual personality. They described some previous research on image-based gender identification, text-based gender identification and a hybrid model which utilized both. They described and compared the above three methods in terms of accuracy and AUC. Their proposition of a hybrid approach yielded an accuracy of 86.54% and AUC of 93.05%.

Geng et. al [3], focused on soft biometrics which is the study of the relationship between gender and study of their virtual personality. They described some previous research on image-based gender identification, text-based gender identification and a hybrid model which utilized both. They described and compared the above three methods in terms of accuracy and AUC. Their proposition of a hybrid approach yielded an accuracy of 86.54% and AUC of 93.05%.

3. METHODOLOGY

3.1 DATASET

The dataset 'Twitter User Gender Classification' from Kaggle [1] was used. The dataset was represented as a CSV file containing approximately 20000 rows and 26 columns like, the text of a random tweet, the author’s perceived gender, confidence value for gender, username, profile description, date of tweet, date of profile creation, number of favorite and retweeted tweets.

3.2 PRE-PROCESSING

3.2.1 Feature Selection

We identified the columns that did not provide useful information for gender classification like, _unit_id, _last_judgment_at, user_timezone, tweet_coord, tweet_count, tweet_created, tweet_id, tweet_location, profileimage and created. These columns were dropped from the dataset. Then, the dataset was filtered according to the column gender:confidence and only the profiles with a gender:confidence = 1 were kept. The dataset contained 4653 males and 5367 females. (Figure 1)

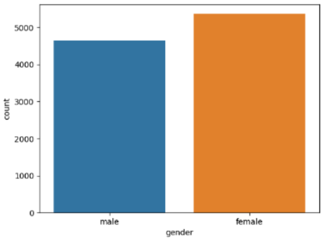


Figure 1: Male vs Female

3.2.2 Dealing with Missing Values

The gender column had the values: male, female, unknown and brand. The ‘unknown’ were the missing values and so we proceeded by deleting the users with the same and, we did not need to include the users having gender as ‘brand’ and therefore, that was removed as well. We inspected the profile_yn colum. "no" value here indicated that the profile was meant to be part of the dataset but was not available when contributors went to judge it [1]. When filtered, the users with profile_yn = no had gender as ‘NaN’. We removed the same and also dropped the profile_yn, profile_yn:confidence and profile_yn_gold, '_golden', '_unit_state', '_trusted_judgments', 'gender_gold' as they were not useful anymore.

3.2.3 Feature Visualization

We used Seaborn and Matplotlib libraries from Python to visualise the features to determine their use. After visualising the attribute sidebar_color(Figure 2), we concluded that, the top 3 colours of both male and female profiles were the same and hence column will not be a useful feature and hence, it was also dropped.

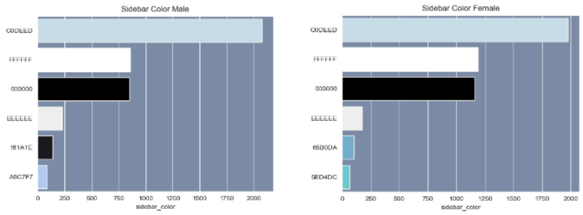


Figure 3: Sidebar Color (Male vs Female)

Visualizing the other remaining features proved that they could be used as features in gender classification.

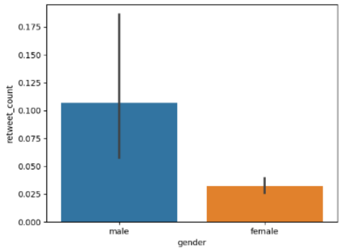


Figure 2: Retweet Count vs Gender

3.2.4 Cleaning the Dataset

We cleaned the text and description columns before using them as features. The punctuations and other html were removed.

Column	Description
gender	Male or Female (target variable)
text	Text of a random one of the user's tweets
description	The user's profile description
fav_number	Number of tweets the user has favourited
retweet_count	Number of times the user has retweeted
link_color	The link colour on the profile, as a hex value
name	The user's name

Table 1 - Viable Features and their Descriptions [1]

After pre-processing, we were left with the features shown in Table 1. Due to time constraints, we decided to use only ’text’ and ‘description’ as our features. Therefore, in our models, ‘gender’ is being used as the dependent variable and ‘text’ and ‘description’ are being used as independent variables.

3.2.5 Feature Normalization

We used the NLTK package to perform feature normalization which included *Stop Word Removal*, *Stemming*, *Calculating TF-IDF Scores* and *Label Encoding*. Stop Word Removal consists of removing the most frequently occurring words as they provide no useful information. Stemming converts words to their root form by removing suffixes like ‘-ing’ and ‘-s’.

After Stop Word Removal and Stemming, we made use of the TF-IDF vectorizer from the scikit-learn package to create TF-IDF scores for all the remaining words. The higher the TF-IDF score, the rarer the term. In Label Encoding,

3.2.6 Label Encoding

we encoded male as 0 and female as 1.

3.3 MACHINE LEARNING ALGORITHMS

We used Logistic Regression, Support Vector Machines(SVM) and Naïve Byes Classifiers to fit the data. Hyper-Parameter Tuning was done using GridSearchCV technique where a grid search was created using 4-fold cross validation method and the best hyper-parameters were found. We used the sci-kit learn package to implement these algorithms.

For Logistic Regression, Hyper-parameter C ranged from 0.001 - 10 with 2x and 5x incremental variations and the Penalties used were L1 and L2.
For Support Vector Machines, C ranged from 1 – 1000 with 10x increments), Kernels used were Linear and RBF and Gamma values were 0.001 and 0.0001
For Naïve Bayes, alpha ranged from 0.001 - 10 with 2x and 5x incremental variations

4. RESULTS AND DISCUSSION

4.1 METRICS AND RESULTS

We used Accuracy and F-1 Score as metrics to evaluate the performance of our models.

4.1.1 Accuracy

Represents what fraction of the samples were correctly predicted by the algorithm. It is given by the formula:

$$\text{accuracy}(y, \hat{y}) = \frac{1}{n_{\text{samples}}} \sum_{i=0}^{n_{\text{samples}}-1} 1(\hat{y}_i = y_i)$$

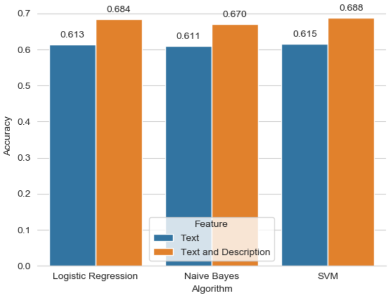


Figure 4: Accuracy of the various algorithms based on number of features used

4.1.2 F1-Score

Provides a balance between precision and recall. It can be represented as:

$$F_\beta = (1 + \beta^2)\frac{\text{precision} \times \text{recall}}{\beta^2\text{precision} + \text{recall}}.$$

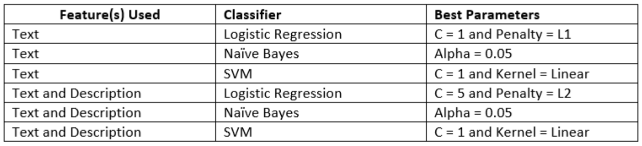
The F-1 scores obtained for the different number of features are mentioned in Table 2

Features	Logistic Regression	Naïve Bayes	SVM
Text	0.61	0.60	0.61
Text and Description	0.68	0.67	0.69

Table 2: F1 Scores for the algorithms by varying the number of features

4.2 DISCUSSION

In the end, we performed a comparative analysis of the algorithms used. We first used text as a feature and found out that SVM performed the best with an accuracy of 61.5% followed by Logistic Regression and Naïve Bayes. In the second case, we took text and description as features. We observed at least 6% boost in accuracy in all algorithms, however as we compare accuracies of the respective algorithms we observe a similar pattern with SVM performing the best with an accuracy of 68.8% followed by Logistic Regression and Naïve Bayes. (Figure 4)



Feature(s) Used	Classifier	Best Parameters
Text	Logistic Regression	C = 1 and Penalty = L1
Text	Naïve Bayes	Alpha = 0.05
Text	SVM	C = 1 and Kernel = Linear
Text and Description	Logistic Regression	C = 5 and Penalty = L2
Text and Description	Naïve Bayes	Alpha = 0.05
Text and Description	SVM	C = 1 and Kernel = Linear

Table 3: Hyper-Parameter Optimization

From Table 3, we can infer the best parameters for the algorithms used along with features selected for implementation.

From the results obtained, we observed that, even though SVM provides a higher accuracy, its execution time was substantially longer than Logistic Regression's execution time and therefore Logistic Regression is a better algorithm to use keeping in mind our dataset and computing power in hand.

In some related works based on twitter gender classification such as [2], SVM performed better than Logistic Regression and Naïve Bayes every time in terms of accuracy. This trend could also be observed in our unbiased comparative analysis between SVM, Logistic Regression and Naïve Bayes where SVM performed the best for both text and description. However, in [3] Random Forest algorithm provided a more robust solution as compared to Support Vector Machine.

5. LIMITATIONS AND OUTLOOK

The limitations of the dataset were that, the dataset contained a lot of NaN (not a number) values and useless features. Moreover, the dataset contains low quality profile images which made them unusable as a feature.

Also, our model uses only text and description as features instead of all the viable features we found due to our time constraints.

We believe that the model can be refined in the future with the help of:

- features like name, link color, retweet count and favorite count.
- Using a dataset with higher quality profile pictures along with Face++ API to generate gender confidence values.
- Using presence of emoticons from tweet text and descriptions as another feature

Moving forward on the project, we would make use of more features along with the features already used. Possible candidates for features are username, link color on the profile, profile image and link color.

ACKNOWLEDGMENT

This work was conducted as part of 2018/19 Machine Learning module CS7CS4/CS4 404 at Trinity College Dublin [7]

REFERENCES

[1] https://www.kaggle.com/crowdflower/twitter-user-gender-classification

[2] Daniela Fernández, Daniela Moctezuma, Oscar S. Siordia Features combination for gender recognition on Twitter users *2016 IEEE International Autumn Meeting on Power, Electronics and Computing (ROPEC 2016). Ixtapa, Mexico*

[3] Li Geng, Ke Zhang, Xinzhou Wei, Xin Feng. Soft Biometrics in Online Social Networks:A Case Study on Twitter User Gender Recognition. 978-1-5090-4941-7/17 $31.00 © 2017 IEEE DOI 10.1109/WACVW.2017.8.

[4] Burger J. H.J., G.K, G.Z., "Discriminating Gender on Twitter, Conference on Empirical Methods in Natural Language Processing, 2011

[5] K. C. Iliya, Predicting Gender On Twitter, Charles Iliya Krempeaux Blog. Software available http://changelog.ca/

[6] Marquardt James, Farnadi G., Vasudevan G., Moens MF, Davalos Sergio Age and Gender Identification in Social Media, PAN 2014, Amsterdam, 2014.

[7] Joeran Beel and Douglas Leith. Machine Learning (CS7CS4/CS4404). Trinity College Dublin, School of Computer Science and Statistics. 2018.