Supervised Learning Approach on Twitter Users' Gender Classification

Geeratigan Arsanathong*, Minjian Li+
School of Electrical Engineering and Computer Science, Washington State University
*g.arsanathong@wsu.edu
+minjian.li@wsu.edu

***Abstract*—The increase of Online Social Networks in society become more and more important since it one of the largest platforms that contain a vast amount of data. Twitter is one of the popular social media systems that provide a lot of useful information. In this paper, we aim to classify the user gender by using text posted on twitter by considering three models; SVM, Logistic Regression, and Naïve Bayes among several models in machine learning. We use training dataset to train on our model. Then, we improve our model based on the data we have and the model we use. Finally, we empirically analyze our result and report our findings**

***Keywords*—Machine Learning, Twitter, Gender Classification, Naïve Bayes, Logistic Regression, Supervised Learning, SVM**

I. Introduction

The popularity of Online Social Networks (OSNs) has been dramatically increased over the past decade. Notably, this can be considered as the biggest community that exists since there is an interaction among people or even someone who has never met before. Twitter is one of the most powerful tools for the research since a massive amount of information is provided with the users. There are about 500 million tweets shared per day [5]. Even though most of the messages given on twitter are unstructured and anonymous, it still can be able to identify the gender, age, country, language, or even the background of the user.

Due to the increase of twitter users and information, the approach to the problem of gender classification by using text message is hugely required, especially in the marketing industry. In this paper, we aim to answer how well our model can perform to classify user gender by using the text that was posted on twitter. However, most of the Social Networks resources contain unstructured data, which rarely have an analytical method that can directly work with [3]. Another challenge is the limitation of gender classification. There is an existence of class that cannot identify as either a woman or a man. Particularly, there is no specific technique to handle this problem.

In our paper, we apply three supervised learning approaches; SVM, Logistic regression, and Naïve Bayes to classify gender using text from twitter. Those three models are primarily aimed at bag-of-words representation. Also, it seems to perform well for the small number of datasets.

In the sequel, we report our empirical results on the three different models and evaluate our model using metric for classification model such as accuracy, recall and precision. We found that the testing accuracies of selected classification models are approximately close to 50%. According to the experiment result in this project, we found that the issue on gender classification in twitter has been making little progress in developing into fruition.

II. Problem Setup

In order to address the limitation problem of gender classification on tweets, this project mainly focuses on exploring and analyzing to what extent the three selected classification models perform on this problem. If the performance among them is similar, we will try to modify several parameters at the algorithm, try to modify the size of each dataset — training, validation, and testing, and/or try to nullify the correlation among features, to observe whether the classification models achieve a better goal or not.

III. Approach

Classification is a requirement for a machine to become more intelligent as if a compulsory course that a student requires to take. In the classification area, there are multiple techniques (usually called classifiers) such as perceptrons, decision trees, and random forests which are applied to train a machine to learn a dataset. Typically, perceptron is popular approach for beginners to train a machine, since it is the easiest way to be implemented in programming. However, perceptrons have several flaws. For example, in binary classification, Fig.1 illustrates perceptron have a hard margin (green line), while other classifiers have a soft one.

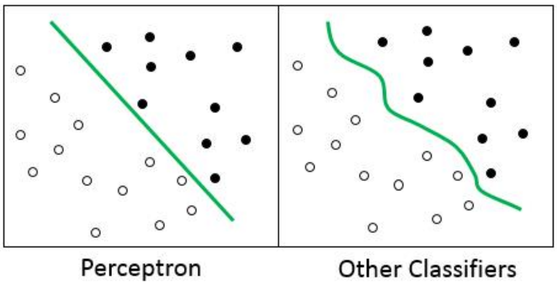


Fig.1 different margin between perceptron and other classifiers

Moreover, in order to push the margin to any position, a biased sample feature x_0 is always added to all training examples, otherwise, the margin of perceptron will always go through a specific point no matter how many iterations it takes to optimize its margin. In other words, perceptron might have no chances to generate a margin to separate samples in certain cases.

In this project, the selected classifiers are implemented to classify a tweet dataset, rather than perceptrons, to observe their performance.

A. SVM

SVM is a supervised learning model, standing for the support-vector machine also referred to as support-vector networks, which decision boundary is to solve the maximum-margin hyperplane of training examples [1]. SVM outstandingly deals with the high-dimensional classifications other than perceptrons. Considering the decision function of binary SVM,

$$\boldsymbol{sign}\left(\textstyle\sum_{i=1}^{n} \boldsymbol{y_i \alpha_i K(x, x_i)} + \boldsymbol{b}\right) \tag{1}$$

in (1), where *n* is the total number of training examples, $\boldsymbol{y_i}$is labels of sample, $\boldsymbol{\alpha_i}$is support vectors, and *K* is a kernel function, SVM only based on a couple weight vectors, instead of dimension of the sample space, to predict the labels. Also, with a controlling parameter, SVM is able to maximize the smoothness of its margin in order to reduce the generalization error and training error.

B. Logistic Regression

It is worth noting that linear regression is not only a regression model but also is a classifier, especially logistic regression. Logistic regression is a general linear regression analysis model widely used in practice in terms of data mining, automatic diagnosis, and economic forecast [4]. A logistic regression to solve multi-class classification problems is called multinomial logistic regression [6]. The following formula:

$$\hat{y} = score(X_i, k) = \beta_k \cdot X_i \tag{2}$$

is the prediction function of a multinomial logistic regression classifier. In (2), X_i denotes the *i*-th vector of feature in vector space *X*, *k* is the number of categories, and β_k is a vector of weights corresponding to the *k*-th categories. Basically, certain features affect significantly on categories, but some are not. By observing the prediction function (2), multinomial logistic regression provides each category a vector of weight, and the vector of weights are independent with each other, which means the prediction will not be affected by the noise samples in a dataset. Therefore, logistic regression works much better than perceptron on balancing the effects of features on each category.

C. Naïve Bayes

In the training process, perceptron needs to be repeatedly trained in several iterations to improve its accuracy. However, Naive Bayes is a speedy probabilistic learning model that only needs to be trained once. The mechanics of Naive Bayes classification is to calculate the probabilities of each category using Bayes rules. For example, in this project, suppose that a sentence *S* is "She's broke up with him". If a Naive Bayes classifier tries to classify the gender of someone who said this sentence, it will compute the probabilities amount different genders as following,

$$predict(y_k) = \alpha \cdot P(y_k) \textstyle\prod_{i=1}^{n} P(x_i \mid y_k), k \le n \tag{3}$$

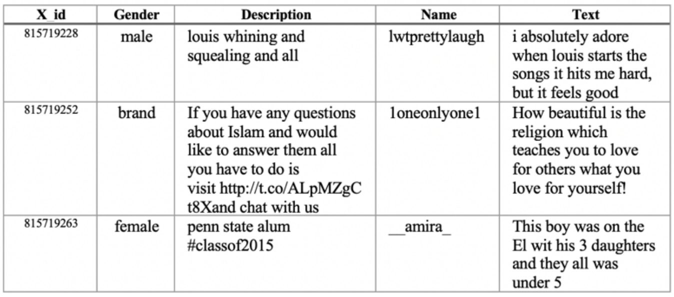
where, in (3), y_k is labels, α is evidence but it does not need to be calculated, *n* is the number of words in the given sentence, and $P(x_i \mid y_k)$ is representing the probability of someone who said the word x_i, is gender y_k . After computing all $predict(y_k)$, the classifier will pick the maximum $predict(y_k)$, as its output. Thus, Naive Baye is a fast-learned classifier.

Moreover, the accuracy of Naive Bayes approximately remains stable, while the size of dataset changes on a large scale. Even if the size of training set is too small but contains enough features, Naive Bayes can still perform pretty well.

IV. Experiment and Results

We collect the Twitter User Gender dataset Kaggle. The dataset contains 20,050 rows with 26 columns. This dataset includes several features such as user's id, gender, user's profile description, text, or color of the profile sidebar. We observed the 'Gender' column as labels that contain Band, Male, Female, and Unknown strings. We present found some exciting features offer as Table I shows:

TABLE I
THE SAMPLE OF DATA SET



X_id	Gender	Description	Name	Text
815719228	male	louis whining and squealing and all	lwtprettylaugh	i absolutely adore when louis starts the songs it hits me hard, but it feels good
815719252	brand	If you have any questions about Islam and would like to answer them all you have to do is visit http://t.co/ALpMZgCt8Xand chat with us	1oneonlyone1	How beautiful is the religion which teaches you to love for others what you love for yourself!
815719263	female	penn state alum #classof2015	__amira_	This boy was on the El wit his 3 daughters and they all was under 5

In this paper, we will train data with vocabulary features which we collect from text features in our original dataset by a base learning algorithm, which are Support Vector Machine, Logistic Regression, and Naïve Bayes.

A. Evaluation Methodology

According to our three models, SVM, Logistic Regression, and Naïve Bayes Classifier, we first consider the classification accuracy (4), which is the number of correct predictions made overall kind prediction made.

$$Accutacy = \frac{Number\ of\ Correct\ predictions}{Total\ number\ of\ predictions\ made} \quad (4)$$

This is a basic and essential measurement when the target variable classes in the data are nearly balanced. On the other hand, if we use this accuracy measure on the imbalance data when there is a majority of one class, the prediction will fall into false sense even the model can achieve high accuracy.

The second metric is precision (5). This measure tells us what proportion of users that we classify as the specific gender is actually be. Precision tells us about its performance respect to false positive.

$$Precision = \frac{TruePositives}{TruePositives+FalsePositives} \quad (5)$$

The last one is recall, in (6), which tells us what proportion of user that labeled as specific gender was classified by the algorithm as having the same gender as labeled. Recall informs us about a classifier's performance with respect to a false negative.

$$Recall = \frac{TruePositives}{TruePositives+False\ Negatives} \quad (6)$$

In this section, we also present the optimization of the hyper-parameters for the algorithm we used to improve the model's generalization performance. In this case, we use grid search which is one of the sample techniques to select the values for a model's parameters that maximize the accuracy of the model.

In SVM algorithm, we consider SVM with linear kernel, so we consider the regularization parameter, C. We set six different setting; 0.001, 0.01, 0.1, 1, 10, 100 for C. We trained validation data with our six different C and selected the best one with the best accuracy to test on testing dataset. Also, in Logistic Regression, we consider six different settings; 0.001, 0.01, 0.1, 1, 10, 100 for C, and we also tune the penalty l1 and l2 in this model. However, Naïve Bayes has nothing to do with grid search.

B. Baseline Approach

Our approach can be summarized as follows:

- We obtained the dataset from Kaggle.
- We applied the R studio tools for data preprocessing and tokenization step.
- We split data into training, validation and testing dataset.
- We trained, tested and validated our dataset using three difference model.
- We created the evaluation metric to check the performances of our model.

In the first step, we gathered the dataset from Kaggle. The dataset contains several features that we found text feature is necessary for us. Since we would like to predict user gender by their text posted on twitter, we dropped most of the features and considered only text features in this project.

In Step 2, we used R studio for the data preparation step. After we knew what feature we would like to use, we created the new data frame with our feature. We found that the dataset contains some missing values and unknown data, so we dropped the rows with those issues. In this step, we need to split our text into tokens, so our data need to be clean. We need to remove all invalid characters such as HTML tags, links, punctuation, and symbols. Also, we create the new label class, which 1 for brand (non-human), 2 for female, and 3 for male. The sample of cleaned dataset present as Table II below:

TABLE II
THE SAMPLE OF DATASET WITH CLEANED TEXT

User_id	Gender	Text
815719228	3	i absolutely adore when louis starts the songs it hits me hard but it feels good
815719252	1	how beautiful is the religion which teaches you to love for others what you love for yourself
815719263	2	this boy was on the el wit his 3 daughters and they all was under 5

In the tokenization step, we use one of the most straightforward techniques to represent text is Bag of Words numerically. In this step, we will make the list of unique words in the text corpus called the vocabulary list. We served each sentence as a vector with their word, which 1 represents for present and 0 for absent from the vocabulary list. Then, we removed stop words such as "the", "is", etc., which do not have specific semantic, remove sparse words that less than 0.3% of the document, and also do stemming the words that are likely to denote the similar context. As a result, we got 344 vocabularies (features) with 18,836 documents.

In step 3, we shuffle the dataset by generating a random number for each example and then reorder the dataset according to the random numbers (from large to small). We split data into three datasets, which we have 70% for training, 20% for validation, and 10% for testing data as shown in Table III.

TABLE III
THE NUMBER OF CATEGORIES

Brand	Female	Male
5942	6700	6194

Then, we ensure that the proportion of all labels in each group is approximately equal and large enough as the following Table IV.

TABLE IV
THE PROPORTION OF CATEGORIES IN EACH GROUP

Dataset	Brand	Female	Male
training	4185	4712	4288
	70.5%	70.3%	69.2%
validation	1172	1309	1286
	19.7%	19.5%	20.8%
testing	585	679	620
	9.8%	10.2%	10.0%

In step 4, we use the essential library in Python, such as scikit-learn [6] to build our model, which are SVM, Logistic Regression, and Naïve Bayes Classifier and test on our dataset. We try to produce the best result by adjusting some of the hyperparameters needed for the specific model.

Finally, in step 5, we print out the confusion matrix and some statistical results after we have produced a model that generates the best predictions. Then, we will discuss our empirical results in the following section.

C. Results

In this section, we train our three models with training data, evaluate it on validation data. Then, we picked the best parameter by tuning its parameter. Finally, we come up with the accuracy of training, validation and testing data with the best parameter it calculated.

TABLE V
THE PERFORMANCE OF DIFFERENT CLASSIFIERS

Model	Score		
	Training	Validation	Testing
SVM	0.511339	0.474117	0.479830
Logistic Regression	0.508912	0.476507	0.483015
Naïve Bayes	0.493440	0.465890	0.453291

Table V shows the accuracy of training performance of three models with the best performance. In this training, we picked C = 10 for SVM since it provided the best result when we used the grid search. Also, we used penalty = l1 and C = 1 for logistic regression. We did nothing for Naïve Bayes since it does not work with the parameter.

In this paper, we also present the confusion matrix shown in table 6, 7, 8 and statistical result which better provide an information to describe the performance of a classification model.

TABLE VI
THE CONFUSION MATRIX OF SVM

	Predicted		
Actual	Brand	Female	Male
Brand	271	135	179
Female	92	338	249
Male	72	253	295

In Table VI, we also calculate the precision for brand, female and male equal to 0.62, 0.47 and 0.41, respectively. Also, we have recall equal to 0.46, 0.50 and 0.48 for brand, female and male, respectively.

TABLE VII
THE CONFUSION MATRIX OF LOGISTIC REGRESSION

	Predicted		
Actual	Brand	Female	Male
Brand	513	272	387
Female	143	693	473
Male	184	513	589

Table VII informs us about the precision and recall of logistic regression as follows: 0.61, 0.47 and 0.41 for precision in the order of brand, female and male. Also, we have recall for brand, female and male as 0.44, 0.53 and 0.46, respectively.

TABLE VIII
THE CONFUSION MATRIX OF NAÏVE BAYES

	Predicted		
Actual	Brand	Female	Male
Brand	427	462	283
Female	87	843	380
Male	95	705	486

From Table VIII, we will get precision about 0.70, 0.42, 0.42 in sequence of brand, female and male and recall is 0.36, 0.64 and 0.38 for brand, female and male in this order.

The three confusion matrixes above show how our classification model in confused when it makes a prediction. The main diagonal gives the correct prediction. It clearly sees that male gives the highest wrong prediction in Naïve Bayes. However, if we look more profound thought those confusion matrixes, we will see the wring prediction seems to fall to female instead of brand. These are the result using different classifiers and our findings.

V. CONCLUSION

The testing accuracies of selected classification models are close to 50%. According to the experiment result in this project, none of them indeed break the limitation problem of gender classification on tweets. The main facts are twofold. The data set is not large enough. Since the dataset is too small to contain the entire vocabulary system, classifiers are not allowed to learn deeper. Another possible hypothesis related to the result is that the link between features and labels is week. In other words, the words that the tweets contain are not related to genders, which means, there is no gender discrimination on such tweets. Thus far, researchers still try hard to surmount this tricky topic. However, none of them have an effective way or algorithm to nullify the limitation in gender classification. Perhaps in the future, a more powerful classifier should be created to resolve this problem as the quantum computing techniques are emerging, or this problem can never be solved since its solution belongs to a higher-dimensional space time.

VI. Acknowledgement

We would like to express our special gratitude to Professor Doppa for his professional advice and guidance on this difficult project. We would also like to thank the librarians in our university. They were nice, friendly, and always willing to help us. Thanks to our university for its wonderful library with colossal amounts of resources and materials for researchers.

References

[1] Cortes, C. & Vapnik, V. “Support-vector networks”. Machine Learning. pp. 273-297

[2] “Scikit-Learn: Mahine Learning in Python.” Available at: https://scikit-learn.org/stable/

[3] S. Tripathi (2018), “Analytics On Unstructured Data - Twitter, Facebook And Social Media,” *Analytics Training*. Available at: https://analyticstraining.com/analytics-on-unstructured-data-—-twitter-facebook-and-social-media/

[4] Tolles, J. & Meurer, W. J. “Logistic Regression: Relating Patient Characteristics to Outcomes”. *JAMA*. (2016). vol. 316, no. 5, pp.533–534. doi:https://doi.org/10.1001/jama.2016.7653

[5] “Twitter by the Numbers (2019): Stats, Demographics & Fun Facts,” *Omnicore*. Available at: https://www.omnicoreagency.com/twitter-statistics/.

[6] Wright, R. E. (1995). Logistic regression. In L. G. Grimm & P. R. Yarnold (Eds.). *Reading and understanding multivariate statistics*. pp. 217–244.