

Section 3. Psychology

<https://doi.org/10.29013/YSJ-22-1.2-18-22>

*Sikun Gan,
Coventry Christian School
High School Student*

BIG DATA EMOTION CLASSIFICATION

Abstract. Affecting our mind in terms of decision-making, influencing our moods and behaviors, emotion makes up a major part of our daily lives. People's physical and mental health and work status can be adversely affected by persistent negative emotions, but positive emotions can enhance subjective well-being and promote physical and mental health. I then formulated a question that can positive/negative emotion be automatically classified using a model, i.e, does the sentence contains enough information for the computer to make a sentiment judgment. With a massive amount of text data, I here build up an automatic emotion classification model that could read and distinguish sentences with negative emotions from sentences with positive emotions. Specifically, I studied the penalized logistic regression model with Stanford movie review data as the input. The AUC metric is used for model evaluation and outputted a promising out of sample score of 0.96.

Keywords: Logistic regression, big data, sentiment classification.

1. Introduction

Emotion makes up a major part of our daily lives. It affects our mind in terms of decision-making, influences our moods and behaviors, and so forth. To name a few, there are investment strategies developed based upon the investor's emotion gathered from stock comments. There are automatic question and answering systems that feed users with different answers based upon the emotion in the online chat window. In general, understanding one's emotions can extrapolate his behaviors, actions, and mental health condition. With the recent advancement of technology, most people can access the Internet and social media. The majority can share their emotions via many different platforms, such as Twitter, Instagram, and Tik Tok. Therefore, by utilizing user-generated content in the correct

manner, we will be able to gauge people's mental health. It could be possible to predict mental health levels and depression by mining the content from social media platforms. Depression is a serious medical condition that hampers the ability to perform normal daily tasks such as working, studying, eating, sleeping and having fun. Thanks to the rapid growth of computer utility, we can develop automatic tools for classifying and identifying emotions behind the posts today by incorporating databases and algorithms.

2. Stanford Movie Review Dataset

To study the emotion classification, I adopt the Stanford Movie Review Dataset which contains 25,000 reviews for popular movies. The data is in text format with the following print out examples:

label: 1 review: For a movie that gets no respect there sure are a lot of mem
 label: 1 review: Bizarre horror movie filled with famous faces but stolen by
 label: 1 review: A solid, if unremarkable film. Matthau, as Einstein, was won

The dataset has been used and cited many times in natural language processing field.

3. Data Preparation

The data is in the text format which requires me to conduct data cleaning first. One way of extracting the emotion information from the text is to count how many times does the positive emotion related words show up in the text. One way to count the occurrence of the keywords in a sentence is to use the CountVec-torizer function from python package sklearn.

However, such a method is not ideal because some of the keywords with high frequency are mean-ingless in the sense that it appears in 99% of the text. For example, the words “the” and “a” are commonly used in all contexts of English language. To better ex-tract the useful keywords summarizing the sentence emotion, I adopt the term frequency-inverse docu-ment frequency, also known as TF-IDF. Specifically, Term frequency is how many times does a word ap-pears in the comment, which is defined:

$$TF = \frac{\text{time of occurrence}}{\text{total number of words in the article}}$$

To calculate IDF, we need a corpus that contains every possible word. The formula should be

$$IDF = \log\left(\frac{\text{the number of document}}{\text{the number of document contains a word} + 1}\right)$$

Finally, we can get TF-IDF by multiplying TF and IDF, $TF * IDF$. By adopting TF-IDF, we can summarize the information within the movie comments with a list of important keywords. The Sklearn package also provide a function TfidfVec-torizer, which facilitate my research for TF-IDF computation.

4. Model

With the TF-IDF numerical data matrix as the input, I adopt the logistic regression model for emotion classification. Because the data contains too many parameters, I applied L1 penalization.

4.1 Model setup

The logistic regression models the probability of the labeled data. It has wide and successful applica-tion in statistics. The model starts by modeling the probability of the comment to be positive with the following equation:

$$\text{Log} \frac{p}{1-p} = \beta_0 + \beta_1 X_1 + \beta_2 X_2 + \dots + \beta_p X_p$$

where:

- p is the probability for the comment to be positive;
- X_1, X_2, \dots, X_p are p keywords appearing in the columns of the TF-IDF matrix.

The model normalizes the RHS equation to do-main $[0,1]$ but one can always obtain the probabili-ty for a comment to be positive with the following equation:

$$p = \frac{\exp(\beta_0 + \beta_1 X_1 + \beta_2 X_2 \dots + \beta_p X_p)}{1 + \exp(\beta_0 + \beta_1 X_1 + \beta_2 X_2 \dots + \beta_p X_p)}$$

4.2 Binomial Probability

To connect the data to the probability model, the logistic regression assumes to use $Y = 1$ to indicate the emotion is positive and to use $Y = 0$ to indicate the emotion is negative. For example, suppose we observe a sentence/comment with positive emotion, the probability for that sentence being positive is p and the probability for that sentence being negative is $1-p$, which can be rewritten in the following form:

$$P(Y) = p^Y (1-p)^{1-Y}$$

If we observe more than one sentence, assum-ing each of the sentence being independent, we have their joint probability being modeled by the product of each individual probability:

$$P(Y_1, Y_2, Y_n, \dots, Y_n) = \pi_{i=1}^n p_i^{Y_i} (1-p_i)^{1-Y_i}$$

Intuitively, we know that depending on the text used in the review sentence, the probability for its emotion to be positive should be different from

sentence to sentence. This coincides with the product of the binomial probability above because the model assumes the probability for the i -th sentence being positive to be p_i . From the previous section, we know that p_i can be modeled with the following equation:

$$p_i = \frac{\exp(\beta_0 + \beta_1 X_{i1} + \beta_2 X_{i2} \dots + \beta_p X_{ip})}{1 + \exp(\beta_0 + \beta_1 X_{i1} + \beta_2 X_{i2} \dots + \beta_p X_{ip})}$$

4.3 Model solution

The above model is now defined on a set of parameters β . The model solution is found by maximizing the probability $\pi_{i=1}^n p_i^{y_i} (1-p_i)^{1-y_i}$ due to the assumption that the probability model we propose should best explain the data by letting the observed data have the highest probability. To facilitate the maximization, the model is usually solved by taking the log operation, after which we could apply the chain rule from calculus to solve for the maximum point:

$$\operatorname{argmax}_{\beta} \frac{1}{N} \sum_{i=1}^n y_i \log(p_i) + (1-y_i) \log(1-p_i)$$

For my research, I used python Sklearn package to solve for $(\beta_0^*, \beta_1^*, \dots, \beta_p^*)$ that maximize $\pi_{i=1}^n p_i^{y_i} (1-p_i)^{1-y_i}$

4.4 Penalized Logistic Regression

Ideally, we should know that some of the $(\beta_0^*, \beta_1^*, \dots, \beta_p^*)$ to be zero because the TF-IDF matrix contains a lot of 0s. To force some of the parameters to be 0, the model can be extended to penalized logistic regression, which minimizing the following equation instead of solely maximizing the joint probability.

$$\operatorname{argmin}_{\beta} -\frac{1}{N} \sum_{i=1}^n y_i \log(p_i) - (1-y_i) \log(1-p_i) + \lambda \left(|\beta_0| + |\beta_1| + \dots + |\beta_p| \right)$$

The first term in red is essentially the negative of the log probability, minimizing of which is equivalent to maximizing the negative of itself. The second summation is a penalization term where if some of the β_i s are not zero, then it will bring up the optimization function value. Essentially, the model is looking for a solution of $(\beta_0^*, \beta_1^*, \dots, \beta_p^*)$ that maximizes

the binomial probability while using the fewest number of β s.

5. Result Analysis

5.1 Train-Test Dataset Split

After we find our model solution $(\beta_0^*, \beta_1^*, \dots, \beta_p^*)$, given a new sentence X_{new} , we will be able to find the positive sentiment probability by

$$p_i = \frac{\exp(\beta_0^* + \beta_1^* X_{i1}^{new} + \dots + \beta_p^* X_{ip}^{new})}{1 + \exp(\beta_0^* + \beta_1^* X_{i1}^{new} + \dots + \beta_p^* X_{ip}^{new})}$$

We could simply adopt 0.5 as the cutoff value to make emotion classification decisions. Specifically, if the probability is above 0.5, we set the sentence emotion to be positive, otherwise, we set the sentence emotion to be negative. To better evaluate my model, I used 75% of the data for finding my parameters $(\beta_0^*, \beta_1^*, \dots, \beta_p^*)$ and pretend the rest 25% of the data as new observations for model evaluation. Statistically speaking, the 75% data used is called the training set and the rest 25% of the data is called testing set.

5.2 ROC Curve

To evaluate the model performance, we could simply use the accuracy score

$$\text{Accuracy} = \frac{TP + TN}{TP + TN + FP + FN}$$

Where TP = True positive; FP = False positive; TN = True negative; FN = False negative.

However, such a evaluation might not be accurate if the data has 98% of the review comments to be positive, in which case, one naïve model of rating all the comments to be positive would give us 98% accuracy. To comprehensively evaluate the model performance, I study the False Positive Rate(FPR) and True Positive Rate(TPR):

$$TPR = \frac{TP}{TP + FN}, FPR = \frac{FP}{FP + TN}$$

Under the FPR evaluation, if a model simply rates all the comment to be positive, it will have a low FPR.

Another issue with the model evaluation is that adopting 0.5 as the cutoff value makes intuitive sense but might not be the best approach. In reality, we

might want to be conversant by setting the cutoff value to be 0.6, or even 0.9. Each choice of the cutoff value can thus give us different FPR and TPR. To comprehensively measure the model performance with different cutoff value, I study the ROC curve defined below.

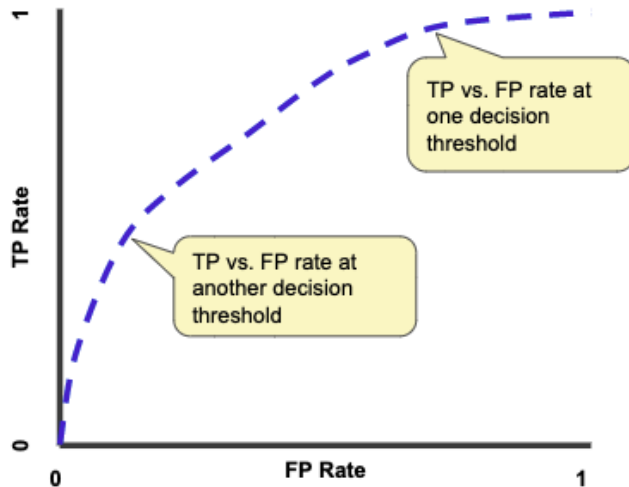


Figure 1.

Where the x-axis is the FPR and the y-axis is the TPR. For each cutoff value, we can obtain a pair of (TPR, FPR) and label it on the above coordinate. If we keep change the cutoff value from 0 to 1, we get many dots and if we connect those dot, it gives us the ROC curve. A good model thus should have the curve close to the top-left corner because it indicates a lower FPR and a high TPR. We know that the Area of the coordinate is 1 and the higher the area under the ROC curve, the better the model is. Consequently, the Area Under the Curve (AUC) is a natural judgement on the classification model performance.

I finally evaluate my model with the 25% testing set on the ROC curve and obtain the following ROC plot, which has AUC score of 0.96.

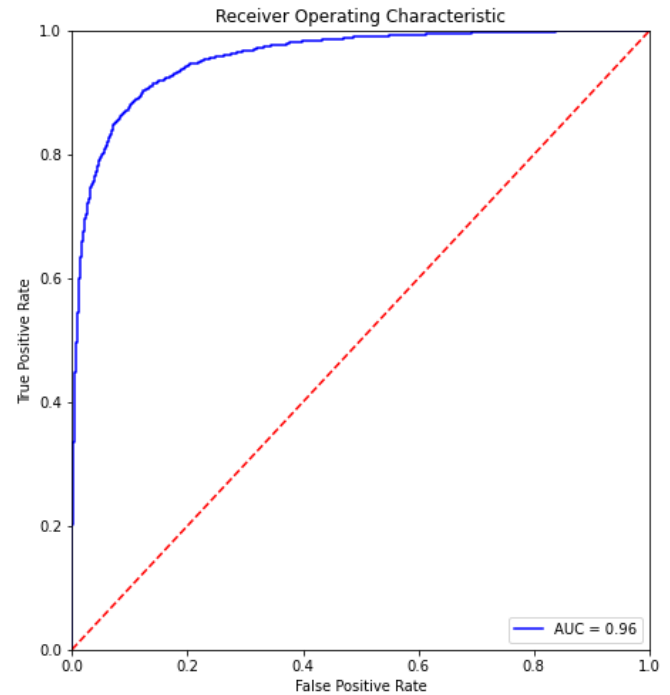


Figure 2.

6. Conclusion

With the TF-IDF metric to summarize the sentence information and with penalized logistic regression to model the sentiment probability, I build up a probability model for sentiment classification. To take the sample imbalance into consideration, I also researched the model evaluation by studying the confusion matrix and ROC/AUC metrics. The out of sample AUC score, 0.96, indicates that my penalized logistic regression can distinguish the negative movie comments from positive comments with high accuracy. With the preliminary exploration of emotion classification, I can further research to create a model that is able to classify different types of emotions beyond mere two polarities, thus can estimate people's depression levels with more confidence and maintain their well-being.

References:

1. All of Statistics: A Concise Course in Statistical Inference, Larry A. Wasserman. 2004.
2. Linear Regression Using R: An Introduction to Data Modeling, David J. Lilja. 2016.
3. Sentiment Analysis. URL: <https://ai.stanford.edu/~amaas/data/sentiment>
4. Learning Word Vectors for Sentiment Analysis, Proceedings of the 49th Annual Meeting of the Association for Computational Linguistics: Human Language Technologies, 2011.

5. Scikit-Learn: Machine Learning in Python – Scikit-Learn 1.0 Documentation. URL: <https://scikit-learn.org/stable>
6. Classification: ROC Curve and AUC | Machine Learning Crash Course. Google Developers. URL: <https://developers.google.com/machine-learning/crash-course/classification/roc-and-auc>