Facebook Like Predictor Within Your Friends

Kevin Chen
Northwestern University
KevinChen2016@u.northwestern.edu

Basil Huang
Northwestern University
BasilHuang2014@u.northwestern.edu

Brittany Lee
Northwestern University
BrittanyLee2015@u.northwestern.edu

Abstract

Over a billion users each month share their lives on Facebook [1]. As one of the largest social networking websites, Facebook has easily become the go to place to learn about someone's colleagues and friends. One of main features of Facebook is the ability to like someone's status, which shows general approval of what that person wrote. Maybe the status was funny, shared an important accomplishment or was relatable which resulted in a popular post with many likes. We attempt to learn what contributes to a popular Facebook status in order to predict how many likes, a particular status will receive.

1. Introduction

Social media has become a pervasive part of our everyday lives that affects how we stay in touch with others as well as how we express ourselves. At the apex of all the different types of social media is Facebook, where users create an account and view of a news feed of posts by their friends. The user has the ability to post a status, which can be seen by all of their friends. Friends can respond to these posts by liking or commenting on them, where the number of likes is generally a good indicator of a popular post. Friends like statuses for a variety of reasons, but typically to show positive feedback. As a user, you want friends to like your statuses because it implies that friends are in support of what you share.

2. Project Goal

Our goal is to predict the popularity of a Facebook status based on that user's personal Facebook data. Popularity, measured by the number of "likes" a post gets, is important to users because it eases the anxiety that users feel when sharing on social networking sites. Facebook is one of the core ways people express themselves, but users often struggle to decide whether their thoughts are worth sharing. By accomplishing our goal we ensure that users feel comfortable that their post will be approved.

3. Data

Our dataset includes 49,216 instances of Facebook statuses including six features. The data was gathered using each of one of our group member's Facebook tokens and included data from that individual and all of their friends. The features that we collected include the number of friends, age, and gender of the user who posted the status, the time of the status (month and times of day), the time since the last status (hours), and the "score" of the Facebook status itself. To find the score of a Facebook status, a dictionary was built with individual scores of keywords from statuses in our entire dataset. The individual score was calculated by averaging the number of likes that a status with that word receives. Once the individual scores for each word is calculated and stored in a dictionary, the status is given a score by averaging those values for each word in the status. The feature that we are predicting is number of likes on each status, which is included in the training and validation set. We created three sets: dictionary builder, training, and testing. The dictionary builder set contained 22,108 statuses and was used to build the word score dictionary. The training set included 22,108 instances and was used to train our model and the testing set contained 5,000 instances used to evaluate the model's results.

4. Approach

Starting off, we selected features by listing all of the different attributes that each status had. Beyond the status, we also listed external factors such as the time of day that the status was posted or various characteristics of the user who posted the status. Once these features were gathered for the statuses, we chose a classifier.

4.1 Features

Our classifiers considered the following features about the user who posted each status and the status itself:

- User's number of friends: This number could help indicate the range of values that a user's status could receive, since this would be roughly the maximum number of likes.
- User's age: The user's group of friends may consist mostly of other user's of the same age, and people of this age may have certain interests.
- User's gender: Statuses about certain topics may be viewed differently if posted by one gender.
- Time the status was posted (month and time of day): Posting statuses about time-sensitive events will garner more likes if posted in a timely manner. Additionally, posting statuses when more people are online will also increase visibility.
    - Times of day included:
        - post_midnight: 12AM - 4AM
        - early_morning: 4AM - 8AM
        - morning: 8AM - 11AM
        - noon: 11AM - 2PM
        - afternoon: 2PM - 5PM
        - evening: 5PM - 9PM
        - night: 9PM - 12AM
- Time since the last status (hours): People may be less inclined to like the status of someone who spams.
- Average number of likes: If a person get a high number of likes on every status, their statuses will get like more in general.
- Score of the status based off of the word score dictionary: Certain words and topics can contribute more likes to a status than others. To calculate the score, we remove common words, such as "the", "was", "am", etc, and assign each of the remaining words a score of (the natural log of the number of likes in this status) / (number of remaining words). The score for each word is updated and averaged in the word score dictionary. We take the natural log so that statuses that have the same number of likes within an order of magnitude will be classified similarly.

4.2 Classifiers

The classifiers that we considered included:

- ZeroR: To give us a baseline to compare other classifiers, we used ZeroR to predict the number of likes. ZeroR in this scenario outputs the average number of likes for all of the statuses.
- IBk (k-Nearest Neighbor): We chose to use a k-Nearest Neighbor algorithm to predict the number of likes because we hypothesized that the nearer neighbor instances would contribute more weight than the further ones. In terms of our problem, we hypothesize that statuses with more similar features would result in similar numbers of likes. IBk, instance based k-nearest neighbor, is a k-nearest neighbor implementation in Weka.
- Linear Regression: To contrast the nearest neighbor algorithm, we used Linear Regression to predict the number of likes.

5. Results

After extensively testing various combinations of attributes and analysis by Weka, we found several specific smaller selections of attributes that consistently performed the best. Our initial tests examined the correlation between individual attributes, which can be seen in Figure 1.

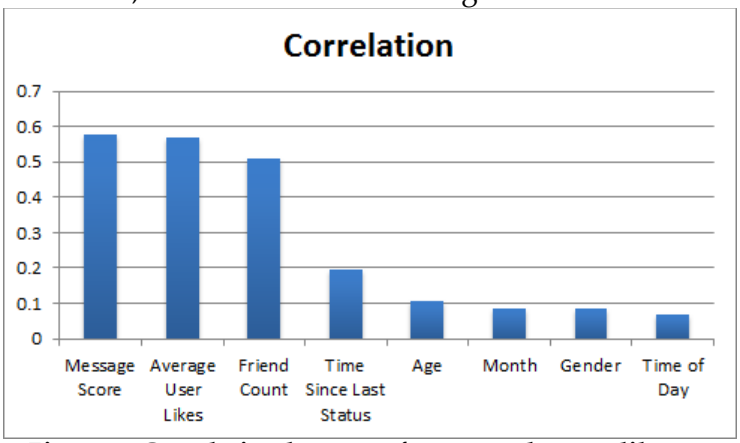

Figure 1. Correlation between feature and status likes

In particular, we found that message score, average user likes, and friend count had the highest correlation (all in between 0.5 and 0.6). These were similarly better in terms of root mean squared error (RMSE) values as well, as seen in Figure 2.

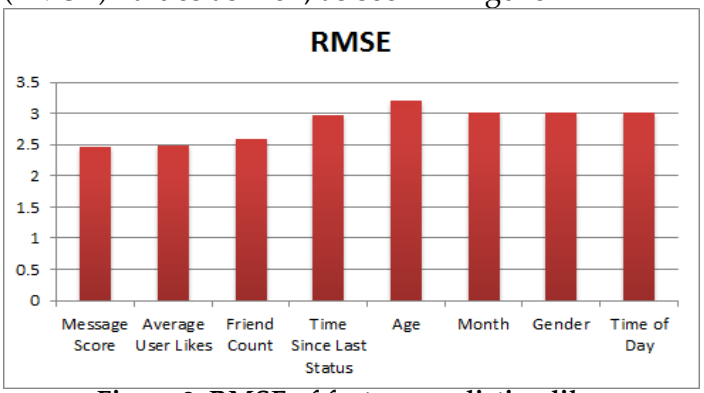

Figure 2. RMSE of feature predicting likes

Interestingly, choosing all three of these attributes and using either the Linear Regression or IBk 3NN algorithms had worse results than choosing only message score and average user likes (correlation 0.6945 to 0.7416). The best combination (found by CfsSubsetEval Exhaustive Search) included gender, message score, month, time of day, and average user likes, which resulted in a 0.7473 correlation for IBk 3NN. As expected from any sort of nearest neighbor algorithm, the correlation and RMSE worsened noticeably when we validated our model with our test set (correlation to 0.4513 from 0.7416, RMSE to 1.0526 from 0.7337). To contrast, linear regression was about the same: correlation to 0.5482 from 0.5318, RMSE to 0.9443 from 0.935. Compared to our baseline of ZeroR, both regression and nearest neighbor performed vastly better (originally 0 correlation and 1.129 RMSE).

6. Discussion

Though our model greatly improved over the baseline, it lacks conclusive results and does not consistently accurately predict the number of likes a status gets. The finding that including number of friends as a factor decreases the performance of our algorithms is also counterintuitive. Further, since we modeled based off of the natural logarithm of the number of likes, our presented RMSE values are deceptively good. Though this isn't meaningful for many statuses, small errors in logarithmic predictions for statuses with potentially large numbers of likes (say 60) will almost always be wildly off. We think that a good portion of this error results from our approach to scoring the actual status's content -- we score words in a fairly simple manner that does not take into account many potential semantic subtleties of statuses or significant events occurring at the time (ex. statuses about a football would be likely to garner more likes in February during the Superbowl, but we do not take this into consideration). Linear regression seems to be the overall more consistent predictor for this problem compared to nearest neighbor -- though nearest neighbor makes intuitive sense to map similar statuses from similar people to similar numbers of likes, linear regressions performance was generally better and not misleading as we went from training to testing, as seen in Figure 3.

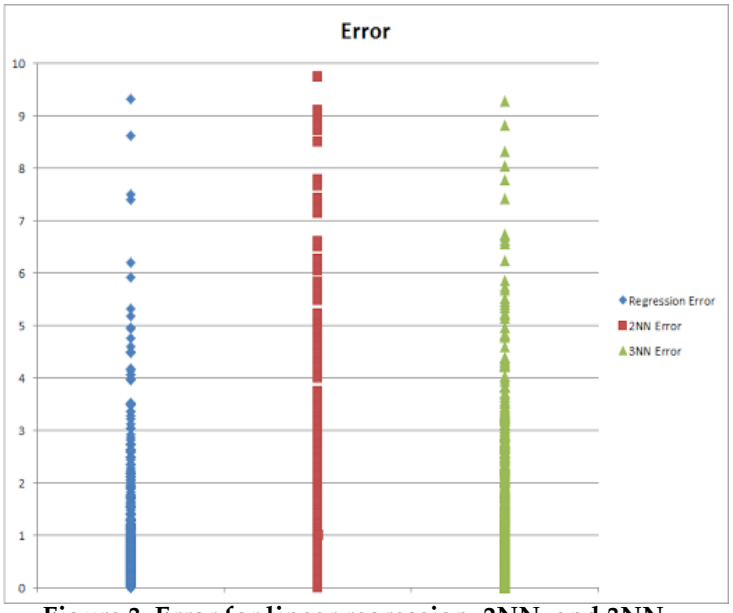

Figure 3. Error for linear regression, 2NN, and 3NN classifiers on the test set

7. Ideas for improvement

To improve our results, we can attempt to predict buckets for the number of likes, instead of trying to predict the exact number of likes. The buckets could be 0 likes, 1-10 likes, 11-100 likes, and 100+ likes. This approach would take advantage of the fact that we give words their score based on the natural log of the status's like and resolve issues we were having with inaccurate predictions for high numbers of likes.

Another method for improving the results of our predictor would be to perform more analysis on the actual status. One such improvement would be to categorize the statuses into buckets, such as technology, politics, business, sports, science, and entertainment. This approach would improve the effect that the actual content of the status has on its score.

8. References

[1] Kiss, J. (2014, February 04). Facebook's 10th birthday: From college dorm to 1.23 billion users. Retrieved March 13, 2014, from http://www.theguardian.com/technology/2014/feb/04/facebook-10-years-mark-zuckerberg