Predicting the Popularity of Food-Related Tweets Using Machine Learning

Giovanni Coronica[1] and Thomas Rossi Mel[2]

[1] problem statement, solution design, solution development, data gathering, writing
[2] problem statement, solution design, solution development, data gathering, writing

January 15, 2023

1 Introduction

In today's digital age, social media platforms like Twitter, Snapchat and Instagram have become an important source of information and communication for millions of people around the world. One popular topic on Twitter is food, with users daily sharing reviews of their meals, as well as information about recipes and restaurants. With the vast amount of food-related content being shared on Twitter, it's important to be able to predict which tweets are likely to be popular.
In this work, we propose a machine learning-based approach that uses a custom popularity index to quantitatively predict the virality of text-only tweets. Previous research [1] has addressed the problem of popularity prediction of tweets by treating it as a classification problem. In contrast, this study focuses specifically on food-related tweets and introduces a regression model, focusing on the prediction of tweet's popularity using only tweet's texts, without requiring any information about the user who posted it.

2 Data collection

The dataset for this research was collected using the official Twitter search API [4]. Over $25,000$ tweets posted during 2022 were selected based on the following criteria:

- Containing the hashtag "food" in the text. Additional food-related hashtags could also be included.
- Being posted at least 1 month before the data collection date. This was done to avoid too recent tweets that may still have large variations in likes over time.
- Not containing any images, videos, or were link-only content.
- Being written in English (for simplicity).

- Not being retweets, as the aim was to predict the virality of tweets based only on their text, and retweets share the same information.

For each tweet, the following features were obtained: the number of likes, retweets, replies, and the full text. Additionally, a few more important features were added to create a popularity index used to label the dataset (see subsection 3): the mean and standard deviation of the user's posts for likes, retweets, and replies.
Specifically, for a tweet and its associated user, the mean and standard deviation were calculated using all of the user's published posts within a 15-day range before and after the retrieved tweet. This was done to ensure that the statistics were based on tweets from the user that were not too far in time from the retrieved tweet, and were therefore not largely influenced by changes in the number of the user's followers.
Tweets from users with less than 30 published tweets that met the previously mentioned criteria were not included in the dataset, as they did not provide an accurate estimate for the mean and standard deviation.

3 Popularity index

As previously introduced, the aim of this project is to predict the popularity of text-only tweets, regardless of the number of followers of the user who posted them. The main idea is to label each tweet by comparing it to the other tweets posted by the same user. Given a real number x representing the number of likes a tweet has received, a potential measure of popularity could be $P(X \leq x)$, where X is a random variable representing the number of likes of a tweet by the author in question.
A first estimate of $P(X \leq x)$ can be obtained by counting the number of posts by the user with less than x likes, and dividing this count by the total number of posts by the user. However, after analyzing multiple distributions of likes for posts of different users (e.g. in Figure 1), a more robust estimator can be defined by assuming that X follows a gamma distribution with parameters α (or shape parameter) and β (or rate parameter). That is, $X \sim \Gamma(\alpha, \beta)$, where α and β can be calculated by solving the system in two equations from the well-known formulas for calculating mean and variance of the likes, retweets, and replies of user posts. Thus, the like-based popularity index is defined as

$$index(x, \alpha, \beta) := \int_0^x f(u; \alpha, \beta) du = \int_0^x \frac{u^{\alpha-1} e^{-\beta u} \beta^\alpha}{\Gamma(\alpha)} du = \int_0^x \frac{\gamma(\alpha, \beta x)}{\Gamma(\alpha)} du \quad (1)$$

where $f(u; \alpha, \beta)$ is the density function of the gamma distribution. Analogously, the distribution of retweets and replies is similar to that of likes, so the same concept can be applied. The final Twitter popularity index is calculated as a weighted average of the three scores, with likes receiving the most weight and replies receiving the least.

$$popularity\_index := \frac{1}{2} likes\_index + \frac{1}{3} retweets\_index + \frac{1}{6} replies\_index \quad (2)$$

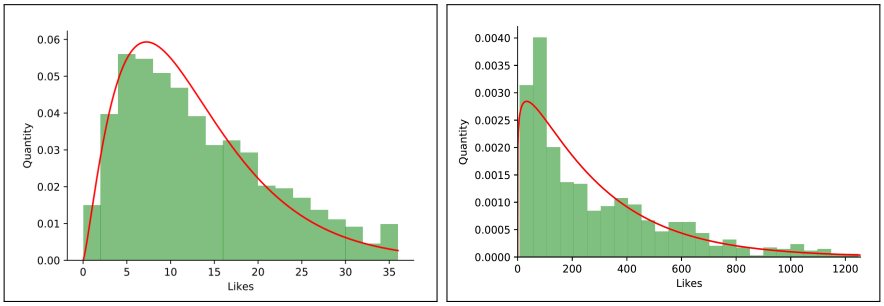

Figure 1: The distribution of likes on posts for two Twitter users, having 1,000 and 30,000 followers respectively

4 Data preprocessing

Common data preprocessing were applied to the text, specifically:

- Removing links (as they are not relevant for the regression problem)
- Removing stop words (common words in a language providing low relevance), special characters (excluding "#" and numbers, common in hashtags) and emojis. This approach yielded similar results while utilizing less memory.
- Converting all text to lowercase, as this had minimal impact on the results.

Additionally, the technique of word stemming was utilized, which involves replacing words with their root form (e.g., "flying" becomes "fly"). Then, multiple datasets were created for different experiments, each containing the k most frequent n-grams present in all the tweets, with $k \in \{1000, 2000, 4000\}$ and $n \in \{1, 2, 3\}$. The n-gram technique was used to give importance to the sequence of words rather than the individual words alone. Each tweet was then represented by k features, one for each n-gram, and the corresponding value was chosen to be the TF-IDF score. The TF-IDF measure, introduced by Spärck Jones (1972) [2], increases in proportion to the number of times a word appears in the document, but is offset by the frequency of the word in the corpus, which helps to account for the fact that some words are more common than others. To calculate the TF-IDF scores, each tweet was tokenized and the formula was applied.

5 Model and metrics

The popularity index provides values within the range $[0, 1]$, indicating that the problem at hand should be treated as a regression problem. A random forest model was thus chosen as an appropriate machine learning technique for this task. The model consisted of 500 trees, a commonly used parameter value, and was trained using mean squared error as the criterion. Various hyperparameters were tested, and the best results were obtained with 1 as the number of elements in a leaf node of a single tree, and using a third of the dataset's features for the training of each tree.
Two metrics were utilized to evaluate the model: mean absolute error (MAE) and a "custom" area under the curve (AUC). MAE is a commonly used metric for regression problems, which calculates the average absolute difference between the predicted values

and the actual values. In this specific problem, the data is heavily skewed towards the value of 0, thus MAE can be compared to the value of 0.268, which was obtained using a dummy classifier that predicts the mean output label. AUC, on the other hand, is typically employed for classification tasks. In this case, AUC refers to the probability that the model will rank a randomly chosen tweet as more viral than another randomly chosen tweet with a lower popularity index, according to the second definition. In a 2009 paper, Rendle et al. [3] mathematically demonstrate the analogies between AUC and this approach.
The model was then evaluated using 10-fold cross validation, which is a commonly used method for small datasets and tends to yield more reliable results. The results of this evaluation are summarized in Table 1, where n and k are, respectively, the dimension of n-grams and the number of features.

n	k (x1000)	MAE	AUC
1	1	0.2202	0.6379
	2	0.2191	0.6400
	4	**0.2184**	**0.6444**
2	1	0.2203	0.6313
	2	0.2199	0.6387
	4	0.2190	0.6431
3	1	0.2212	0.6235
	2	0.2195	0.6362
	4	0.2193	0.6413

Table 1: Results of the model for MAE and AUC

6 Conclusions

In conclusion, this study proposes a machine learning-based approach that uses a custom popularity index to quantitatively predict the virality of text-only tweets related to food. A random forest regression model was developed to achieve this prediction, and data preprocessing techniques were utilized to eliminate any dependence on the user's profile. The proposed approach was evaluated using mean absolute error and a "custom" area under the curve metric, and the results showed that the model was able to accurately predict the virality of tweets with a high degree of accuracy.
However, it was also found that the performance of the model deteriorated as the value of n in the n-gram increased, possibly due to the fact that higher-order n-grams capture more context and meaning but may also introduce more noise and complexity into the model, requiring more data to accurately predict tweet popularity. On the other hand, the performance of the model improved as the number of features (k) used in the model increased. This is likely because datasets having higher values of k contain the same features as lower values of k, in addition to additional features.
Overall, the custom popularity index introduced in this study allows for the prediction of tweet popularity based solely on the tweet's text, making it a useful tool for understanding the factors that contribute to the success of food-related content on social media platforms like Twitter.

References

[1] Arpan Kumar Kar P. Vigneswara Ilavarasan Nimish Joseph, Amir Sultan. *Machine Learning Approach to Analyze and Predict the Popularity of Tweets with Images.* Springer, Cham, 2018.

[2] K. Spärck Jones. *A Statistical Interpretation of Term Specificity and Its Application in Retrieval.* journal of Documentation, 1972.

[3] Zeno Gantner Steffen Rendle, Christoph Freudenthaler and Lars Schmidt-Thieme. *BPR: Bayesian Personalized Ranking from Implicit Feedback.* UAI, 2009.

[4] Twitter API v2. *https://developer.twitter.com/en/docs/twitter-api.*