Classifying Twitter Food Trends

Amanda Bertschinger
ambertsc@uvm.edu

Alaina Carstensen
ancarste@uvm.edu

Jek Kuhn
jkuhn@uvm.edu

ABSTRACT

Foods used to be items that were marketed minimally, because everybody needs to eat. Nowadays, with so many foods and diets, it is one product that everybody needs so the market is very competitive. Supermarkets need to be able to know what food people want and for how long they will want it for. This is a very well researched topic by companies, but not much is available publically. We used machine learning to analyze food mentions on social media to determine if foods were "trending" and "popular".

KEYWORDS

Twitter, Food, Trend, Popular Foods, Trending Foods, Neural Networks, Twitter Trend, Machine Learning, Data Science

1. INTRODUCTION

Food is one the three most basic needs and is essential to survival. Without proper nutrients every biological being would die. Many people do not get this basic need met, while others have an abundance of food and let it go to waste. With the recent COVID-19 pandemic, many people in "First World" countries are experiencing food shortages for the first time as people stockpile food preparing for shutdowns. With many shelves in supermarkets empty and more time spent at their homes, people have begun to cook and bake more often. Since many people are baking and cooking more, people consult the internet to see what food to make. This means that some foods are being made for a couple of months while others are being made routinely.

Along with the pandemic, there has been a growing trend to eat more of a plant based diet in the United States of America. Both the varying health benefits and the positive impact on climate change, contribute to this shift.

In order to reduce waste and enable shelves to be properly stocked. Popular foods will be in higher demand and will commend more shelf space especially when trending. We will classify whether a food will be "popular" or "unpopular" and "trending" or "not trending".

2. PROBLEM DEFINITION AND ALGORITHM

A. Task Definition

We analyzed past food trends and tried to predict what food trends are in the process of happening or have happened. We used the frequency with which foods are mentioned as well as the rank they achieve on twitter in order to determine what is "trending" and what is "popular".
A certain food can be looked up to find its current and past popularity and whether or not it is trending. Knowing what is trending and popular gave us insight into what food people were interested in consuming during a certain time period, up to and including the present.

Since we could not find any work that is doing what we are doing it is inherently important. This could help supermarkets have the correct supply of food, see how quickly people are switching to plant based diets that help the environment. Other applications include food companies and restaurants being able to also figure out what to post on their social media or how to modify their menus. Food is something that everybody needs on a daily basis so there will always be a need for this.

B. Algorithm Definition

For our machine learning algorithm, we made an ensemble of several different methods. We chose to make an ensemble as we felt that it would help give our final algorithm the benefits from each method while mitigating some of the issues that the individual methods suffer. We used logistic regression, support vector machine, decision tree, random forest, and multilayer perceptron neural network methods. We chose to use a decision tree and a random forest as they are some of the most accurate learning algorithms available, especially for more complex non-binary classification problems like ours. However, both methods have a tendency to overfit (as can be seen in Figure 1), so we added in the logistic regression method to help allay this overfitting.

Figure 1: Fully grown decision tree for our algorithm

Neural networks were chosen as they are a good method to use with large and somewhat complex data sets like ours. With an even number of methods, a tie was possible in the ensemble, so we added in SVM as a tiebreaker, as it provides a good intermediary between low accuracy and overfitting.

The logistic regression method was the scikit-learn logistic regression function, which uses L2 penalty. The SVM method used was the scikit-learn support vector classification function, with a radial bias function as the kernel. We chose a radial bias function as it gave the best results, and made the most sense given our data. The decision tree method was the scikit-learn decision tree classifier function. We chose a random splitter for our DTC and used the Gini impurity, as this helped reduce overfitting. The random forest method used was the scikit-learn random forest classifier function. We used 100 estimators and Gini impurity, as 100 estimators helped to make it accurate without overfitting too heavily. For the multilayer perceptron neural network, we used the MLP classifier function from scikit-learn. We used 8 hidden layers, early stopping, and an activation function of ReLU. To choose this neural network, we experimented with building our own using the sequential function from TensorFlow and various parameters with the sklearn MLP classifier. This particular set of parameters was chosen as we felt that it gave the best accuracy without overfitting too heavily.

After getting the predicted results for all of these methods, we put each method's predicted result into a table indexed by the data point that gave that prediction. We then found the mode of the predictions for each data point, and assigned that mode as the final predicted result.

C. Data

The data, from 2009 to present, is from StoryWrangler, which pulls data from twitter as two separate JSON files (pictured in Figures 2 and 3). One of the JSON files has the rank data, and the other JSON file the frequency data. On Twitter, "rank" means the position the word achieves based on the number of times the word appears. The "frequency" is the number of times that word appears in a day over the total number of tweets in that day (and standardized by the Computational Story Lab to account for the steady rise in total number of tweets in a day over the last 11 years). We then combined these two JSON files so that the data has the following features: date/time, rank, frequency for each food we pulled. We took 6 different food mentions: "pizza", "eggs", "ice cream"," horseradish", "sardines", and "brussel sprouts". We have 4142, 4142, 4142, 4140, 4141, and 4126 data points for each of the foods respectively. This culminates in 24,833 total data points from the last 11 years.

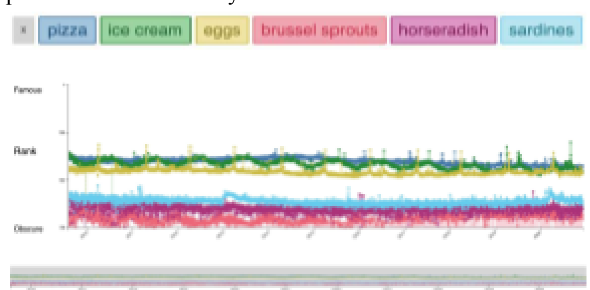

Figure 2: Rank data on food related words on the Storywrangler website

Figure 3: Frequency data on food related words from the Storywrangler website

3. METHODOLOGY

Before we could start using our data to fit the pieces of our ensemble method, we needed to clean and label the data. To clean the data, the first part was making sure that all data points had values for both frequency and rank. As there are days when a certain word will not be tweeted about at all, there were some data points in the JSON file that had NaN as a value. As we would not be able to work with this, we had to fill this. For the frequency data, it was simple to replace any NaN with a 0, indicating that it did not appear that day. However, for rank it was a bit more complicated. In rank, a lower number is better (a rank of 1st is better than a rank of 500th). Therefore, we were not able to simply replace a NaN with 0. The equivalent would be to replace a NaN with infinity (an infinitely low ranking, meaning it did not appear). However, this was not possible as infinity cannot be used in computations any more than NaN can be. Therefore, we looked through all rank data to find the lowest value. This minimum turned out to be around 900,000, so we set any NaN value to lower than that value (at 1,000,000), indicating that it did not appear that day. After we ensured that all data was filled with values that can be used in calculations, we separately normalized the rank and frequency so all values lay between 0 and 1.

After the data cleaning, we had to label the data. It needed to be labeled due to us using supervised learning methods, but we were able to partially automate this by setting up rules for the labeling. There were 4 separate labels:

- 0: Not popular and not trending
- 1: Not popular and trending
- 2: Popular and not trending
- 3: Popular and trending

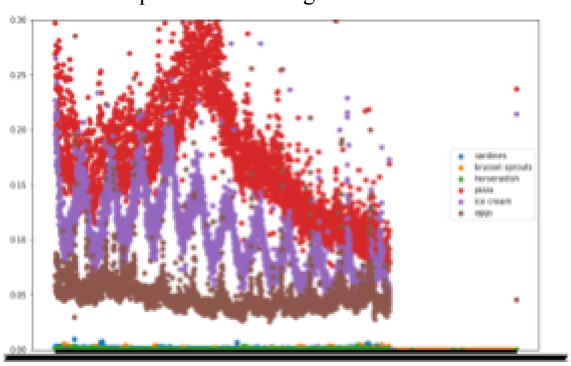

Figure 4: Normalized frequency data, cut off at 0.3 to show the clear gap between "popular" and "not popular"

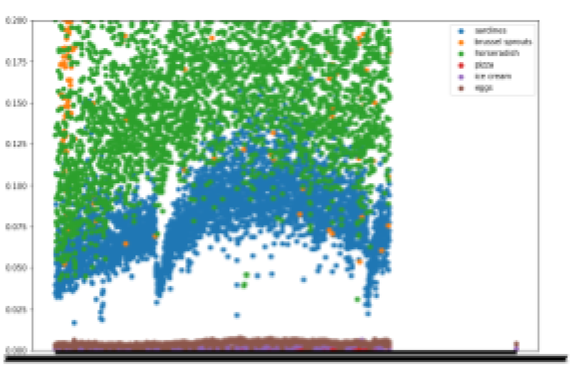

Figure 4: Normalized rank data, cut off at 0.2 to show the clear split between "popular" and "not popular"

In order to label data as popular or not popular, a threshold value was found for both frequency and rank. This threshold value was obtained by observing the data. In the data there was a clear separation between the foods with better average frequency/rank and those with worse average frequency/rank, as can be seen in Figures 4 and 5. If the data point was above this threshold value for frequency and below it for rank, it was labeled as popular. If it was below the threshold for frequency and above it for rank, then it was labeled as not trending. In the event that the rank and frequency gave different results (above/above or below/below respective to frequency/rank), then the absolute value of the difference between the rank/frequency and the threshold was compared, and the larger value was used to label the data (if above, using the above logic).

In order to label data as trending or not trending, the rank and frequency value of each data point was compared to running averages of the seven days previous to the data point. If the frequency of the point was above the average and the rank below its average, then the point was trending. If below the 7 day average frequency and above 7 day average rank, the point was not trending. Similar to the logic used for "popular", if the frequency/rank was below/below or above/above, then the one with the greatest absolute difference took preference for labeling.
Once these "popular/not popular" and "trending/not trending" values were obtained, they were used to label the data 0-3 according to the key above. We also used the calculations used for "trending" to add a change in rank (Δrank) and change in frequency (Δfrequency) to each data point as two additional features in order to help with the classification by the algorithm.

Once the data was cleaned and labeled- and all features were added- the 5 machine learning methods that made up the ensemble were able to be fitted. For each method, we tested a variety of hyper parameters in order to determine the best one for our ensemble. This testing was mostly experimentation and trial/error with various hyperparameters (e.g. activation function, number of hidden layers, and early stopping were explored for the neural network). The main statistic used to evaluate the methods was their accuracy, though the confusion matrix for each method was also studied. For each method, the accuracy and confusion matrix were used to determine the set of hyperparameters that gave the highest accuracy while not overfitting the data too severely in order to maximize accuracy while attempting to minimize variance as much as possible. The hyperparameters that were chosen for each method are discussed in the algorithm definition section (2B).

4. RESULTS

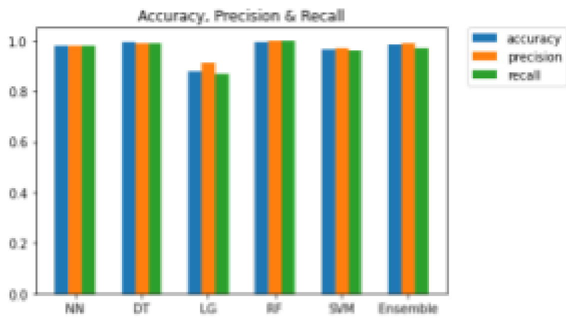

Figure 6: The individual method statistics for the machine learning algorithm trained on twitter food data

As seen in Figure 6, the SVM model has an accuracy of 0.97, and an average precision of 0.97 and average recall of 0.96. The decision tree classifier has an accuracy of 0.99 with an average recall and precision of 0.99. The logistic regression model has an accuracy of 0.88, as well as an average precision of 0.91 and average recall of 0.87. The random forest classifier has an accuracy of 0.996, as well as an average precision and recall of 1. Finally, the MLP has an accuracy, average precision, and average recall of 0.98. The comparatively lower accuracy, precision, and recall of the logistic regression helps to compensate for the overfitting of the random forest and decision tree classifiers; as was the intent.

	0	1	2	3
0	1366	50	0	0
1	10	1706	0	0
2	0	0	1681	10
3	0	0	45	1341

Figure 7: Ensemble method confusion matrix

	precision	recall	f1-score	support
0	0.99	0.96	0.98	1416
1	0.97	0.99	0.98	1716
2	0.97	0.99	0.98	1691
3	0.99	0.97	0.98	1386
accuracy			0.98	6209
macro avg	0.98	0.98	0.98	6209
weighted avg	0.98	0.98	0.98	6209

Figure 8: Ensemble method classification report

Once the ensemble method was put together, the confusion matrix from Figure 7 and classification report from Figure 8 were obtained. As can be seen in the confusion matrix, there were 60 false negatives and 55 false positives predicted. However, out of the 105 falsely predicted values out of our 6,209 testing points is very low, only 1.69% of the testing data being incorrectly classified. This gave rise to the high accuracy of 0.985 that the final ensemble method achieved. While a high accuracy score is desirable, between the very high accuracy and the high precision and recall values, it is clear that our method suffers from high variance. This level overfitting is lower than that seen in the random forest classifier and decision tree classifier individually, but does still exist.

5. DISCUSSION

It is clear from the confusion matrix that the largest difficulty our algorithm has is in determining the difference between judging trending and not trending. It made no errors with classifying popularity, all the false positives and false negatives were in classifying the trending/not trending. This is likely because of how a data point was classified as trending or not trending, versus the information given to the algorithm. In order to classify this, we looked at the data point in comparison to a 7 day average. However, while the algorithm was given the difference between a point and its average, it was not given the average itself as a threshold to judge the difference by. This likely accounts for the false negatives and false positives seen between the classifications of 0 and 1 as well as those between 2 and 3.

Our largest and smallest precision values were .99 and .97 respectively. Our largest and smallest recall values were .99 and .96. Since all recall and precision values were that high, we know that overfitting occurred. DT and RF are machine learning methods that are notorious for overfitting, that coupled with how we only took 6 different foods, leads us to believe that the machine learning models were constrained by the foods we chose.

6. RELATED WORKS

From our research into academic and research papers on food, most articles are about the health effects of different foods and diets. Not much is done in the academic world for analyzing food trends. However, there have been a lot of private companies who make money by selling their predictions to vendors [1], if their work was publicly available then there would be no need for people to pay and they would not make any money.

The academic work that is most closely related to ours, is a paper about the book “The Tastemakers: Why We’re Crazy for Cupcakes but Fed Up with Fondue” [2][3]. The paper states the book is broken up into three types of sections: “types of trends, the people behind them and the impact of these trends". We looked at specific foods instead of the general trends and classify those foods as "trending" and " popular". Another work takes a holistic approach of how foods came to be [4], while another looks at how people who are obsessed with eating healthy food drive supply and demand in supermarkets [5]. A different paper analyzes already known food trends using image recognition for different countries [6]. We looked at existing Twitter data in order to train an ML algorithm on whether or not something is (or ever was) "trending" or "popular". Once this is classified we can see how long the "trend" lasted.

7. CODE AND DATASET

For the code and dataset, please see the attached zip folder. The zip folder contains two .JSON files with the dataset and the Jupyter Notebook with the code. In order to run the code, you will need to have the two .JSON files in the same directory.

These JSON files and the code can also be obtained at the following git repository: https://github.com/ancarste/TwitterFoodTrends

Our dataset came from the StoryWrangler website: https://storywrangling.org/ courtesy of the University of Vermont Computational Story Lab.

8. CONCLUSION

This model has near perfect classification based off of how we labeled it. Looking at our results, we know that popular foods must be kept well stocked all the time as they are constantly being consumed, whereas trending foods only need high stock for a limited time period. Popularity level is able to give an indication of baseline demand level and, therefore, an idea of necessary inventory. Trending level tells people there will be increased demand and, in turn, the supply should temporarily be increased. Fads, or foods that trend briefly, are a particular place of caution. For the time that they are trending, the demand for the food will be higher, but that demand will not last for long. Thus, it is crucial to only increase supply for a short duration of time. The current model vastly overfits the data and would struggle to adapt to a new data set.

9. FUTURE WORK

Our research focused classifying if something was trending and popular and didn't look much at predicting whether a food would trend. The first way to extend out research is to look at the trends for certain foods throughout the year and see if they are cyclic. "Eggs" are always trending around Easter and "ice cream" is more searched during the host summer months are two examples of these cyclic patterns. After that getting a lot more data of different foods would help our model since we only had six different foods to begin with. Once we got these foods we classify those into different groups, they could be vegetables, dairy, meat, etc., and see if these groups have any patterns since there has been a notable shift to plant based diets in America lately. The penultimate step would be to determine two new categories of "lasting trend" and "temporarily trend".

In the future, we would also be able to extend our research by having the program automatically bring in the live Twitter data for food based keywords. Currently the program requires downloading a JSON file from StoryWrangler and importing it to our algorithm's code by hand. Using access to the StoryWrangler API and the Twitter Dechahose (courtesy of the Computational Story Lab and Vermont Advanced Computing Core), we would be able to feed the current day's data into our algorithm and have it automatically predict what category food-related words are in real-time.

10. ACKNOWLEDGMENTS

Acknowledgements to the University of Vermont Computational Story Lab and the Vermont Complex Systems Center for allowing us access to the Twitter Decahose and for the data from the StoryWrangler website. Our calculations were not done on the Vermont Advanced Computing Core, but calculations for the StoryWrangler website and API are, so our thanks to the VACC as well.

11. REFERENCES

[1] Tastewise. (2017). *Food Trends, Analytics And Statistics*. Tastewise. https://www.tastewise.io/foodtrends

[2] Laurent, Christopher. (2014). "The Tastemakers: Why We're Crazy for Cupcakes but Fed Up with Fondue." *Cuizine*, Vol. 9, no. 2, 2019, doi:10.7202/1055223ar.

[3] Sax, David. (2014). *The Tastemakers: Why We're Crazy for Cupcakes but Fed up with Fondue*. Signal.

[4] Festi, Giacomo. (2015). "Semiotics And Qualitative Research: The Case Of Food Trends." *New Semiotics. Between Tradition and Innovation* doi:10.24308/iass-2014-064.

[5] Duarte, Paulo, et al. "Orthorexia Nervosa: Can Healthy Eating Food Trends Impact Food Companies Marketing Strategies?" *Journal of Food Products Marketing*, vol. 25, no. 7, 2019, pp. 754–770., doi:10.1080/10454446.2019.1671931.

[6] Amato et al. (2017) "Social Media Image Recognition for Food Trend Analysis". Association for Computing Machinery, pp. 1333–1336, doi:10.1145/3077136.308414