TechPoint AI Model Overview

Predicting Foodx Orders

By William Chen

Date: 10/16/2023

New Jersey Institute of Technology

Table of Contents

Table of Contents

[Model Considerations 3](#_Toc148392036)

[Model Choice: Random Forest 3](#_Toc148392037)

[Confusion Matrix 4](#_Toc148392038)

[Training and Testing: 4](#_Toc148392039)

[Preparing the Data: 4](#_Toc148392040)

[Ai Model Results 5](#_Toc148392041)

[Additional Considerations 5](#_Toc148392042)

[Suitable Course of Action 6](#_Toc148392043)

# Model Considerations

Picking the Right Tool (Model Selection):

* **Look at the Data**: We have mostly category-type info, like different food orders.
* **Best Tools for the Job**:
  + **Decision Trees**: Like a flowchart – asks questions about data and makes decisions.
  + **Random Forest**: Many decision trees combined – better decisions by working together.
  + **Gradient Boosting**: Corrects its own mistakes. Strong performer but needs more computer power.
  + **Logistic Regression**: Simple tool, checks how factors, like the university someone studies at, might influence their order.
* **Others We Didn’t Pick**:
  + **SVM & Neural Networks**: Great for complex problems but might be too much for this.
  + **KNN**: Like asking neighbors for advice. Not great for big data.
* **How We'll Judge Success**:
  + We’ll see how often the model’s guesses are right, and also how well it spots true positives (like correctly guessing a specific food order).

# Model Choice: Random Forest

1. **Nature of the Data**: Our dataset predominantly consists of categorical-type information, like different food orders. Random Forests are especially adept at handling categorical data because they split on feature values at each node of the decision trees.
2. **Versatility**: Random Forests can automatically handle missing values, outliers, and other data irregularities, which reduces the pre-processing steps we need to undertake.
3. **Avoiding Overfitting**: While a single Decision Tree can be prone to overfitting (i.e., it might perform very well on training data but poorly on unseen data), Random Forests counteract this by averaging the results of many trees.
4. **Feature Importance**: Random Forests provide an inherent capability to rank features by their importance. This can be crucial if we want to understand which features play a significant role in determining the food order.
5. **Stability & Robustness**: With multiple trees, Random Forests are less sensitive to noise in the dataset and changes in the input data, making them a more stable and robust classifier.

# Confusion Matrix

1. **Detailed Performance**: Instead of just providing an accuracy rate, a confusion matrix gives a detailed view of how the model performs for each class.
2. **Identifying Specific Errors**: By examining the off-diagonal elements, we can see which specific classes the model confuses most often, allowing for targeted improvements.
3. **True Positives and Other Metrics**: The matrix lets us extract essential metrics like Precision, Recall, and F1-score. These metrics are crucial when classes are imbalanced, or when the cost of misclassifying one class is much higher than another.

# Training and Testing:

* **Splitting the Data**: We'll set aside some data to train our model and some to test it.
* **Training**: The model learns from known data.
* **Testing**: We see how good the model is with unknown data.
* **Tweaking the Model**: We’ll adjust some settings to get better results.

# Preparing the Data:

* **Cleaning Up**: We remove any errors or gaps in the data.
* **Adding Insight (Feature Engineering)**: Think of patterns. Do students from a culinary college order different foods?
* **Changing Data Types**: The model needs numbers. So, we’ll turn categories (like university names) into numbers.
* **Balancing Data Sizes (Scaling)**: Makes sure no piece of info has too much say in our model's decisions.
* **Handling Sparse Data**: Some universities, like Purdue, have few orders. We might group such universities as "Others" since one or two orders don’t really tell us the full story about all Purdue students.

# Ai Model Results

Based on the data I've analyzed, I've come to several observations regarding the classification of various food items:

* My model seems to perform exceptionally well for "Indiana Pork Chili" as indicated by its high precision and recall. This suggests that when my model predicts this class, it's usually correct, and it also captures a significant majority of the actual instances of this class.
* On the other hand, the "Breaded Pork Tenderloin Sandwich" appears to be a challenging class for my model. The low recall value indicates that my model often misses instances of this class. This is an area I'll need to look into further, perhaps by examining the features associated with this class or considering additional data.
* The confusion matrix reveals a few misclassifications that I should be concerned about. Particularly, many items that are truly in class 0 have been predicted as class 4. This kind of misclassification can be problematic, especially if there's a significant difference between these two classes in a practical setting.

# Additional Considerations

1. **Feature Examination**: I should revisit the features used in the classification. Are there any features that can be added, removed, or engineered to better represent the classes, especially the ones where the model seems to struggle?
2. **Algorithm Exploration**: While the current algorithm has provided decent results, trying out different algorithms or tweaking the parameters might offer better performance.
3. **Dataset Balance**: It might be beneficial to check the distribution of classes in my dataset. If some classes are underrepresented, it could lead to the model not learning them well. Techniques like oversampling, undersampling, or using synthetic data might be considered.
4. **Practical Implications**: I need to consider the real-world implications of misclassifications. If, for instance, misclassifying a specific food item could lead to significant issues or complaints, I should prioritize refining the model for that class, even if it means compromising slightly on others.

# Suitable Course of Action

Top of Form

Bringing our RandomForest model to maturity for FoodX's order prediction demands a thoughtful appraisal. One of my foremost concerns is the model's performance. While RandomForest is renowned for its robustness and ability to handle a variety of data types, ensuring its accuracy consistently across diverse user inputs is vital. From our initial tests, the model has shown promise. However, real-world deployment could bring unforeseen challenges. I've noticed that while the model excels in certain predictions, there are instances where it might not be as precise. Regular model retraining with fresh, diverse data could address this, but it's a continuous commitment.

Data biases also weigh on my mind. The current dataset, though comprehensive, is primarily student-driven, which means our model may have inclinations towards certain food preferences common among students. If FoodX is considering a broader demographic in the future, we might need to re-evaluate and retrain our model. The success of our current promotion heavily leans on students' college experience data, but scaling this up means diversifying our data sources and ensuring the model remains unbiased.

Lastly, the ethical, business, and technical implications can't be ignored. From an ethical perspective, the data collection process needs transparency. Users should be informed about how their data aids predictions and assured of its confidentiality. On the business front, while the 10% discount on incorrect predictions is a catchy promotion, the financial implications of frequent mispredictions, especially with growing customer volume, could be substantial. Technically, while the **.predict()** method streamlines order predictions, integrating this into the FoodX app requires seamless backend development to ensure real-time, accurate predictions. Considering all these aspects, while our RandomForest model shows potential, its full-scale deployment requires a mix of continuous model improvement, ethical data practices, and efficient technical integration.

Top of Form

Bottom of Form

Bottom of Form