Optimizing Cookie Recipes for Ratings Using Machine Learning and Deep Vector-to-Sequence Recurrent Neural Models

Mackenzie O'Brien and Dr. Pablo Rivas

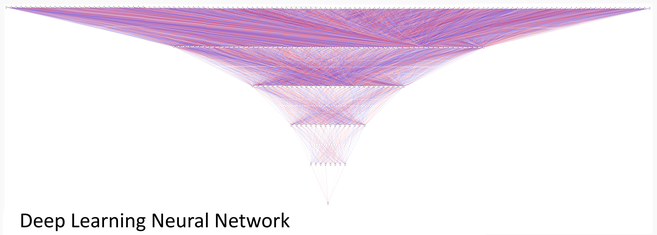
Our Idea

This project employed **machine learning** concepts and processes in an attempt to teach a computer an algorithm that outputs a **cookie recipe** that is comparable to human-made recipes.

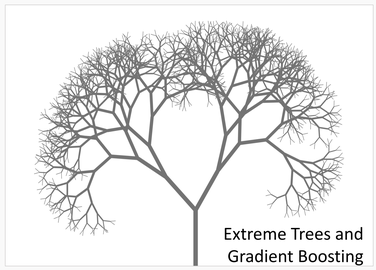
The process involved showing the algorithms a data set of cookie recipes that included details such as ingredients, ratings, and calories. A **deep LSTM** algorithm was used to construct instructions for a recipe once ingredients were selected based off of instructions from the dataset of recipes.

The recipes were gathered by scraping the website allrecipes.com for chocolate chip cookie recipes that were compiled in order to "teach" the computer **to recognize and create** tasty cookies by looking at tactors such as reviews, ingredients, and calories.

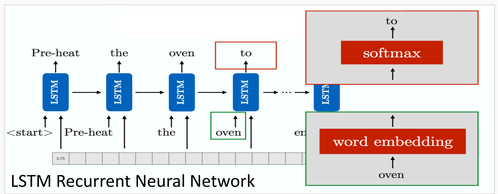
Machine Learning Architectures



Deep Learning Neural Network



Extreme Trees and Gradient Boosting



LSTM Recurrent Neural Network

References

Clifford, Catherine. "Alphabet Billionaire Eric Schmidt: Google Used A.I. to Find the Perfect Chocolate Chip Cookie Recipe." CNBC, CNBC, 29 Jan. 2018.
Kochanski, Greg et al. "Bayesian Optimization for a Better Dessert." (2017).
Naik, JitendraB.. "Cuisine Classification and Recipe Generation." (2015).

Methodology

Gathering Recipes for Data Input

250 chocolate chip cookie recipes were taken from allrecipes.com. This data was then cleaned and used to create ingredients and instructions.

Creating Training Models For Ingredients Vectors

Deep Learning Neural Network (DNN)

This a nonlinear artificial neural network with multiple layers between the input and output layers. Each mathematical manipulation of the data is considered a layer, and complex DNN have many layers.

Extremely Randomized Trees

The algorithm is a variant of the Random Forest regression that picks a decision boundary at random by using the entire sample at each step of the extreme tree.

Gradient Boosting

The gradient boosting algorithm applies the concept of modifying weak learners to become better at identifying good outcomes more efficiently.

ML Error Rating Prediction via Mean Square Error

$$\frac{1}{n}\sum_{i=1}^{n}(y_i - y_i^*)^2$$

Vector-to-Sequence Models to Get Instructions

Long Short-Term Memory Recurrent Neural Network

We used each generated recipe as an attention mechanism, which when combined with LSTM, trains the model to output a sequence of words that it believes are good recipe.

Minimizing Cross Entropy Loss

$$\frac{1}{N}\sum_{n=1}^{N}\sum_{c\in C} d_{cn}^* \ln d_{cn} + (1 - d_{cn}^*)\ln(1 - d_{cn})$$

Testing the Cookies

After evaluating the models, we chose 5 recipes to test. We did taste tests of each machine learning cookie against the same control cookie, serving 40 of each cookie on campus per test. The typical rate of response was around 30 for each cookie.

Quality Metrics of Cookie Recipes

Simplicity

$||y - y^*|| + ||\mathbf{w}||$

Uniqueness

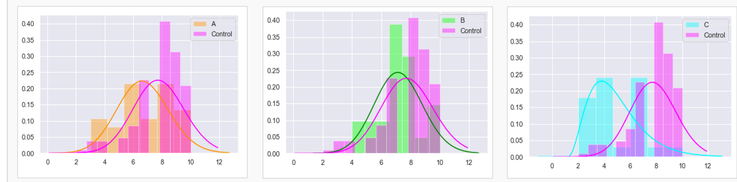
$$\min_{i=1,\ldots,N} ||\boldsymbol{x_i} - x^*||$$

Survey Questions

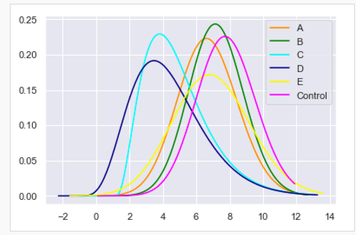
1. Appearance on a scale of 1 (unfit for consumption) to 5 (Excellent)
2. Aroma on a scale of 1 (unfit for consumption) to 5 (Excellent)
3. Taste on a scale of 1 (unfit for consumption) to 5 (Excellent)
4. Texture: crunchy, chewy, gooey, juicy, soggy, creamy, other
5. Overall Satisfaction on a scale of 1 (Hated It) to 10 (Loved It)

Graphs

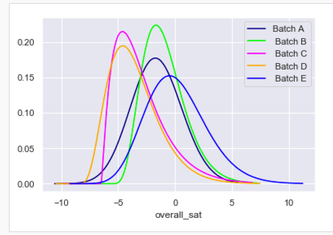
Individual Batches vs Control



All Batches vs Control



Δ: Machine Learning – Control



Cookies

A B C D E Control

Results

Ingredients

Batch B had the best results, but none of the recipes beat the control cookie. Overall, the recipes generated had good responses during experiment tastings. **Recipe E had to be modified**, as it included no wet ingredients, and thus would not correctly form dough. The **best algorithm** out of those used was **Extremely Randomized Trees.**

Vector-to-Sequence Instructions

Recipe instructions were created, **however some were not viable for use**, and none generated a sequence that included every ingredient in the recipe fed into it. This may be because our dataset was not robust enough, but that can possibly be remedied through using more recipes during training the algorithm in further testing.

Improvements include ensuring that the instructions contain all ingredients in the vector that contain non-zero values and eliminating repeating loops that the algorithm gets stuck on.