Capstone Project Final Report

David Tam

BrainStation

Problem

Over the years, regular home cooking has dipped drastically with each succeeding generation, from 64% with millennials to 55% with Gen Z (Charlebois, 2020). In the same study, 68% of Canadians claimed to want to spend more time with home-cooking (likely higher now with COVID-19). I believe retailers can a play a huge role here – the goal of my capstone project is to address this by building a recommender system that suggests recipes based on the users' shopping patterns and recipes they enjoy. This would benefit both the shopper and the retailer as it offers a personalized shopping experience and an opportunity to increase shoppers' basket size.

Solution

I believe the domain of recommendation systems is seldom used in grocery retail and is what makes it a great problem to address with data science techniques. Retailers offer rewards programs for the purpose of understanding consumer behaviour, yet the data is underutilized to improve shopping experiences. Loblaw and Walmart, two of the largest retailers in Canada, have been making great strides in e-commerce, but do not have a holistic recommendation system. Walmart offers sponsored products whereas Loblaw's online platform offers related items, but neither offers an element of user-tailored experience. Recipe websites like AllRecipes tend to only suggest general popular recipes or what is trending, but also missing the tailored experience. I believe implementing a more coveted recommendation system that combines elements of an ingredient recommender (through basket analysis and network analysis) and recipe recommender (through collaborative filtering) will improve the shopping experience which online retail is currently lacking.

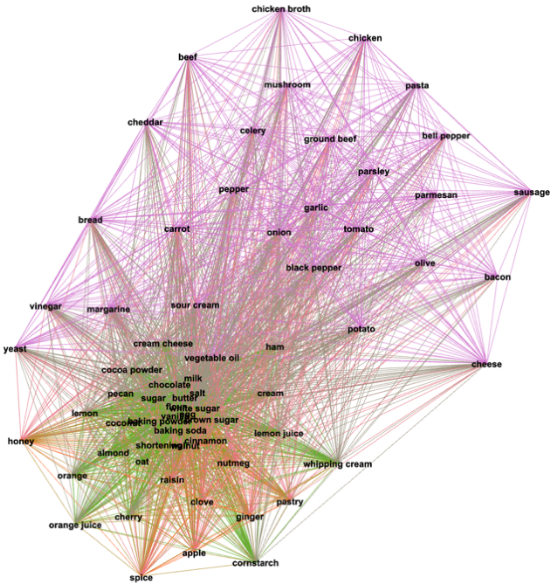
Data Acquisition

I was able to acquire my datasets from a combination of Kaggle and web scraping with BeautifulSoup. In the Kaggle csv datasets, it provided me with two tables: one with user IDs, recipe IDs (foreign key) and the respective user ratings. The other dataset includes a table with recipe IDs (primary key) with varying recipe features such as the author and ingredients. With web scraping, I scraped additional recipe features with the same recipe IDs (primary key) and these include nutrition values like calories, cook times, categories and servings.

Preprocessing, Cleaning, EDA

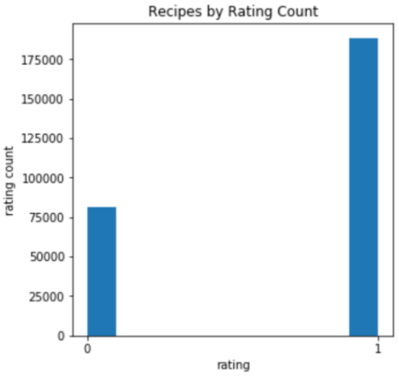
For my recipe recommender, I intended to use Surprise package to build the model, which requires user ratings for each recipe. That meant any recipes that did not have user ratings would automatically be dropped. Additional cleaning addressed missing values and ensuring there were no strings in numeric columns – what began as a dataset with over 7,000 recipes ended up closer to 1,700 recipes.

To address the ingredient recommender element, I leveraged basket analysis and network analysis in order to suggest related ingredients to the user. With that in mind, I had to transform the ingredients column to its own label-encoded ingredient-recipe matrix. This enabled me to apply the a priori function to find frequent item sets (for basket analysis) and unique combinations of item pairs (for network analysis). The ingredient component offers retailers opportunity for incremental revenue and acts as a filter to increase relevancy for users based on that current shopping transaction. Through EDA, I established that 1% would be my support cut off for basket analysis as this captured 25% of the ingredients.



Network Graph of 371 ingredients with 10,161 edges

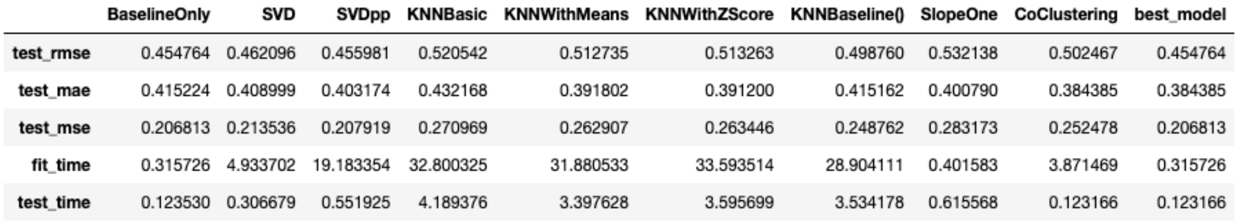
From network analysis, I was able to find a fully connected ingredient network with ingredients that were grouped together in three main communities – desserts, mains, common ingredients. Additionally, there was a noticeable imbalance in rating scores where 70% of votes were 5's and the other 30% spread across 4 and lower. To level off the data, I chose to binarize the data, transforming 5 to 1 and everything else to 0, suggesting that the user either loved the recipe or not.



Binarized rating score distribution

Modelling

For the second component of the recommender system, the recipe recommender, I utilized Surprise (Pyton scikit for recommendation systems) to build my collaborative filtering model. I trained each available algorithm in the package to the trainset, to identify the algorithms with the lowest RMSE and MAE scores – BaselineOnly algorithm had the lowest RMSE score (0.455), and the CoClustering algorithm had the lowest MAE score (0.384). Hence, I performed a gridsearch between the two algorithms to further compare the RMSE and MAE scores. In the end, both models ended up with relative MAE scores while BaselineOnly had comparatively lower RMSE score (0.453 versus 0.495). With that, I chose BaselineOnly algorithm for my final model, which makes predictions based on the average item rating along with user and item biases.



	BaselineOnly	SVD	SVDpp	KNNBasic	KNNWithMeans	KNNWithZScore	KNNBaseline()	SlopeOne	CoClustering	best_model
test_rmse	0.454764	0.462096	0.455981	0.520542	0.512735	0.513263	0.498760	0.532138	0.502467	0.454764
test_mae	0.415224	0.408999	0.403174	0.432168	0.391802	0.391200	0.415162	0.400790	0.384385	0.384385
test_mse	0.206813	0.213536	0.207919	0.270969	0.262907	0.263446	0.248762	0.283173	0.252478	0.206813
fit_time	0.315726	4.933702	19.183354	32.800325	31.880533	33.593514	28.904111	0.401583	3.871469	0.315726
test_time	0.123530	0.306679	0.551925	4.189376	3.397628	3.595699	3.534178	0.615568	0.123166	0.123166

Base model scores between 8 different algorithms

$$\hat{r}_{ui} = b_{ui} = \mu + b_u + b_i$$

BaselineOnly algorithm where u is average rating of recipe, b(u) is user bias and b(i) is item bias

Conclusion

One of the challenges that I discovered about predictive recommendation systems is that it is difficult to assess the accuracy of the model without A/B testing or exploring click-through rate, at least with the Surprise package. The reason is because the model was trained on the full dataset and predictions were made on the anti-test set (recipes not yet rated by the users) so it is difficult to determine if the predictions are accurate without actually receiving user feedback. Instead, I did comparative analysis by exploring the RMSE score between the algorithms. In the end, the final model on anti-test set had an RMSE score 0.140, which I interpreted as a good score given it was much lower than the other RMSE scores I got.

Through this hybrid memory-based recommendation system, I hope it sheds light on the potential retailers have to improve e-commerce shopping experiences while encouraging home-cooking given. Additionally, this is only one approach to this issue as there are many opportunities for further development such as applying content-based methodologies that take into account other features of recipes like cuisines, nutrition values and cook times. Through applying ensemble learning, it can add additional layers to the recommender system that provides a more user-centric experience with more accurate recommendations. An additional element that was not applied to my recommendation system is a feedback loop that allows the recommender to update the database when users rate new recipes, so that it is not suggested again.

References

Charlebois, S. (2020, March 17). With COVID-19, home cooking may get its mojo back. Retrieved June 26, 2020, from http://www.canadiangrocer.com/blog/with-covid-19-home-cooking-may-get-its-mojo-back-93580