Network Recommendation for Small Businesses

Harvard Computational Science and Engineering Capstone Project, Fall 2020

Team members: Catharine Wu, Royce Yap, Zhenru Wang

Video: https://youtu.be/8lYnGJo8iiY

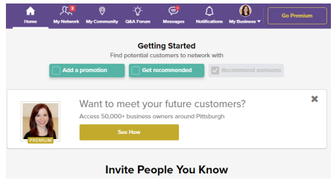
INSTITUTE FOR APPLIED COMPUTATIONAL SCIENCE AT HARVARD UNIVERSITY

Overview

Alignable aims to help business owners and operators grow their networks by finding peers with related professional interests. To build valuable relationships through referrals and recommendations, the key question here is "How do we best recommend potential connections?". Our solution is to build a recommendation system that would generate ranked lists of businesses so that Alignable can send out referral emails accordingly.

Key Question

Who should connect with whom?



Data

To start the project, Alignable provided a thoroughly anonymized dataset.

It contains basic information about around 3 million businesses, including geographical location, business type and scale, and their activities on the Alignable platform.

First, we generated one single table with useful features for connected businesses.

We then converted boolean columns to integers (0–1), and one-hot-encoded categorical columns, such as types (e.g. B2B) and scales (e.g. Regional). All numerical features are normalized between 0 and 1 to avoid scaling issues. We also added geographical distance and number of mutual connections for each business pair.

To make the train-test split, for each business, we used all the connections made before a timestamp (normalized between 0 and 1, here we chose .75) as training, the rest as the test set.

Modeling Approaches

Our strategy is to fit models on a community level. We started building our models with the biggest community in our dataset, LA (42,468 businesses). After the best model is selected, we run it on all available data.

Input and Output

Each row of the input table is a unique source-target pair, the columns are details about the source and target businesses. The output of the model is whether the pairs are connected, with 1 indicating a connection and 0 otherwise.

		Input				Output
source_ business_id	target_ business_id	source_business _profile	target_business _profile	distance	# of mutual connections	connected
2422	7257010	...	...	14.072	2	1
2422	8550114	...	...	1.8673	8	0

Business Profile
type, scale, location, industry, role, profile completeness, time joined, latest activity

Baseline

Our baseline model randomly predict whether two businesses are connected.

Predictive Models

We attempted multiple models used for binary classifications - regressions (linear, logistic), SVM, KNN, and trees (decision tree, random forest, boosted trees). The metric for predictive models is the overall accuracy, and the results are as follows. Random Forest performed the best.

Baseline	Linear Regression	Logistic Regression	Decision Tree	Random Forest	AdaBoost
50%	81.25%	80.36%	96.15%	99.25%	87.43%

Collaborative Filtering

We focused on the *user-based collaborative filtering* approach, namely, recommending items based on preferences of similar users. What is unique in this case is that users (those given recommendations) and items (those being recommended) are both businesses within the Alignable network.

Singular Value Decomposition (SVD)

To run the SVD algorithm, we employ the *Surprise* package, which estimates unknown ratings between certain user-item pairs by performing the minimization of the regularized squared error in the matrix by a standard stochastic gradient descent process.

Alternating Least Squares (ALS)

Theoretically, ALS works really well at solving the scalability and sparseness of the rating data. It is implemented in *Apache Spark ML*, and we are using *PySpark* so that it runs in a Jupyter notebook. It's simple and scales well to very large datasets.

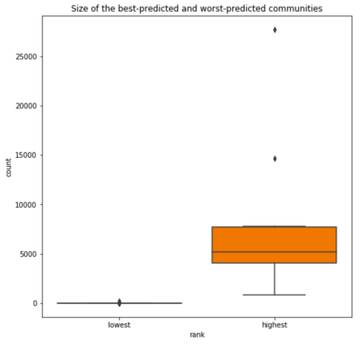
For SVD, using the *Surprise* package, we obtained a cross-validation RMSE of 0.26. For ALS, after tuning the algorithm, the best model results in a validation RMSE of 0.28. This validation RMSE score looks good considering the prediction values are on a scale of 0 to 1.

Evaluation and Interpretation

We used the unordered Normalized Discounted Cumulative Gain (NDCG) as a metric and fit the best performing models, Random Forest and SVD, on three communities, LA (~40,000 businesses), Midtown East in New York (~25,000), and Downtown Boston (~5,000). In the table below, Random Forest out-performed SVD.

	Baseline	Random Forest	SVD
LA	0.0222	0.1178	0.0382
NYC	0.0003	0.0086	0.0073
Boston	0.0023	0.0322	0.0270

Finally, we ran Random Forest on all communities. To mitigate runtime difficulties, we clustered target businesses using K-means, and only recommended those that are in the same cluster as the source business. We achieved an **83.4%** overall accuracy on average among all communities. By plotting the feature importance of our model, we identified significant effects from attributes such as the time a business joined Alignable, when it was most recently active, both businesses' distance and their mutual connections.



By exploring the relationship between accuracy and community size, we see that our model performed better on larger communities.

Final Deliverables

1. A CSV file containing 5 ranked recommendations for each business.
2. A single python file that produces this CSV, such that the client can reproduce the result or update it when new data comes in.
3. Our modeling approaches and analysis.