Predicting Business Successfulness using Predictive Data Analytics

Karan Chaudhari
Pune Institute of Computer Technology,
Pune

Harsha Gandikota
Pune Institute of Computer Technology,
Pune

ABSTRACT
In the modern business landscape, new business institutions are constantly emerging to meet the demands of the free market. However, the mere founding of a business does not necessarily mean that it will be successful among the general populace. A business that appears to be good idea, might in fact not be a good idea at all. So, to be able to appropriately predict the chance of success of a particular business, predictive data analysis and machine learning algorithms can be used to accomplish this task. So to predict the approximate success of a business, recommendations will be used. These recommendations will be generated by applying machine learning to a particular set of data. In this case, openly available yelp data that is set to perform all machine learning tasks will be used. Recommendation generation will also be done using both collaborative and content-based filtering. In this case, two different algorithms will be used, the k-nn nearest neighbor algorithm and dimensionality reduction through single value decomposition.

General Terms
Recommendation Generation

Keywords
Machine Learning, Problem Solving

1. INTRODUCTION
Recommendation Generation is starting to be seen as widely applicable in the world of consumer goods. With the number of consumer goods increasing drastically over the years, finding relevant results amidst the sea of data can prove to be very difficult. To be able to sort or filter through all this raw data, various personalization frameworks have been proposed. These personalization frameworks seek to limit to the overall set of results to a far limited number of relevancies. These personalization frameworks can be of three types. These are Content-based filtering, Collaborative-based filtering[2,4,5,7], and Hybrid-based filtering[5]. Each of these filtering methods will use a different set of rules to adequately generate recommendations or suggestions regarding the user's preference to a business.

Preference to a business will be indicated through the number of stars that the user will assign to a particular business. Higher the number of stars a user is more likely to give, greater the chance of success of that particular business. So, the recommendation engine to be constructed will be predicting the number of stars that a particular user will assign to another particular business, and accordingly help decide how great of a chance the business has at succeeding over the others.

One method of filtering is the collaborative-filtering method. In this method, a database is built that maps the preferences of the user to a particular product. So, say for example there is a user say u1, with a certain set of preferences. What will be done is that u1's given preferences will be used to find out what other users share a similarity with him/her. When a similarity between u1 and other neighboring users are found in the database, other user's preferences to u1 are recommended. This method is used by sites like Facebook, Myspace, LinkedIn to recommend new friends, groups, or other social connections.

Another method of filtering is Content-based filtering[5]. This method is based on a description of the item and a profile of the user preference. So in a content-based filtering, keywords are used to relate to a particular item, and a user profile is built to indicate the type of item the user likes. In particular, various candidate items are compared with previously rated items by user and best-matching items are recommended.

The final method of filtering is the hybrid filtering method. Hybrid recommender systems use elements from both collaborative and content based filtering methods to make recommendations.

2. TYPES OF RECOMMENDATION SYSTEMS

2.1 Hybrid Recommendation Systems
Hybrid Recommendation Systems are basically information filter systems that seek to predict the 'rating' that a user would give to a particular system through the utilization of both collaborative and content-based filtering methods. Hybrid filtration is chosen as it provides a chance to provide more effective results as compared to the individual implementation of content and collaborative based filtering[2, 4, 5].

Examples of the implementation of a hybrid recommender system can be seen in applications such as Netflix. In which recommendations are made by comparing the watching and searching habits of similar users as well as by offering movies that share characteristics that a user has preferred in the past. In this sense, recommendations are made by performing collaborative and content-based filtering separately and then combining the results to give a more effective recommendation result.

Another reason hybrid approach of recommendation is preferred is that the known shortcomings of the pure forms of information filtering can be mitigated. For example, the content based and collaborative based systems both suffer from the problem of cold-start, meaning that say when making new recommendations to a new user, the past preferences of the user are not known since that user is new; hence the making of recommendations is drastically limited.

2.2 The Yelp Dataset

The Yelp dataset is a free and readily available by the American Multinational Company "Yelp". This particular dataset is chosen as it was readily available to be downloaded and was in the easily to parse json format. Another reason why yelp was a preferable option was that it provided a wide array of useful data.

The Yelp data set overall provides data on various objects, these objects include: business, reviews, check-ins, users, and tips. Basically, the overall purpose of the yelp data set is to provide information about businesses, and how the users perceive the businesses through the reviews provided. The yelp dataset can be said to be quite vast as it provides the following:

1. 1.6 million reviews and 500,000 tops by 366,000 users for 61,000 businesses.
2. 481,000 business attributes, e.g. hours, parking availability, ambience.
3. Social network of 366,000 users for a total of 2.9 million social edges.
4. Aggregated check-ins over time for each 61,000 businesses.

This data-set is ideal as it was collected over a large range of cities in varying countries including UK: Edinburgh, Germany: Karlsruhe, Canada: Montreal and Waterloo, U.S.: Pittsburgh, Charlotte, Urbana-Champaign, Phoenix, Las Vegas, and Madison.

So, the data-set has a huge potential of uncovering useful information such as cultural trends, location mining and urban planning, seasonal trends etc.

In the operation of determining the capability of success of a particular business, data-sets regarding particular objects will be used. These objects will also have their own sub-parameters mentioned as well. These objects include:

- Business
- user
- check-in
- and review

2.3 Software Context

Python

The primary language that will be used will be the python programming language. Python is being used instead of the more conventional languages as it offers greater advantages. Python's syntactical interface is much simpler and straight forward as compared to that of C++. It offers a more clean and un-complicated syntax and steps when it comes to handling and processing raw text and other forms of bulk data. Another reason why python was more preferable is that it offers a wider array of standard libraries that can be used when performing operations. Python offers a wide array of easily accessible libraries, these also include the json library that can be used to perform parsing operations on json files.

MrJobs

Another advantage seen in the use of python was the availability of the mrjobs module. The mrjobs module is a module that is capable of performing a large number of map reduce operations on a given data-set. Meaning that, with mrjobs, instead of having to manually divide the json files into discrete chunks and assign jobs to be performed on each individual chunk, map-reduce operations can be performed on a single json file automatically. The mrjobs module will automatically perform a mapping of the overall dataset to various data clusters and will coordinate in the assigning of jobs to each cluster. Mrjobs also allows for a direct map reduce from the raw data to data base software that supports map reduce operations, such as Hadoop.

Json

The dataset that is available to us will be in the Json format or the JavaScript Object Notation. The json format of the data set offers us a greater advantage as compared to the use to other notations such as xml. For one thing a greater cluster of data can be adequately stored within the json as compared to that of xml. Another advantage of json is that it is a format that is easier to parse and read as it has a more simpler syntax. Finally, with the Json format can perform mapreduce operations with a greater ease, and overall operations such as parsing of json files are readily available via the python json library.

2.4 Algorithms Considered

The algorithms that will be considered in the implementation of recommendation system will primarily consist of two central algorithms. The approaches to be used can be specified as the K-NN algorithm, which is used in the process of classification, and the SVD[3,7,14] method, or single value decomposition method, which is used to reduce the dimensionality of the probability matrices considered. The algorithms can be further expanded in the following:

K-Nearest Neighbor Algorithm

The K-Nearest Neighbor Algorithm[4] is an algorithm used for the purpose of classification of a given set of raw entities from the data set. It is a supervised learning algorithm which basically performs the operation of grouping together raw entities which are derived from a source practice data set. The algorithm will decide whether a particular entity belongs to a particular class by the use of a distant metric, or a Euclidean distance can be used in this case. The algorithm will take into consideration all K nearest entities to the given subject entity and based on the average of the number foreign classifications available in the vicinity of the subject entity, and finally decide what class the subject entity belongs to.

The reason for the use of the K-NN algorithm in particular, is mainly due to the utter simplicity of its nature and works very well when it comes to basic classification. Another reason why K-NN is used is because it works more effectively when using a larger training set. This is due to the fact that as the training set becomes larger, more precise classifications can be made as better reading is made when taking the average of all the surrounding K entities.

Matrix Factorization

Matrix Factorization [3]model helps to map users and items to each on other on a matrix using vector semantics.

Accordingly:

Each item 'i' is associated to a vector qi such that: $qi \in Rf$

where, Rf, represents the set of possible vectors of dimensionality 'f'.

Similarly, it can be said the same to each user, 'u',such that,

$pu \in Rf$,where pu is the vector associated to each user.

For a given item 'i', vector qi denotes the extent to which item 'i' possesses

desired factors.

For a given user 'u'. Vector pu measures the extent of interest the user has in items that are high on the corresponding factors.

The resulting dot product: qipu Captures the interaction between user 'u' and item 'i'.

An entirely different approach to estimating the blank entries in the utility matrix is to conjecture that the utility matrix is actually the product of two long, thin matrices.

This view makes sense if there are a relatively small set of features of items and users that determine the reaction of most users to most items.

Singular Value Decomposition

In SVD ie Singular Value Decomposition [3,7,14], the central idea is to factorize a single matrix 'A' into the following:

Amxn=U(mxm)S(mxn)VT(nxn)
Where, A is matrix of mxn, and the following: U and V are orthogonal matrices and S is adiagonal matrix.

Steps of SVD

1. Start out with a single Matrix called 'A'.
2. First, find what the matrix 'U' is, to calculate what A.t(A) is. Suppose D = A.t(A)
3. Next, using D, calculate the corresponding eigenvectors. The result will give us U.
4. Then solve t(A).A is, suppose E = t(A).A.
5. Next, using E, calculate the corresponding eigenvectors. The result will give V.
6. Next, calculate S. To calculate S, then take the square roots, of all non-zero Eigen values and populate the diagonal of S in descending order.

A fun property of machine learning is that this reasoning works in reverse too: If meaningful generalities can help you represent your data with fewer numbers, finding a way to represent your data in fewer numbers can often help you find meaningful generalities. Compression is akin to understanding and all that. One Can easily map this process with that of compression and decompression methods applied to reduce noise in Digital Signal Processing

Given that the SVD somehow reduces the dimensionality of the dataset and captures the "features" that can be used to compare users, what was needed was how to actually predict ratings.

The first step is to represent the data set as a matrix where the users are rows, movies are columns, and the individual entries are specific ratings.

In order to provide a baseline, fill in all of the empty cells with the average rating for that movie and then compute the SVD.

Once SVD is reduced to get the reduced matrix, rating can be predicted by simply looking up the entry for the appropriate user/movie pair in the reduced matrix.

A more better approach is to use regressive values itself as a measure for reducing the matrix. Then much more accurate results can be made when predicting the business ratings .

2.5 Use-Cases

The Use-case diagram shown in Figure 2 depicts two actors, one is the user and the other is the admin. The actor who takes the role of the user are associated with the use-cases called 'search', 'rate', and 'access recommendations'. The User is basically expected to perform searches through the entire entity-set for desired entities. These entities can be varied and can include items such as movies, music, products for sale, etc. The user will then assign ratings to the entities if he chooses to do so, these ratings can be described through the number of stars that he assigns to a particular entity. Based on the ratings and what items he searches, the recommendation system will give him recommendations that bet suit him.

The actor who takes the role of the admin are associated with the use-cases such as 'rate-recommendation', 'gather data', 'make profiles', 'show predictions'. The 'rate recommendation' use-case basically predicts how the user will rate a particular entity based on the data gathered on that user from past entities. Based on the result of the rate recommendation, the recommender system will decide whether to give a particular recommendation to the user. The admin will also perform the operation of gather all necessary data on the particular user or entities. It will then make an appropriate the profile based on the data gathered, and can also then show whatever predictions are made.

2.6 Evaluation

To evaluate the effectiveness of the proposed recommender system, RMSE is used, or root-mean-square error. This measure is used to measure the differences between values predicted by a particular model and those measures that are actually observed. It can basically aggregate the errors of a particular model into a single measure of predictive power. It is considered as a good measure of accuracy within a given system.

The average RMSE value given by yelp for its recommendations is 1.4. The aim is to reduce this value through the use of dimensionality reduction[7] through the SVD method. This reduction in RMSE will basically indicate an reduction in the magnitude of errors that occur and will hence indicate an improvement in the recommendation system as a whole.

2.7 Algorithm

Steps to be taken in the implementation of the recommendation system can be explained through the following:

1) Accept raw data-set in json format.
2) Parse raw data-set from json format to readable format to mongodb through map-reduce as mentioned in Fig.1.
3) Split parsed data into two sets, one is the test data-set, and another is the practice data-set.
4) Perform SVD operation on test data-set and obtain data-matrix representation.
5) Perform Nearest Neighborhood method on practice data-set and obtain Euclidean data representation.
6) From Euclidean data representation and data-matrix representation, predict future ratings by user.

7) Make recommendations to user.

3. FUTURE WORK

If trends can be found in business and user ratings over time,better predict ratings can be made with a movie average. Check-in information is used to find whether differences in prediction accuracy occur with more popular businesses

4. FIGURES/CAPTIONS

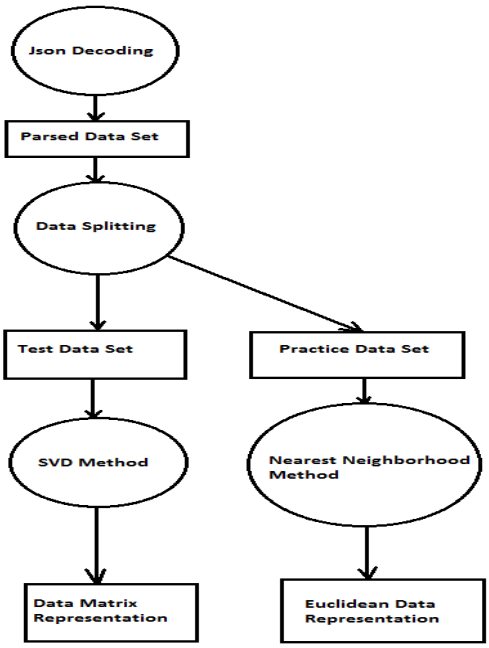

Fig 1: Data Flow Diagram for Proposed Recommender Systems, shows the process of how raw data is effectively divided and applied to different methods to produce different representations.

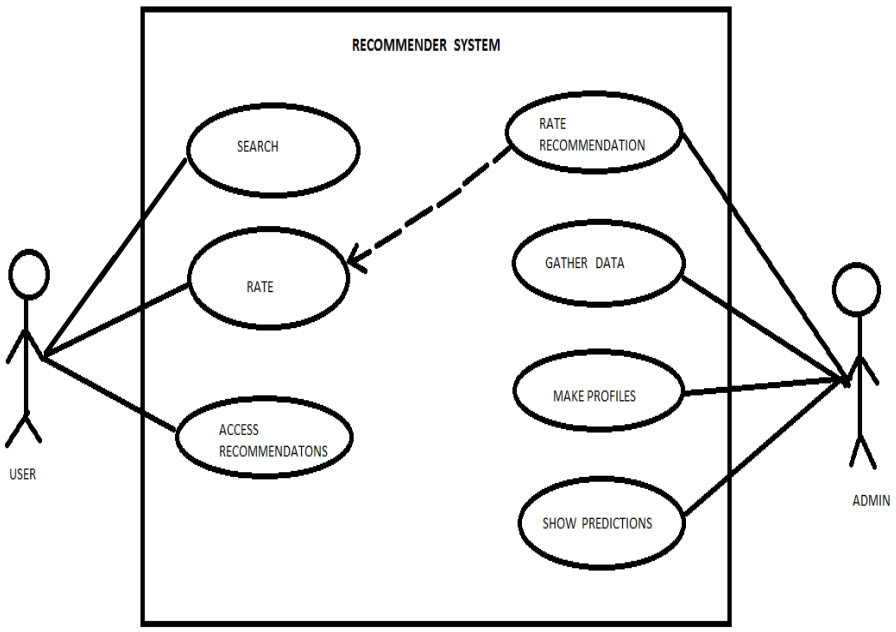

Fig 2: Use Case Diagram for the Proposed Recommendation System, shows the various use-cases and associations between user and the admin.

5. ACKNOWLEDGMENTS

We would like to take this opportunity to thank our internal guides for giving us all the help and guidance we needed. We are really grateful to our teachers for their kind support. Their valuable suggestions were very helpful.

We are also grateful to Prof. G.P.Potdar, Head of Computer Engineering Department, Pune Institute of Computer Technology for his indispensable support, suggestions.

6. REFERENCES

[1] Jiliang Tang, Huiji Gao, Xia Hu and Huan Liu , "Context-Aware Review Helpfulness Rating Prediction ", Recsys-13

[2] Jonathan herlocker,"Evaluating collaborative filtering recommender systems",2004

[3] Yehuda Koren, Yahoo Research, Robert Bell and Chris Volinsky , "Matrix Factorization Techniques for recommender systems" ,IEEE 2009

[4] B. Sarwar, G. Karypis, J. Konstan, and J. Riedl. ,"Item-based collaborative filtering recommendation algorithms"., WWW, 2001.

[5] X. Su and T. Khoshgoftaar.," A survey of collaborative filtering techniques. Advances in AI", 2009.

[6] M.OMahony and B. Smyth.," Learning to recommend helpful hotel reviews".RecSys, 2009.

[7] Sarwar, B. M., Karypis, G., Konstan, J. A., and Riedl, J. (2000). Application of Dimensionality Reduction in Recommender System–A Case Study. In ACM WebKDD 2000 Workshop.

[8] S. Funk, "Netflix Update: Try This at Home," Dec. 2006;http://sifter.org/~simon/journal/20061211.html

[9] Resnick, P., Iacovou, N., Suchak, M., Bergstrom, P., and Riedl, J. (1994). GroupLens: An Open Architecture for Collaborative Filtering of Netnews. In Proceedings of CSCW '94, Chapel Hill, NC.

[10] Sarwar, B., M., Konstan, J. A., Borchers, A., Herlocker, J., Miller, B., and Riedl, J. (1998). Using Filtering Agents to Improve Prediction Quality in the GroupLens Research Collaborative Filtering System. In Proceedings of CSCW '98, Seattle, WA.

[11] Berry, M. W., Dumais, S. T., and O'Brian, G. W. (1995). Using Linear Algebra for Intelligent Information Retrieval. SIAM Review, 37(4), pp. 573-595.

[12] Berry, M. W., Dumais, S. T., and O'Brian, G. W. (1995). Using Linear Algebra for Intelligent Information Retrieval. SIAM Review, 37(4), pp. 573-595.

[13] Joseph A. Konstan, John Riedl "Recommender Systems: from Algorithms to User Experience" Springer Science+Business Media B.V. 2012

[14] Carlos Cobos, Orlando Rodriguez, Jarvein Rivera, John Betancourt, Martha Mendoza, Elizabeth Leon, Enrique Herrera-Viedma. "A hybrid system of pedagogical pattern recommendations based on singular value decomposition and variable data attributes.