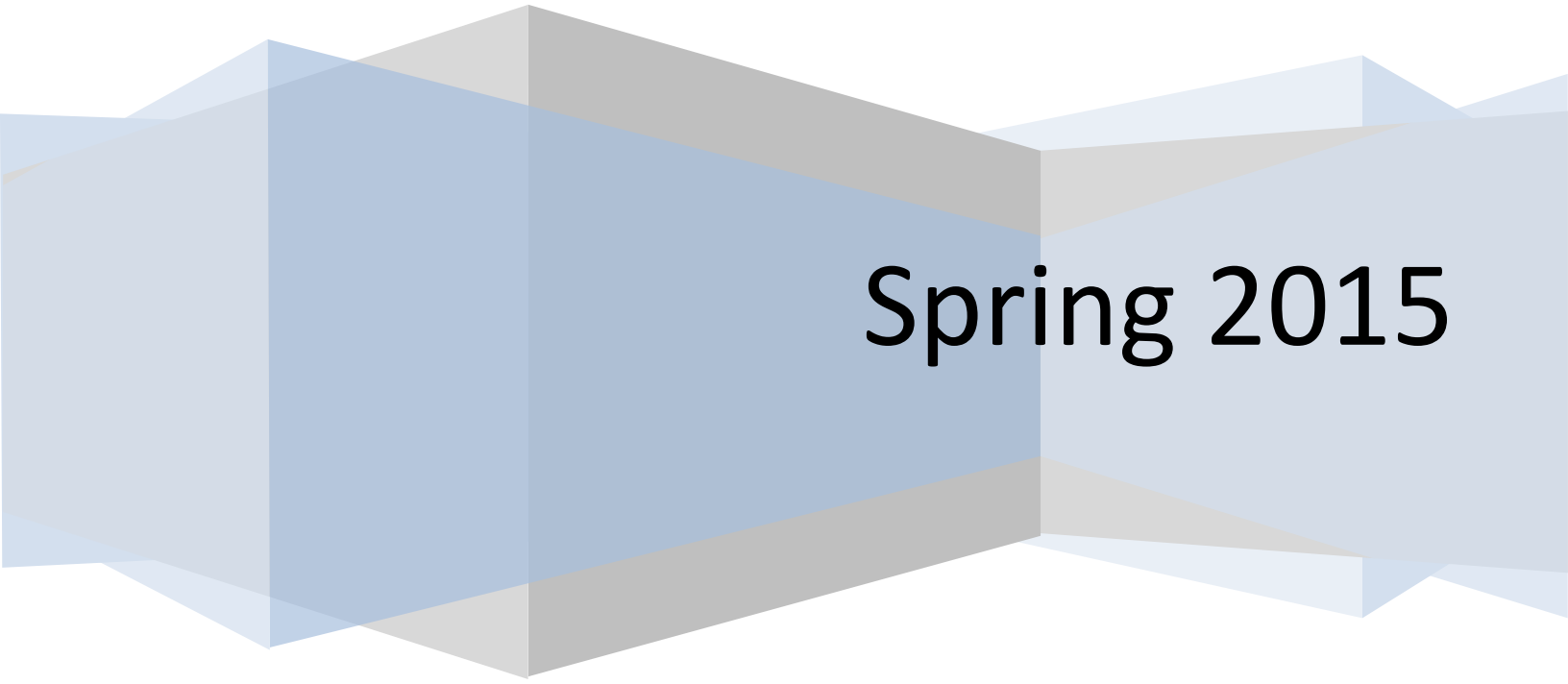
University of St. Thomas - Data Mining

Sentiment Analysis & Rating Predictions of Reviews in Yelp Dataset

SEIS 734-1

Driesch, Robert
Jois, Suraj
Mandhale, Shraddha
Miller, Todd
Parastoo, Karacic
Princewill, Eneh



Spring 2015

I Project Overview

Yelp is a corporation that was formed in 2004 and hosts a website (Yelp Home Page) which publishes crowd-sourced reviews about local businesses. They routinely track over 100 million unique visitors each quarter and contain over 77 million local reviews within their database.

In order to continue to add value to the data they have collected and to continue to push the frontiers of data science research, Yelp has been hosting multiple Yelp Dataset Challenges for the past two years. These challenges offer a subset of their local business data from multiple cities and across different countries that can be mined for a variety of targeted challenges.

We have chosen to base our Group Project around the data provided by the Yelp Dataset Challenge and focus upon our ability to predict a review's rating from its text alone. More specifically we want to build a model using the algorithms that we have discussed in class to help predict the star rating a business is likely to receive based upon previous customer reviews. Additionally we want to see what changes can be observed while taking into account differences within the data corpus for different geographies.

II Algorithms

In order to mine the business review data from the Yelp Dataset, we have chosen to concentrate upon the following classification algorithms: C4.5 Decision Trees; Naive Bayes Classifiers; and Back Propagation Neural Networks.

We chose the first two classification algorithms because we have had experience with both of them during the course work for this class. The last algorithm was chosen to see how the algorithms that we have already had some experience with held up against an algorithm that was not covered in the course work and is considered an advanced topic for a future course.

C4.5 Decision Trees

In a general sense, a decision tree is used as a classifier for determining an appropriate action or decision when dealing with an existing set of known previous actions. It helps you identify what attributes to consider and how each attribute has affected the outcome of previous decisions. It makes use of a tree like structure of conditions and branches that identify each action and consequence. The bottom or leaf nodes will identify the resulting decision or action the decision tree is attempting to model.

The C4.5 algorithm that we use within SAP HANA (SAP HANA Predictive Analysis Library, 2014) builds a decision tree from a set of attributes contained within some set of training data. The attributes of the data are interrogated by C4.5 algorithm at each node within the tree as it is being built and it makes its decision based upon the information entropy that splits the data into separate classes or branches. This process is repeated recursively on the smaller subset until the some exit or stopping criteria is met.

Naive Bayes Classifiers

The Naive Bayes algorithm to build a decision tree is based upon the Bayes theorem which treats all of the attributes as if they are conditionally independent of one another. One of the main advantages of the Naive Bayes classifier is that is generally only requires a small amount of training data in order to build an effective model (SAP HANA Predictive Analysis Library, 2014).

Back Propagation Neural Networks

Back Propagation Neural Networks build a model that is similar to the nervous system of a human. It builds a network structure of neurons and sets the connection weights between these neurons based upon the training method used for the network (back propagation in our case).

During training, the neurons take in the signals (attributes) from the outside dataset and transform them into a specific value for output. The connection weights will continually get adjusted as the network compares the generated output from the training data against the desired output.

III Tools Selection

For this project we chose to make use of a variety of different tools to help assist us in the various tasks outlined in our basic Workflow (see figure xxx). While we did try and limit our analysis primarily to SAP HANA, we did make use of a number of additional tools to help in our preliminary analysis, data preparation and visualization.

Data Preparation	For data preparation and cleansing, we made use of custom scripts written in Python.
Preliminary Analysis	DQ Analyzer was used for early profiling of the Yelp dataset to help us determine the scope and focus of our project.
Data Mining	For the actual data mining, we made use of SAP HANA to store and process the mining algorithms against our database. Additionally, we made use of the SAP HANA Application Function Modeler (AFM) to generate the procedures needed to process the individual algorithms.
Visualization	The XLSTAT add-in for Excel was one of the tools used to help transform our results into a visual medium. Additionally we also made use of Matlab for creating some of our visualizations.

Data Preparation

The dataset that we chose to work with from the Yelp Dataset Challenge was delivered in a JSON format that first needed to be converted into comma separated vales (CSV) to allow for easy importation into our SAP HANA database. While numerous examples can be found on the internet to perform this type of conversion, we found that they all needed a bit of tweaking in order to suit our specific needs.

We chose to perform this JSON to CSV conversion using Python because there were many examples and code snippets showing the basic structure for performing this conversion. There were even some snippets provided by Yelp (Yelp Inc., 2014) to perform this conversion for previous versions of the Yelp Dataset Challenge. However, all of these snippets required additional modifications in order to successfully process this version of the dataset and correctly convert it into a form that could be easily imported into our database.

Since we customized the Python scripts, that also allowed us the opportunity to add logic to help cleanse the data while it was being converted. Some of the more glaring examples of the types of issues that we encountered within the data included embedded quotation marks, lengths of the various reviews and instances of the forward slash character that kept being interpreted as an escape character during importation.

Preliminary Analysis

During the early stages of the Project, we were still attempting to finalize the scope and focus of our research. We used the DQ Analyzer tool (DQ Analyzer) to assist us in this analysis. This free download helped us better understand the characteristics about the dataset provided by Yelp and eventually led to our decision to work with the Review Text and the Star Ratings within the dataset.

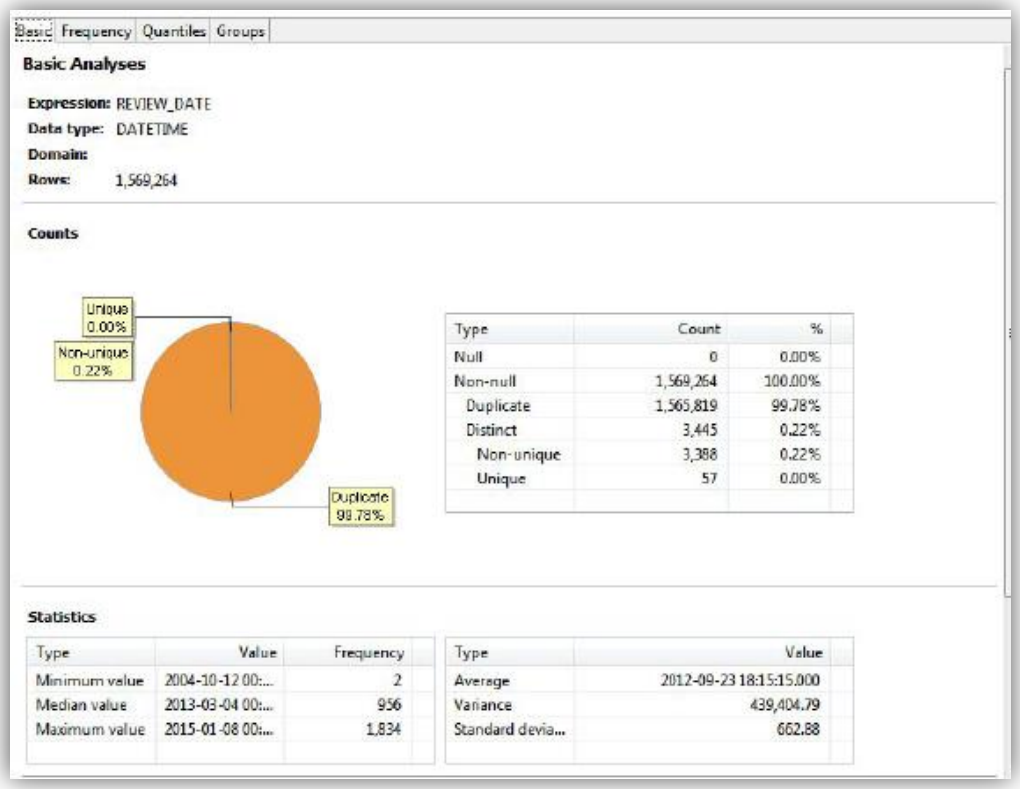


Figure 1 DQ Analyzer - Data Profiling Tool

The tool helped us by profiling the dataset and describing the characteristic and distribution of the various columns. It graphically displays the frequency, quantiles and specific groups within the data contained in the dataset.

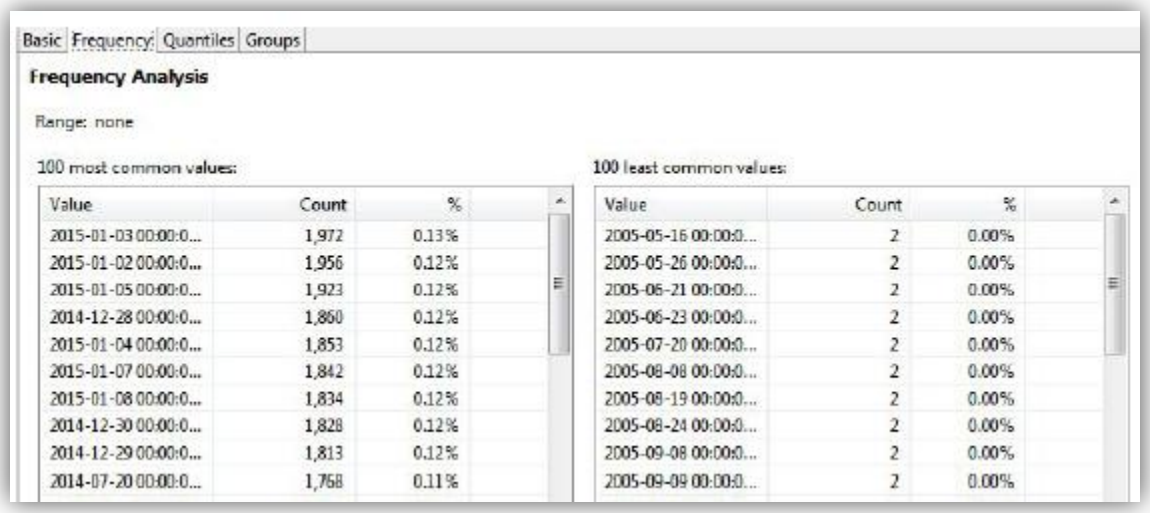


Figure 2 DQ Analyzer - Frequency Analysis

In various instances, we made use of the tool when it helped us discover that the data contained within a column (such as the useful votes in the Review table) were sparsely populated and skewed towards a single value.

Data Mining

We made exclusive use of the SAP HANA Application Function Modeler (SAP HANA Application Function Modeler) to create and generate the SQL Procedures (or flowgraphs) to perform the specific data mining algorithms against our dataset. The AFM is shipped as a part of the SAP HANA Development perspective and can be accessed through either the SAP HANA Development Studio or the equivalent Eclipse plug-in for SAP HANA.

A flowgraph is a development object within the HANA environment and can be used to model the flow of data from tables, views or procedures from the SQL catalog through relational operators and tie in functions from the Application Function Library (AFL) or the Predictive Analysis Library (PAL). When a flowgraph is activated, it generates a procedure to perform the specified operations and stores this information within the catalog.

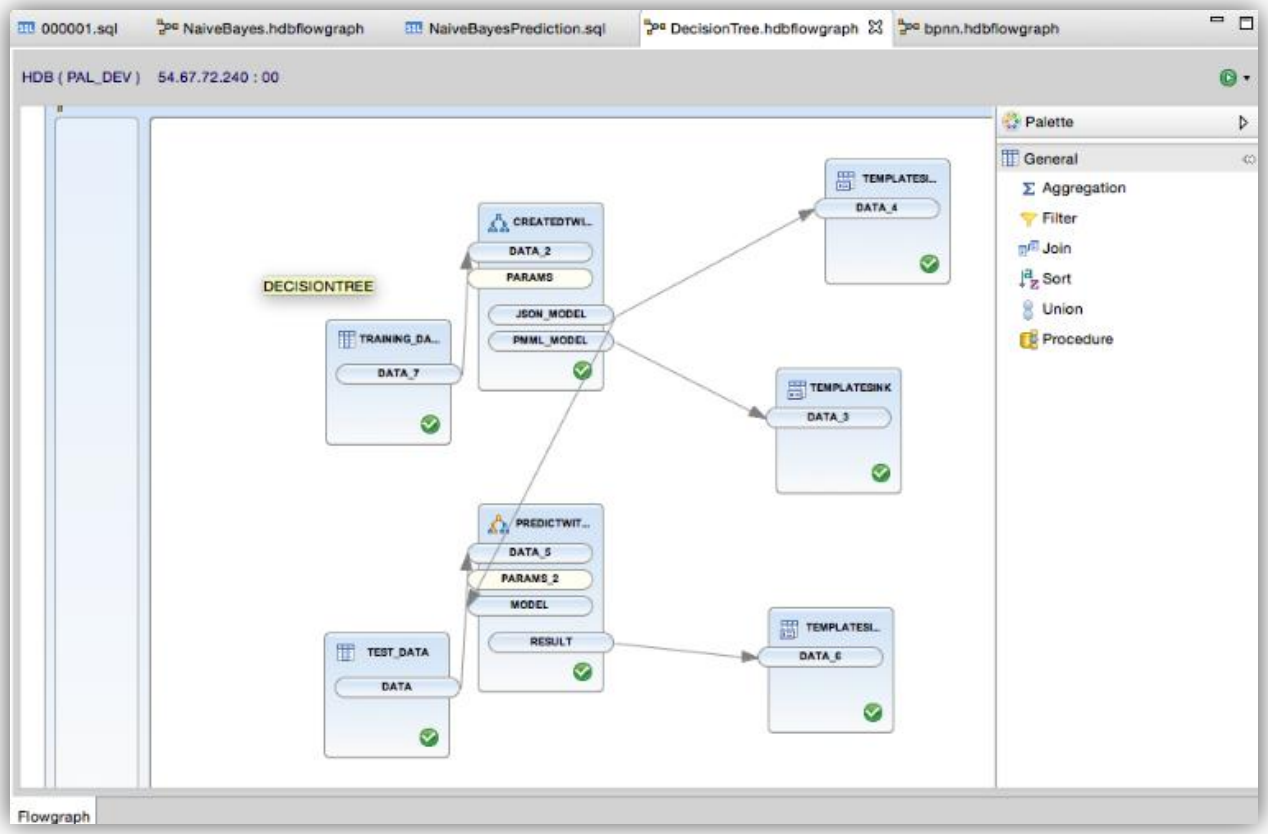


Figure 3 Application Function Modeler - Decision Tree

Using the capabilities of the AFM allowed us to follow a workflow where we could easily *plug-and-play* various permutations of the dataset into our flowgraphs and experiment with the results.

Visualization

We ended up using a number of different tools to assist us in visualizing our results including XLSTAT (XLSTAT Statistical Software for Excel), MATLAB (MATLAB - The Language of Technical Computing) as well as Excel. These tools allowed us to view the results and generate the necessary visualizations for the Confusion Matrix, ROC Curve and the Percent Off Diagonal charts needed for the presentation as well as this paper.

XLSTAT is a statistical analysis plug-in for Excel that contains a wide variety of built in functions that can be used to help analyze different results. We made use of this tool to generate a version of the Confusion Matrix with a heat map and a ROC Curve to show the effectiveness of our generated models.

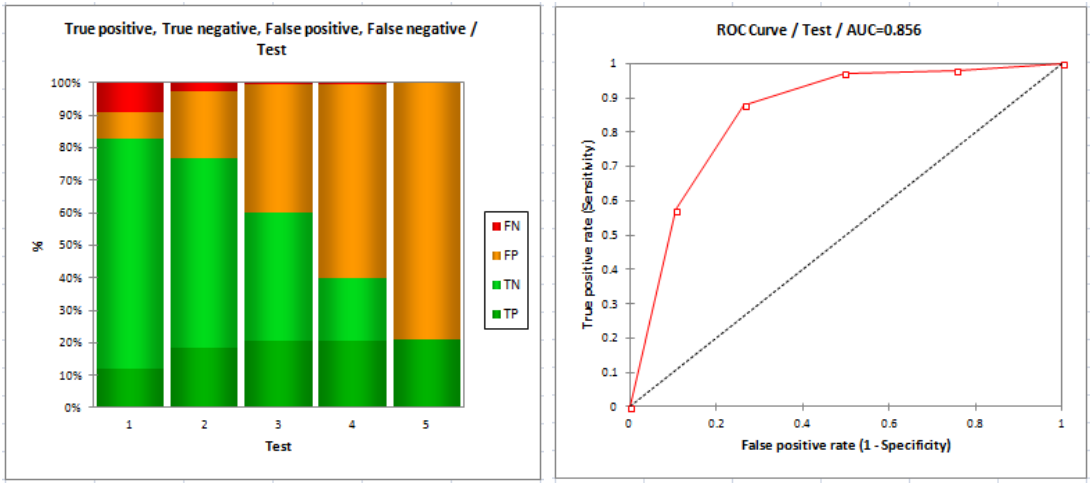


Figure 4 XLSTAT - Confusion Matrix & ROC Curve

We also made use of MATLAB to generate another version of the Confusion Matrix with a heat map that showed the results in a slightly different manner.

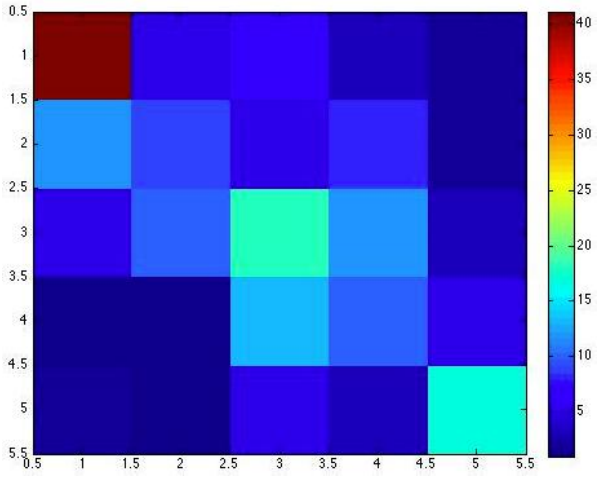


Figure 5 MATLAB - Confusion Matrix

Lastly, we attempted to create our own visualization of a Percent Off Diagonal matrix and Variance Distribution directly within Excel that more closely resembled the charts that we were exposed to within our coursework.

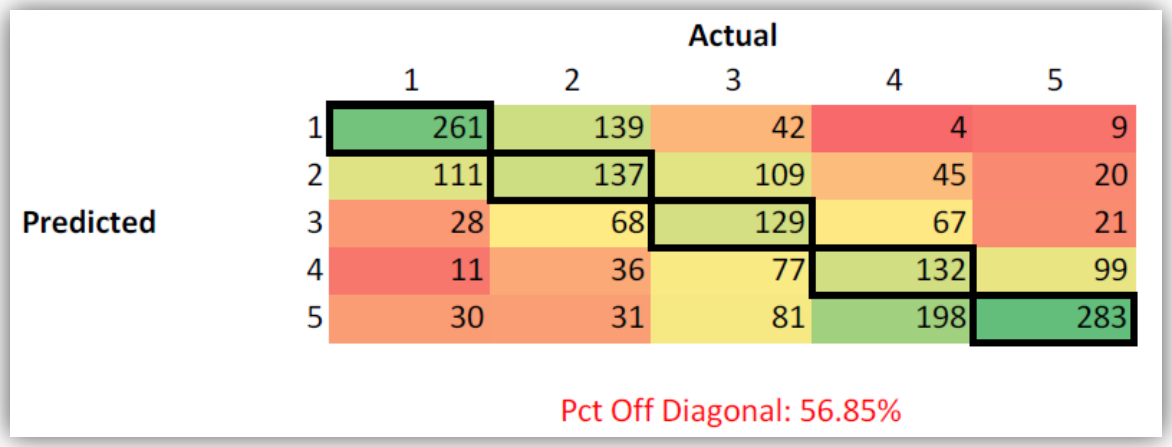


Figure 6 Excel - Percent Off Diagonal

IV Workflow & Methodology

For our project we chose to approach the problem by first performing our analysis across all of the geographies contained within our dataset. We wanted to perform the sentiment analysis against the text contained within the Yelp Reviews table for the different businesses across all of the cities and countries in the sample dataset that Yelp provided us. We would train and test our models against the three different algorithms and compare their results to one another.

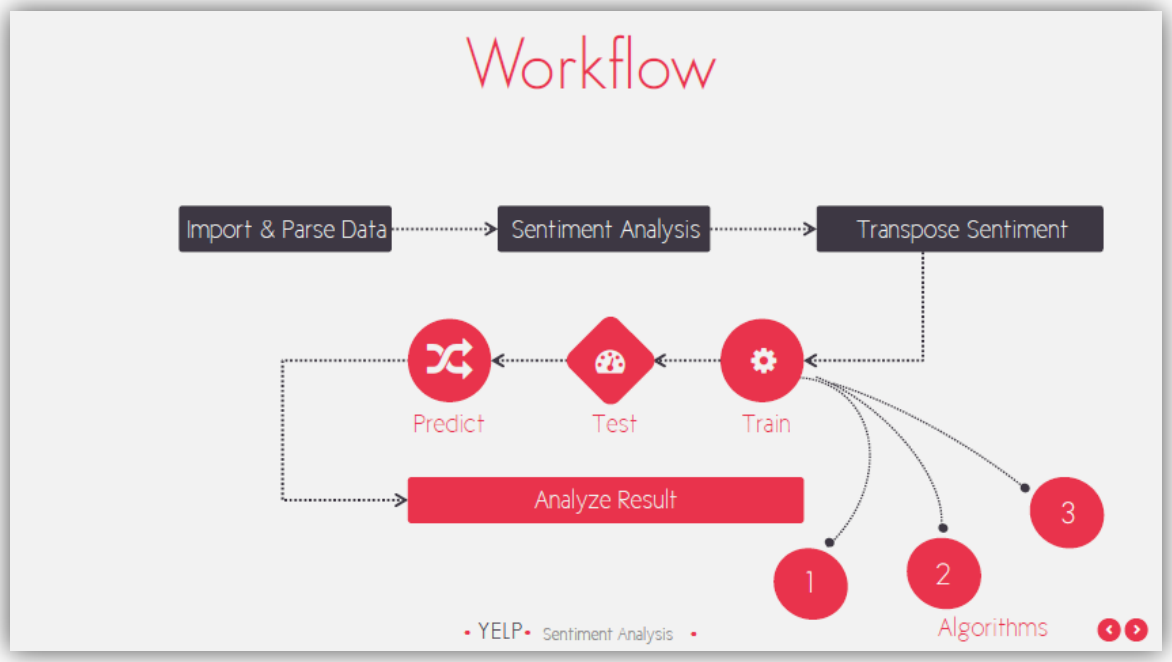


Figure 7 Process Methodology

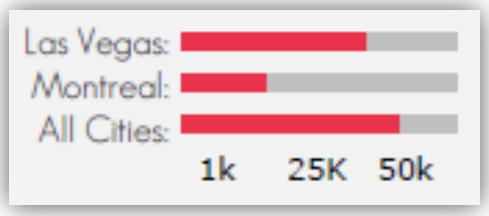
We quickly discovered that the data contained within the Yelp Dataset was heavily skewed towards two specific cities and also skewed towards the highest star ratings. In order to balance out this skewed data, we devised a scheme where we would randomly stripe the data to include a balanced number of geographies and star ratings to try and make the comparisons between the different algorithms and geographies more consistent.

CLUSTER	ONE_STAR	TWO_STAR	THREE_STAR	FOUR_STAR	FIVE_STAR
URBANA-CHAMPAIGN	1,157	1,233	1,901	3,625	4,208
KARLSRUHE	208	261	444	860	919
EDINBURGH	778	1,661	4,891	9,992	6,458
LAS VEGAS	72,523	62,241	100,008	196,349	248,339
MONTREAL	3,211	3,870	7,981	18,150	16,272
WATERLOO	242	278	473	910	781
CHARLOTTE	8,431	9,063	16,270	33,527	30,743
PHOENIX	64,748	51,355	73,622	167,535	233,932
MADISON	3,525	4,308	6,600	14,210	15,055
PITTSBURGH	4,988	6,338	10,529	21,441	22,820
TOTAL	159,811	140,608	222,719	466,599	579,527

Figure 8 City and Star counts within the Yelp Dataset

We also decided to focus our attention on just a small subset of the geographies contained within the dataset in order to keep the scope of the project more manageable. In this regard we chose two cities (Las Vegas and Montreal) in addition to our All Cities permutation for our geography specific analysis. We chose Las Vegas since it provided us with the largest subset of city specific data within the dataset so we felt that this would represent the most *mature* set of reviews available. Montreal was chosen since it represented the largest non-United States city and it also contained reviews from neighboring cities such as Quebec which meant that we also had a mix of English and non-English users generating the reviews.

Again since the data for Las Vegas and Montreal were not evenly distributed across their Start Ratings, we chose to randomly sample and stripe the data from the cities to make sure the results from our geography specific analysis matched our analysis from the runs made against all of the geographies. We ended up with a Training & Testing Dataset of varying sizes for the three different subsets we chose to work with. Within each subset we split the dataset to allocate 75% of the total rows for training and the remaining 25% for testing and verification.



V Data Mining Results

Our methodology indicates that we will first analyze the dataset subset for all of the cities (All-Cities) against the three different algorithms. What follows is a detailed description of the results from those three algorithms against that specific subset of data. We will then follow this section with an analysis of the results from the two geographies (Las Vegas and Montreal) and summarize how they compare to the results we observed against All-Cities.

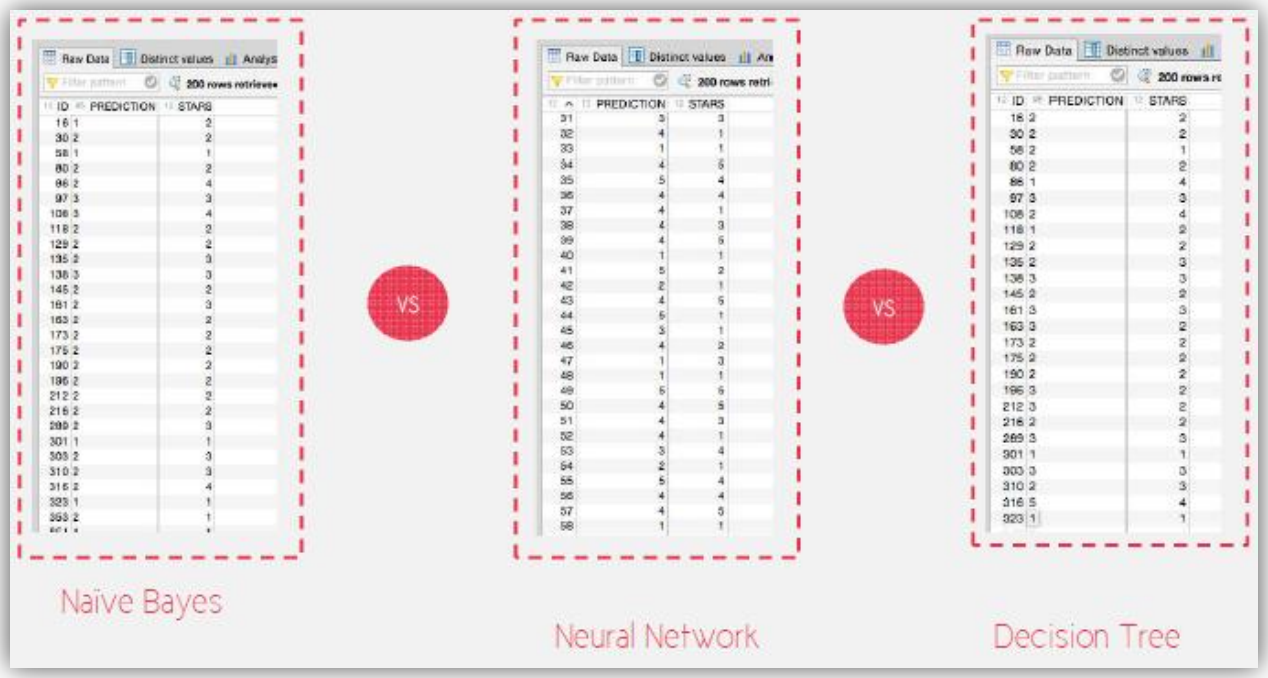


Figure 9 Results Comparison for each Algorithm

All-Cities Dataset Results

For the All-Cities analysis, we started with the C4.5 Decision Tree and built up a number of graphs to help us illustrate the results and to measure how accurate our predictions were. We focused upon the Confusion Matrix and ROC Curve as generated by the XLSTAT tool as our primary measurements. However, since the Confusion Matrix generated by the XLSTAT tool lists the values vertically we additionally generated a more traditional confusion matrix where we also listed the Percent Off Diagonal calculation.

The first two graphs show how the C4.5 Decision Tree performed against our test dataset. You can see in the Confusion Matrix a high degree of green that indicates that we accurately predicted a good share of True Positives (TP) and True Negatives (TN). This is especially true of the negative star ratings whereas the amount of green for the positive star ratings was quite small. The ROC Cure has a fairly shallow curve and gave us an Area Under the Curve (AUC) score of 0.833.

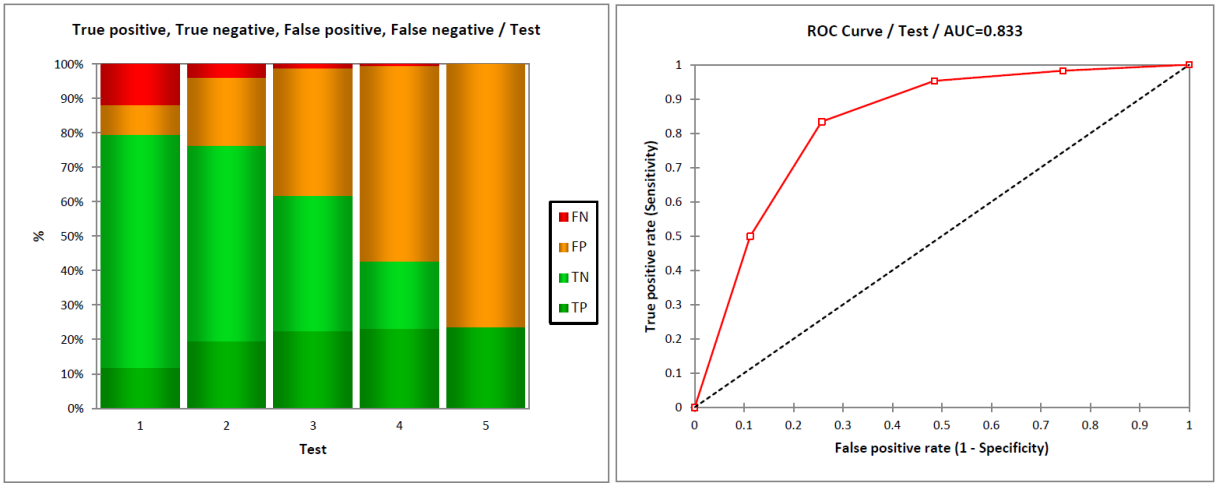


Figure 10 All-Cities: Confusion Matrix & ROC Curve for C4.5 Decision Tree

Initially we were pleased with the shape of the bars within the matrix as it indicated to us a fairly consistent slope that we mistook to mean that we were fairly close to the diagonal measurement we make with the traditional confusion matrix. However upon inspection of the more traditional confusion matrix we could see that even though our Percent Off Diagonal score was 61.25% there were cases where our predictions were off by 1 star (as in the case of the 5 stars).

ALL - Decision Tree

Predicted \ Actual	1	2	3	4	5
1	255	171	61	15	9
2	91	87	83	54	23
3	53	96	130	72	30
4	13	23	82	88	90
5	29	34	82	217	280

Pct Off Diagonal: 61.52%

Figure 11 All-Cities: Traditional Confusion Matrix for C4.5 Decision Tree

For the Naive Bayes Classifiers, we generated the same three graphs and were again confronted with very similar looking results. We had a high number of green showcased in the confusion matrix especially for the negative star ratings and we again appeared to be off by a single star on much of the higher star ratings.

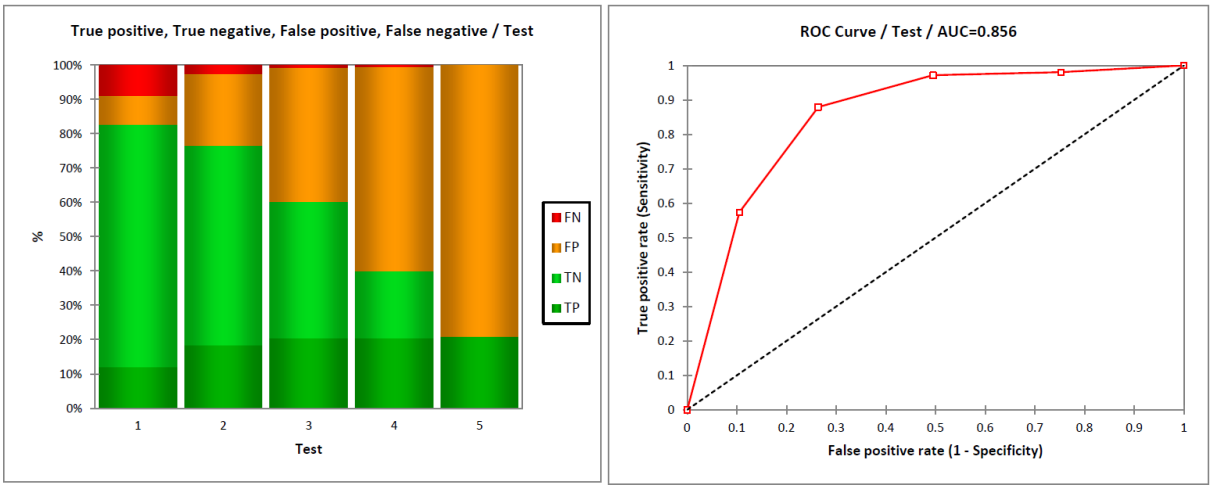


Figure 12 All-Cities: Confusion Matrix & ROC Curve for Naive Bayes Classifiers

The predictions scores for the Naive Bayes did improve over the C4.5 Decision Trees. We now have an AUC score of 0.856 and a Percent Off Diagonal 56.85%. Our initial conclusion was then that both types of decision trees behaved in a very similar manner for our All-Cities dataset even though the Naive Bayes did consistently perform slightly better.

ALL - Naive Bayes

Predicted \ Actual	1	2	3	4	5
1	261	139	42	4	9
2	111	137	109	45	20
3	28	68	129	67	21
4	11	36	77	132	99
5	30	31	81	198	283

Pct Off Diagonal: 56.85%

On Diagonal	Total	Off Diagonal
942	2183	1241

Figure 13 All Cities: Traditional Confusion Matrix for Naive Bayes Classifiers

The graphs generated from the results of the Back Propagation Neural Network again showed the same pattern. A high degree of green for the predicated negative star ratings and a much lower degree of green for the predicated high star ratings. The individual scores for the AUC (0.813) and the Percent Off Diagonal (63.12%) were also the lowest we had seen of the three different algorithms.

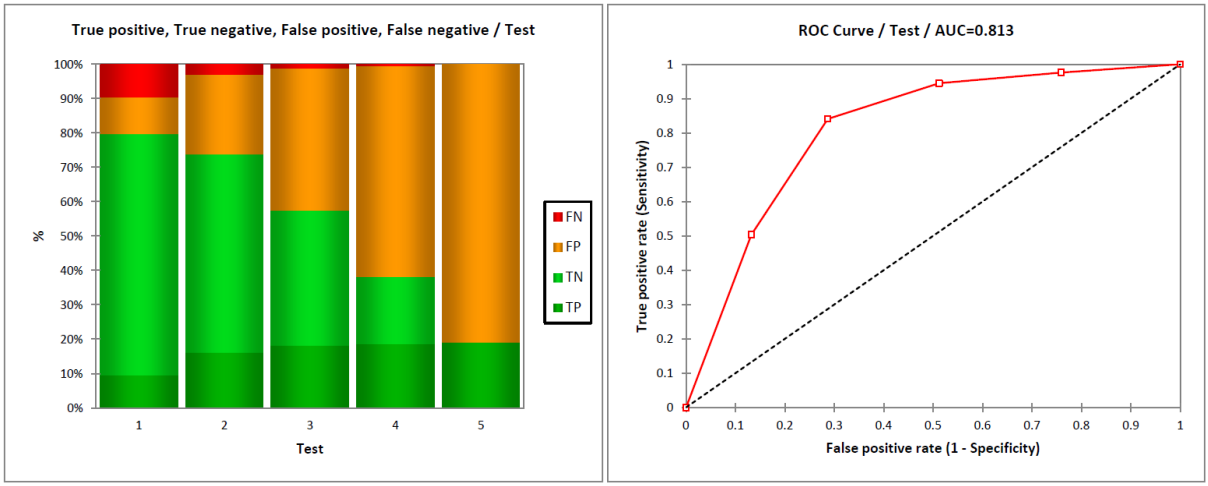


Figure 14 All-Cities: Confusion Matrix & ROC Curve for Back Propagation Neural Networks

Our conclusion from processing these three algorithms against our All-Cities dataset was that each of the algorithms showcased the same pattern. A high degree of accuracy in predicting the negative star ratings and consistently off by a single star in the higher star ratings.

ALL - Neural Network

Predicted \ Actual	1	2	3	4	5
1	209	140	43	13	10
2	63	88	77	51	19
3	115	96	178	99	66
4	22	37	58	74	81
5	32	50	82	209	256

Pct Off Diagonal: 63.12%

On Diagonal	Total	Off Diagonal
805	2183	1378

Figure 15 All Cities: Traditional Confusion Matrix for Back Propagation Neural Network

We still concluded that we would rank the three algorithms with Naive Bayes performing the best, followed by C4.5 Decision Trees and lastly followed by the Back Propagation Neural Network.

Comparison Against the Geographies Results

When we ran the analysis against the remaining two datasets for the different geographies and compared the resulting graphs with the ones generated for our All-Cities analysis see continued to see the same shape and pattern to our predicted results.

The C4.5 Decision Trees generated for the three different datasets all indicated that we were much more accurate predicting the negative star ratings rather than higher positive ones.

Las Vegas | All-Cities | Montreal

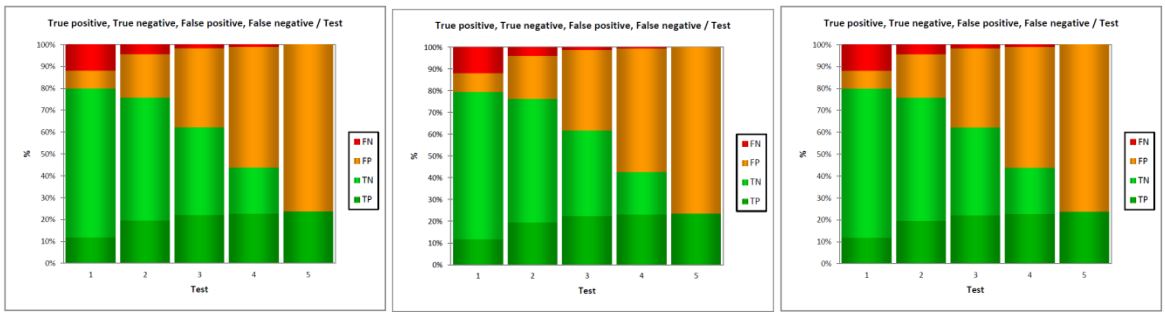


Figure 16 Confusion Matrix Comparison of C4.5 Decision Trees

The ROC Curves for the C4.5 Decision Trees also continued their similar pattern and nearly identical AUC scores. This indicated to us that there was not a high degree of difference in our predictions regardless of the geographies that we generated and processed it against.

Las Vegas | All-Cities | Montreal

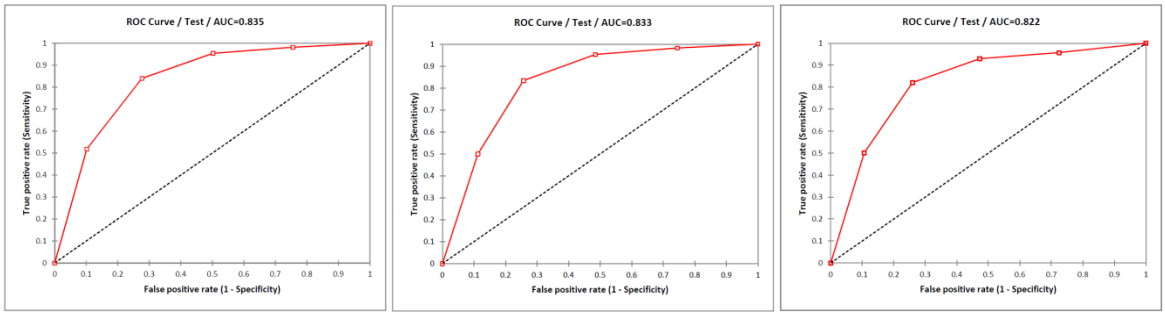


Figure 17 ROC Curve Comparison of C4.5 Decision Trees

The Naive Bayes Classifiers generated graphs for the three dataset continued to show the same pattern within the Confusion Matrixes that was established with the previous comparisons. The same general shape, slope and pattern of the green values indicating our strength in the negative star rating predictions rather than higher positive ones.

Las Vegas | All-Cities | Montreal

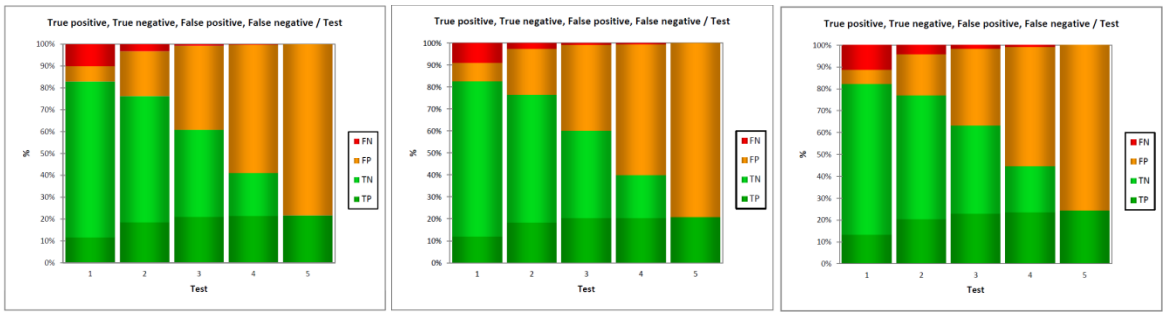


Figure 18 Confusion Matrix Comparison of Naive Bayes Classifiers

The ROC Curves for the Naive Bayes Classifiers again followed the established pattern. However it is worth noting that regardless of geography, the Naive Bayes predictions were always the *Top of Class* of the three different algorithms tested against each dataset.

Las Vegas | All-Cities | Montreal

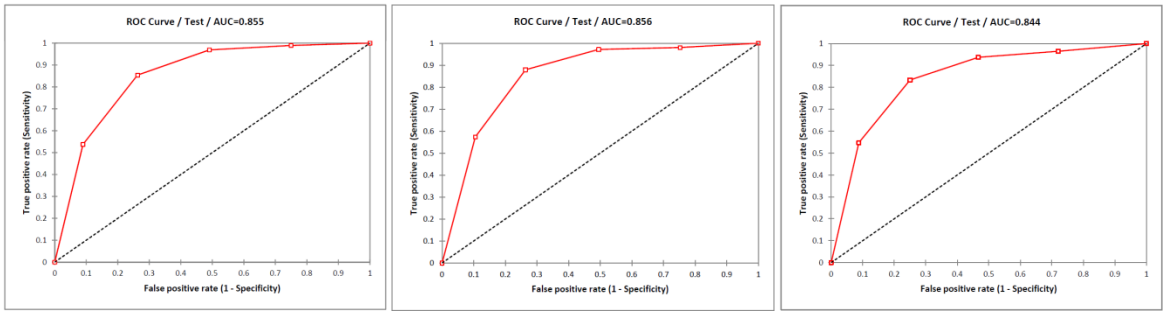


Figure 19 ROC Curve Comparison of Naive Bayes Classifiers

Lastly, the graphs generated for the Back Propagation Neural Network algorithm against all of the geography datasets continued to follow the pattern set forth by the previous two algorithms and showed the same patterns within the Confusion Matrixes.

Las Vegas | All-Cities | Montreal

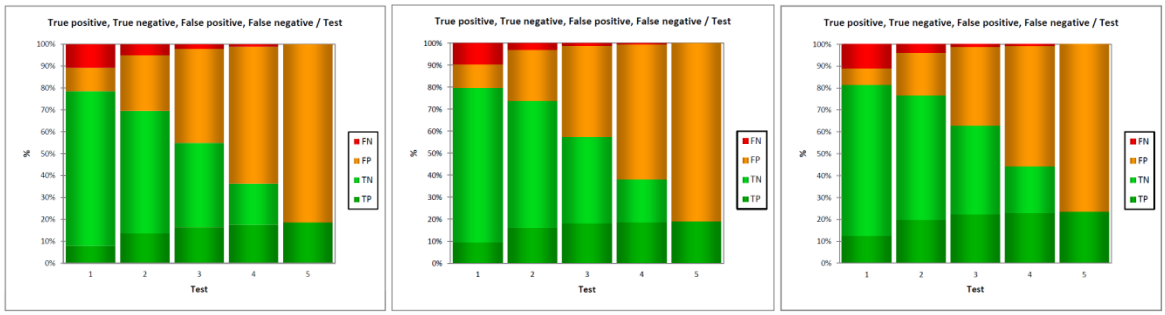


Figure 20 Confusion Matrix Comparison of Back Propagation Neural Network

Additionally, the ROC Curves for the Back Propagation Neural Networks continued to follow the established pattern. Separately within each geography, the ranking of the algorithms was identical according to the AUC scores with Naive Bayes always scoring the best and the Back Propagation Neural Network always scoring the worst of the three algorithms.

Las Vegas | All-Cities | Montreal

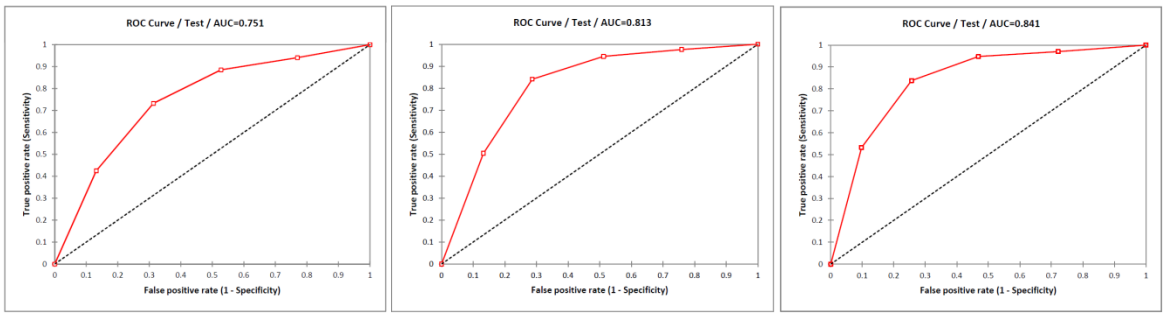


Figure 21 ROC Curve Comparison of Back Propagation Neural Network

Binning of the Star Rating Results

Once all of our results had been processed and analyzed, we reanalyzed our results taking binning or bucketizing into account. By grouping together all of the 1 and 2 stars into a single group and performing the same with 4 and 5 stars we hoped to see an improvement in our ability to predict the higher star ratings that our confusion matrixes continued to show that we had trouble with previously. Not surprisingly we did see a marked improvement in our results across the spectrum of different algorithms especially in our ability to predict the higher star ratings.

As the different confusion matrixes in following figure illustrate, our result in the Percent Off Diagonal dropped in each instance giving us an average improvement of close to 40% from our previous measurements. However, the binning did not change which the rankings of the different algorithms. Naive Bayes stilled scored the best of the three algorithms regardless of any binning we applied.

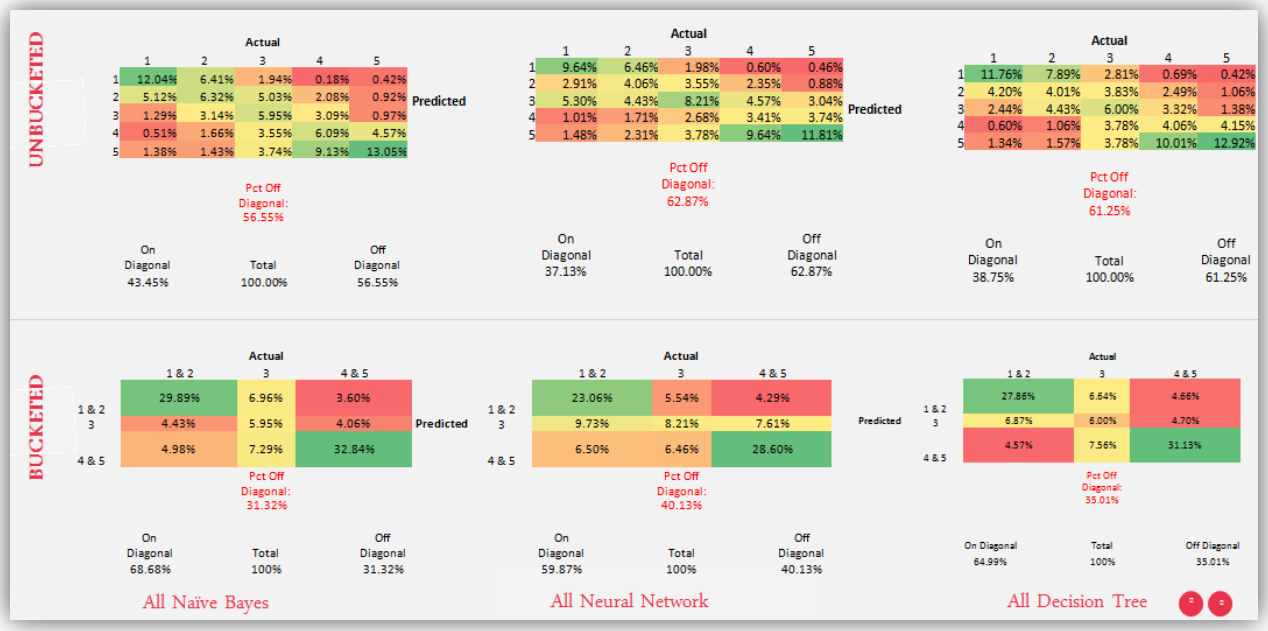


Figure 22 Comparing Results for Un-Binning vs. Binning

The results of the binning analysis confirmed the information presented in class (Lai, 2014) about a previous project against the Yelp Dataset in that the intermediate values (stars) are much harder to predict and that by binning the results we would gain a more accurate prediction.

VI Lessons Learned

Throughout the course of this project we encountered a number of different challenges that we had to learn from and eventually overcome. Probably the biggest challenge and most stark observation is the amount of time and effort that goes into planning, data preparation and cleansing prior to any analysis being performed.

A simple task of importing data into a database gets compounded and complicated when the data must first be converted into an understandable format by the target database. We discovered that we are at the complete control of the previous owners in charge of the dataset. Whatever problems that they have left behind in the dataset are now ours to first discover and then to resolve. This can turn into a very time consuming iterative process and can easily cause you to scrap any previous results as you must reanalyze all of your results with the most current (correct) set of data.

Additionally, we discovered that our results can be biased greatly by the skew that is naturally present within the dataset and this required us to devise schemes to partition and stripe the data so that all of the training data would be treated equally.

The last problem that confounded our team was narrowing the numerous possibilities to pursue for this project down to a small manageable problem space that could be contained within the timeframe that we had been allocated for the project. The Yelp Dataset provided an enormous number of possible problems and permutations that we could have attempted to solve and focusing on a single problem was harder than we originally anticipated.

Finally, upon completion of our project we discovered that there were still a number of possible problems, oddities and variations upon our solution that we could have continued to pursue. For instance we could have allowed other attributes within the dataset to help us score or weigh the reviews that we based our model upon. We could have also worked with different word sets within the dictionary to see how they might have affected the sentiment analysis and the overall results we observed. So the general observation that these types of data mining problems can easily turn into an on-going perpetual project was easily observable

VII Conclusions

We very consistently observed within our models across all of the algorithms and all of the different geographies that we were able to predict the negative star ratings with more accuracy than the higher star ratings. This led to the observation that we too were being affected by the *Off By One Star* issue that was observed by the previous Yelp student project discussed in our coursework.

Of the three algorithms that we chose to pursue Naïve Bayes performed the best against the models that we generated regardless of any variation taken into account for different geographies or cultural differences. This was a surprising result to us since we expected to see some variation because word choice differences in the English speaking city of Las Vegas and the French/English speaking city of Montreal.

Much of the similarity can be explained away in the results between geographies because we chose to use the standard (default) configuration file and dictionary for the sentiment analysis. A more divergent set of answers might have been possible if we configured the dictionary with more (geography) specific terms for the sentiment analysis.

VIII References

A list of references that have contributed to the preliminary ideas and topics for the project are included below. Many of these references have been used to help formulate the scope of the project as well as to how best proceed with analyzing the dataset that we have been provided.

DQ Analyzer. (n.d.). Retrieved March 2015, from Ataccama: https://www.ataccama.com/products/dq-analyzer

Huang, J., Rogers, S., & Joo, E. (2013). *Improving Restaurants by Extracting Subtopics from Yelp Reviews.* Retrieved March 2015, from Yelp Dataset Challenge Winners: http://www.yelp.com/html/pdf/YelpDatasetChallengeWinner_ImprovingRestaurants.pdf

Hung, K., & Qiu, H. (2014, Dec). *UCSD - Yelp Dataset Challenge 2014 Submission.* Retrieved March 2015, from http://kevin11h.github.io/YelpDatasetChallengeDataScienceAndMachineLearningUCSD/

Martin, T., & Kita, R. (2014). Retrieved March 2015, from Good Food Bad Service: http://www.goodfoodbadservice.com/

SAP. (n.d.). *SAP HANA Application Function Modeler.* Retrieved April 2015, from SAP Help Portal: http://help.sap.com/saphelp_hanaplatform/helpdata/en/29/de6754ef9646999b6261819bd802cb/content.htm?frameset=/en/93/b3e3191ae34508a4d92dff9b6d350c/frameset.htm&current_toc=/en/34/29fc63a1de4cd6876ea211dc86ee54/plain.htm&node_id=212&show_children=false

XLSTAT Statistical Software for Excel. (n.d.). Retrieved April 2015, from Addinsoft: http://www.xlstat.com/en/

Yelp Inc. (n.d.). *Yelp Dataset Challenge*. Retrieved March 2015, from Yelp: http://www.yelp.com/dataset_challenge

Yelp Inc. (2014, Nov). *Yelp's Academic Dataset Examples.* Retrieved March 2015, from GitHub Repository: https://github.com/Yelp/dataset-examples/blob/master/json_to_csv_converter.py

IX Source Code Repository

Copies of all source code and supporting materials have been uploaded to a public repository at GitHub with the URL of github.com/RDCooter/SEIS-734-Spring-2015.

…/data/JSON - Originals	Contains compressed (7-Zip) copies of the original Yelp Dataset in its JSON format. Each file is contained separately to make sure the files remain under the file size limit imposed by GitHub.
…/data/CSV - Loadable	Contains compressed (zip) converted copies of the original JSON dataset into a CSV format. GitHub does not allow larger files to be stored within their repository so this is not an inclusive collection of the tables.
…/data/ORACLE - DDL and Pre-Processing	Contains copies of SQL DDL and scripts to create and load the tables into an Oracle 11g database so simple data cleansing can be performed.
…/data/HANA -DDL	Contains copies of the SQL DDL statements to create the Yelp tables into a SAP HANA database.
…/data/YelpAcademicDataset	Contains information relating to the Yelp Academic Dataset that was downloaded from their website such as the Challenge Description; the Terms and Conditions; Usage Agreement and a JSON Data Model Description of the dataset as well as a quick ERD of the dataset.

…/src/AFM - Flowgraphs and SQL	Contains copies of SQL scripts used to setup the tables and views for processing into the AFM flowgraphs. Partitions and stripes the data based upon the geography and size desired for training and testing of the resulting model.
…/src/Python - JSON to CSV	Contains copies of the Python script to convert the original Yelp JSON data files into corresponding CSV files.

…/results/AFM - Screenshots and CSV Files	Contains copies of the resulting CSV files generated by the AFM as well as screenshots of the AFM interface within Eclipse.
…/results/Excel - Variance Distribution and Pct Off Diagonal	Contains copies of the AFM results in Excel spreadsheets that have been processed into Variance Distribution and Percent Off Diagonal graphs.
…/results/Preliminary Analysis	Contains copies of the preliminary analysis results performed against the Yelp Dataset through DQ Analyzer and other tools.
…/results/XLSTAT - Naive Bayes, Decision Tree, Neural Network	Contains copies of the AFM results processed through XLSTAT in order to generate the Confusion Matrix and ROC Curve graphs.

Appendix A Data Source / Selection

The data for our project has been gathered from the dataset made available by Yelp for their ongoing Dataset Challenge (Yelp Inc.). They have made this dataset available for research purposes for the past two years and it encompasses a detailed snapshot of their enormous database of business and user submitted reviews.

The current version (5) of the dataset includes a variety of different cities (10) across multiple countries (4). It includes the following highlights:

- 1.6 million reviews and 500k tips submitted by 366k different users for 61k businesses
- 481k business attributes (e.g. hours, parking availability, ambience)
- Social network of 366k users for a total of 2.9 million social edges
- Aggregated check-ins over time for each of the 61k businesses

Figure 23 Locations included in the Yelp Dataset

- A detailed description of each of the fields (attributes) for the different tables can be found in the Appendix attached to this document.

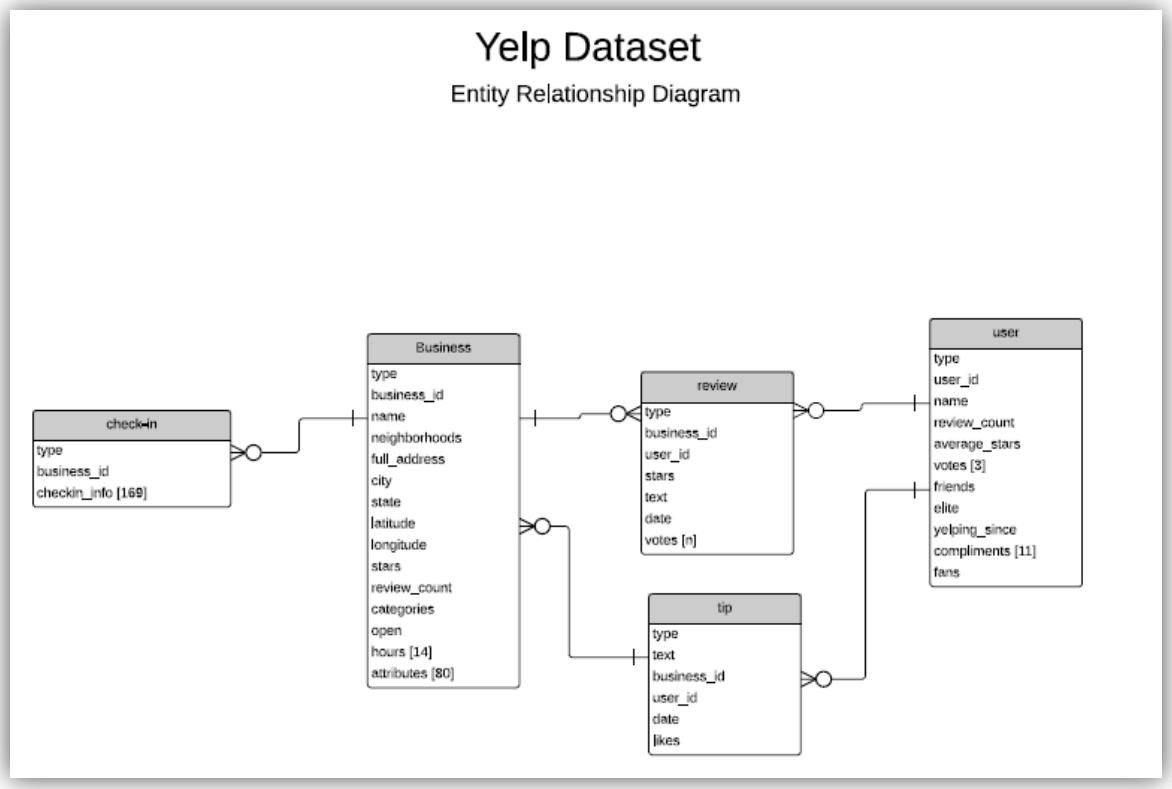


Figure 24 Simple ER Model for Yelp Dataset

Appendix B Yelp Academic Dataset — Business (yad_business.csv)

The *YAD_Business* table consists of 159,030 records that contain a description of various businesses that exist within the dataset and contain information submitted by various users.

Column Name	Column Description
type	Enumeration of the type of record contained within the table. For this dataset, this value will always be set to 'business'.
business_id	Encrypted identifier to uniquely identify the business associated with this record.
name	Name of the individual business.
neighborhoods	A comma separated list of neighborhoods that this business is commonly associated with.
full_address	The localized address of the business.
city	The city where a business resides.
state	The state where a business resides.
latitude	Latitude coordinates of the business location.
longitude	Longitude coordinates of the business location.
stars	A numeric star rating associated with the business. Rounded to the nearest half-star.
review_count	A numeric count of the number of *reviews* associated with this business.
categories	A comma separated list of different categories that this business is associated with.
open	Enumeration of the current state of the business [True or False]. False represents that the business has permanently closed.
hours[14]	An enumeration of the day of the week along with the time the business opens and closes each day. There are currently 14 types represented by this dataset.
attributes[80]	A True/False enumeration of the various attributes used to describe a business within the Yelp dataset. There are currently 80 types represented by this dataset.

Table 1 YAD_Business.csv

Appendix C Yelp Academic Dataset — Review (yad_review.csv)

The *YAD_Review* table consists of 6,304,124 records that contain the information about different reviews contributed by the users for the various businesses that exist within the dataset.

Column Name	Column Description
type	Enumeration of the type of record contained within the table. For this dataset, this value will always be set to 'review'.
business_id	Encrypted identifier to uniquely identify the business associated with this *review*.
user_id	Encrypted identifier to uniquely identify the user who submitted this *review*.
stars	A numeric star rating associated with the review. Rounded to the nearest half-star.
text	Text of the response (*review*) left by the user.
date	Date when the review was originally submitted. Formatted in YYY-MM-DD style.
votes[*n*]	An enumeration of the type of vote submitted as well as the numeric count of the number of times that users voted for the same value. There are currently *n* types represented by this dataset.

Table 2 YAD_Review.csv

Appendix D Yelp Academic Dataset — User (yad_user.csv)

The *YAD_User* table consists of 366,716 records that contain a description of various users who have registered and submitted reviews and tips about a number of different businesses.

Column Name	Column Description
type	Enumeration of the type of record contained within the table. For this dataset, this value will always be set to 'user'.
user_id	Encrypted identifier to uniquely identify the user within this dataset.
name	First name of the user.
review_count	A numeric count of the number of reviews submitted by this user.
average_stars	A numeric (floating point) average of the number of stars that this user has submitted.
votes[3]	An enumeration of the number and types of votes that this user has submitted. There are currently 3 types represented by this dataset.
friends	A comma separated list of user_id's that are friends with this user.
elite	A comma separated list of years that this user has achieved Elite status.
yelping_since	Date when the user first created their account on Yelp. Formatted in YYY-MM style.
compliments[11]	An enumeration of the number and types of compliments that this user has received. There are currently 11 types represented by this dataset.
fans	A numeric count of the number of *fans* that this user currently has.

Table 3 YAD_User.csv

Appendix E Yelp Academic Dataset — Check In (yad_checkin.csv)

The *YAD_CheckIn* table consists of 45,167 records that contain a description of various users who have registered and submitted reviews and tips about a number of different businesses.

Column Name	Column Description
type	Enumeration of the type of record contained within the table. For this dataset, this value will always be set to 'checkin'.
business_id	Encrypted identifier to uniquely identify the business this *check-in* associated with this record.
checkin_info[169]	An enumeration of the date and time when the check-in occurred at the business. There are currently 169 types represented by this dataset formatted in [HH-DAY#] style.

Table 4 YAD_CheckIn.csv

Appendix F Yelp Academic Dataset — Tip (yad_tip.csv)

The *YAD_Tip* table consists of 495,108 records that contain various user supplied tips about the different businesses contained within the Academic Dataset. Other users can review these *tips* and indicate that they *like* its contents and found them informative and useful.

Column Name	Column Description
type	Enumeration of the type of response left by the user. For this dataset, this value will always be set to 'tip'.
text	Text of the response (*tip*) left by the user.
business_id	Encrypted identifier to uniquely identify the business this *tip* was left for.
user_id	Encrypted identifier to uniquely identify the user that submitted this *tip*.
date	Date when the tip was originally submitted. Formatted in YYY-MM-DD style.
likes	A numeric count of the number of times this *tip* has been seen and liked by different individuals.

Table 5 YAD_Tip.csv