BizTrender - CS526 Final Project (Group 9)

Abhishek Bhatt (NetID : ab2083)
Praneeth Chandra Thota (NetID : pt357)
Chaitanya Sharma Domudala (NetID : cd817)

ABSTRACT

Over the past decade, there has been a great advancement in algorithms and distributed systems that allow retrieval, transformation and management of huge volumes of data from varied sources, with different frequencies and granularity levels. On the other hand, industries have realized the importance of data-driven decision making. Businesses across all sizes and segments, have started utilizing data mining and visual analytics, to mine insights from data like past sales, product pricing and customer behavior that was not considered valuable earlier. This is significant since the trends discovered from these data sets are crucial in aligning business strategies to maximize revenues by organizations.

This has led to two major developments. Firstly, there is a need to streamline massive data across multiple business units and channels (data *in-silos*), define entity relationships amongst them and cleanse it for storage as per a well-defined data model in a centralized data lake. Distributed frameworks and scalable databases like Spark, Kafka, Elasticsearch, Hive and MongoDB make this data engineering process feasible. Secondly, there is increased collaboration between automated data mining and human decision making to optimize end results. Large scale data pipelines compute and produce results on top of the data lake, while analysts leverage this *business intelligence* to discover patterns, predict market trends and make accurate decisions. This is a remarkable shift from conventional data analytics for small-sized data, where all analysis and reporting was done by analysts manually using tools like MS Excel.

In this work, we describe a search and visualization solution to streamline the data of various businesses listed on *Yelp* and discover trends on top of it. Yelp is a business directory service which publishes crowd-sourced reviews about businesses through their website and mobile application. We build an end-to-end data pipeline to ingest, transform, search and aggregate publicly available data sets from the *Yelp Dataset Challenge* on the backend, and display useful information or visualize key trends on a GUI frontend. The user driven system enables interactivity between data and analysts, assisting the latter to draw insights from the data and make accurate business strategy.

KEYWORDS

Visual Analytics, Business Intelligence, Yelp

1 INTRODUCTION

The proposed solution, *BizTrender*, is a web application. Its backend comprises of raw data ingestion into a centralized data lake, and REST endpoints used to query or aggregate results from the data lake when triggered. A User Interface consumes the output served by the REST APIs, to display search results or plot interactive visualizations of key metrics. The UI is interactive and driven by user inputs or mouse actions for updates.

The motivation to build this solution came after going over the work described in [5]. Below we elaborate on the various aspects of the project design: data, problem statement, mode of processing, visual representation and interactivity, trends discovered and future possibilities.

2 THE DATA

We build this solution on top of the publicly available Yelp data set[7], from the The Yelp Dataset Challenge. The data contains a subset of businesses listed on Yelp website or mobile application, besides data on users, visits, user ratings and reviews.

For this project we used 5 JSON files from the dataset[6] - business.json, review.json, user.json, checkin.json, tip.json. The files add up to 8.6 GB of data when uncompressed. The Yelp Dataset Challenge is a yearly challenge. As a result, the data set is usually updated once a year for each new contest. Thus, considering the low frequency of updates, we take the data to be *static* for our development. The entities present in the data are described briefly below:

- business.json : This file contains attributes that define a business, for example, its category, location, rating, current active status and business hours. This serves as the *master data* for defining the entity relationships. Each business is uniquely identified by a business_id field.
- review.json : The file contains text reviews on a business by users, including star rating, votes on the review and the date when the review was added. Each business is linked to a review by business_id, while a review and a user are linked by a user_id field.
- tip.json : This file contains quick suggestions written by a user for a specific product or service within a business. This is different from a review in the sense that it does not apply on the business as a whole. Each record also contains date and the number of compliments received on each tip.
- user.json : This file contains user metadata that is identified by a unique user_id field. Each record also defines *friends* of a particular user, which are other users.
- checkin.json : It contains check-in (*visit*) timestamps for a particular business. These records are indicators of customer traffic for a particular business over a given period of time.

All the entities described in figure 1 are identified by unique IDs which can be used as keys for joins and aggregations. Time is a crucial component to visualize trends in check-ins and reviews (user sentiment) over time.

3 THE QUESTIONS

On a high level, the problem statement and the use-case we want to address can be formulated under three questions as discussed below.

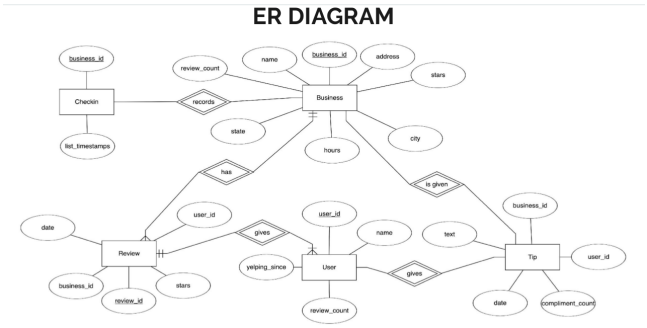

Figure 1: Entity Relationships Diagram

- WHAT? : BizTrender is a web application that provides a centralized platform to search and perform visual analytics on businesses listed on Yelp. It is important to note here that this is not an automated data mining solution. For example, a recommender system, that is capable of taking a decision to suggest items to users based on their activity. BizTrender on the other hand, supports the *human* decision making process in a data-driven manner. Hence, it leverages the perception and cognitive abilities of an analyst to drive end results.
- WHY? : To understand the motivation behind building such a platform, we need to make a clear distinction between a *customer* and an *analyst*. The Yelp website and mobile application do a great job of helping *customers* choose the right business for them in terms of location, category and quality. For example, while choosing a restaurant to eat out, users can view all the restaurants near them, see the ratings and reviews, and find specialities of that restaurant. Accordingly, they can decide which one to visit. However, our solution is targeted for business *analysts or consultants* who would like to study the performance of a business or group of businesses, with respect to various metrics, identify correlations that drive those performances, and infer trends to devise business strategies. It is important to understand this difference, as the results derived from BizTrender might not mean anything for a person who simply wants to decide where to order Pizza from. At the same time though, it can help analysts save or make many dollars for businesses. The term *user* used henceforth in this work shall imply a business analyst or consultant, unless otherwise stated.
- HOW? : We propose two components in the application design to help address the use-case discussed above. One is search functionality, where a user can type in a search query and get all *relevant* businesses in the result. This provides a starting point and defines a *domain of analysis* for the analyst. The second component is visualizations and exploring trends through them. The various plots help observe any relationships between different entities, comparisons between businesses or groups of businesses for a metric, and user behavior/sentiment against time or average. Various insights can be derived through these visualizations that are summarized towards the end of this report. The visual analytics on the dataset is performed at both at a business level and over the whole dataset. Both the search and visualization components are connected to a centralized data-lake on the backend. The application draws data on-demand from the data-lake to display results on the front-end.

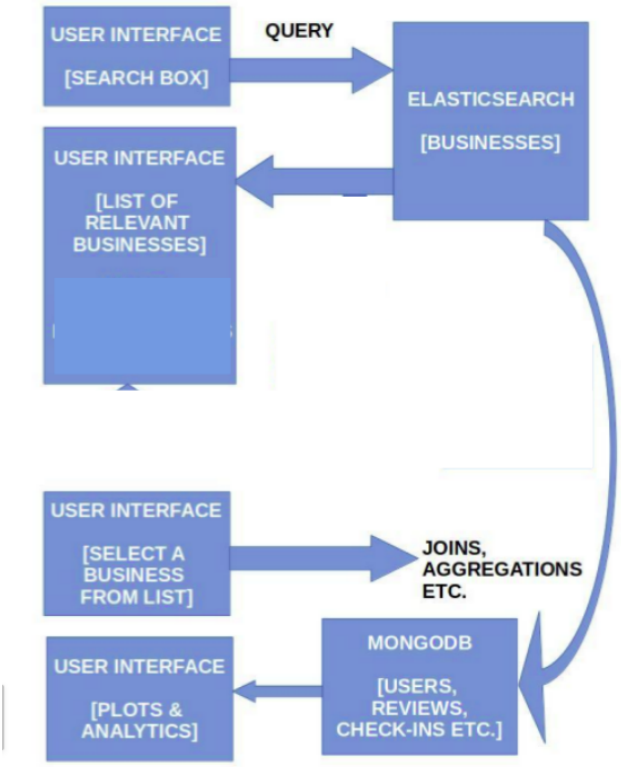

Figure 2: BizTrender Conceptual Flowchart.

3.1 Users

As highlighted previously, the target users of the application are primarily business analysts or independent consultants advising businesses on strategy and planning. The solution is also useful for small businesses having owners themselves at the helm of planning and management. Lastly, since the insights drawn from BizTrender are also useful for planning the launch of new businesses based on certain metrics, it can be a platform of interest for real-estate consultants.

3.2 Data Representation

The raw data set is in JSON format. The master data, i.e. businesses, is represented as an index on *Elasticsearch v7.6.2*[1], for the search functionality. The search functionality allows a user (analyst) to type, say any location or business name, in a search bar and retrieve all relevant results. The other data attributes are represented as *facts* and *dimensions* stored in a database, which can be pulled to join with the master data and perform any aggregation/transformation for visualization purposes. This is the case when the user selects one of the search results above for detailed view or analysis, or visits the overall *Trend Summary* page. To maintain the scalability of the

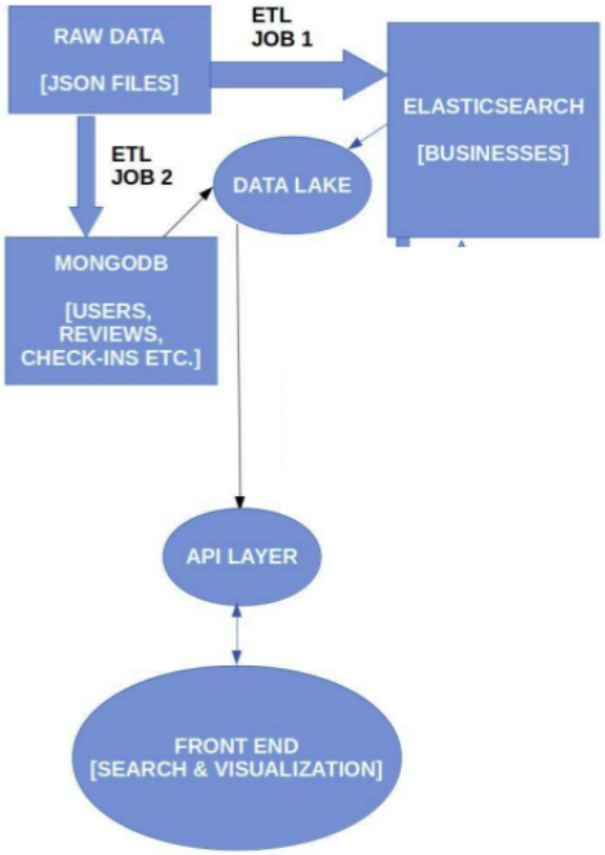

Figure 3: BizTrender Data Pipeline.

solution for larger data sets in future, and in compliance with the JSON format, we used *MongoDB v3.6.17*[4] as the database to store facts and dimensions besides the master data. These operations and the usage of the application are shown in figure 2.

4 MODE OF PROCESSING

The data pipeline for BizTrender is shown in Figure 3. Our main goal while processing the data on the backend is to minimize the time taken to answer queries from the instant they are triggered from the UI. This can be achieved by minimizing the time taken to query the database, and by minimizing the time taken to return the response of the API calls. This is crucial for giving spontaneous results based on user interactions on the UI.

The raw data did not require any pre-processing, as it did not contain any garbage or missing values in the required fields. The business.json file was loaded to an Elasticsearch index using the Elasticsearch Bulk API, implemented through a Python script. All JSON files were loaded to MongoDB using mongoimport bash commands. The data processing mechanisms are summarized below:

- Raw data : JSON files.
- Data ingestion : Elasticsearch Bulk API (Python) and Mongoimport (Shell).

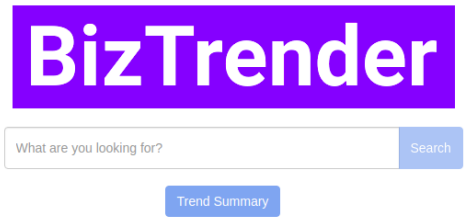

Figure 4: BizTrender Landing page.

- Data Lake : Elasticsearch and MongoDB.
- Database indexes created for faster queries on ID fields of each collection, using Mongo shell commands.
- Calls to REST endpoints from UI for data movement between data lake and visualizations. The REST API queries, filters or aggregates *static* data in the data lake.
- Caching : API responses, that are not too large in size, are cached in MongoDB GridFS. This maintains a history of past requests, and retrieves the response for an API call that was also triggered in the past from the cache. Caching helps reduce the number of open sessions created with the database and faster resolution of queries.
- Streaming HTTP Response Body : API responses, that are too large in size (order of 40000 records or more), are streamed to the UI rather than a single chunk of results. As a result, the user gets spontaneous result on the UI in the form of few data points, while rest of the points keep adding incrementally. This is very significant to maintain user engagement in the application.

4.1 Software Stack

- Backend : JAVA Spring Boot, MVC Architecture
- Frontend : Javascript, AngularJs, Highcharts[2] (includes built-in features for saving/exporting visulizations)
- Scripting/Commands : Bash, Mongo shell, Python
- Deployment : Localhost; some experimental attempts to use AWS EC2 and Elaticsearch Service

5 VISUAL REPRESENTATION

Figure 4 shows the home page for the web application. From here the user has the option to type in a search query in the search box to get *specific* results, or visit the Trend Summary page which contains *generic* insights on the overall dataset. The generic to specific navigation manifests a hierarchical abstraction in the system.

The Trend Summary page provides two filters in the form of drop-down menus (figures 5 and 6) that allow the user to control the amount of data that should be sent to the screen to render the visualizations. By default, the first options in each filter are selected.

Based on the selection of businesses through the categories and/or ratings filters, the figures 7 and 8 show visualizations of

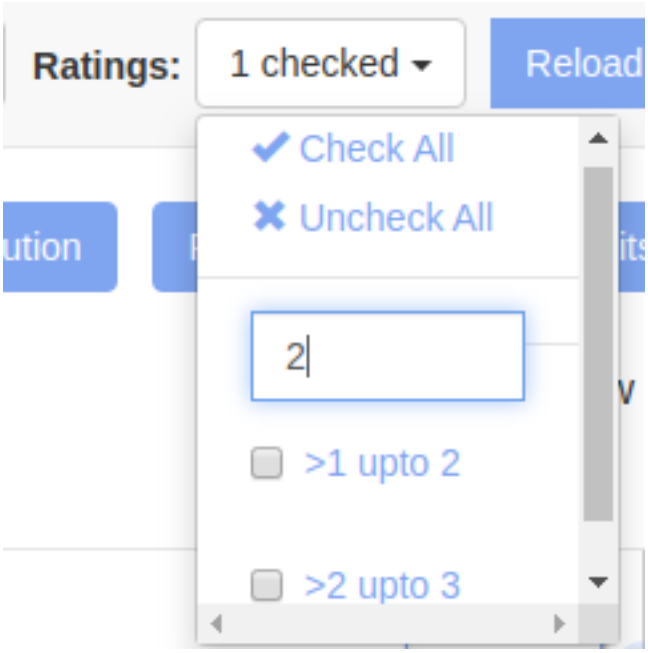

Figure 5: BizTrender Filtering by rating.

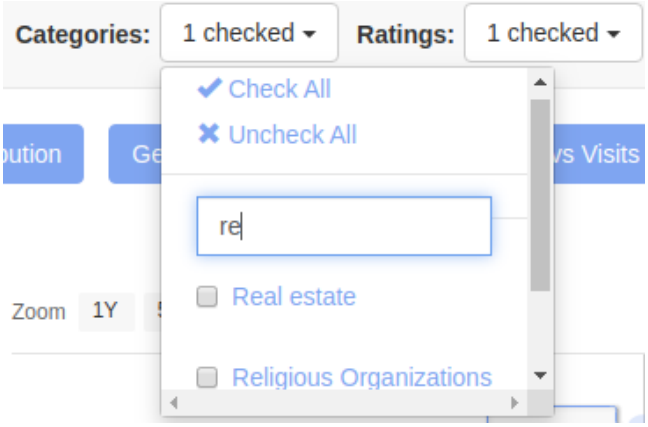

Figure 6: BizTrender Filtering by categories.

Figure 7: BizTrender Rating distribution by category.

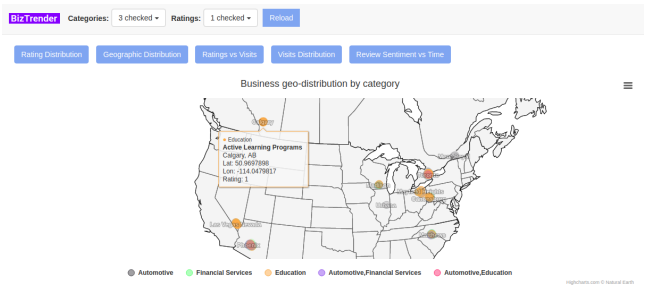

Figure 8: BizTrender Geographic distribution of businesses.

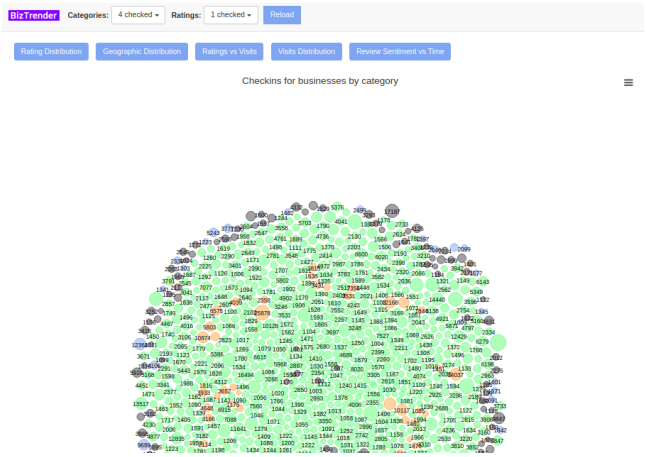

Figure 9: BizTrender Visits by categories.

how ratings are distributed across categories, and geographical distribution of the businesses respectively.

In figures 9 and 10, businesses are represented by bubbles while their color identifies their category. The area of each bubble is proportional to the visits received by the business it represents.

In figure 11, the visits received for all the selected businesses (based on the filter selection) are distributed by day of the week. For the same selection of filters, visits distribution by hour of the day is shown in figure 12.

In figure 13, we visualize the review sentiment score of reviews against time. We query for all the reviews that were given to businesses selected using the filters. Each review has a star field that has a value from 1 to 5. We baselined it to zero, which represents a neutral sentiment. Scores above zero imply a positive sentiment while a negative score corresponds to a negative sentiment. The extent of positivity or negativity is quantified respectively, by how higher or lower the value is from the neutral line (zero).

From the landing page of the application, user can also perform a search by typing in a query. The returned search results are displayed as in figures 14 and 15. The user can navigate across multiple pages of results using the links at the bottom. By default, the first result is selected, and its details displayed on the left pane in figure 14. User can change the selection by a single click on any business.

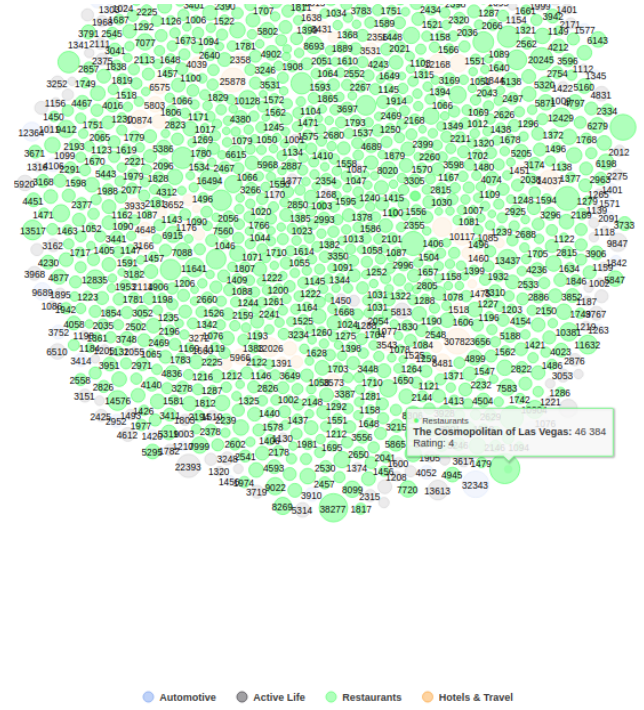

Figure 10: BizTrender Visits by categories.

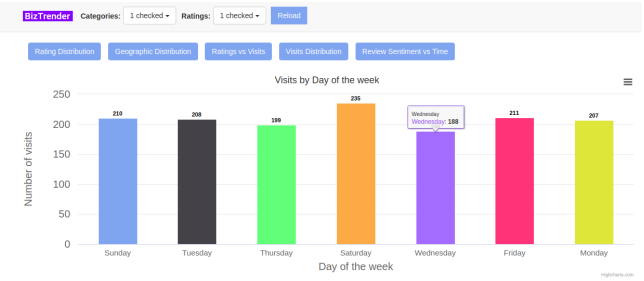

Figure 11: BizTrender Visits distribution by day.

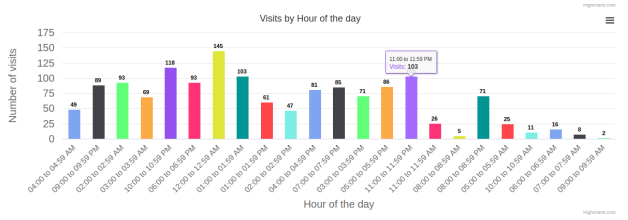

Figure 12: BizTrender Visits distribution by hour.

Double clicking on a business opens a popup window with insights visualized for that specific business. The first view in the popup window where the user lands is shown in figures 16 and 17.

Figures 16 and 17 show the distribution of review sentiment for the selected business, both in terms of actual data points and frequency distribution (histogram) respectively.

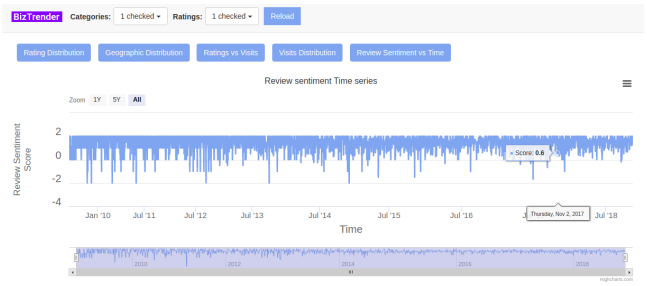

Figure 13: BizTrender Time series of review sentiment.

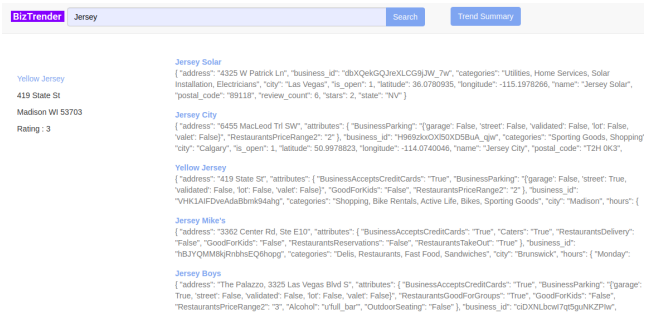

Figure 14: BizTrender Search results page.

Figure 15: BizTrender Search results page.

Figure 16: BizTrender Distribution of review sentiment for a business.

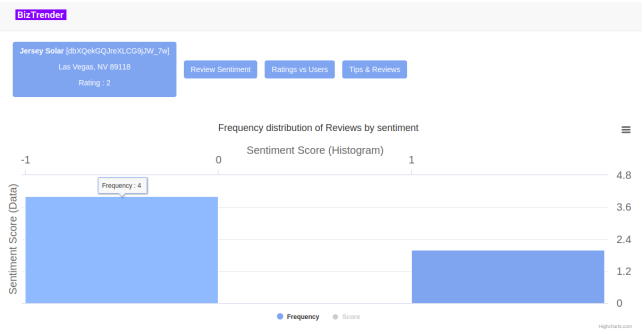

Figure 17: BizTrender Distribution of review sentiment for a business.

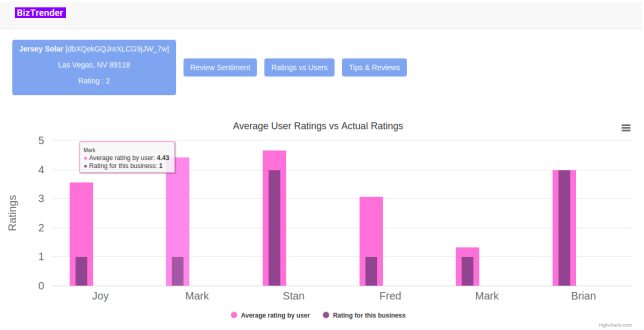

Figure 18: BizTrender Average vs actual rating by users for a business.

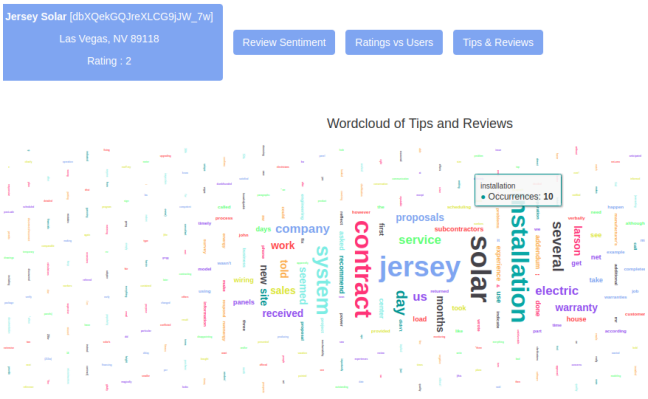

Figure 19: BizTrender Tips and reviews word cloud for a business.

The ratings given by users for the selected business are compared against their average ratings in the plot displayed in figure 18. For this specific example, it can be clearly seen that the four negative reviews shown in figure 16 correspond to users Joy, Mark, Stan and Fred in figure 18, who rated this business significantly below their average ratings.

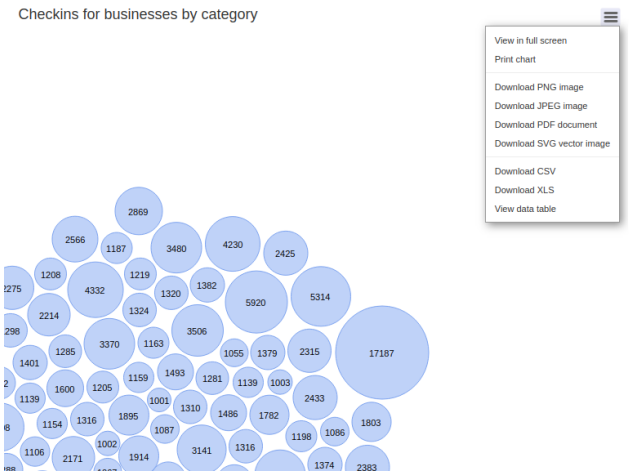

Figure 20: BizTrender Built-in options.

Lastly, figure 19 shows the word cloud for all the reviews and tips written for the selected business. The size of a word is proportional it its frequency across the tips and reviews.

The visual metaphors used in the project are summarized below.

- Use of standard UI fonts, sizes, color and texture palettes.
- Elements : Search box, click buttons, dropdown menus, unordered lists, navigation buttons, canvas divisions for plots.
- Interactive plots: 3D column chart with stacking and grouping, map[3] (US and Canada) with latitudes/longitudes, packed bubble chart, dynamic column charts, dynamic time-series line chart, histogram, fixed placement column chart, and word cloud.
- Multi-colored labels and legends.

6 INTERACTIVITY

BizTrender is a user-driven, interactive system. The results are updated/adjusted based on user inputs and selections. The goal is to achieve minimum response time for the user. The solution depends on a combination of human intelligence and data engineering techniques, to facilitate decision making, based on the discovered insights. Key interaction mechanisms used in the project are summarized below.

- Hierarchical abstraction : Generic (trend summary) and specific (search) views.
- Interface interaction mechanisms :
  - Bidirectional vertical sliders.
  - Both textual and graphic representations for answered user queries.
  - Mouse clicks, mouse hovering, mouse selection and their combinations.
  - Menu driven plots and tabs.
  - Mouse controlled zooming and panning.
- Built-in options for full-screen view and exporting plots (figure 20).
- Different views linked through navigation using mouse clicks.

7 INSIGHTS

Some key trends discovered in the dataset using the BizTrender application are listed below. The solution is flexible in the sense that it allows discovery of similar trends based on the categories of interest of the user.

- The largest number of businesses in the dataset are either restaurants (approximately 23000) or food-based (approximately 20000). Shopping centers come next at approximately 16000. This is followed by Home Services (approximately 8000), Nightlife or, Beauty and Spas (approximately 7800), Health and Medical (approximately 7600), and Local Services (approximately 6500). Active Life, Event Planning and Services, and Automotive each have between 4000 to 6000 businesses. Professional Services, Arts and Entertainment, Education, and Hotels and Travel are next in order with approximately 2000 to 3500 businesses each. Pets and Financial Services are the next categories with 1000 to 1500 businesses in each. Lastly, the categories in the dataset, with fewer than 1000 businesses under them, are Religious Organizations, Public Services and Government, Local Flavor and Mass Media.
- Religious organizations is the only category that has no business with rating 1 (least possible rating).
- For any category, most of the businesses are rated between 2 and 4 (on a scale of 1 to 5). It is clear that the ratings center heavily around 3 or the *middle* of the rating scale. This is followed by number of businesses rated between above 1 but upto 2.
- For each category, the number of highly rated (greater than 4) businesses is significantly smaller than the total number of businesses in that category that are rated above 1 or below 4. However, the smallest number of businesses in each category are the ones that are rated the worst at 1.
- In terms of geographic distribution, the dataset is sampled in such a way that all businesses lie in one of the clusters within following states or provinces - Alberta, Quebec or Ontario in Canada, and Nevada, Arizona, Illinois, Wisconsin, Ohio, Pennsylvania, North Carolina or South Carolina in the USA. Thus, the insights drawn from the visual analytics on this dataset should not be assumed or generalized for the whole country.
- It was also observed that many businesses with lower ratings have higher number of visits than other businesses with higher ratings. We therefore conclude that higher rating does not imply more visits for any business.
- For poorly rated (1 to 2) restaurants, the number of visits across the week is comparable, with no significant difference between weekdays and weekend.
- For average or highly rated (above 2) restaurants, the number of visits during weekend is higher than during weekdays. However, the number of visits during weekdays only is comparable.
- Religious organizations receive the highest number of visits on Sunday, which is significantly higher than the visits received on any other days. The number of visits on days other than Sunday are comparable.
- Across all categories, it is observed that businesses receive more visits during late evening or early morning as compared to daylight hours. However, this should be left as an open question instead of assuming to be a generic insight, since it is possible that the dataset was sampled in a way that introduced this bias.
- For poorly rated restaurants (1 to 2), the review sentiment is mostly negative with few positive scores in between. Also, during the initial years, the sentiment fluctuates between positive and negative a lot. Over time as more reviews come through, the sentiment time series, however, tends to stabilize around a negative value below zero (neutral).
- For middle-rated retaurants (2 to 3, the review sentiment is mostly centered (roughly symmetrical) about zero (neutral), with frequent fluctuations above and below it.
- For restaurants with good ratings (above 3), the review sentiment is mostly positive with few negative scores in between. Also, during the initial years, the sentiment fluctuates between positive and negative a lot. Over time as more reviews come through, the sentiment time series, however, tends to stabilize around a positive value above zero (neutral).
- For a specific business, the number negative reviews corresponds to the number of users rating the business below their average user rating.
- For a business with fewer tips and reviews, most of the words in the word cloud occur repetitively across those tips and reviews. On the other hand, in the word cloud for a business with more number of tips and reviews, there are only few repetitive words. Instead in this case, we find that there is a large variety of words with comparable frequencies.

8 CONCLUSION AND FUTURE WORK

Through BizTrender, we successfully demonstrated how visual analytics can be used to support the process of mining trends in a large dataset. We also used many visual metaphors, interaction mechanisms and abstractions learnt in class to build the final project.

Some future possibilities and updates that will further enhance the solution are listed below.

- Analyzing user-user relationships to discover insights into how one user's ratings affect other user's ratings.
- Adding user profiling and security to the application, with ability to assign different levels of visibility to each user.
- Adding location based data like population, demographics, income etc., and analyzing them against existing dimensions to discover more interesting trends.
- Make the solution generic by giving the user the ability to add data from multiple sources, so that the centralized datalake is further enriched and complex insights can be discovered.

9 ACKNOWLEDGEMENTS

We express our gratitude to Prof. James Abello, who instructed the 526 course in a very elaborate manner. We learned useful visual analytics tools and techniques in this course. Such takeaways from the class were crucial while building this solution.

We would also like to thank the course Teaching Assistant Mr. Fangda Han and the course Grader Mr. Nishant Kumar, for their useful review and feedback on our assignments as well as the course project, throughout the semester. We could only take BizTrender to its current deployable state because of multiple phases of feedback and peer-reviews.

REFERENCES

[1] Elastic. 2020. *The heart of the free and open Elastic Stack.* https://www.elastic.co/elasticsearch/
[2] HIGHCHARTS. 2020. *Highcharts Demos.* https://www.highcharts.com/demo
[3] HIGHCHARTS. 2020. *Highmaps Demos.* https://www.highcharts.com/maps/demo
[4] mongoDB. 2008. *The MongoDB 3.6 Manual.* https://docs.mongodb.com/v3.6/
[5] Sayantan Satpati. 2016. *YELP BUSINESS INSIGHTS.* http://people.ischool.berkeley.edu/~sayantan.satpati/yelp/
[6] Yelp. 2020. *Yelp Dataset JSON.* https://www.yelp.com/dataset/documentation/main
[7] Yelp. 2020. *Yelp Open Dataset.* https://www.yelp.com/dataset