Spark Practical Work – Report

Institution: UPM
Lecture: Big Data
Due Date: 21st January 2018

Authors: ECHAVARRIA CABALLERO, LADY CAROLINA
HUSSNAETTER, MICHAEL

Group: 22

Workflow implemented

1. Dataset Analysis
2. Problem Characterization
3. Data Task and abstraction
4. Interaction and visual encoding
5. Algorithmic implementation

Dataset Analysis

The selected dataset corresponds to the Yelp Dataset, which is a service that allows users to review different business. The Dataset itself is divided into different datasets:

- Business
- Review
- User
- Checkin
- Tip
- photos

Problem Characterization

Issues of the application domain and end users involved.

The main use of the platform is to consume information, but we are facing with two different users:

1. The first user group wants to know which one is the best restaurant to go in a certain location? When should users go to that selected restaurant regarding to the number of check-ins it has? Has the place improved according to the number and sentiment of the reviews? Those end users want to know which restaurant to select according to the location and the rates. Then, see the number of check-ins per hour to select the best time to visit the place, and see the amount of reviews per year for that specific place to ensure or reject the selection according to the rating in the reviews, which are directly related to the quality the restaurant.

2. The second users are those concerned by human behaviour or interested in marketing strategies, and are be interested in knowing the relationship between the number of stars and the amount of reviews. The initial hypothesis is that users are more likely to post reviews in an “extreme” emotional state. I.e. when are extremely satisfied or extremely

disappointed. Those users are also concerned about the detailed review ratings by state, showing the percentage of each share.

Data and Task abstraction

Basically is why the visual analytic tools are used for?

Given that we are facing with two different kinds of users, as mentioned previously:

In the first case, the visualization tool is used to consume information about restaurants, in order to select the best place to go according to the users preferences.

Task – Identify the tasks required by end users in their workflow

- Explore/search the restaurants according to the location in the map and number of stars.
- Filter restaurants in: Free WIFI availability, if takes reservations, take out and caters
- Select the ideal time to visit the restaurant according to the number of check-ins in a day of the week and the hour
- Explore the amount of reviews in a period of time according to the number of stars (rates), in order to see the behaviour of the user's comments and conclude about the improvements of service on the place over the time
- Select a place according to gathered information

Data – Determine the representation that best fits user's needs

- Location of restaurants in a map, using coordinates and a map server
- Filter restaurants using check boxes according to WIFI, etc
- Show plots with number of check-ins per day of the week and hour
- Show plots with number of reviews per year and number of stars

In the second case, the visualization tool is used also to consume information about restaurants, but focused on the behaviour of the reviews with respect to the rate of the place (or number of stars) and the average ratings per state.

Task – Identify the tasks required by end users in their workflow

- Visualize the number of reviews per business according to the average rating scores.
- Explore the average review ratings by state and visualize more detailed proportion of review ratings by state

Data – Determine the representation that best fits user's needs

- Plot the amount of reviews versus the rating scores.
- Distribution grid of average rating score per state

Interaction and visual encoding

Determine the specific design choice for creating and manipulating the visual representation of the abstract data.

As a first view, users will see in the tab "Look for a restaurant", where the identity channel is related to the main objective which is to provide information about identity and location of a business. Furthermore, using a heatmap representing the number of check-ins addresses the identity as well as the magnitude channel. All of these layers are shown on top of a cartographic arrangement of the

restaurants grouped by location, with leaflet as a Mapping Library proving a world map as background.

- The Heat map shows the number of check-ins over the map, and gives to users an overview of the number of check-ins distributed on the area, which is related with the most or less visited areas, and it will help the customers to select an area according to his preferences. Each number of check-ins is represented by two-dimensional marks to which has been assigned a sequential two-colour scale that goes from yellow to red: yellow represents areas with less check-ins and red the areas with more check-ins. A sequential colour scheme was chosen, since it serves the representation of magnitude.

- The clusters of restaurants are shown using area marks as circles with colours green, yellow and orange expressing how many restaurants are grouped: In the case of orange circles, clusters with more than 100 restaurants are shown; yellow circles represent clusters between 10 and 100 restaurants and green, cluster with less than 10 restaurants. The restaurants are grouped with the purpose of having a general idea of the location of restaurants without annoying the users with a big amount of points located in the map. Every time users zoom in on a certain area in the map, the circles are updated with more detailed clusters, obtaining at the end the single restaurants seen as point marks.

Given that the first kind of users (mentioned in the section Problem Characterization) want to select a restaurant that fits their interests, in the described first view, the customers are able to change the restaurants they are seeing on the map by filtering of the data according to: The number of stars from 1 to 5, due to some users want to select restaurants with an specific rating score; free WIFI availability, if takes reservations, take out and caters. Once the filter is done, the map is updated with the new amount of restaurants and the right panel will also show the number of restaurants in the current view from the total amount of restaurants.

Once the users have chosen a restaurant, by clicking on the selected location, they will be able to obtain detailed information about the place:

- A bar chart was chosen to visualize the number of check-ins as quantitative value attribute and the hour as categorical key attribute. Because of the users need to select the day and time to visit the restaurant according to the amount of visits, the number of check-ins is a good indicator since it has a close relationship with the amount of visits per hour.

- To lookup the amount of reviews per year, a stacked bar chart was chosen, because it allows consulting values accessing with: two keys which are year and number of stars; and one quantitative attribute equivalent to number of reviews. The secondary key, in this case the number of stars, is used to build the one-dimensional vertical structure on the bars, that shows the part-of-whole relationship between the number of stars and the amount of reviews per year. The information offered by this plot, gives to users an idea about the behaviour of customer perception over the time, and helps the users to ensure or reject the selected restaurant.

- A normalized stacked bar chart stretches each rating scores on the stack to the maximum possible length, in this case 5, representing percentages instead of absolute values as the stacked bar chart explained before. The current plot helps users to understand the contribution

of each rating scores or number of stars to the whole numbers of reviews, over the time. And tries to eliminate the fact that yelp.com became more popular and consequently the number of reviews increased.

For those users concerned by human behaviour or interested in marketing strategies, mentioned in the section “Problem Characterization”, the second tab of the visual analytic tool, called “Inspect dataset” presents general information:

- To visualize the behaviour of the reviews with respect to the average rating scores, a scatter plot was chosen because it is very effective to characterize the distributions encoding the variables using the horizontal and vertical spatial position channels to express the number of stars and number of reviews, respectively, with point marks.

- A grid plot showing the detailed information about the amount of reviews for each state, according to the number of stars, through the magnitude channel to provide quantitative information using area marks and luminance colour: the proportion is shown with circles, where the size of the circle represents the percentage of reviews, and the colour gradient represents the average score. In this distribution grid of average rating score, each column indicates a proportion of each score by state while each row indicates a proportion at a specific review score in each state.

The information derived from the visual analytic tool becomes a tool for experts to analyse and make conclusions about the behaviour of reviews according to the average rating score and state.

Algorithmic implementation

Efficient implementation to achieve what was designed in the previous steps.

General objectives

The first step for generating a quickly starting and reasonably fast responding application is pre-processing the data such that the format is quickly processable for the desired tasks. Hence, the input data should contain only relevant data which is used in the app. Considering this maxim the size of the relevant json files (business, checkin and review) was converted and compressed from approximately 4 GB to 100 MB. In the upcoming sections “*.dat” files always denote a pre-processed file.

Mark and cluster available businesses at world map

The businesses are filtered according to the user’s selection (Stars, Free-WiFi, Take-Out, Cater, Takes Reservations). Then for each entry left in business.dat a marker is created on the map and depending on the zoom level the markerClusterOptions() function of leaflet is used to cluster the markers. Note, that all updates for the markers are done only for the currently visible restaurants to save compute time.

Number of Check-ins per business heatmap layer

The heatmap layer plots at the latitude and longitude of every business by the number of visits encoded by color. All three values are easily looked up in the pre-processed business.dat and no further calculation must be done. This is possible as “num_checkins” column was added to the business.dat to save query time. Furthermore, the heatmap is always updated only considering the restaurants which are currently visible on the screen. Therefore, position and zoom-level of the map are exploited to increase performance. Additionally, a reactive legend is displayed to show the user the current color mapping of the number of check-ins.

Check-ins per hour of a selected business bar chart

This bar chart visualizes the checkins.dat which was flattened in the pre-processing step to have a single row containing all information required to plot this chart. This means that the checkins.dat has 24*7 + 1 = 169 columns. One for the business_id as index and all the others storing the number of check-ins for each hour of the week.

Reviews-by-year stacked bar chart

To create this chart the most important value is the amount of reviews with a certain rating for a selected business by year. This value is received by first filtering reviews.dat for the selected business, then grouping the data by rating and finally counting the number entries for each rating by year. To increase the performance a little the “year” column replaced the date column in review.dat as the full date was not required for the plot. Since the generation of the plot did not have a major delay, no further optimizations were performed. Nevertheless, the aforementioned queries should be precalculated and stored in dedicated columns in case the dataset increases and the creation of the plot takes more time.
For the normalized version of the plot no other lookups are necessary, just a little more calculation effort to normalize the total number of reviews for every year.

Stars by number of reviews scatter plot

This scatter plot is a graphic representation of all stars /review_count pairs available in the business.dat file. These values are immediately available and do not require further measures to increase plot performance.

Average Ratings-By-(US-)State grid plot

For this graph which is the most complex plot of the app, first, all business without a valid entry in the “state” column in business.dat are filtered. I.e. empty and the value “01” are not considered to be a representation of a state with the US. Subsequently, the data frame is grouped by “state” and “stars”.
Thereafter, a new data frame is created which contains a summary for each state. Every row contains the state name, the rating, the number of times this rating was given, the total amount of ratings for this state and finally the percentage totalByStar/total.
For the final plot the state column is the x-Axis, the stars column is the y-Axis, the percentage column is the size and the stars column is color of the point.

Final comments and remarks

1. All code and pre-processed files are available at: https://github.com/mihuss/bigDataTask2

2. For executing the pre-process.R file note that it expects the following input files:
   - business.json
   - checkin.json
   - splitReviewA.json
   - splitReviewB.json
   - splitReviewC.json

   Due to RAM issues review.json is split with the following bash command: “split -1600000 ./review.json ./split” and renaming the file according to the specification above.

3. Visual output is optimized for screens with a resolution of 1920x1080 and more.