Cluster and Cloud Computing

Australian Cities Analytics Report

Team 5

Da Zhang 665442
Tianying Cui 664885
Xiangyu Zhou 690709
Kankai Zhang 689196
Xiaoxuan Tang 692782

1. Introduction

Twitter is a one of the most popular online social networks around the world. It allows users to post and read short messages called “tweet”. Analysing these tweets may obtain lots of information about people’s behaviour and emotions. And from the analysis , some valuable information can be presented during the process. Our project can be separated into two parts, respectively the system design and implementation part and the scenarios analysis part. This project firstly harvests tweets from cities of Australia, utilizing automatic configuration of harvest instances, and then data stored in Couchdb gets processed with NLP(Natural Language Toolkit) library for sentiment analysis. Both Apache server for front-end and ReSTful API server for backend are implemented to provide data visualization service. The second part develops 5 scenarios and finally analyses them combined with data from Australian Urban Research Infrastructure Network (AURIN). This report, introduces both two parts in detail, containing the system design, architecture, error handling and the statistics supporting the scenarios analysis.

2. System Functionalities

The system has four main functionalities consisting of tweet harvesting, data pre-processing, data analysing and data visualisation. To be more specific, each functionality can be exactly divided into several sub-functionalities. For example, data harvester depends on the sub-functionality, automatic deployment implemented by the “Ansible” in different instances. Data processing, inserting attribute of sentiment into the database utilizing the NLP library and 5 main views in CouchDB are created to support the query about the 5 scenarios adopting the Map/Reduce method embedded in CouchDB. Data analysing part is regarded as both info formatter and API provider. The specific scenarios in web application requests the ReSTful API (http://115.146.89.147:8080/scenarios/1) for charts statistics and Data visualization support the function to render the JSON format data from API into charts and maps to illustrate statistics vividly.

3. System Architecture

The system uses Service Oriented Architecture and is deployed in the NeCTAR Research Cloud. It contains three parts which is main server, instance cluster and web application respectively. Figure 3.1 shows the architecture of the whole system.

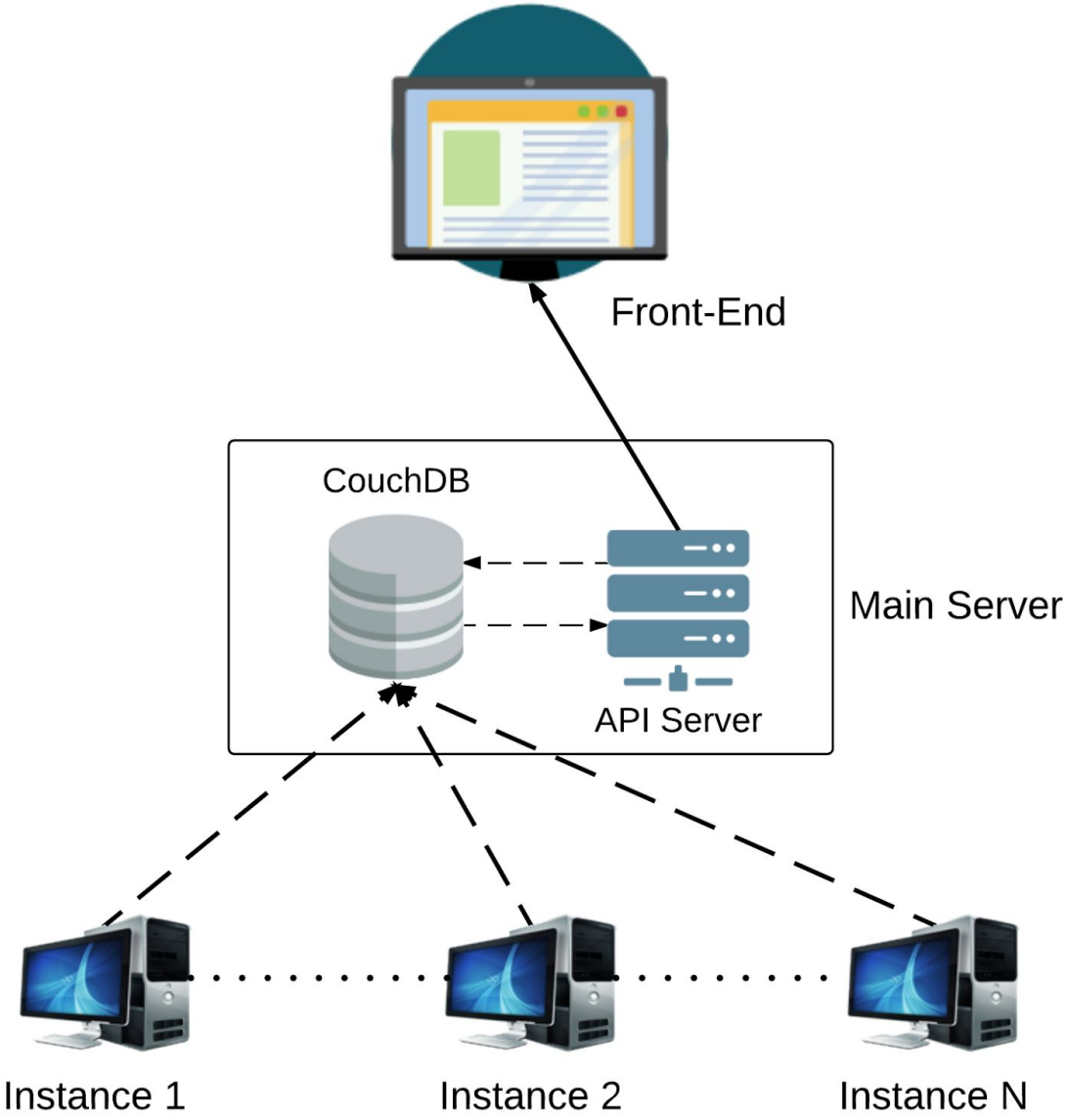


Figure 3.1 System Architecture

The main server consists of database server and API server. The system uses CouchDB as the database. CouchDB is a NoSQL database and stores JSON object directly as record without any transformation. Thus it is convenient to store tweets that are returned from twitter API. Meanwhile CouchDB can be deployed in cloud environment, which is easy to process data simultaneously and good for expanding system in the future. Using embedded Map/Reduce functions in it, CouchDB can create and store data view for specific scenarios. The API server is used to get the data view from CouchDB and then sends them to the web application so that the web application will form these data into diagrams for users to compare and analyse them.

The instance cluster is used to run the process of collecting tweets. It has three instances and each instance runs a process to collect tweets from specific area of Australia. With loose-coupling design, the system allows different processes to harvest tweets concurrently and avoids influence between different instances, which leads to a high-efficient harvesting process of the system. All the tweets that collected from all three instances will be stored into the same database.

The web application can provide diagrams and statistics according to the data that API server sends in order to help people analyse scenarios.

4. System Design

The figure 4.1 depicts the structure of main components.

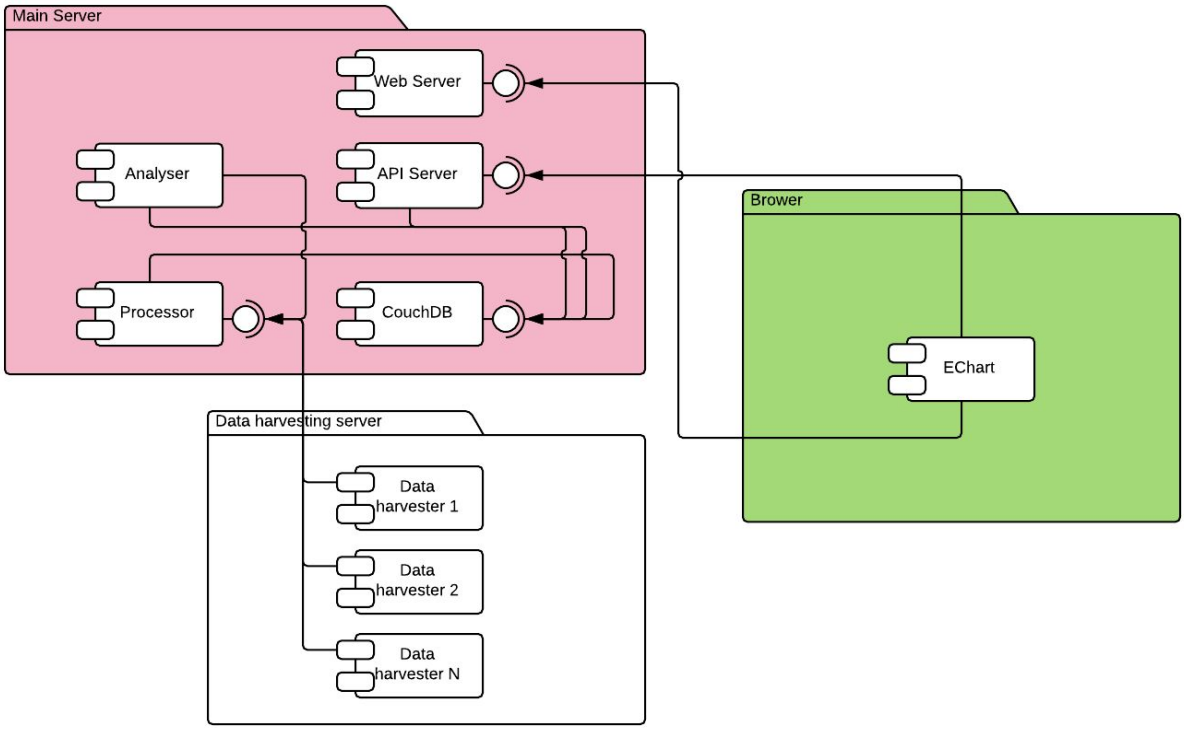


Figure 4.1 Component diagram

4.1 Tweet Harvester

Distributed Harvester, 1 Master 3 Slaves

As the system design shows, we deployed the harvest twitter instance on Nectar. Since the maximum allowed quota is 4 for the project, we can deploy up to 3 slave instances to harvest the raw twitter data.

We defined six main cities in Australia which are Melbourne, Sydney, Adelaide, Perth, Brisbane and Gold Coast to harvest twitter from. Due to the number of expected running instances is dynamic, the six regions are dynamically allocated to the instances when deployed using Round-robin algorithm(Anon, 2016).

Data Collection Distribution

To analyse the scenario and information from tweets, we used AURIN system to help our analysis. AURIN is the Australian Urban Research Infrastructure Portal which provides tons of data set in many areas for research. We created a program to do a statistics of the specific data set in AURIN and combined the result with the analysis of tweets in order to achieve a more precise outcome.

Data Size Data pre-process (sentiment)

The sentiment analysis of Tweet text is conducted once a Tweet is harvested. We do not implement sentiment analysis by ourselves, but taking advantages of existing library, which is TextBlob. This is a pretty good python sentiment analysis library that will take a lot of factors into consideration when analysing. For example, different punctuations will lead to different sentiment polarity(exclamatory mark implies much more emotion than full stop). It can not only process pure text, but tell emoji and emoticons as well(e.g.: :-) and xD). This is quite helpful for this assignment because Twitter users tend to often use emoticons in Tweets.

However, TextBlob still has one shortcoming, which is quite common for most of sentiment analysis technology, that it is not able to identify sarcasm. In most of cases, a text with sarcasm contains negative sentiment and the technology will say it positive. This is a quite difficult issues to be conquered because it requires not only to understand text lexically, but also understand overall meaning of given text.

4.2 Web Application

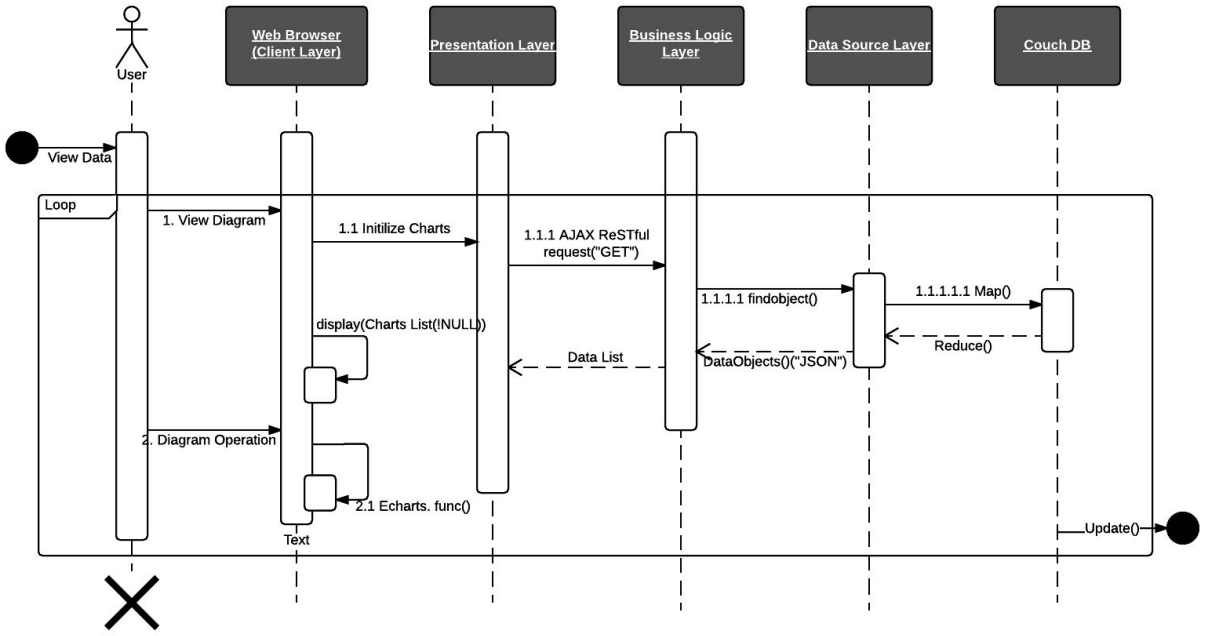


Figure 4.2 Sequence Diagram View Charts

Web Application adopt the traditional layer architecture design to implement the web application part, Couchdb is absolutely the db server used in this system. To provide the demonstration, some views are implemented using the Map/Reduce depending on the mechanism of Couchdb itself. The Couchdb here take responsibility as the Data Source Layer in the architecture.

And there is also a backend server to process the data and format the data as the presentation needs, therefore this backend server works as the domain logic layer for the whole system, providing ReSTful API. The API in our system is defined concentrating on the resource (scenarios) itself. Although the method implemented for the data visualization is only http “GET” other than using “PUT” for updating, “POST” for creating resource and so on. The data set for the presentation are only the scenario we need to present in the lecture, therefore, the number of scenario are defined to navigate the resource. ReSTful API, indeed, decreases the complexity when establishing interface for component. And it is obviously make better use of HTTP protocol (methods). Simultaneously, transparency of resource can be obviously presented by the url itself (Richardson and Ruby, 2007).

To loose coupling between each layer, the presentation layer itself only access resource from the API server, and javascript lib (JQuery) get used to send the ajax request to the API server and receive the response from server rather than making the server render the template website. As web application the presentation should be able to communicate with domain logic layer. To improve the data visualization interface, some javascript libs for establishing charts are selected to provide better user experience and obvious comparison among huge tweets for example, nightingale rose chart which presents proportion directly by the size of different areas, can be accessed in our web application for showing different source of tweets. Simultaneously, the implementation of partial charts can support transferring from existing type of chart into another type by clicking on the buttons in toolbar on the top of each diagram. Certainly the data we indicate in web application can be edited directly and user can decide to view which part of legend by selecting the items as well.

5. Fault Tolerance

As described above, we have 4 instances running, one takes the role as main server, the other three run as equivalent data fetching instances which responsible for splitted jobs.

Here are some situations where fault may occur, and their corresponding strategies of handling them:

5.1 Flooded useless data

As original data are more meaningful than those retweeted, and when something big event happens or someone famous says something, flooded retweeted tweets will have large impact on statistics, we will filter retweeted tweets based on the scenarios we will build.

5.2 Failures of data fetching server

These kind of failures will result in partial missing of data. To tolerate it, the main server keeps an map of the instance ips and their jobs (To be more specifically, the geographic region each instance should watch), and the main server keeps asking the liveness of each instance every 10 minutes, if instance is out of response, it will force shutdown that instance and restart it through boto client API, meanwhile, the main server will take over the responsibility of the failed instance temporarily till the failed server back online.

5.3 Failures of main server

If the main server crashes, the whole system breaks, it can not be recovered automatically, hence, the data fetching instances will monitor if the the main server is alive by monitor if the data is successfully sent to main server, otherwise, notifications will be sent to administrator by email.

6. System Discussion

The system is deployed in the NeCTAR Research Cloud which is an online infrastructure that supports research work. Running system in the NeCTAR has many advantages. First, there is no need for onsite hardware and related expense. NeCTAR provides hardware for users and it is easy to upgrade its hardware resources according to the user's requirements. Second, system deployed in NeCTAR is easily scalable. Modules and resources can be modified dynamically as demanded. Third, users can access to their system or data at anywhere anytime with internet connection. NeCTAR helps users work more flexibly. Finally, system can be backed up in the cloud which significantly improves system reliability.

However, NeCTAR also has some disadvantages. Without internet, users cannot access to the NeCTAR cloud so that everything based on it cannot be functional. Also, the same situation may happen when NeCTAR service is down. During the project developing process, NeCTAR had got a service failure that no one can login to the system. The project was postponed for a whole day until the NeCTAR service was recovered. Furthermore, user experience of NeCTAR is limited by the speed of internet connection and data security is another drawback that needs to be improved.

Regarding the tools for the Nectar service, we use boto which is a powerful substitution for the management website. With boto, querying instance status, running instance, creating container and uploading file to container can be maintained in a single python file. Once the script is properly written, the deployment can be done without human interference. The side fact of boto tool is there would be chance to delete an instance by mistake without any alerting, which would lead to a severe consequence.

The instance task management tool Ansible is another powerful when we deploy our instances. The new created instance initialisation like updating repository, installing python libs, creating twitter harvest service can be executing through Ansible. The

demerit we have confronted when using Ansible is that it requires to use a valid key pair to ssh the target instance. If the public key is not well added to the target instance, the Ansible will fail.

7. Scenarios

Scenario 1 : The emotion statistics among different users from divergent source

According to the picture below, it is obvious that IOS are most popular among Australia Twitter users. 50% of users choose to post twitter via iPad and iPhone. Web application might be the second popular method for twitter usage which presents proportion of almost 24%. In Australia's main cities, only 16% users choose to post tweet via Android platform. However, from the statistics of Smartphone OS market share table (Kantarworldpanel.com, 2016), there should be more Android devices sold in Australia, 55% of the smartphone os should be consumed by Android system, while IOS accounts for 38% in 2016 Feb. By the contrast of two charts below Android twitter users seems to use Twitter less frequently.

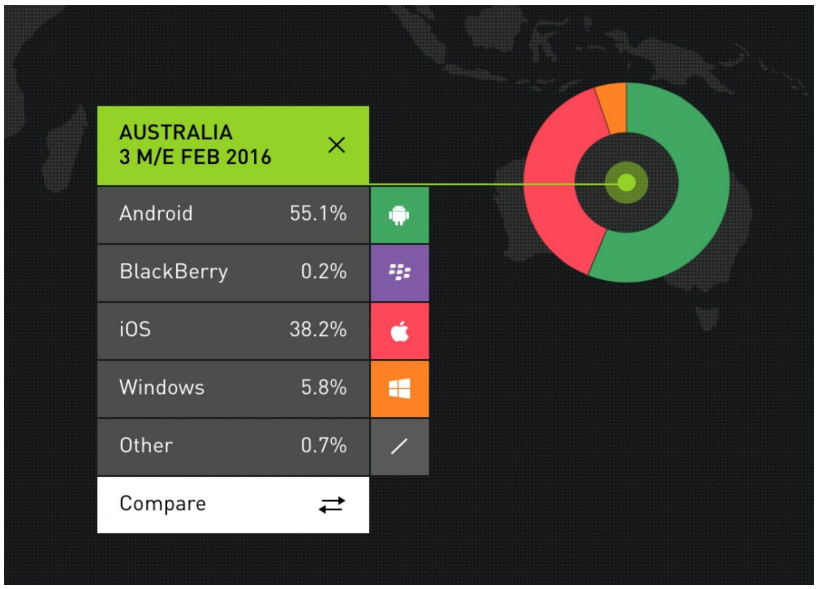


Figure 7.1 smartphone market share

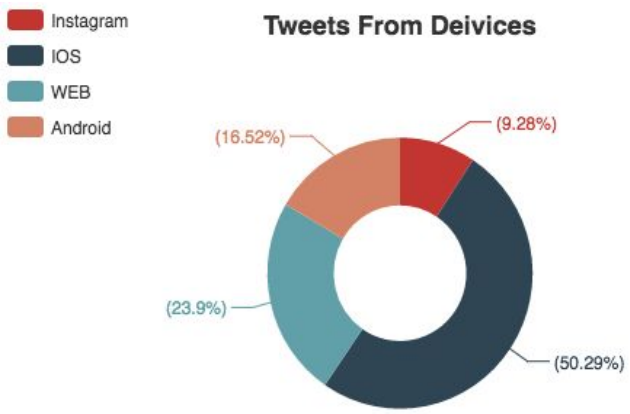
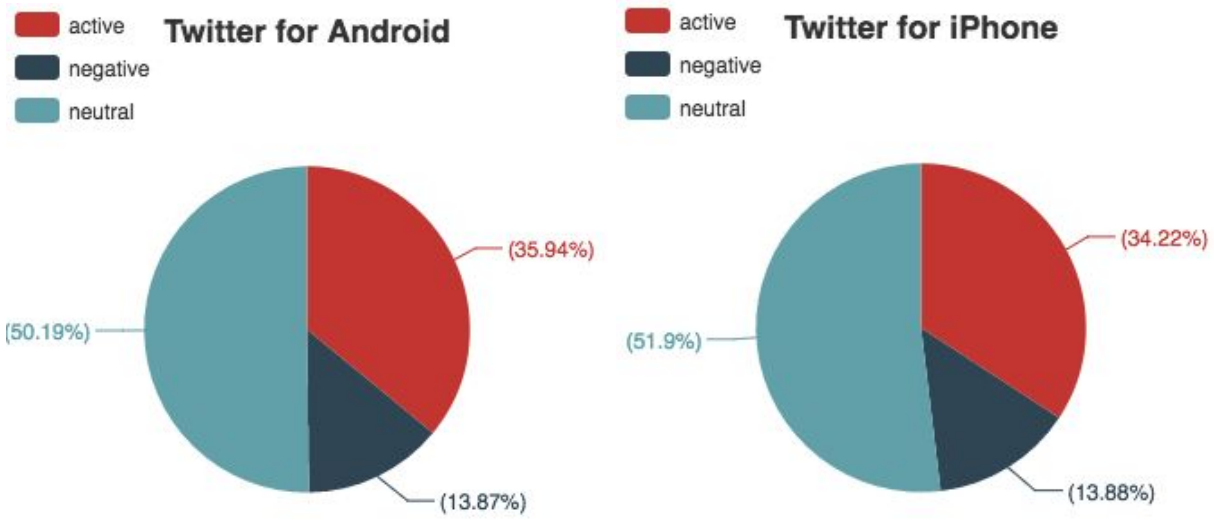


Figure 7.2 Tweets From Different Devices

First Reason for the special situation is that the Android devices might be even more popular in suburb areas rather than the main cities collected in our system. Secondly the user experience for Android might be worse than IOS, which increases the usage frequency of IOS users.

Besides, the other parts, less than 10% users post the tweet via third-party platform Instagram, which must get attached with pictures. This is the main third-party platform of twitter service.



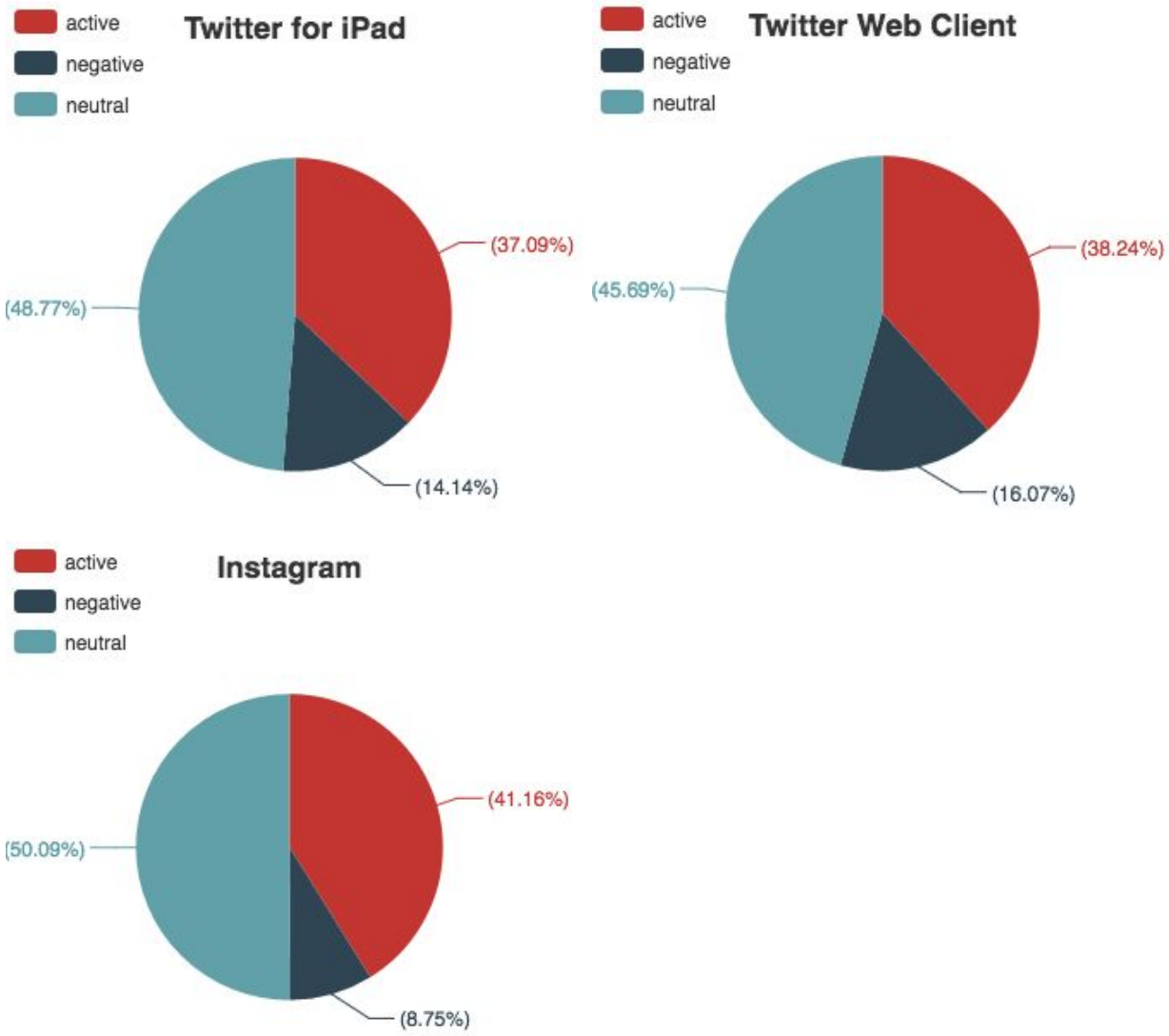


Figure 7.3 Sentiment Distribution in Devices

After adopting the NLP libs for analysing both natural language and some expression such as (":), >0<") , the additional attribute get inserted by the processing script as "sentiment". From this attribute we can easily find the emotion of tweets via its text. Although the neutral emotion seems to consume the largest portion and It is apparent that people attend to share the active emotion via twitter. The percentage of negative tweets sharing accounts for less than 15% in most devices.

From the pie charts above, interesting result can be concluded. The tweets shared from platform which is likely to contain both text and picture seems to be more active than the tweets with only text. The percentage of negative emotion from Instagram accounts for merely 8% of all, one time shorter than the negative percentage from other devices. By

comparison with other source, users from Instagram seems to be happier and active towards the daily life.

In conclusion, tweets with picture selected elaborately tends to present positive emotions. On the contrary, if users with negative emotion want to express their feelings, it seems unlikely to spend time selecting compatible pictures. The negative emotion usually bursts with dirty words in plain text, if people try to find a compatible picture to describe their negative emotions, it seems to be easier getting rid of this awful feelings during the selecting time. Therefore, to contribute better internet environment, and to keep a good mood, try to select an interesting picture before posting a tweet.

Scenario 2: Relation between the sentiment of people using twitter and the employment rate in different regions

According to the AURIN data set “Employment Vulnerability Index for Australia 2011 Data Profile”, the average unemployment rate of six cities are listed in Figure 7.8. It is obviously that Gold Coast has the highest unemployment rate which is 7.3%. In contrast, the unemployment rate in Perth is the lowest one among these six cities which is 4.3%. Generally speaking, people may be happier if they live in an area with low unemployment rate. Thus, we analysed sentiment of these tweets and have got the result listed below.

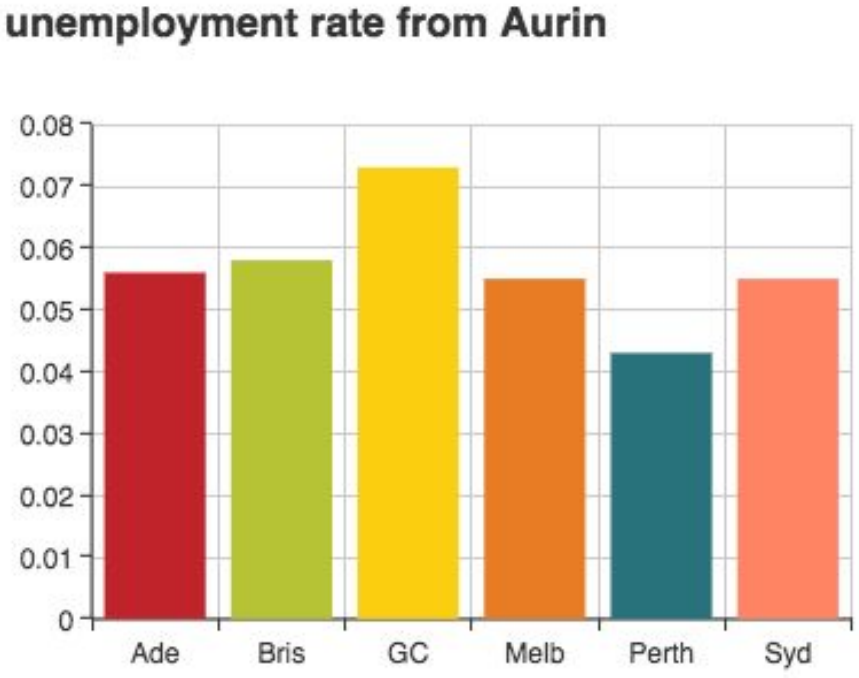


Figure 7.4 Unemployment rate in different cities

Figure 7.9 shows the ratio of the number of tweets with positive sentiment in each city. From the figure, it is clear that the ratio of Gold Coast is 33% which is the lowest one among six cities and Perth has the highest ratio, 37.6%, of positive tweets. However, in this case the result may be affected by tourist tweets. Tourists may post tweets and this does not have any relation with unemployment rate because tourists do not work in the city. Since these six cities are all famous city that attracts lots of tourists from all over the world, we need to eliminate the effect caused by these tourist tweets. We remove those tweets that are not posted by local people and then recalculate the ratio. The result shows in FIgure 7.10. The ratio in Gold Coast is 34.3%, which is still the lowest one and in Perth, the ratio becomes 36.5%.

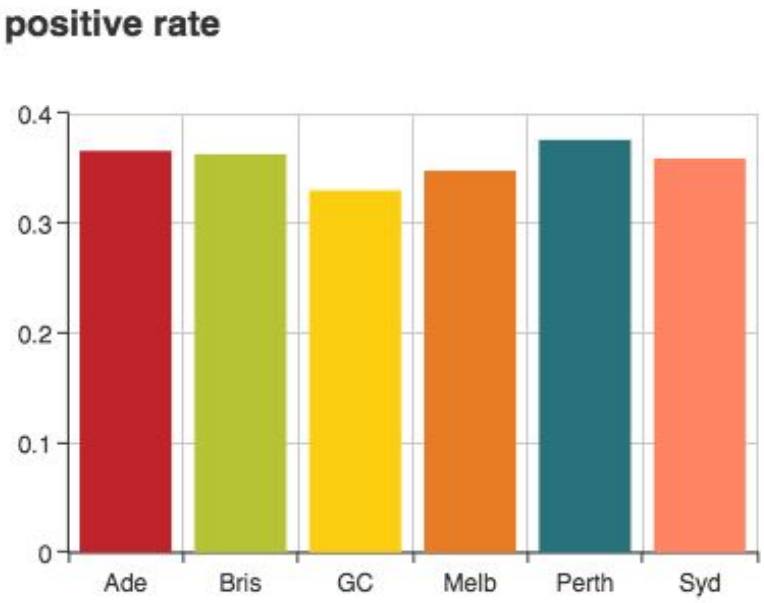


Figure 7.5 Ratio of positive tweets in different cities

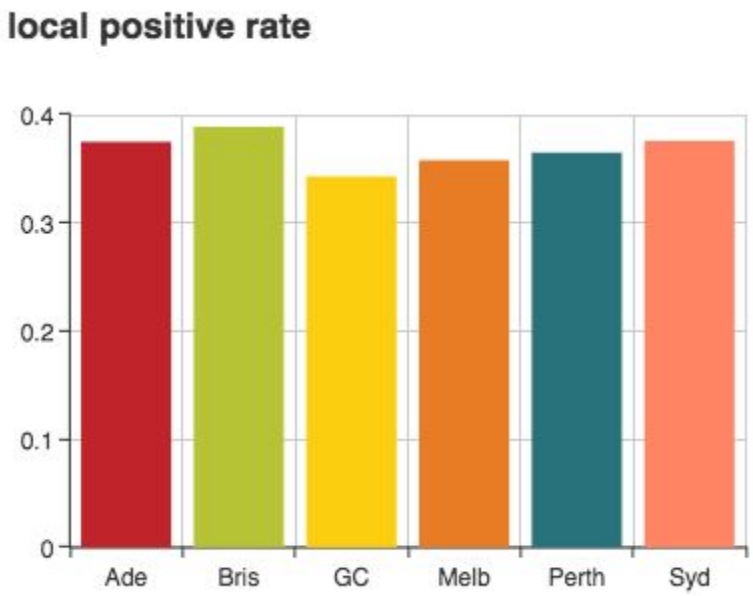


Figure 7.6 Ratio of local positive tweets in different cities

Based on the above information, it turns out that the lower the unemployment rate of a city, the more positive tweets people may post and the happier people who are living in that city are. Because the AURIN data set we use only contains unemployment rate data in 2011, the data result may not align with the pattern perfectly.

Scenario 3 : Relationship between sentiment of tweet text and twitter user followers count

Our hypothesis is based on the twitter user follower count which is the more a twitter user got followed, the more positive sentiment tweets would be posted.

The following figure displays the tweet text sentiment analysis over user follower of 0-200, 200-400, 400-800, and 800+ followers.

Figure 7.7 percentage of positive, negative and neutral sentiment

As it shows, the percentage of neutral sentiment is getting lower which indicates twitter user with more followers would post more meaningful and emotional tweets (either positive or negative). Such tweets will attract other users discussion and interaction.

The second figure shows the positive sentiment percentage over negative one.

Figure 7.8 percentage of positive and negative sentiment

After unclicking the neutral sentiment, this figure shows clearer relation between the positive sentiment and negative one. A trend can be seen that the positive part is

getting higher, which can support our initial hypothesis that user with more followers would post more positive tweets.

The last figure shows the negative and neutral sentiment.

Figure 7.9 percentage of negative and neutral sentiment

As it shows, there is no explicit difference between neutral and negative after the positive is excluded. This can be explained that the possibility of user post a negative tweet is not highly related to the user follower number.

To conclude, the more follower a twitter user has, the more positive tweets will be posted but this will not influence the possibility of negative post sentiment.

Scenario 4 : Tweet length in accordance with local time

We would like to discover the correlationship between local time and the average length of characters in content, based on our common sense, the content of tweets are more likely a reflection of people in real world at the moment they tweeted. We just take the length of content as an indication of entropy, and we collect data from Melbourne and Sydney to see the differences between two cities. The final graphic is listed below:

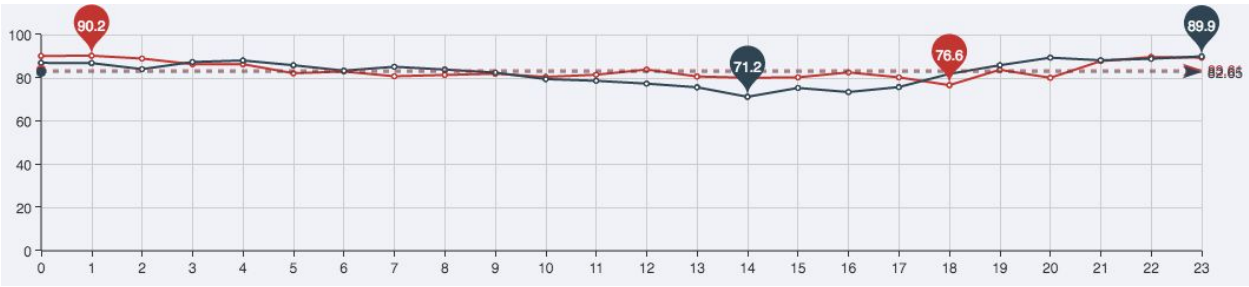


Figure 7.10 Local time and average tweet length

As you can see, the length of tweet is relatively stable across the day, and two cities are very close to each, that is, around 80 characters. What more, if we take a close look at the patterns of two cities, differences can be found, let us zoom in a little bit.

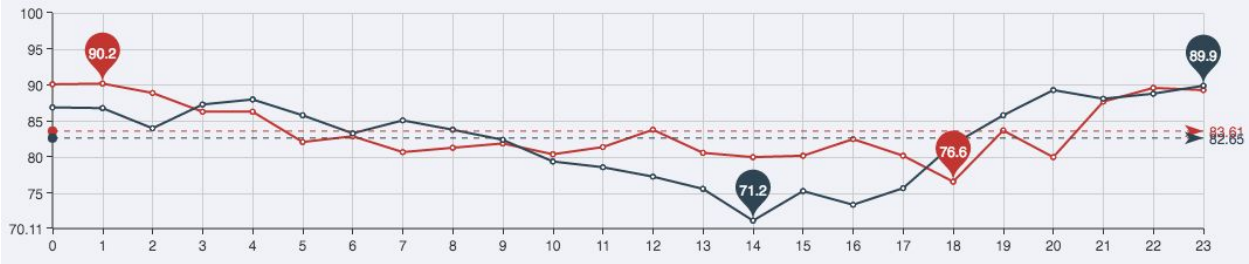


Figure 7.11 Local time and average tweet length(zoomed in)

When zooming into the field between length 70 and 100, some interesting patterns appear:

1) The average length of tweets in Sydney starts to decrease from 7:00AM and reaches the lowest point at 2:00PM, and then starts to increase till reaches the highest point at 8:00PM, and other time region the average length fluctuates randomly.

2) The average length of tweet in Melbourne does not have apparent patterns, but we can see that both Melbourne and Sydney have top average content length around midnight.

As we consider that the average length of tweet content is an indication of entropy, in order to find out the inner connection, we can simply make inferences based on the underneath equations:

$$\text{Entropy}_{\text{average}} \propto (\text{Twitter length})_{\text{average}}$$

$$\text{Entropy}_{\text{avergae}} = (\text{Time spent})_{\text{average}} \times (\text{Entropy generation rate})_{\text{average}}$$

$$(\text{Entropy generation rate})_{\text{average}} \propto (\text{People's energy level})_{\text{avergae}}$$

And we have some inferences:
1) People write more before they go to sleep, most likely they spend more time on typing.
2) People in Sydney type least at their afternoon coffee time, most likely they are busy doing other stuff and put less time on typing.
3) People in Sydney have more regular timetable than Melbourne, in other words, live in Melbourne is more casual than that in Sydney.

Scenario 5 : Language distribution in different Australian cities

As Australia is an immigrant country, we would like to figure out the culture distribution in some target Australian cities. However, Twitter can not, and do not provide the information of user's culture out of the box. As a compromise, we decide to take preferred language each user selected to identify user's culture. We picked five cities out of all, from which we can catch the most Tweets among Australia, which are Adelaide, Brisbane, Melbourne, Perth and Sydney. The overall result can be found in Figure 7.7.

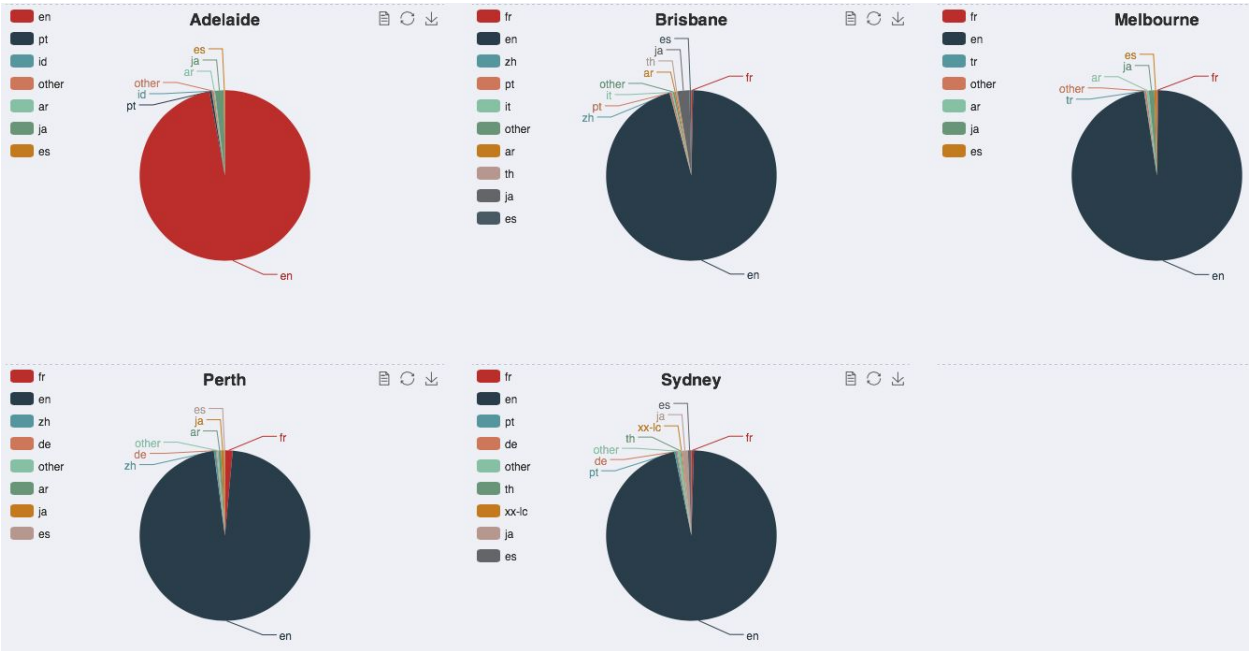


Figure 7.12 The distribution of Twitter language in selected cities

It is obvious that English, as Australian national language, dominates the Twitter language usage in all of the five cities and the other languages just share only a little usage. To reveal more, it is ideal to remove English and observe the distribution of the rest languages.

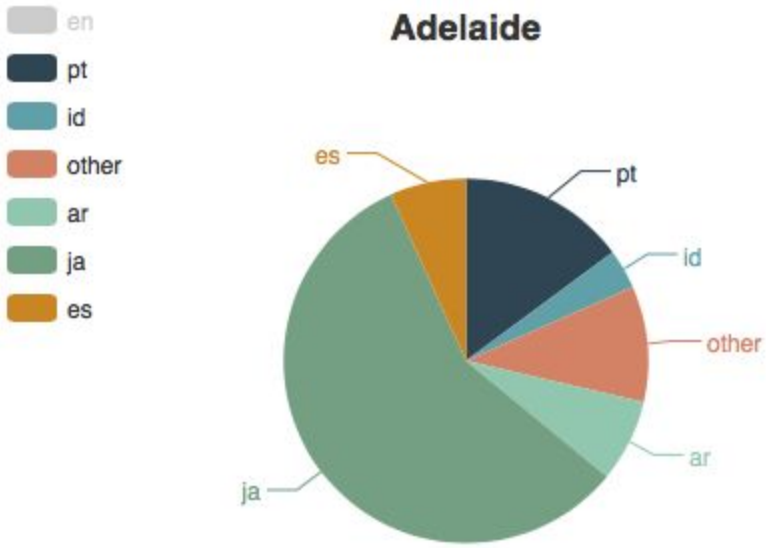


Figure 7.13 The distribution of Twitter language in Adelaide(English excluded)

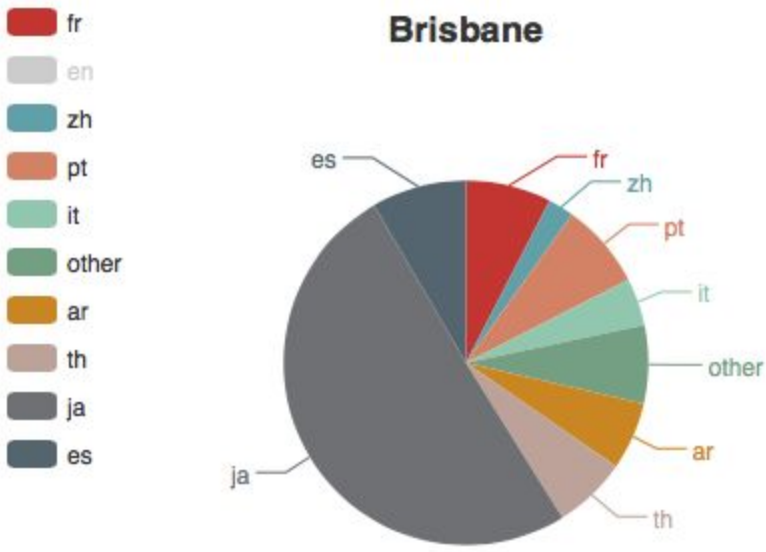


Figure 7.14 The distribution of Twitter language in Brisbane(English excluded)

Figure 7.15 The distribution of Twitter language in Melbourne(English excluded)

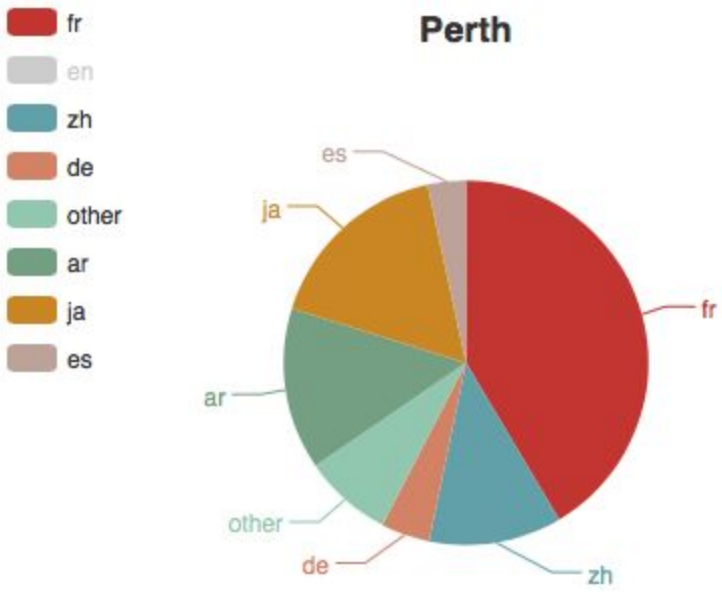


Figure 7.16 The distribution of Twitter language in Perth(English excluded)

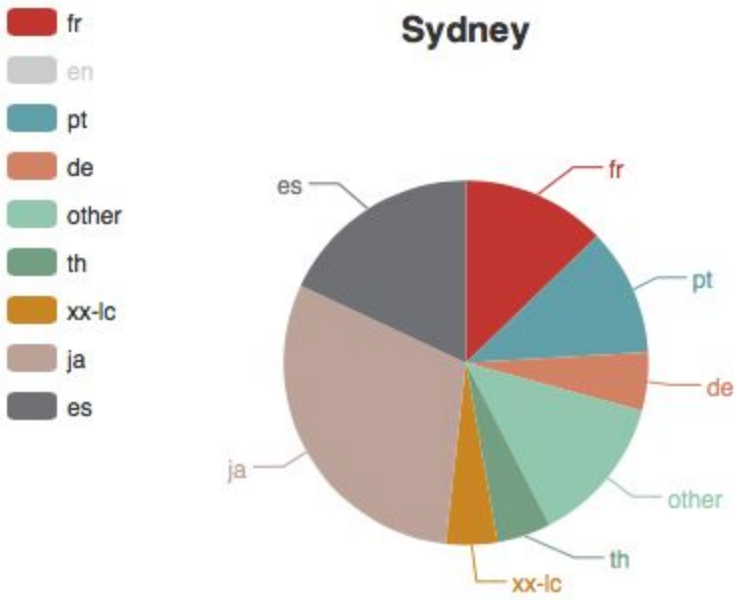


Figure 7.17 The distribution of Twitter language in Sydney(English excluded)

It is quite out of expectation that Japanese is the most frequently used language in Twitter besides English in four cities(Adelaide, Brisbane,Melbourne and Sydney) and the second following French in Perth. Chinese only occupies minority of minority in all of the five cities. This is quite against the fact, as according to total population, Chinese is always one of the top three non-English language used in Australia and as for Japanese, it can never be found in top 10 in any state.

The reasons for this conflict could be various:

1) The usage of Twitter language do not have direct proportional relation to the usage of language in real life.
2) The attraction of Twitter to people from different countries(cultures) differs. Maybe Chinese or Greek prefer to use other SNS rather than Twitter.

8. User Guide

The system is designed to deploy automatically. To deploy the system, the user should follow the steps below.

1. A python 2.7 environment is required to run the scripts
2. Get the code package and cd to scripts folder
3. Run ./run.sh

4. The terminal will ask you the instance number to be created (the number should be greater than 0 otherwise it will reject)
5. Wait for the script complete running
6. Access http://115.146.89.147:5984/_utils/ to monitor the DB which is called “comp90024”
7. Access http://115.146.89.147/ to view the web application portal
8. Backup youtube url https://youtu.be/sbwjqbOYA0I

Bibliography

Kantarworldpanel.com. (2016). *Smartphone OS sales market share – Kantar Worldpanel ComTech*.[online]
Available at:http://www.kantarworldpanel.com/global/smartphone-os-market-share/ [Accessed 11 May 2016].

Richardson, L. and Ruby, S. (2007). *RESTful web services*. Farnham: O'Reilly.

Australia language spoken at home, profile.id,[online]
Available at: http://profile.id.com.au/australia/language [Accessed 11 May 2016].

Anon, (2016). [online] Available at: http://pages.cs.wisc.edu/~remzi/OSTEP/cpu-sched.pdf [Accessed 11 May 2016].

Fengxiang, F. (2013). Text Length, Vocabulary Size and Text Coverage Constancy. Journal of Quantitative Linguistics, 20(4), pp.288-300.

Woodward, P. (1957). Entropy and negentropy (Edtl.). IRE Transactions on Information Theory, 3(1), pp.3-3.