COMP90024 - Cluster and Cloud Computing

Assignment 2 - Australian City Analytics

Team 40

Yuyang Wang	1075732	System Setup/Automation & Front end
Chongzheng Zhao	1064561	System Setup/Automation & Front end
Harshal Shah	1020849	Data Harvesting & CouchDB with Map Reduce & Visualization
Kamakshi Banavalikar	1039197	Data Harvesting & CouchDB with Map Reduce
Sophia Nelson	1113928	Backup Coder

GitHub Link: https://github.com/hhshahum/CLUSTER-AND-CLOUD-ASSIGNMENT-2

YouTube Link: https://www.youtube.com/watch?v=qrtjx8jl9Ec

Table of contents

1. Introduction

They say Melbourne is one city where you get everything you wish for. The Economist Intelligence Unit's 2017 awarded Melbourne as “The most livable city” for the past seven years. However, since the last two years Melbourne has been the second most livable city, overtaken by Vienna. This could be due to crime concerns in Melbourne. ("Melbourne is NOT the most livable city in the world for the second year running", 2020). According to the multicultural youth australia census status report 2017/18, the social indicators says that the diversity and discrimination are most significant issues in Australia followed by education and job.(*Multicultural youth australia census status report 2017/18*, 2018).

Like everything else this country also has a negative side to it. Since last few years, the crime rate and discrimination has been increasing tremendously. It has also been noticed that a lot of residents are being vocal about the discrimination and crime they have been facing in their day to day life on social media platforms like Twitter. Therefore, the questions to be analyzed are: **How many people face crime on a daily basis? Does language have any significance with respect to discrimination?**

Social media platforms like Twitter give direct access to one's thought process. These tweets open the gate for the huge data set which can be further analyzed in different ways. This helps in exploring the different behavioral patterns. The following system has been created by cloud analytics tools such as Python Libraries, CouchDB, Map Reduce functionalities, Google Cloud API, NeCTAR, Front-End technologies and Grafana as the cloud monitoring service. In the following report the story is narrated with two scenarios, crime, and discrimination.

The report begins with the process of extracting the huge amount of data and eventually storing it in the database for further processing and analysis. The findings are later compared to the dataset obtained by the AURIN to have a better understanding of the mentioned scenarios.

2. Architecture

The architecture of the system is very simple to understand. From allocated NeCTAR resources (the Melbourne Research Cloud) 4 instances were used. In the first instance there are three services running namely, Apache Server, Cloud Monitor and Tweet Harvester. The CouchDB cluster is set up on the remaining three instances numbered as 2,3 and 4. The three instances have been used for load balancing since the data is in Gigabytes. In total 200 GB were allocated, each CouchDB instance consists of 50 GB volume and the first instance also consists of 50 GB Volume. The flow of the architecture can be understood easily by the following diagram. Each component of the architecture has been explained in detail in the corresponding subsections.

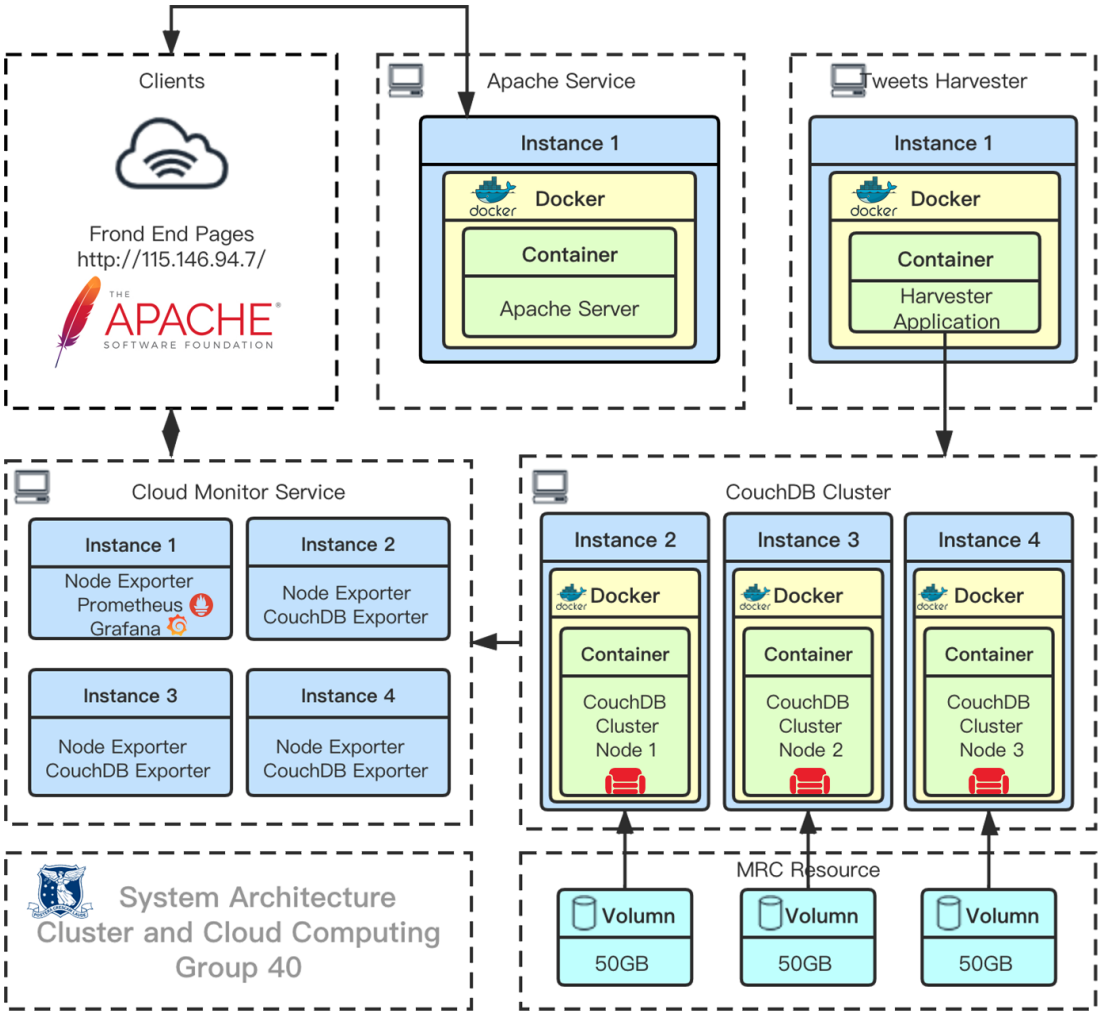

Fig 1 : Architecture Diagram

2.1 Tweet Harvesting

In order to harvest tweets, a harvester application was set up on the first instance. For tweet streaming five Twitter's stream API authentication keys were used. These APIs would fetch the tweets and manipulate them according to the python script *Tweet_Harvesting.py* and later on store in the CouchDB database named *harvest_tweets*. In the following process the "tweepy" library in python was used to harvest tweets with respect to the location constraints. The motive of using location constraints was to collect tweets only with respect to Australia. Parallel processing concept is being implemented in the harvesting script as there are multiple API's used for tweet fetching and it was implemented using the multiprocessing (Pool) library in python. The AURIN (Australian Urban Research Infrastructure Network) majorly has historic data and to compare the historic data with the most recent ones is not possible, hence the historic dataset for Melbourne provided by the lecturer was used. To import the huge historical dataset provided by the lecturer, into the CouchDB (database name: *tweets_from_richard_db1*) SPARTAN HPC was used for smooth implementation. The AURIN data was imported onto the database under the name *aurin_discrimination_data , aurin_sa_crime , aurin_vic_crime , aurin_vic_cirme_with_location*.

Handling Duplication of Tweets:

In the initial harvesting method, it was realized that the duplicates were being stored in the CouchDB database '*tweets_using_tweepy_for_australia'.* There were approximately 1.7 Million tweets stored in the above-mentioned database. As soon as the error was noticed, it was rectified by implementing preprocessing in the same python script and started storing the fresh tweets in the new database called '*harvest_tweets'.* In the CouchDB it was noticed that the duplicate tweets had the same '*id_str'*. The '*_id'* generated by the CouchDB was replaced with '*id_str'* and further checked if this '*id_str'* exists in the database, so that only unique tweets are being stored.

Exceed the request limits:

There were continuous 420 errors from the Twitter API, which comes up when the API exceeds the request limits. If the number of requests crosses the certain threshold limit then that particular API is refrained from using the service for a dedicated time frame and the waiting time is linearly proportional to the number of requests sent by the API. This was also dealt in the preprocessing step. As per the new twitter API rules the '*wait on rate limit'* was set to '*true*' since the number of requests which an API can send in a certain time span is limited.

2.2 CouchDB cluster

The CouchDB clusters are set up on 3 instances utilizing a tri-cluster of docker containers. CouchDB was preferred over other database systems since CouchDB has more advantages from the rest, as it stores the data in the form of documents and our requirement was fault tolerance and avoiding single point of failure. Every node has been backed up in the cluster; if any node goes down the backup node would take over. The default value for the number of replicas will be the number of nodes, in our case it is 3. The CouchDB inbuilt functionality makes it easy to replicate data on the different instances which otherwise is a tedious task since the Twitter data has a huge volume. CouchDB lets us export or import data in JSON format since it is a JSON based document-oriented database. Couch helps in storing the data in the appropriate format (in our case it is JSON + document format) to run the MapReduce queries for further processing. The shards and replicas are features of CouchDB. Expansion of the database is linearly to the number of replicas of the shard. The replica adds fault resistance, as some nodes can go offline without crashing. In the implementation, the number of nodes are 3 and the number of shards are 8.

Map Reduce:

The main advantage of CouchDB is that it allows the users to define their own views, write their own map reduce queries to analyse the data. To query and report on the documents in the CouchDB database, 21 different views have been created with different combinations of the date, location, and language of the tweets for the two stories (i.e. Crime and Discrimination). These are being written in the script *'CouchDB_Vews.py'*. The default programming language for writing map reduce queries in CouchDB is JavaScript. The map function takes the document as the input and divides it into key value pairs based on the given condition and then passes the results to the reduce function for further aggregation.

To filter the tweets with respect to crime, the following keywords were used *"crime, criminal harassment, robbery, assault, rape, blackmail, extortion, robbery, graffiti, deception, theft, offence, drugs, homicide, abduction, arson, burglary, bribe, trafficking, breach"* .

In the same manner to filter the tweets with respect to discrimination, the keywords used are "*discrimination, niggas, sexual discrimination, age discrimination, sexualism, gay liberation, job discrimination, fair employment, okboomer, racial, racism, nigro, currypeople, lookism, fattism, sizeism, sexist, misogynist, heterosexism, ageist, menaretrash, paygap, classism, ableism*".

Further to this, the date of the tweet was converted to the desired Locale format using the '*Date*' module in JavaScript and were used as the keys for further grouping procedure.

In the following example the discrimination words are targeted and sorted in terms of date and location of the tweet. They are later reduced by the *count*. In a similar fashion, all other views have been created and map reduce operations are performed thereafter.

Fig 2 : Map - Reduce Function Example

key	value
["04/29/2020", [140.840501, -11.654941]]	2
["04/29/2020", [144.593742, -38.433859]]	9
["04/29/2020", [150.520929, -34.118347]]	10
["04/29/2020", [151.555002, -33.120457]]	2
["04/29/2020", [152.99692, -26.853342]]	2
["04/30/2020", [115.587654, -33.440339]]	2
["04/30/2020", [115.617614, -32.675715]]	18
["04/30/2020", [138.44213, -35.34897]]	5
["04/30/2020", [140.999475, -37.50506]]	5
["04/30/2020", [144.018211, -38.04229]]	2
["04/30/2020", [144.290649, -38.216797]]	1
["04/30/2020", [144.593742, -38.433859]]	76
["04/30/2020", [144.680455, -37.603654]]	2
["04/30/2020", [144.991371, -37.812143]]	1

Fig 3 : Map - Reduce Function Output Example

In Fig 3, key is the combination of the date and location (co-ordinates) of the tweet and value is the count. This query is reduced at group level 2 to merge both the keys into a single group and count is calculated.

2.3 Cloud Computing Monitor

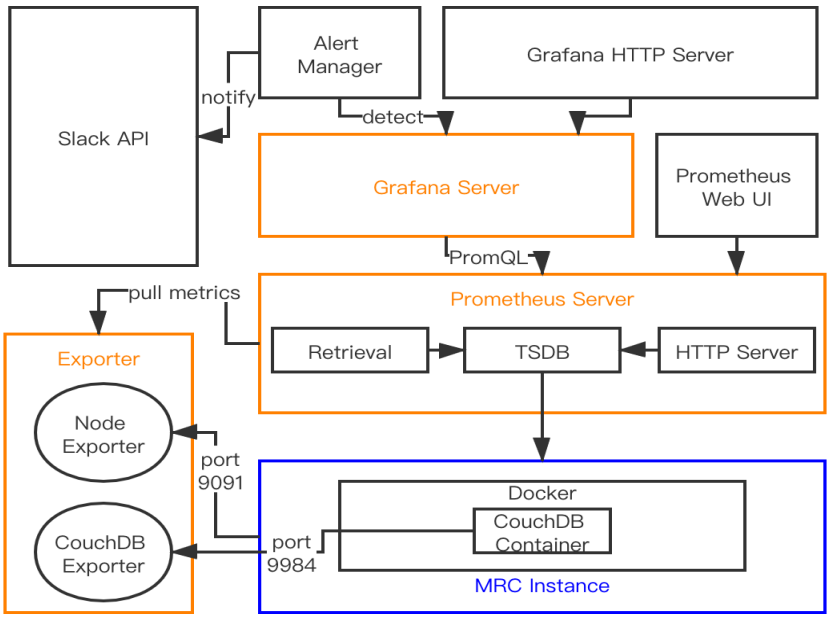

Fig 4 : Cloud Computing Monitor Architecture

As seen from the above diagram the base layer is Exporter. This layer consists of the CouchDB exporter and the node exporter. The exporter pulls metrics from the Prometheus Server.

Grafana is a user-friendly graph monitor server and is written in go language. It is mainly used for visual display of large-scale data indicators. The role of Grafana is to monitor the harvester fetching rate. It also gives an approximation of days in which a few millions of tweets can be achieved. Prometheus Query Language can be well supported by Grafana. We pre-configure the platform to establish a dashboard. The dashboard can add, delete, modify any time-series graphs as per requirement. Grafana does not support direct fetch metrics so a middle layer has been added.

Prometheus is a time series data processing server and it fetches the metrics from the exporter and processes the data. As seen in the architecture diagram, Prometheus acts as a middle layer.

CouchDB Exporter monitors the CouchDB, fetching any information from CouchDB Cluster. The CouchDB metrics exporter requests the CouchDB stats from the /_stats and /_active_tasks endpoints and exposes them for Prometheus consumption. The exporter can be configured via program parameters, environment variables, and configuration file. ("Overview | Prometheus", 2020). Node Exporter monitors the instance, the Memory, the Network, and the Disk IO.

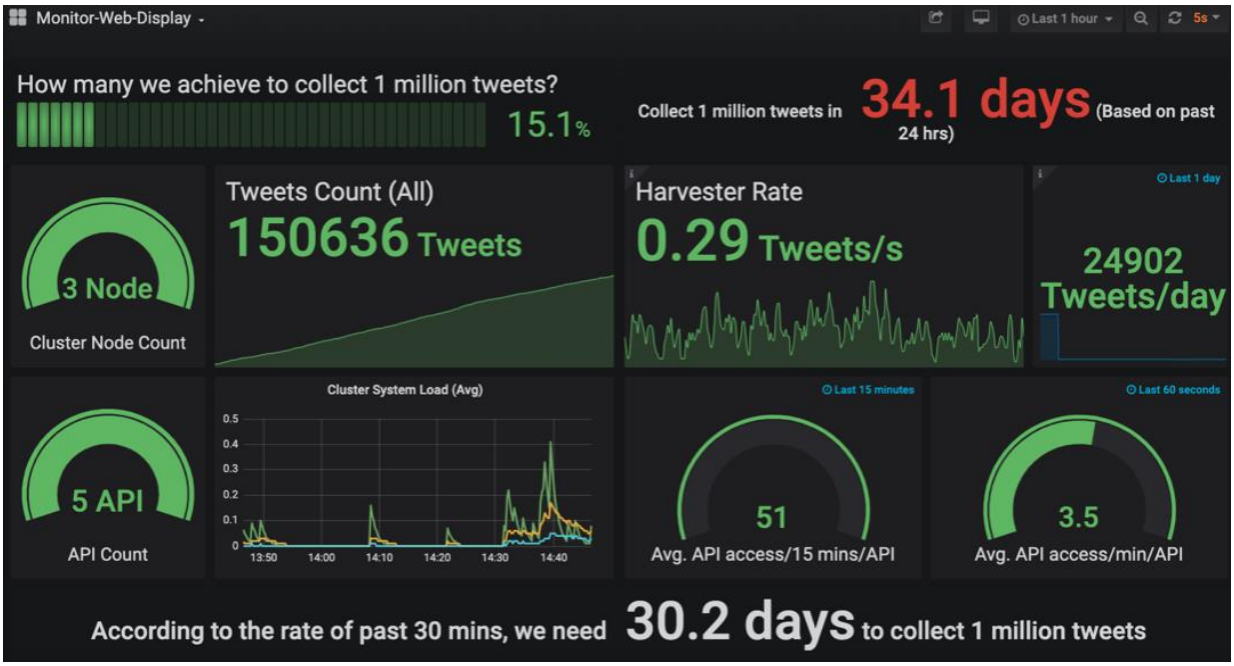

Fig 5 : Customized Monitor Overview

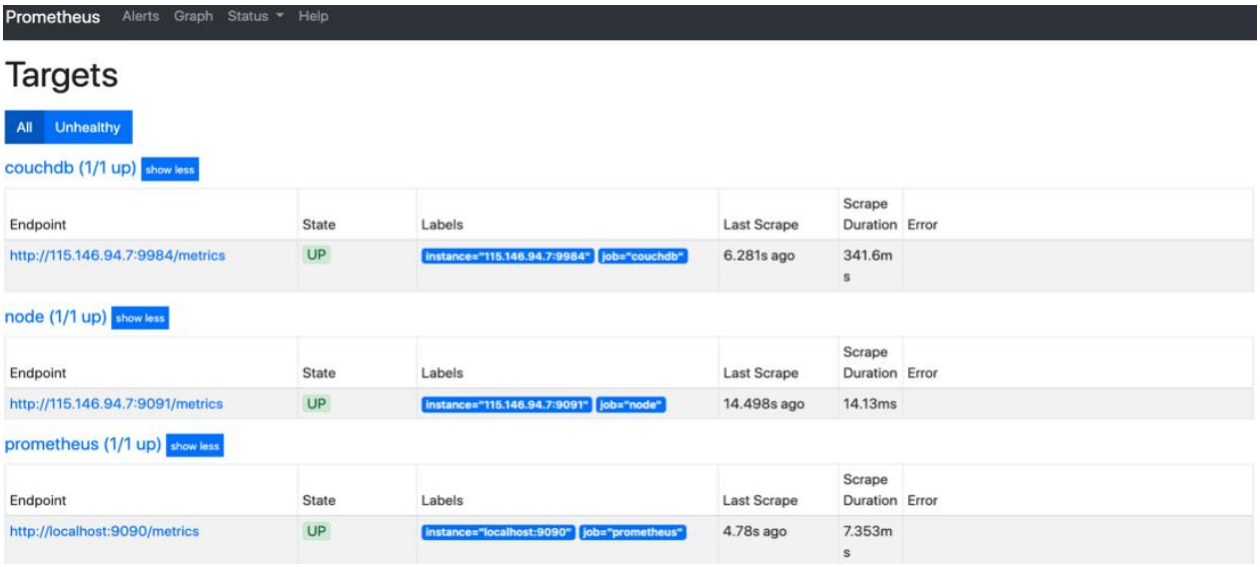

Fig 6 : Prometheus Targets

2.4 Docker

Docker is an open source application container engine that allows developers to package their applications and their dependencies and libraries into a single package to have a portable container. It can also be virtualized. The containers use the sandbox mechanism and there is no common interface between them. A complete Docker consists of the following parts: Docker Client, Docker Daemon, Docker Image, Docker Container.

Docker is being configured by Ansible. The procedure includes installing the Docker CE and container, creating a configuration file, setup proxy for docker, flush changes and restart the docker. Our project uses docker technology, which provides elastic cloud services. It is because docker works on the principle of scalability. The CouchDB cluster along with the docker services makes it easy to expand the database whenever needed. If there is a need for more data storage, it can dynamically increase the number of CouchDB containers. The CouchDB cluster will evenly distribute data ensuring the reliability of data access.

2.5 AURIN data

Australian Urban Research Infrastructure Network is abbreviated to AURIN. The main reason for using AURIN is to compare the data manually collected from AURIN to the tweets stored in the CouchDB. The data was filtered by using keywords such as crime and discrimination, to obtain the relevant dataset. The dataset was further exported in JSON format. Since AURIN's data is historic, we used a database provided by the lecturer for comparison. In the following section of analysis, the output of the comparison is displayed. It can easily be understood in the heat maps as well.

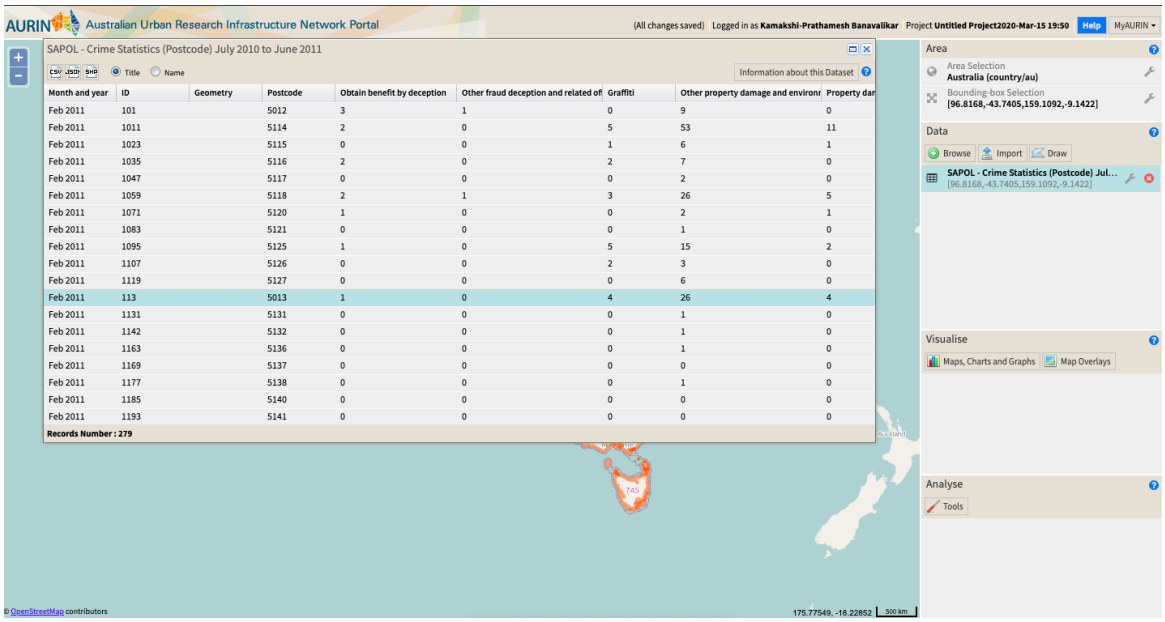

Fig 7 : AURIN Console

2.6 Web App

2.6.1 Architecture

The website consists of four segments. The first segment is the introduction page, which introduces our entire team. It also briefly describes the story being narrated .The second segment is the monitoring desk, that displays the real-time update about tweet harvesting as per fig 5. The third and fourth segment focuses on scenarios related to crime and discrimination, respectively. This data is further reflected in the form of maps and charts. The architecture of the web app is as follows.

1. The twitter data is collected from the Twitter API and the AURIN data is collected from the AURIN official website.
2. This twitter data is then stored in the CouchDB and accessed by the monitor. Furthermore, both these datasets are stored in google storage so that the Maps of Crime and Discrimination can fetch the data.
3. This Architecture was especially chosen so that the website can easily access the data from the web dynamically.

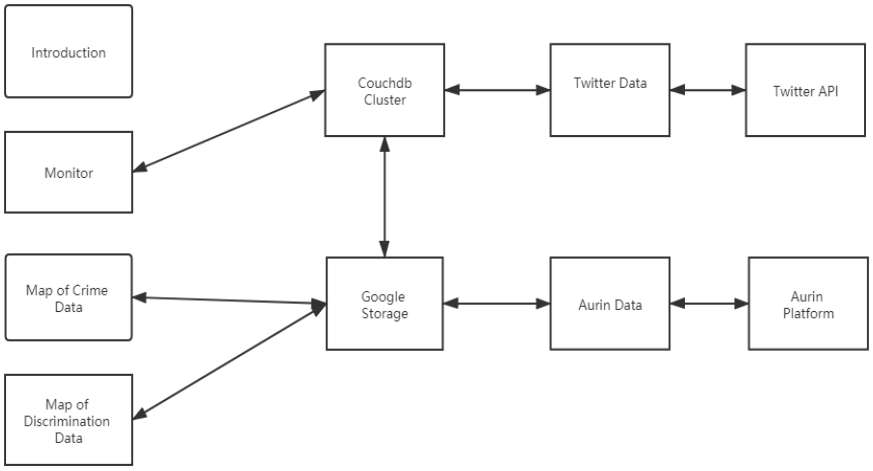

Fig 8: Front End Architecture

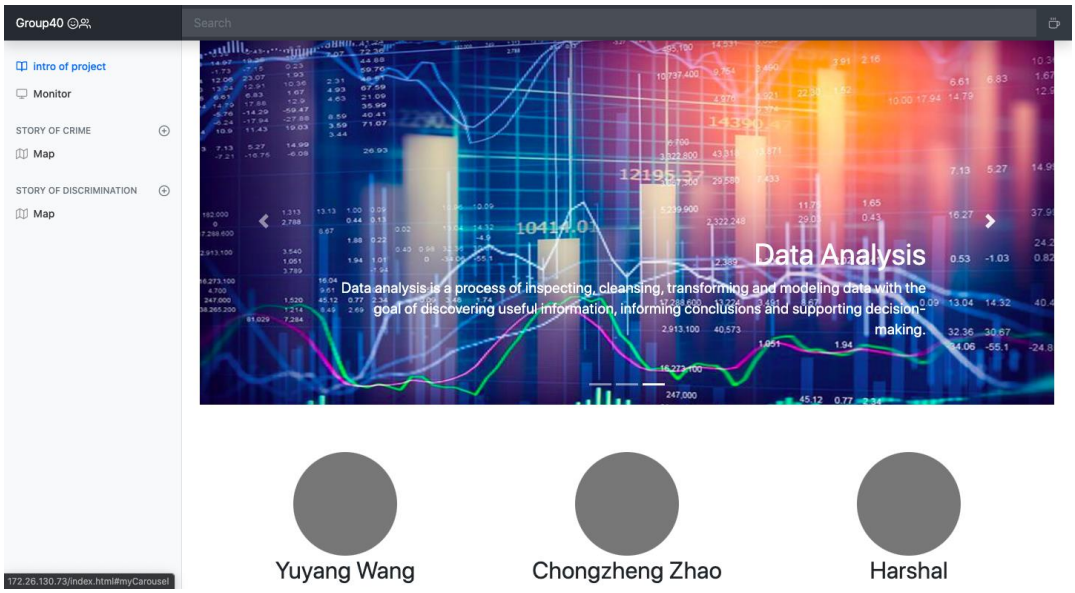

Fig 9 : First Page of the Website

2.6.2 Google Maps

The Google Maps platform uses real-time data and live images to create immersive location experiences that allows users to make more appropriate business decisions. The main advantage of Google Maps is that it allows the users to embed the 'Map' into the website without setting up their own map servers. Heat maps were used to show data from Twitter and AURIN (geojson files). When a click event is triggered, the data for the region selected by the user is converted in the form of pie chart or bar charts, and the user can see these charts for better understanding of the data. These charts show the different types of crime and discrimination present in Australia based on the AURIN dataset. By passing the coordinates of the formatted twitter data into Google's heat map API, the heat map is loaded quickly. If any area is more prominent in red, it implies more data (more crime or discrimination) is present at that particular location. Conversely, if the area is green, it means that there is very little data present in that location. When compared with the AURIN geojson data, users can clearly understand the relationship between the two. This helps in finding meaningful correlation between AURIN and Twitter datasets. Moesif CORS Plugin is a chrome plugin and is needed for efficient working of the web app (Fig 10, Fig 11, Fig 14, and Fig 15).

Echarts, is an open-source visualization library implemented using JavaScript. It relies on the vector graphics library ZRender, which are intuitive, and interaction-rich, as well as highly customizable data visualization charts. After getting the data from google storage, the data is further formatted. To maintain the efficiency, redundant data is deleted.

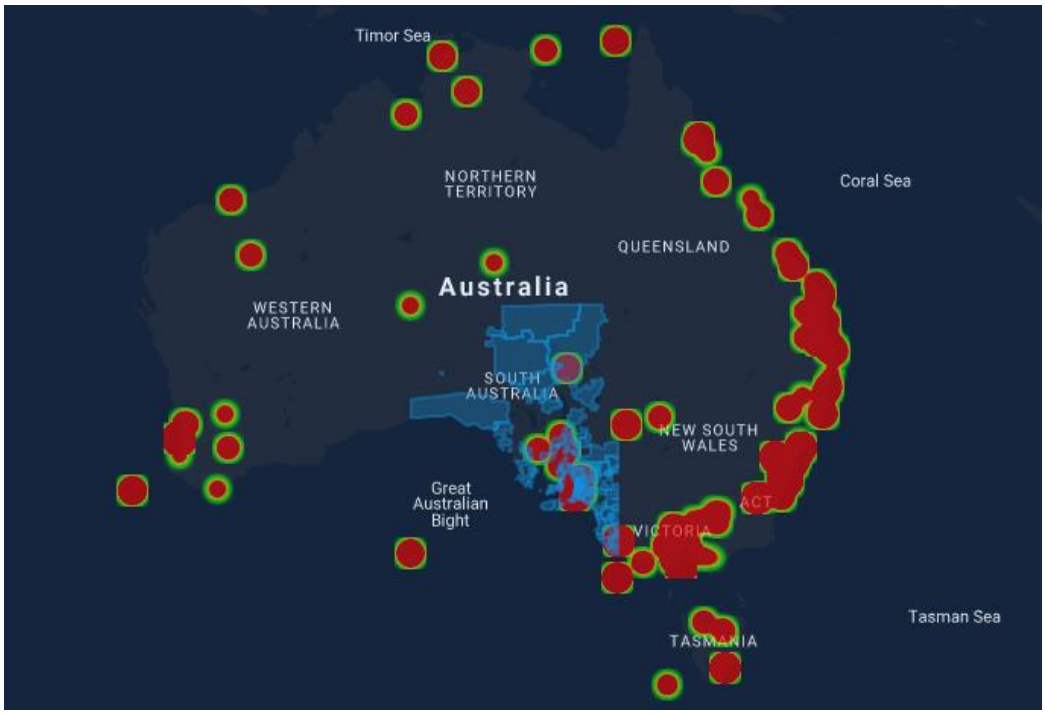

Fig 10 : The heat map of crime data for South Australia

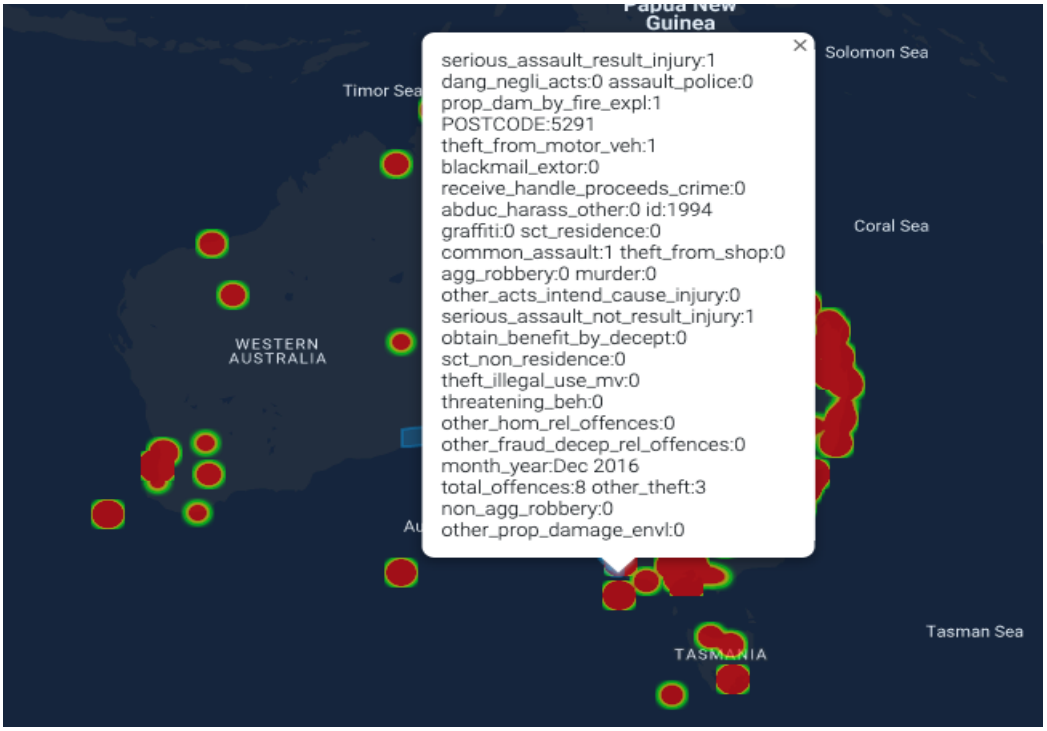

Fig 11 : The detailed regional data of crime for South Australia

Fig 12: The bar chart of crime data

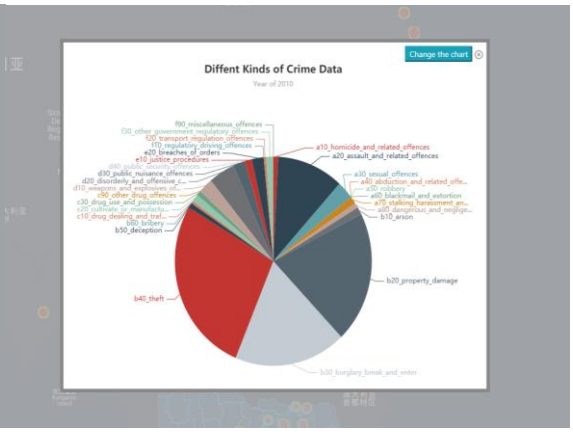

Fig 13 :The pie chart of crime data

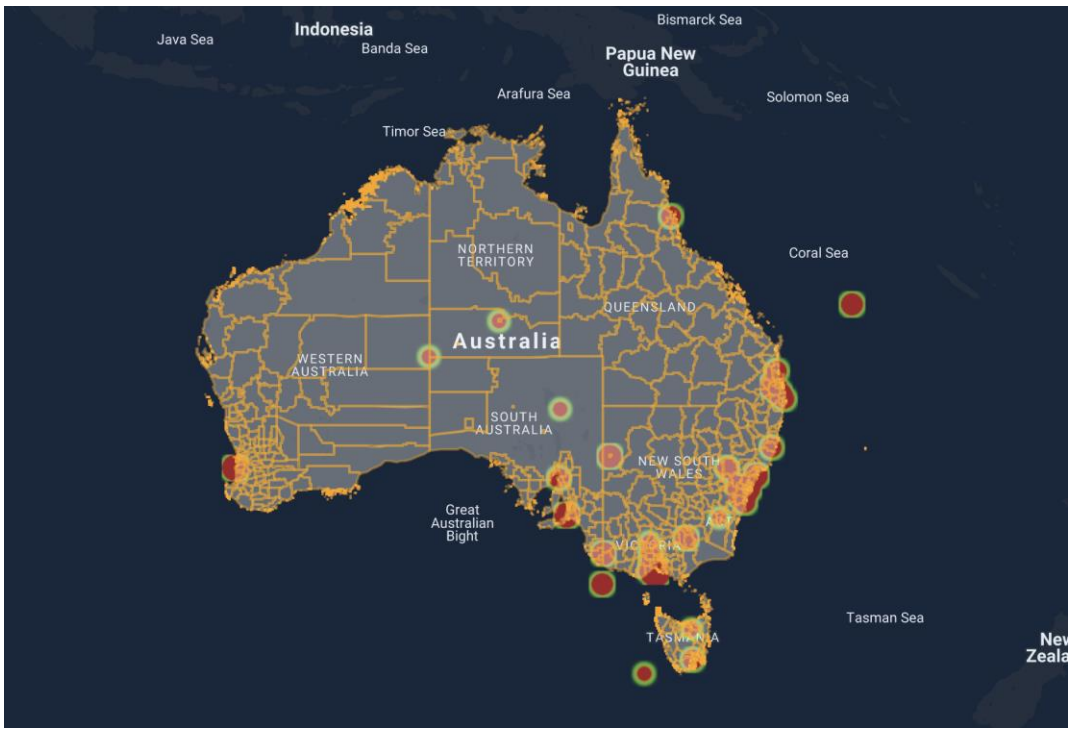

Fig 14 : The heat map of discrimination data for Australia

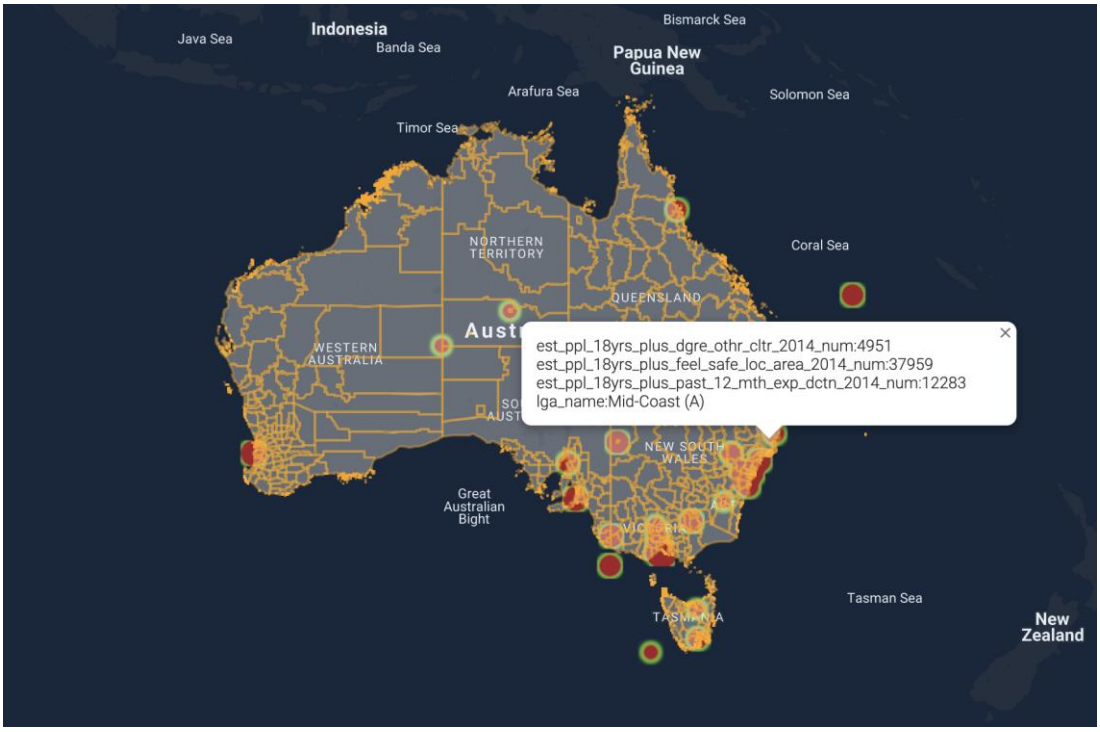

Fig 15 : The detailed regional data of discrimination for Australia

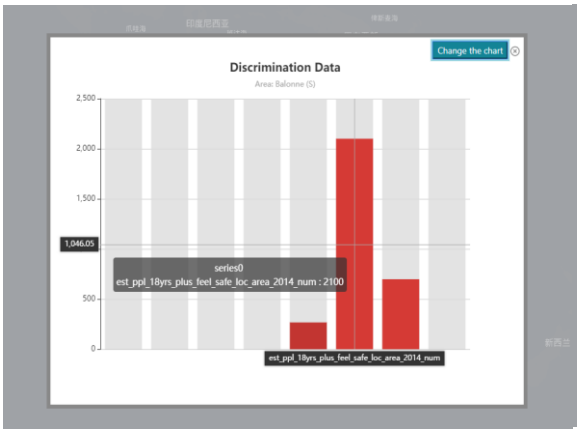

Fig 16 : The bar chart of discrimination data

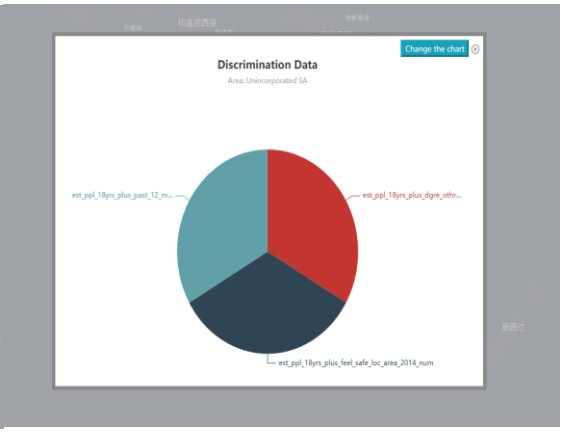

Fig 17 : The pie chart of discrimination data

2.6.3 Deployment of Website

The apache server was deployed by using ansible script and docker was used to automate the installation. The resources such as web pages in the apache directory were placed, with the corresponding security group port number of 8080 open for access.

3. Melbourne Research Cloud

Melbourne Research Cloud as the name suggests, is an on-demand computing services provider for researchers both at the University of Melbourne and affiliated institutions. In order to deploy our cloud-based solution, UniMelb Research Cloud platform has been used to both harvest and store the tweets from Twitter. As part of the project, 4 Instances were provided, 8 virtual CPUs and 200 Gb of Volume Storage. All 4 instances have been utilized and have almost 50 Gb of Volume attached with each instance along with employing all the CPUs to continuously harvest tweets from Twitter and analyze it to gather important insights regarding the crime and discrimination life in Australia.

Security is a very important part of any system that restricts non-authorized people to access the system and make edits or compromise the integrity of the data. So to maintain the security of our system, Security Groups are attached with our Instances with set rules that restrict access and ssh-key-pair to ensure that only people with the private key will be able to access the system on the cloud. All this is automated using an ansible script.

Ansible:

Ansible is based on the Python language and is implemented by three key modules: Paramiko (a concurrently connected ssh host library for python), PyYAML and Jinja2. Ansible is used to set up the instances, CouchDB cluster and web server. For setting up the instance, the ansible automatically creates the volume, security group, and the instances based on the assigned image. Initially the ansible installs the docker, then installs the CouchDB, and auto sets up the cluster. For the web set up, the ansible installs the apache first, then uploads the website folder to the root folder of the web server. The ansible playbooks include several yaml files and a host file. In the design process, all commands that are needed to be executed are placed in one yaml file. It executes this file once on each host, and all tasks can be completed. It is like scripting to complete multiple tasks.

4. Tweet Scenarios

4.1 Hypothesis

The main motivation with respect to the project was to study the increase in the crime rate and the percentage of non-natives facing discrimination while they try and acclimatize in Australia. Our hypothesis checks whether the residents really face crime and discrimination like they tweet. We further check if the language of the tweet has any significance with respect to the discrimination criteria. This hypothesis has been tested by comparing it with the official AURIN data.

4.2 Visualization & Data Analysis

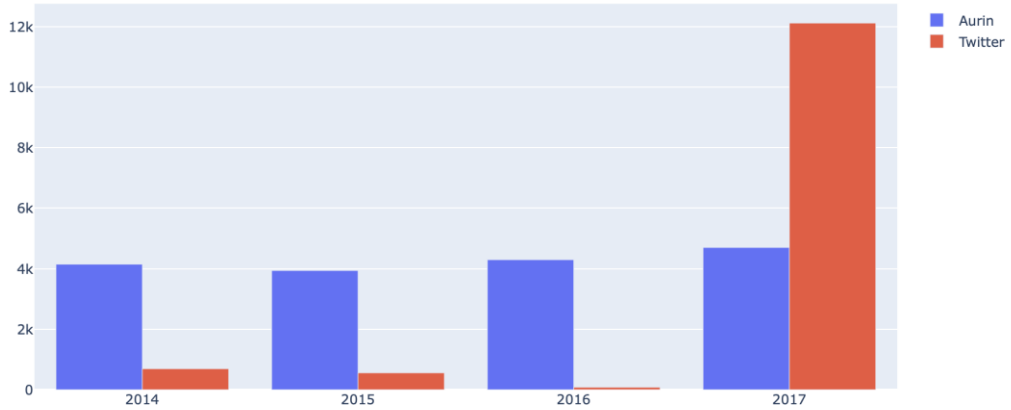

Fig 18 : The Twitter vs AURIN bar graph for crime data (2014-2017)

The AURIN data collected is until 2019 whereas the tweets harvested are the most recent ones hence we use the historic tweets from the year 2014 - 2017 and map it with the AURIN data to find correlation.

In the above graph, we see an overall increasing trend with crime as the central focus. It is observed that in the year 2017 the twitter data has smoothly overtaken the average Aurin data.

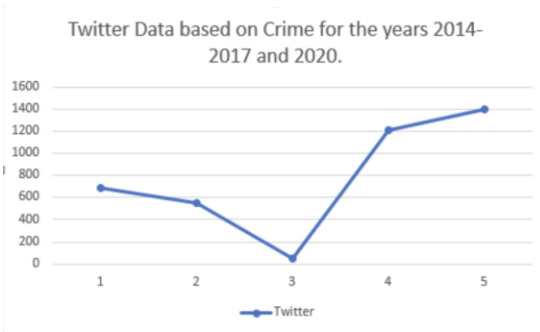

Fig 19 : Line chart for Crime data (2014-2017 & 2020)

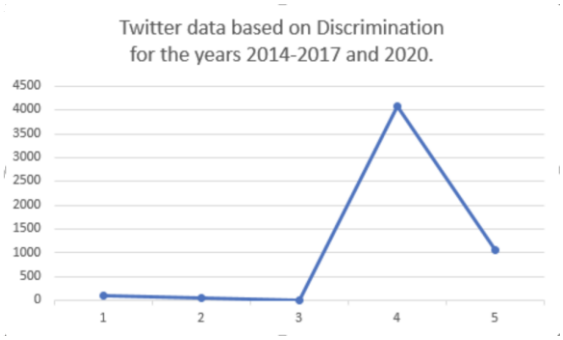

Fig 20 : Line chart for Discrimination data (2014-2017 & 2020)

In the above graphs, for the year 2017 both crime and discrimination are at the highest peak as compared to the other years. It has also been noticed that there is a linear increase in the crime rate whereas there is a vast decrease in the discrimination rate. This can be further proved by the shadow report notes a visit in 2017 by the UN Special rapporteur on Indigenous people that "found deeply disturbing the numerous reports on the prevalence of racism against Aboriginal and Torres Strait Islander Peoples' . ("Australia is still fighting racism and it's time we faced up to it", 2020). The article 4510.0 - Recorded Crime - Victims, Australia, 2017 mentions the rise in criminal

activities such as murder, sexual assault, robbery etc. in the year 2017. ("4510.0 - Recorded Crime - Victims, Australia, 2017", 2020).

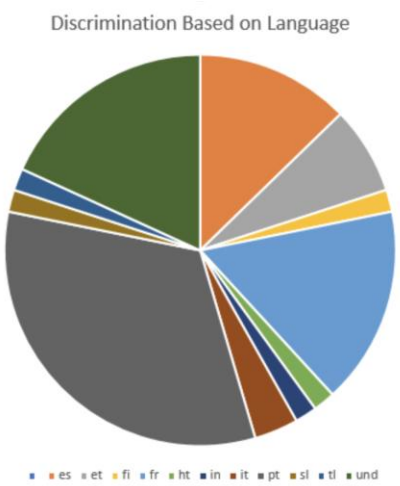

Fig 21 : Pie chart for discrimination with respect to language data

From the historic tweet database, the number of tweets associated to discrimination based on English language are overpowering the other languages by the count of 4179 whereas the other languages sum up to 55. The following pie chart shows distribution of languages other than English.

Date Range	Language	Total Number of tweets	Total Number of Crime tweets	Total Number of Discrimination
16/05/2020 - 25/05/2020	English (en)	153982	633	120
16/05/2020 - 25/05/2020	Other	49998	69	14

Table 1 : Statistics for recent tweets in Australia based on English language

The table 1 shows the number of tweets with respect to language in the entire Australia. From the total number of tweets related to crime 90% of the tweets are written in English. Also, 90% of the total tweets related to discrimination are tweeted in English. Further, Table 2 shows the basic statistics of the number of tweets harvested. It also mentions the number of tweets in context of crime and discrimination on that particular day.

.

Date	Total number of tweets	Total number of Crime tweets	Total number of Discrimination tweet
16/05/2020	3478	18	-
17/05/2020	22713	81	13
18/05/2020	22735	76	14
19/05/2020	23133	80	16
20/05/2020	23746	78	20
21/05/2020	24368	96	18
22/05/2020	24509	81	14
23/05/2020	22912	100	12
24/05/2020	22780	44	15
25/05/2020	13606	48	12

Table 2 : Statistics for recent tweets in Australia

4.3 Conclusion

Life in Australia is very beautiful as it sounds. Though there is abundant diversity, it has been noticed that there has been increase in the discrimination and crime rates in the past few years. Our analysis prove that language is not the only criteria that deals with discrimination. English speaking localities face a lot of criticism with respect to Aboriginal and Torres Strait Islander groups. Though Melbourne has been the second most livable city, crime rate shows a linearly increasing trend. The crime and discrimination rates have been at their peak in the year 2017 leading to Melbourne being the second most livable city in the world, as the major number of offences are in Victoria. This sums up to that, life in the most livable city does not make one's life always happy as the title may suggest.

The research project gave us hands-on experience in making a cloud-based system. The technologies used while making the system gave the team a clear view of what goes on behind the whole cloud computing technology, thereby learning the pros and cons of the technologies used and facing the challenges and solving them. Time was a major constraint during this project, the team would have come up with an advanced system had there been sufficient time.

5. User Guide of the System

System Requirement

1. Python Version >= 3.0
2. Openstack RC File is placed in the root folder
3. Comfortable with Bash Scripting
4. Connect to the University of Melbourne VPN

Deploy the System

1. Run SETUP.SH in command line:

```bash
bash SETUP.SH
```

2. Follow the steps, type in the Openstack Password.
3. (Optional) Type in the system password of your laptop

Access to The Platform

1. Web Visualization and Project Homepage
   172.26.130.73 (No authorization needed)
2. Cloud Monitor
   115.146.94.7:8888 (User: admin, Password: group40)
3. CouchDB Cluster:
   Node 1: 172.26.131.165:5984 (User: admin, Password: group40)
   Node 2: 172.26.130.73:5984 (User: admin, Password: group40)
   Node 3: 172.26.132.193:5984 (User: admin, Password: group40)
4. Software Glance of The System
   Server1: 115.146.94.7
   Jupyter Notebook/ jupyter_notebook:latest
   Node Exporter/ node_exporter:latest
   Grafana/ grafana/grafana:latest
   Server2: 172.26.131.165
   CouchDB/couchdb:2.3.0
   Node Exporter/ node_exporter:latest
   CouchDB Exporter/ couchdb_exporter:latest
   Server3: 172.26.130.73
   CouchDB/couchdb:2.3.0
   Node Exporter/ node_exporter:latest
   CouchDB Exporter/ couchdb_exporter:latest
   Apache/ apache2:latest
   Server4: 172.26.132.193
   CouchDB/couchdb:2.3.0
   Node Exporter/ node_exporter:latest
   CouchDB Exporter/ couchdb_exporter:latest

6. References

1. Melbourne is NOT the most liveable city in the world for the second year running. (2020). Retrieved 24 May 2020, from *https://www.3aw.com.au/melbourne-is-not-the-most-liveable-city-in-the-world-for-the-second-year-running/*

2. Melbourne Youth Australia Project Team. (2018). *Multicultural youth australia census status report 2017/18* [Ebook] (p. 41). Retrieved from Melbourne is NOT the most liveable city in the world for the second year running. (2020). Retrieved 24 May 2020, from *https://www.3aw.com.au/melbourne-is-not-the-most-liveable-city-in-the-world-for-the-second-year-running/*

3. Overview | Prometheus. (2020). Retrieved 25 May 2020, from *https://prometheus.io/docs/introduction/overview/*

4. Australia is still fighting racism and it's time we faced up to it. (2020). Retrieved 25 May 2020, from *https://www.abc.net.au/news/2017-11-27/australias-race-relations-will-be-examined-by-un-in-geneva/9198272*

5. 4510.0 - Recorded Crime - Victims, Australia, 2017. (2020). Retrieved 25 May 2020, from *https://www.abs.gov.au/ausstats/abs@.nsf/Lookup/by%20Subject/4510.0~2017~Main%20Features~Vicitms%20of%20Crime,%20Australia~3*