UNT

Department

of Computer Science and Engineering

CSCE 5300: Introduction to Big Data and Data Science

Chapter 7 (7.1): Practical Applications

3 November 2024

Qutline

- **7.1.1. Introduction**
- **7.1.2. Internet information retrieval, parallel sorting and rank-order filtering**
- **7.1.3. Mining public opinions**
- **7.1.4. Exploring Twitter sentiment analysis and the weather**

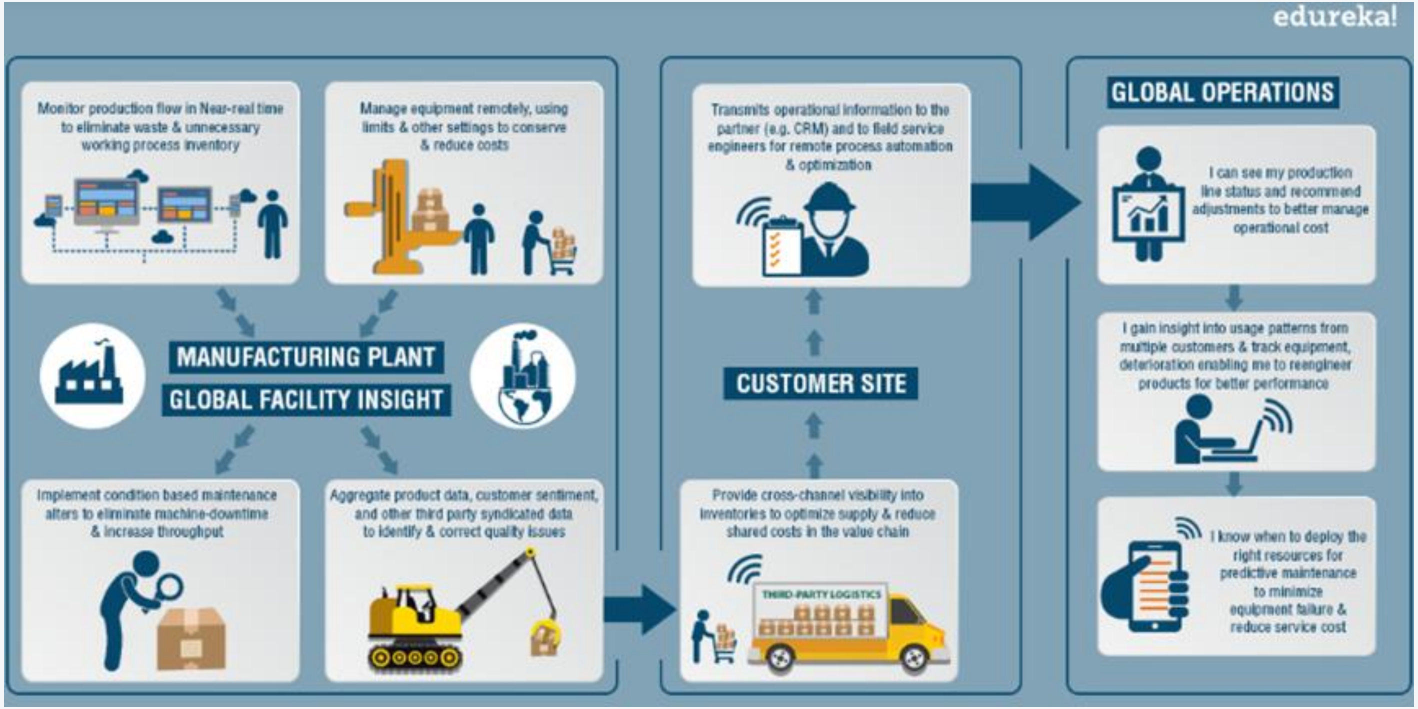
7.1.1. Introduction

- Big Data has many applications. In particular, it is used for tracking customer spending habit and shopping behavior, in recommendation systems, smart traffic systems, secure air traffic systems, auto driving cars, as virtual personal assistant tools, in education, in energy sector, in media and entertainment Sector , and in many other applications [1].

- Examples of using Big Data in different applications are presented in Fig. 7.1 - Fig. 7.3.

- [1] R. Buyya, R.N.Catheiros, A.V. Dastjerdi, *Big Data. Principles and paradigms, Elsevier,* Cambridge, MA, 2016.

7.1.1. Introduction



Monitor production flow in Near-real time to eliminate waste & unnecessary working process inventory

Manage equipment remotely, using limits & other settings to conserve & reduce costs

MANUFACTURING PLANT

GLOBAL FACILITY INSIGHT

Implement condition based maintenance alters to eliminate machine-downtime & increase throughput

Aggregate product data, customer sentiment, and other third party syndicated data to identify & correct quality issues

Transmits operational information to the partner (e.g. CRM) and to field service engineers for remote process automation & optimization

CUSTOMER SITE

Provide cross-channel visibility into inventories to optimize supply & reduce shared costs in the value chain

GLOBAL OPERATIONS

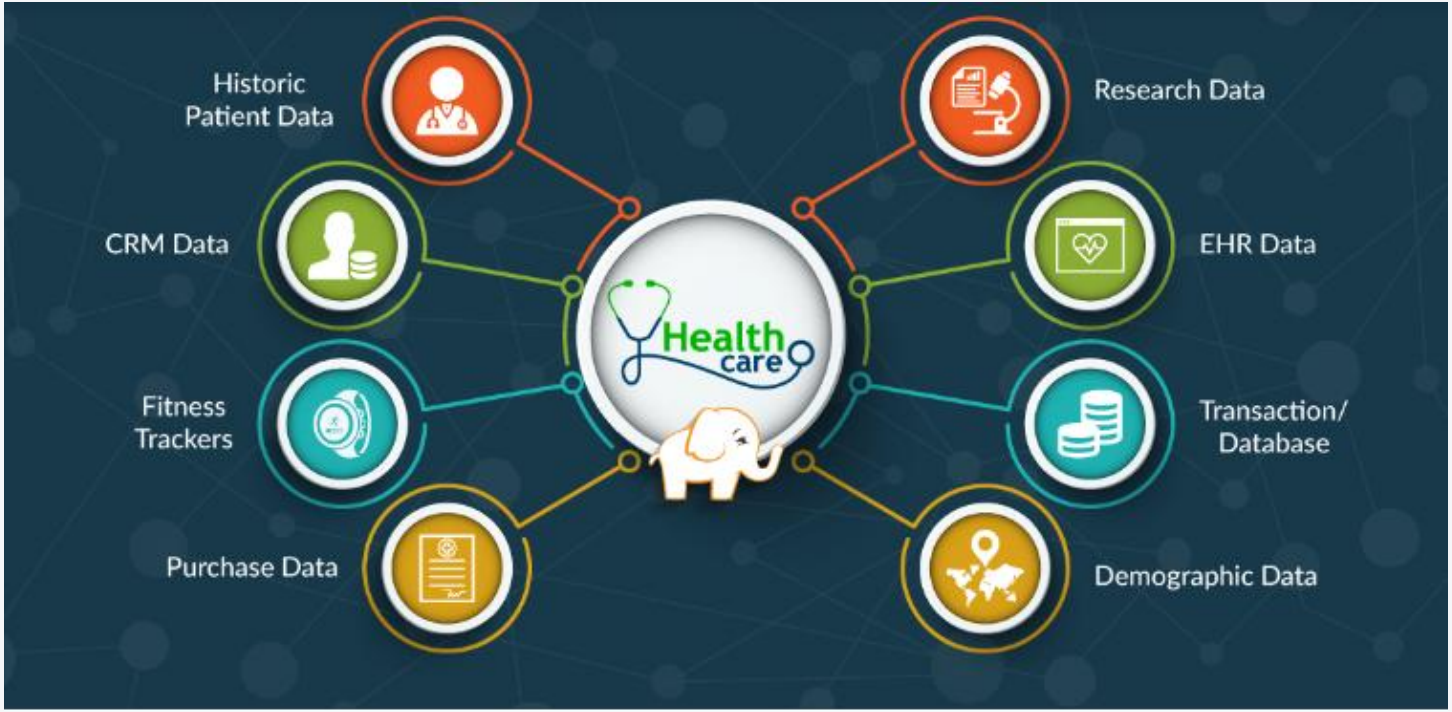
I can see my production line status and recommend adjustments to better manage operational cost

I gain insight into usage patterns from multiple customers & track equipment, deterioration enabling me to reengineer products for better performance

I know when to deploy the right resources for predictive maintenance to minimize equipment failure & reduce service cost

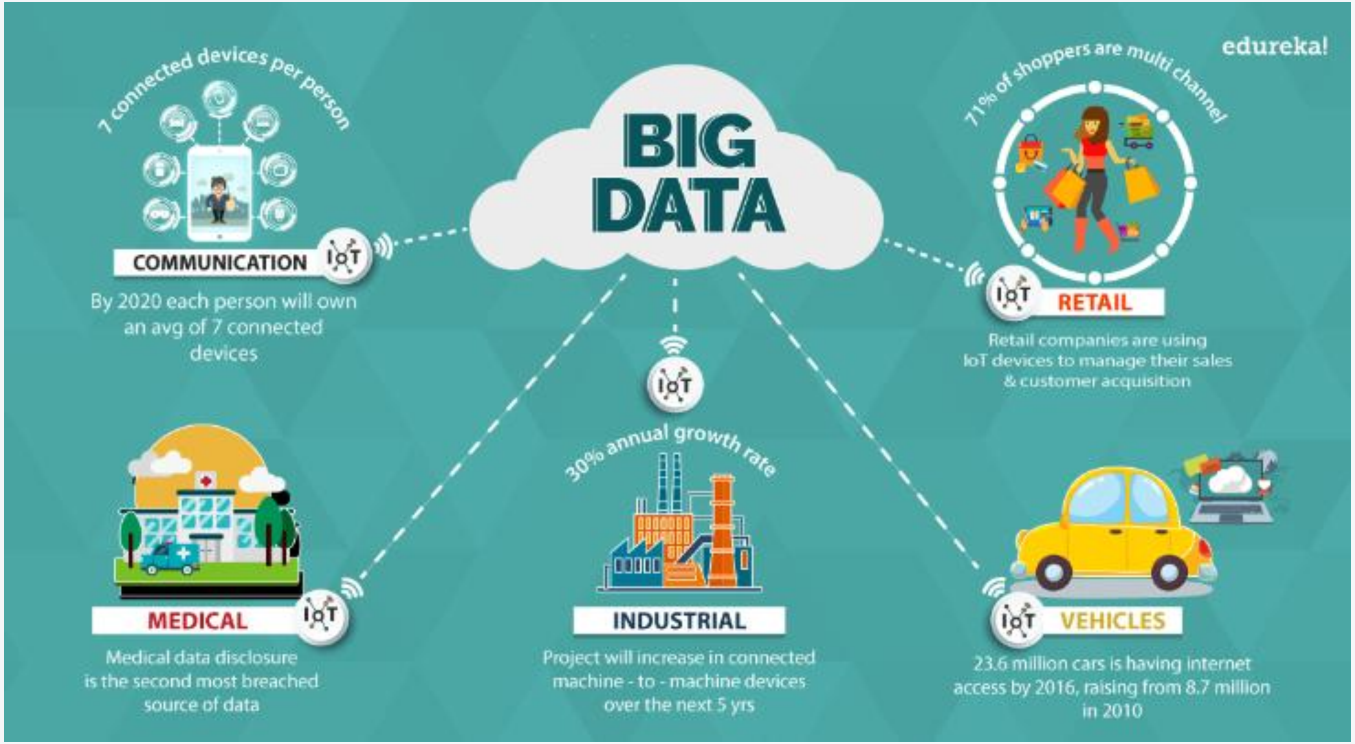
- Fig. 7.1.

7.1.1. Introduction



- Fig. 7.2.

7.1.1. Introduction


edureka!
BIG DATA
7 connected devices per person
COMMUNICATION
IoT
By 2020 each person will own an avg of 7 connected devices
71% of shoppers are multi channel
RETAIL
Retail companies are using IoT devices to manage their sales & customer acquisition
30% annual growth rate
INDUSTRIAL
Project will increase in connected machine - to - machine devices over the next 5 yrs
MEDICAL
Medical data disclosure is the second most breached source of data
VEHICLES
23.6 million cars is having internet access by 2016, raising from 8.7 million in 2010

- Fig. 7.3.

7.1.2. Internet IR, Parallel Sorting and ROF

- Let us consider in detail application of Big Data for Internet information retrieval using artificial neural networks.

- A main part of Section 7.1.2 material is available in [2].

- [2] P. Tymoshchuk and D. Wunsch, “Design of a K-winners-take-all model with a binary spike train,” *IEEE Trans. Syst. Man. Cybern. B, Cybern.*, vol. 49, no. 8, pp. 3131-3140, Aug. 2019.

7.1.3. Mining Public Opinions

- **XDOM**
- **DATA SOURCES**
- Data were collected from four different data sources: Twitter, Facebook, Foursquare and Pantip, as described in Table 7.1.
- These social network data were collected in the Bangkok area. Different data sources require the use of different connectors.
- For Twitter data, Search API provided from Twitter Inc. was used to collect tweets without any keywords.
- Approximately 15 million tweets were collected, or, about 12GB of uncompressed data [10].

- [3] *Big Data. Principles and Paradigms*, Ed. by R. Buyya, R.N.Calheiros, A.V.Dastrejdi, Elsevier, Cambridge, MA, 2016.

7.1.3. Mining Public Opinions

- Table. 7.1.

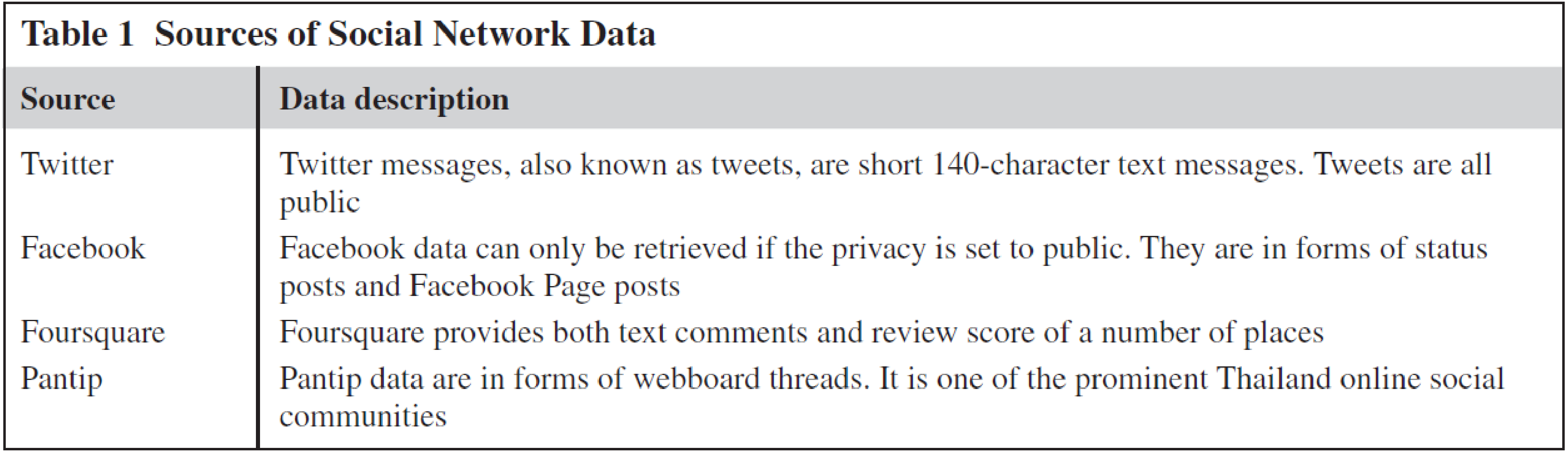


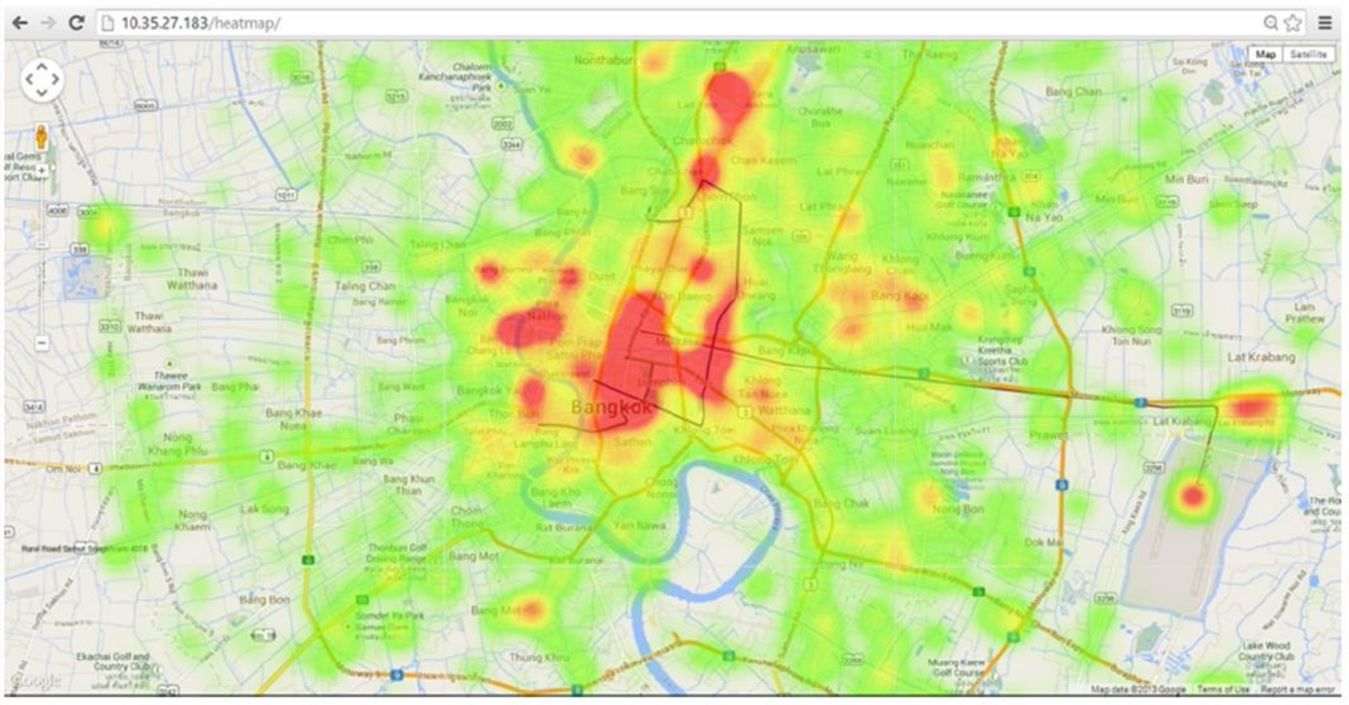
Table 1 Sources of Social Network Data

Source	Data description
Twitter	Twitter messages, also known as tweets, are short 140-character text messages. Tweets are all public
Facebook	Facebook data can only be retrieved if the privacy is set to public. They are in forms of status posts and Facebook Page posts
Foursquare	Foursquare provides both text comments and review score of a number of places
Pantip	Pantip data are in forms of webboard threads. It is one of the prominent Thailand online social communities

7.1.3. Mining Public Opinions

- Each tweet contains multiple data fields, including time, username, user followers, retweet, count, location, and the textual comment.
- The data collected from twitter can be represented as an activity heat map (Fig. 7.4).
- For Facebook, Graph API developed by Facebook Inc was used.
- Unlike Twitter, one can only request and collect data from the Facebook fan page, which consists of posts and comments on specific topics.
- Facebook data were collected from about 5,000 messages, which is approximately 4 MB
- per fan page. Graph API provides attributes including time, username, number of Likes, location, and textual comments for each message.
- For Foursquare, the situation is like Graph API.
- Foursquare providessome useful APIs, named Venues and Tip search API [9], for developers to gather data.
- Foursquare provides comments of places.

7.1.3. Mining Public Opinions

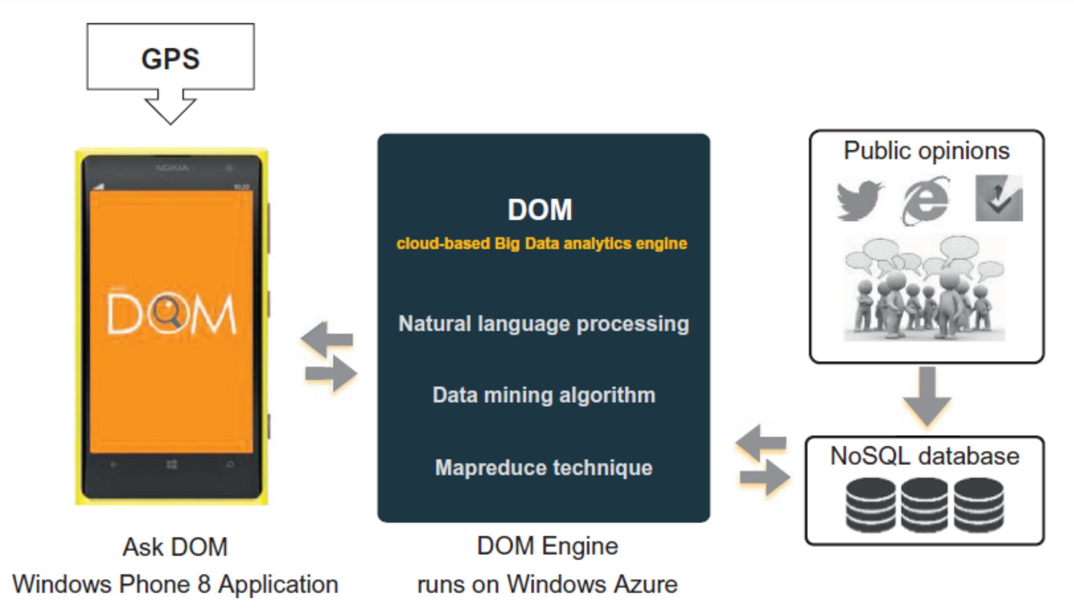

10.35.27.183/heatmap/

- Fig. 7.4.

7.1.3. Mining Public Opinions

- **DOM SYSTEM ARCHITECTURE**
- The original version of DOM is composed of two basic modules: server-side and client-side modules.
- The architecture design of the DOM framework is shown in Fig. 7.5.
- The components of the DOM engine are classified into the server-side, which is a cloud-based cluster.
- The DOM engine is responsible for collecting, analyzing data and distributing the analyzed data to the client-side.
- AskDOM components are client-side.
- The client-side requests the analyzed data, queries, and displays them to end users.

7.1.3. Mining Public Opinions



- Fig. 7.5.

7.1.3. Mining Public Opinions

- Workflow of our framework is as follows.
- Public messages are collected from social networks, blogs and forums using DOM’s crawler module.
- All collected messages are stored in MongoDB, a NoSQL database.
- After that, each message is processed using the basic Natural Language Processing (NLP) technique to parse the text data, categorize its topic, compute its sentimental score, select its representative text in order to form a summary and analyze its influences.
- DOM also uses the MapReduce technique based on the Apache Hadoop framework to reduce processing time.
- DOM periodically processed the data to compute their sentimental score as well as to summarize their opinion text.
- Finally, AskDOM, the mobile application, gets the analyzed data, queries, and displays the information to users according to the inquired-upon topics.

7.1.3. Mining Public Opinions

- The core functions of the DOM engine were designed to support dynamic data.
- There are several features that could be added or further developed to provide additional functionality (e.g, adding more data sources, supporting other languages).
- Since DOM is a cloud-based engine, scalability is also available.
- Furthermore, DOM can be easily applied in various types of usage, on either the community side or the commercial side.

- The current version of DOM consists of five modules, which are MapReduce framework, sentiment analysis, clustering-based summarization framework, influencer analysis, and AskDOM mobile application.

7.1.3. Mining Public Opinions

- **MAPREDUCE FRAMEWORK**
- Since huge data are involved in this project, MapReduce is used.
- If the data is processed sequentially, the processing time would be too large for the practical application.
- The MapReduce technique on the Apache Hadoop framework is therefore the best way to accelerate the analysis speed.

7.1.3. Mining Public Opinions

- **SENTIMENT ANALYSIS**
- Words targeted in which opinions are expressed in each sentence.

- A simple observation was that these sentences always contain sentiment words (eg, great, good, bad, worst).

- To simplify the process, if the sentences do not contain any sentiment words, their sentiment values will be neutral (non-opinions).

- So, the framework was designed to classify the sentiment of each sentence based on

7.1.3. Mining Public Opinions

- To classify each message as positive, neutral or negative, a lexicon-based algorithm to measure the sentiment score of each message was employed.

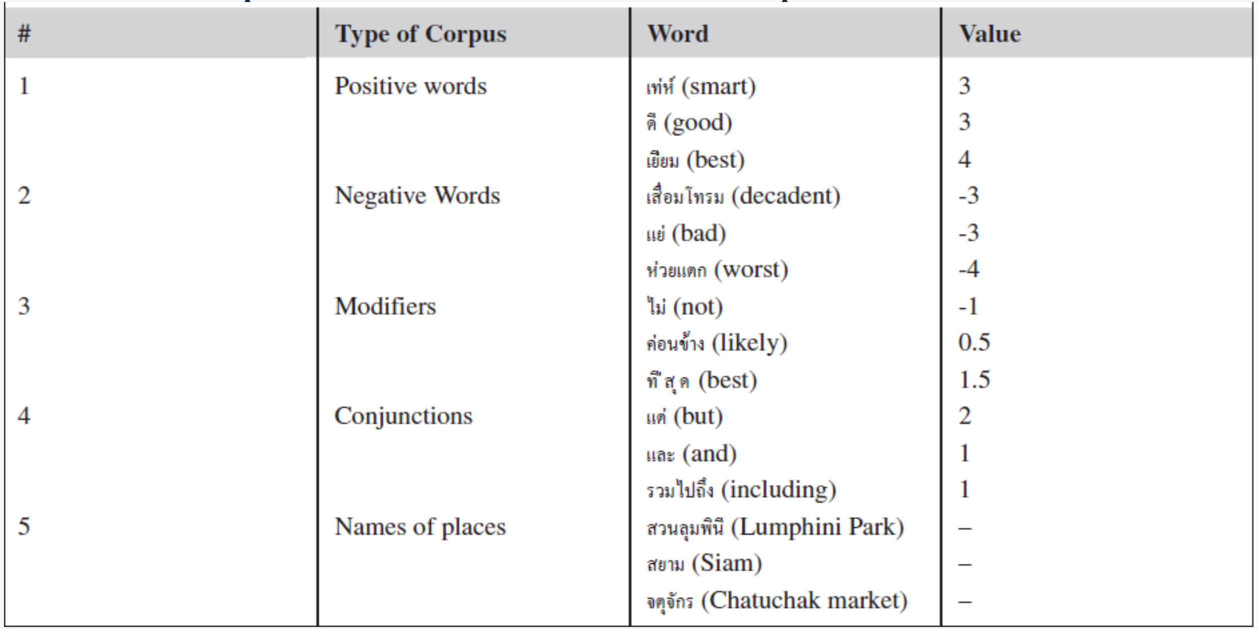
- Five corpora were designed, including positive words, negative words, modifiers, conjunctions, as well as the names of points of interest.

- Each word in the two sentiment corpora, positive words and negative words, contains sentiment ratings ranging from −5 to 5.

- The examples of our corpuses are shown in Table 7.2.

7.1.3. Mining Public Opinions

- Table 7.2. The Examples of Sentences in the Corpora.



#	Type of Corpus	Word	Value
1	Positive words	เท่ห์ (smart)	3
		ดี (good)	3
		เยี่ยม (best)	4
2	Negative Words	เสื่อมโทรม (decadent)	-3
		แย่ (bad)	-3
		ห่วยแตก (worst)	-4
3	Modifiers	ไม่ (not)	-1
		ค่อนข้าง (likely)	0.5
		ที่สุด (best)	1.5
4	Conjunctions	แต่ (but)	2
		และ (and)	1
		รวมไปถึง (including)	1
5	Names of places	สวนลุมพินี (Lumphini Park)	–
		สยาม (Siam)	–
		จตุจักร (Chatuchak market)	–

7.1.3. Mining Public Opinions

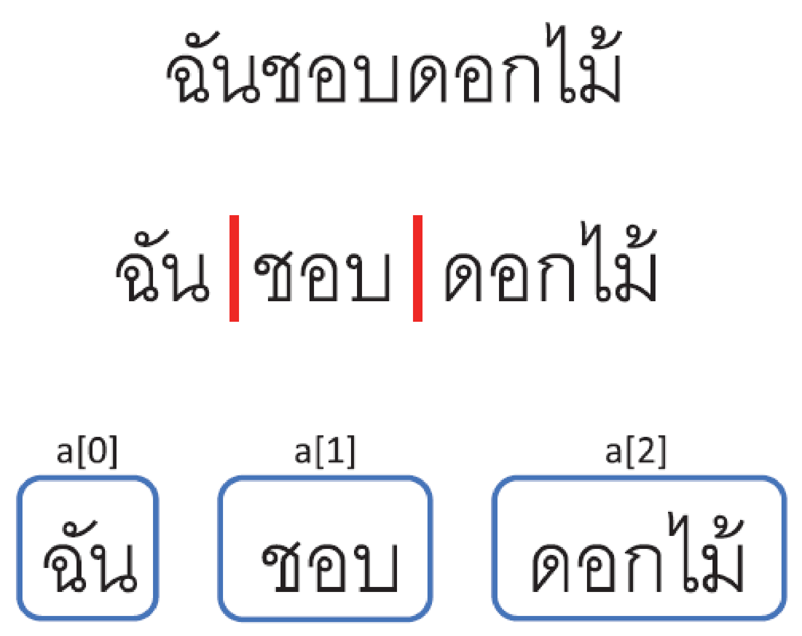
- DOM detects and matches words and their sentiment polarity by using these corpora.

- Since the nature of Thai sentence structure is continuous without any white space breaks between words, it is necessary to tokenize each sentence into a group of words.
- In this process, “LexTo”, the open source Thai word tokenizer was used, to tokenize words in each sentence and then store them as an array using the longest word matching algorithm [12].

- An example of this procedure is shown in Fig. 7.6.

7.1.3. Mining Public Opinions

- Fig. 7.6. Example of Thai word tokenization.



ฉันชอบดอกไม้

ฉัน | ชอบ | ดอกไม้

a[0] ฉัน

a[1] ชอบ

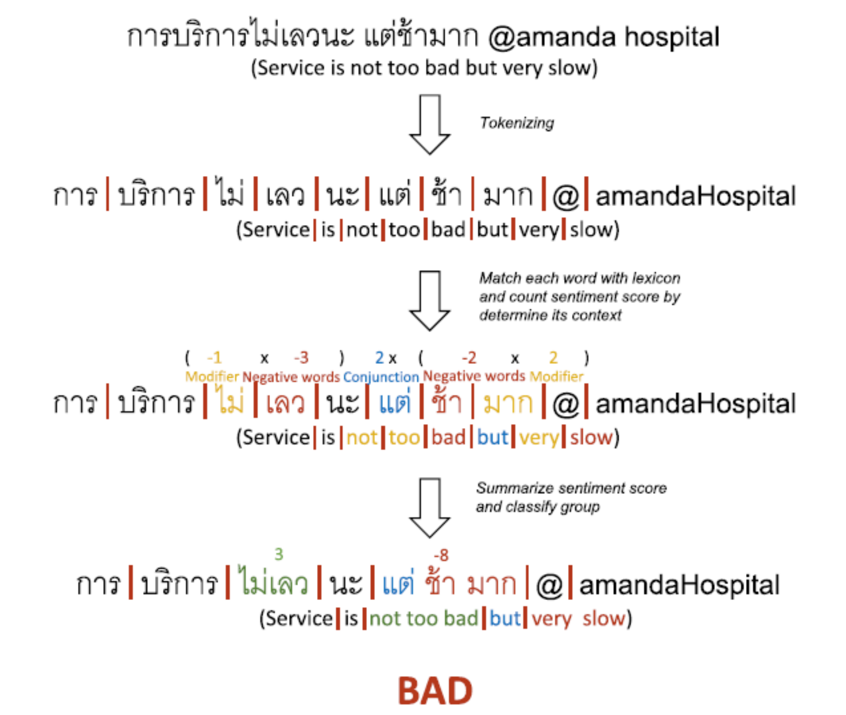
a[2] ดอกไม้

7.1.3. Mining Public Opinions

- DOM generates small jobs to detect words of each sentence in parallel.
- First of all, DOM filters the nonrelated sentences out by matching words with the names of POI corpus.
- After that, only sentences that relate to specific topics of interest (in this case, points of interest) would remain. DOM then iteratively matches sentiment keywords with remaining corpuses.
- If there are sentiment words in an array, DOM collect its sentiment score and summarizes it at the end of each sentence.
- DOM then automatically classifies each sentence into a sentiment group: positive, neutral or negative, depending on its score band (the range of distributed sentiment score).
- DOM not only determines keywords from sentences, but also determines the context of each sentence. The positions of words, modifiers, conjunctions, and emoticons are also determined in our framework. In some cases, these words can be important clues to emphasizing the mood of the sentences.
- Especially for the modifier keywords, they can invert the sentiment score if their positions are adjacent to the sentiment words as illustrated in Fig. 7.7.

7.1.3. Mining Public Opinions

- Fig. 7.7. Example
- of Thai sentiment
- analysis.



การบริการไม่เลวนะ แต่ช้ามาก @amanda hospital
(Service is not too bad but very slow)

Tokenizing

การ | บริการ | ไม่ | เลว | นะ | แต่ | ช้า | มาก | @ | amandaHospital
(Service | is | not | too | bad | but | very | slow)

Match each word with lexicon and count sentiment score by determine its context

(-1 x -3) 2 x (-2 x 2)
Modifier Negative words Conjunction Negative words Modifier
การ | บริการ | ไม่ | เลว | นะ | แต่ | ช้า | มาก | @ | amandaHospital
(Service | is | not | too | bad | but | very | slow)

Summarize sentiment score and classify group

3 -8
การ | บริการ | ไม่เลว | นะ | แต่ ช้า มาก | @ | amandaHospital
(Service | is | not too bad | but | very slow)

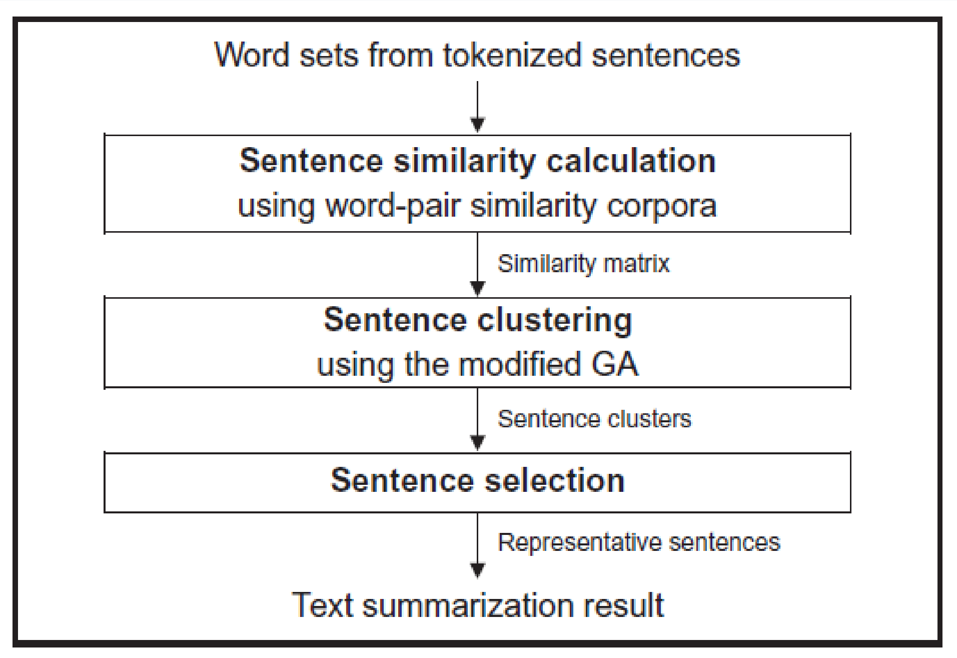
BAD

7.1.3. Mining Public Opinions

- **CLUSTERING-BASED SUMMARIZATION FRAMEWORK**
- Due to the high impact of textual opinions on decision making, a review summarization system is inevitable, as the system provides a shorter version of informative text apart from overall numerical sentiments.
- As a result, readers can quickly understand major authors’ opinions without losing any key points.

- The opinion summarization framework is able to produce a representative and non-duplicate textual summary. Fig. 7.8 presents a framework architecture which is composed of three processes—a sentence similarity calculation, a modified genetic algorithm (GA) sentence clustering, and a sentence selection.

7.1.3. Mining Public Opinions


Word sets from tokenized sentences
Sentence similarity calculation
using word-pair similarity corpora
Similarity matrix
Sentence clustering
using the modified GA
Sentence clusters
Sentence selection
Representative sentences
Text summarization result

- Fig. 7.8. The architecture of our clustering-based summarization framework.

7.1.3. Mining Public Opinions

- After completing the sentence analysis task, each sentence is represented with its tokenized words.
- First, the framework takes these preprocessed sentences to generate a semantic similarity matrix through a sentence similarity calculation process.
- This matrix reflects semantic similarity relations between sentences.
- Unlike existing works, a semantic similarity corpora was created in order to identify similarity levels between Thai word pairs.
- Subsequently, the modified GA assigns sentences into clusters based upon the similarity matrix.

- In a sentence selection process, a final summary is created by selecting a representative sentence of each generated cluster.

7.1.3. Mining Public Opinions

- The following subsections present the details of each component.
- **(1)** Sentence Similarity Calculation
- Since the sentence clustering process aims to assign semantically similar sentences into the same clusters, and vice versa, the similarity values between every sentence pair must be calculated before performing the next process.
- Unlike the original method, a similarity corpora for Thai language was created to determine word-pair similarity scores, which range from 0 to 1.
- The lower values between two comparing words indicate the lower similarity relations.
- The example of the corpora is shown in Table 7.3.

7.1.3. Mining Public Opinions

- Table 7.3. The Examples of the Word-Pair Similarity Corpora.

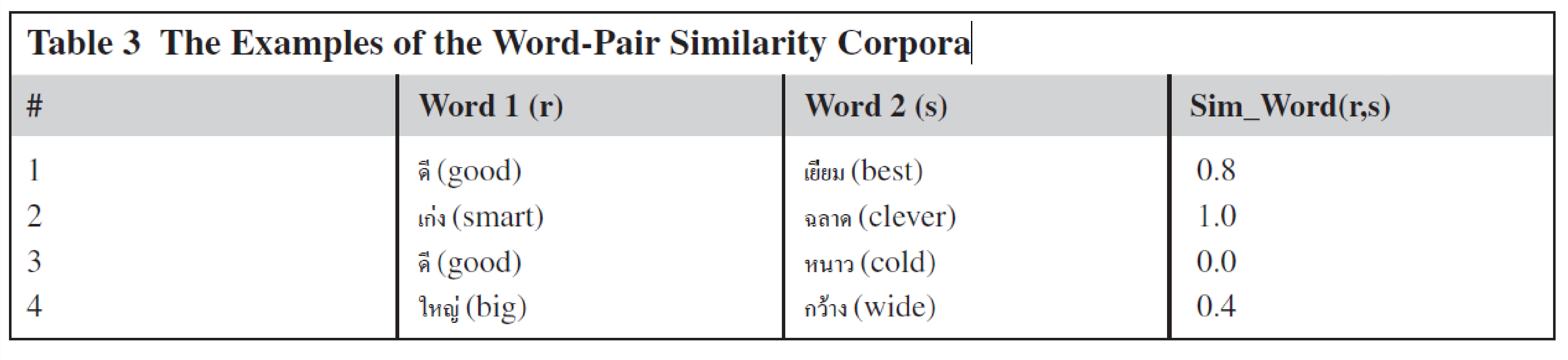


Table 3 The Examples of the Word-Pair Similarity Corpora

#	Word 1 (r)	Word 2 (s)	Sim_Word(r,s)
1	ดี (good)	เยียม (best)	0.8
2	เก่ง (smart)	ฉลาด (clever)	1.0
3	ดี (good)	หนาว (cold)	0.0
4	ใหญ่ (big)	กว้าง (wide)	0.4

7.1.3. Mining Public Opinions

- Each sentence Si is represented with its tokenized words, Wi = {w1,w2,…,wn} where n is number of words in sentence Si.
- The similarity score, sim(S1,S2), is derived from a cosine similarity between two semantic vectors (V1 and V2) which represent similarity relations of a sentence pair (S1 and S2), denoted as
- $$\mathrm{sim}\left(S_1,S_2\right)=\frac{V_1 \bullet V_2}{\| V_1 \| \bullet \| V_2 \|} \quad . \qquad (7.1)$$

7.1.3. Mining Public Opinions

- To create the semantic vectors (V1 and V2), a union word set U is constructed by merging two words sets, U = W1 ∪ W2 = {u1,u2,…,uq} where q = |U|, of two comparing sentences (S1 and S2).
- After that, each element of two semantic vectors, V1 = {v11,v12,…,v1q} and V2 = {v21,v22,…,v2q}, is created from the similarity scores, Sim_Word(r,s), between a word pair (r and s) in the similarity corpora, denoted as
- $$v_{1i} = \begin{cases} 1, \mathit{if}\ u_i \in W_1 \\ \max_{a \in W_1} \mathrm{sim}_{\mathrm{word}(u_i, a)}, \mathrm{if}\ u_i \notin W_1 \end{cases}, v_{2i} = \begin{cases} 1, \mathit{if}\ u_i \in W_2 \\ \max_{b \in W_2} \mathrm{sim}_{\mathrm{word}(u_i, b)}, \mathrm{if}\ u_i \notin W_2 \end{cases}, \quad (7.2)$$

7.1.3. Mining Public Opinions

- where $i \in \{1,2,\ldots,q\}$ and $ui \in U$.

- At the end of this process, the similarity values of all sentence pairs are assembled into the semantic similarity matrix, $M = \{m00, m01,\ldots, mhh\}$ where h is the number of opinion sentences.

- For example, a value of element m02 in a similarity matrix indicates the similarity score between two sentences, S0 and S2.

7.1.3. Mining Public Opinions

- **(2)** Sentence Clustering
- The objective of our summarization framework is to select underlying text from each cluster.
- In order to achieve the expectation, the sentence clustering process assists in dividing semantically similar sentences into the same clusters based on the similarity matrix generated in the previous process.
- In this work, the sentence clustering problem is formulated as an optimization problem which attempts to minimize dissimilarity between sentences in the same clusters.
- To solve the optimization problem, the genetic algorithm is applied to find near globally optimal results.

7.1.3. Mining Public Opinions

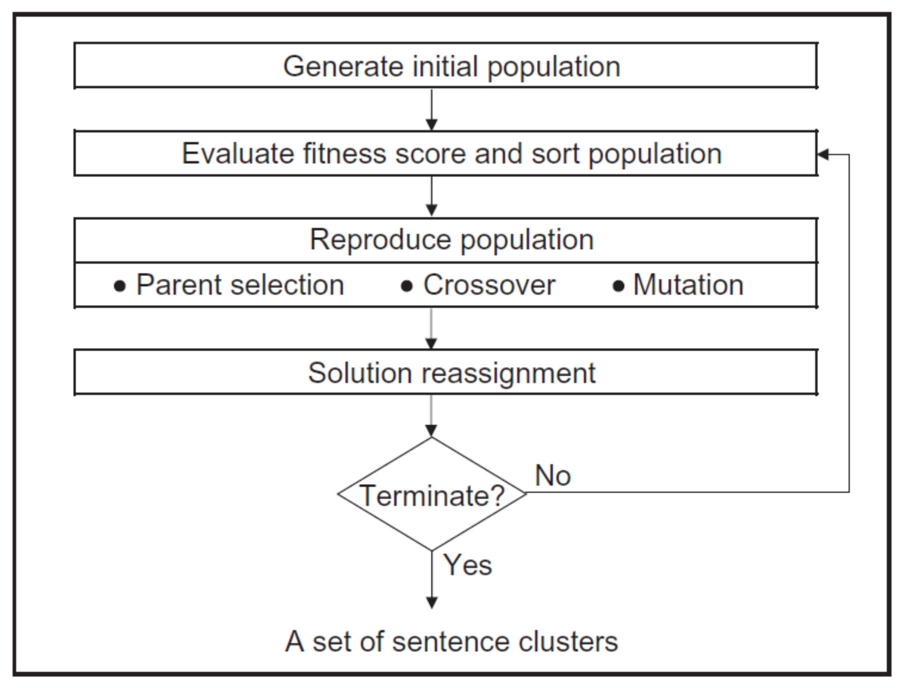
- As the sentence clustering problems contain few good results, the GA suffers from slow convergence.
- To boost up the algorithm, the concept of membership degree in data clustering was utilized to form an additional solution reassignment operation of our modified GA.
- In general, a sentence has different degrees of being a member in any cluster.
- The degree is defined as the similarity level of any sentence to all members in a particular cluster.
- The higher degrees of sentences in any cluster reflect a higher likelihood that they will be in that cluster.
- Thus, a sentence should be assigned to the cluster that has the highest degree of belonging.

7.1.3. Mining Public Opinions

- With this clustering characteristic, all feasible solutions of the modified GA in every generation were reassigned.
- By doing this, the algorithm considers only the solutions which satisfy with this clustering characteristic, resulting in faster convergence.

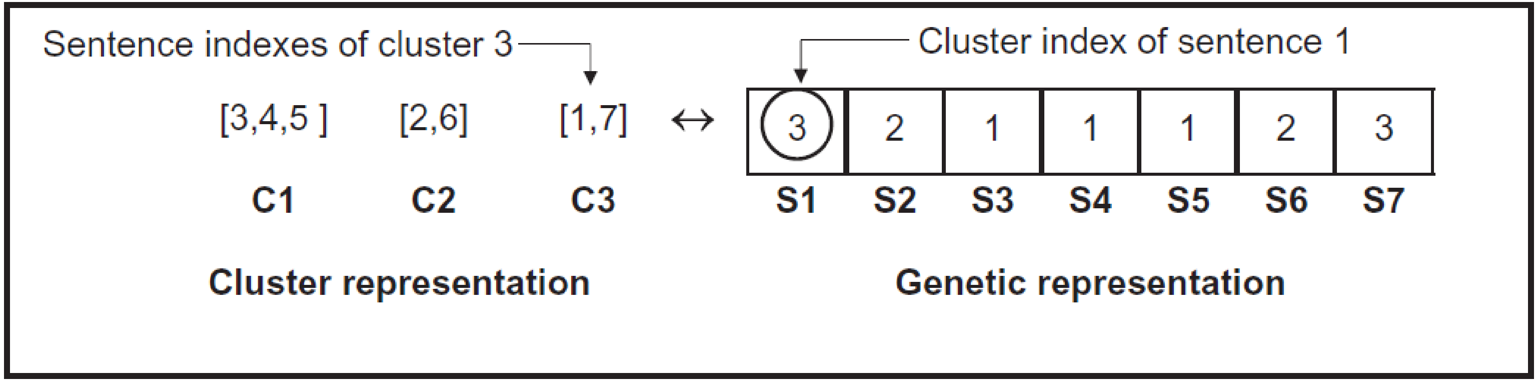
- The overall flowchart of the modified GA is shown in Fig. 7.9.
- First, the algorithm randomly generates feasible solutions (i.e., individuals) and encodes them into the genetic representation.
- To represent a sentence clustering solution, a string of n-digit integers was used, where n is the number of sentences.
- Each digit presents the cluster index of the corresponding sentence, as illustrated in Fig. 7.10.

7.1.3. Mining Public Opinions


Generate initial population
Evaluate fitness score and sort population
Reproduce population
• Parent selection
• Crossover
• Mutation
Solution reassignment
Terminate?
No
Yes
A set of sentence clusters

- Fig. 7.9. The flowchart of the modified genetic algorithm.

7.1.3. Mining Public Opinions



Sentence indexes of cluster 3

Cluster index of sentence 1

[3,4,5]	[2,6]	[1,7]	↔	3	2	1	1	1	2	3
C1	C2	C3		S1	S2	S3	S4	S5	S6	S7

Cluster representation

Genetic representation

- Fig. 7.10. The example of 7-digit integer representation.

7.1.3. Mining Public Opinions

- All generated individuals are then formed to a population.
- After that, a fitness score of each individual is calculated by using the intra cluster dissimilarity function.
- Then, all individuals are sorted by their fitness scores.
- The population of the next generation is derived from the best individuals (ie, elitisms) and the reproduced individuals (ie, offspring).
- To generate each offspring, two individuals from the current population are selected as parents in order to perform a crossover operation.
- In the crossover operation, the parents exchange their genes based on random probability in order to create new individuals (ie, offspring).
- Subsequently, the mutation operation is applied only on offspring in which the random values of mutation exceed the predefined values.
- The mutated offspring are randomly changed in the cluster index at a random point.

7.1.3. Mining Public Opinions

- After the population of the next generation is generated, each individual F represents a clustering solution, $C = \{c1, c2, \ldots, ce\}$ where $e = |C|$, of sentences $S = \{S1, S2, \ldots, Sn\}$ where $n = |S|$.
- To refine the current solution, the individual is then fetched into the solution reassignment process.
- In this process, the value of each digit of an individual is altered according to a reassignment function ϕ(Si) which determines new cluster index for sentence Si.

- This function will reassign a sentence Sx to the cluster Cy that has the highest membership degree, $dxy \in [0,1]$ where x and y are the sentence index and cluster index respectively, denoted as

7.1.3. Mining Public Opinions

- $$\varphi(S_x) = \operatorname{argmax}_y d_{xy} \qquad (7.3)$$

- According to the similarity matrix, each element mij indicates a similarity score sim(Si,Sj) between two sentences (Si and Sj).
- The membership degree dxy is derived from weighted sums of total similarity scores between sentence Sx and all sentence members in cluster Cy, denoted as

- $$d_{xy} = \frac{\sum_{a \in C_y} \operatorname{sim}(S_x, a)}{\sum_{i=1}^{k} \sum_{b \in C_i} \operatorname{sim}(S_x, b)}, \qquad (7.4)$$

7.1.3. Mining Public Opinions

- where *d i x k* = = å 1 1 and *x* ϵ {1, 2, …, *n*}.
- Later, all reassigned individuals are formed to the population of the next generation.
- The algorithm iteratively performs until a termination criteria is met.
- After termination, the final clustering solution is described by the best-scored individual.
- Owing to the large number of opinion texts, it becomes difficult for readers to read all relevant text and draw conclusions.
- Taking the generated clusters from the sentence clustering process, the sentence selection assists in selecting an underlying sentence from each cluster based on a representative
- score.
- The higher scores reflect the sentences that are more similar to other sentences in the same cluster.

7.1.3. Mining Public Opinions

- In this work, the representative sentence of a cluster is defined as the most similar sentence.
- Thus, for each cluster, a sentence which has the highest score as the representative sentence was selected.
- After all representatives are selected, a list of representatives with their cluster sizes is presented in the final summary.
- The size of each cluster can indicate its impact on the opinion data; that is, the larger clusters reflect more impact opinions.
- The textual summary of this framework provides underlying reasons to support the sentiment analysis.
- This additional information helps readers make better and stronger decisions, resulting in business success.
- In other words, the opinion summarization framework is added to increase the reliability of making decisions in a DOM engine.

7.1.3. Mining Public Opinions

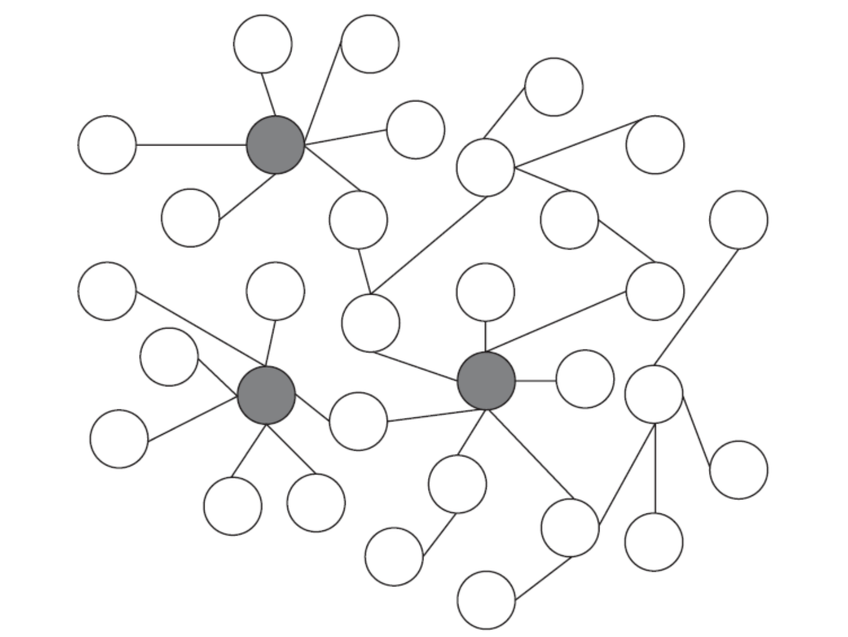
- **INFLUENCER ANALYSIS**
- The rise of social media platforms such as Twitter, with their focus on user-generated content and social networks, has brought about the study of authority and influence over social networks to the forefront of current research.
- For companies and other public entities, identifying and engaging with influential authors in social media is critical, since any opinions they express can rapidly spread far and wide.
- For users, when presented with a vast amount of content relevant to a topic of interest, sorting content by the source’s authority or influence can also assist in information retrieval.
- In the social network community, a variety of measures were designed for the measurement of importance or prominence of nodes in a network.

7.1.3. Mining Public Opinions

- In the following, we will briefly summarize the centrality measure that was used to describe possible candidate indicators for the power of influential in message diffusion.
- For the DOM engine, “Degree centrality” to identify influential users in Twitter’s networks was used.

- Degree centrality is the simplest centrality measure, as illustrated in Fig. 7.11.
- The degree of a node *I* denoted by *ki*, is the number of edges that are incident with it, or the number of nodes adjacent to it.
- For networks where the edges between nodes are directional, it id necessary to distinguish between in-degree and out-degree.
- The out-degree centrality is defined as
- $$C_{D_o}(i) = \sum_{j=1}^{n} a_{ij} \qquad (7.5)$$

7.1.3. Mining Public Opinions



- Fig. 7.11. Simulation of Influencer network graph in the Twitter’s networks.

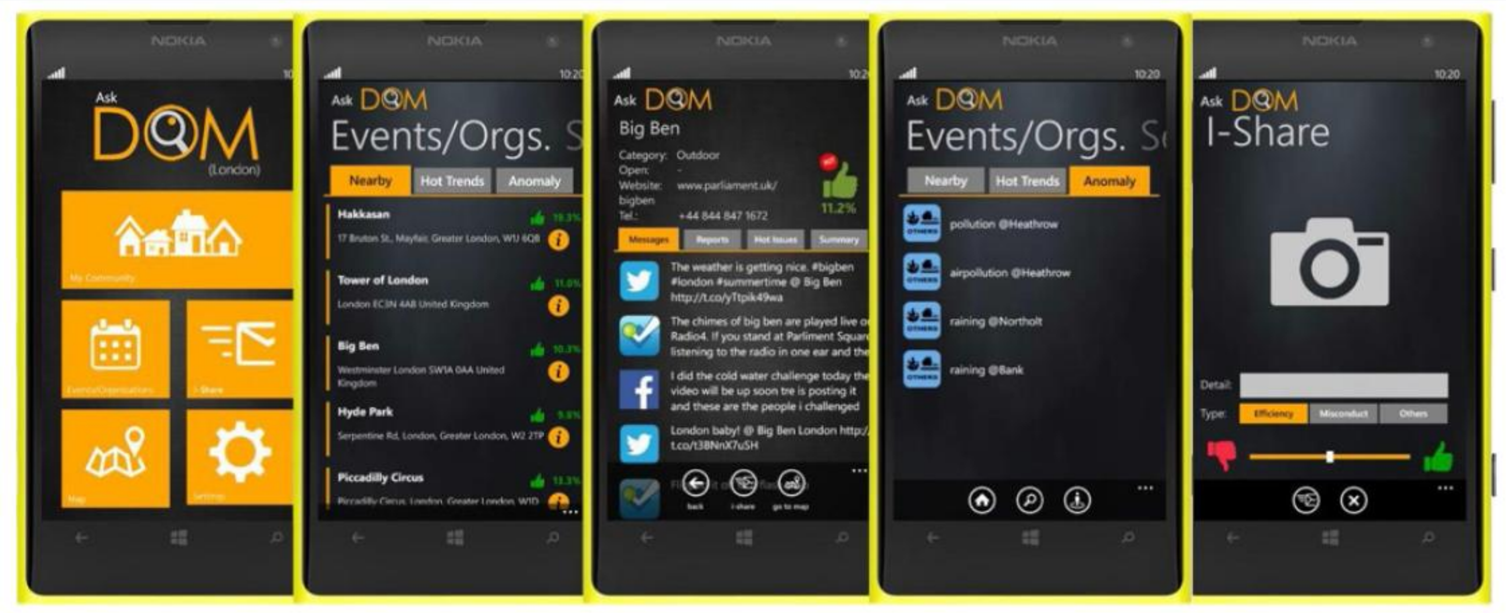
7.1.3. Mining Public Opinions

- where *aij* is 1 in the binary adjacency matrix A if an edge from node *i* to *j* exists, otherwise it is 0.
- Similarly, the in-degree centrality is defined as
- $$C_{D_1}(i)=\sum_{j=1}^{n} a_{ji} \quad , \qquad (7.6)$$
- where *i* describes the node *i* and *aji* is 1 if an edge from node *j* to *i* exists, otherwise it is 0.

7.1.3. Mining Public Opinions

- **AskDOM: MOBILE APPLICATION**
- To utilize DOM to its fullest extent, AskDOM (Fig. 7.12) was developed, a mobile solution designed to use DOM to provide a means for the general public to help improve their own communities by providing reviews, feedback, and ratings of service providers automatically analyzed from public opinions on social networks (Twitter, Facebook, Pantip and Foursquare).
- AskDOM comprises two important modules: (a) a front-end interface with features designed to connect users to service providers such as I-Share (direct feedback), Map (traffic and incident map), Anomaly (abnormal situations reports), and (b) the DOM Engine, the back-end system that periodically gathers and processes social network data, performs public sentiment analysis, discovers underlying textual opinions, determines relationship influencers, and conducts natural language processing for both Thai and English.

7.1.3. Mining Public Opinions



Hakkasan
17 Bruton St., Mayfair, Greater London, W1J 6QB

Tower of London
London EC3N 4AB United Kingdom

Big Ben
Westminster London SW1A 0AA United Kingdom

Hyde Park
Serpentine Rd, London, Greater London, W2 2TP

Piccadilly Circus
Piccadilly Circus, London, Greater London, W1D

Big Ben
Category: Outdoor
Open: -
Website: www.parliament.uk/bigben
Tel.: +44 844 847 1672
11.2%

Messages | Reports | Hot Issues | Summary

The weather is getting nice. #bigben #london #summertime @ Big Ben http://t.co/yTtpik49wa

The chimes of big ben are played live o Radio4. If you stand at Parliment Squar listening to the radio in one ear and the

I did the cold water challenge today the video will be up soon tre is posting it and these are the people i challenged

London baby! @ Big Ben London http:/ t.co/t38NnX7uSH

pollution @Heathrow

airpollution @Heathrow

raining @Northolt

raining @Bank

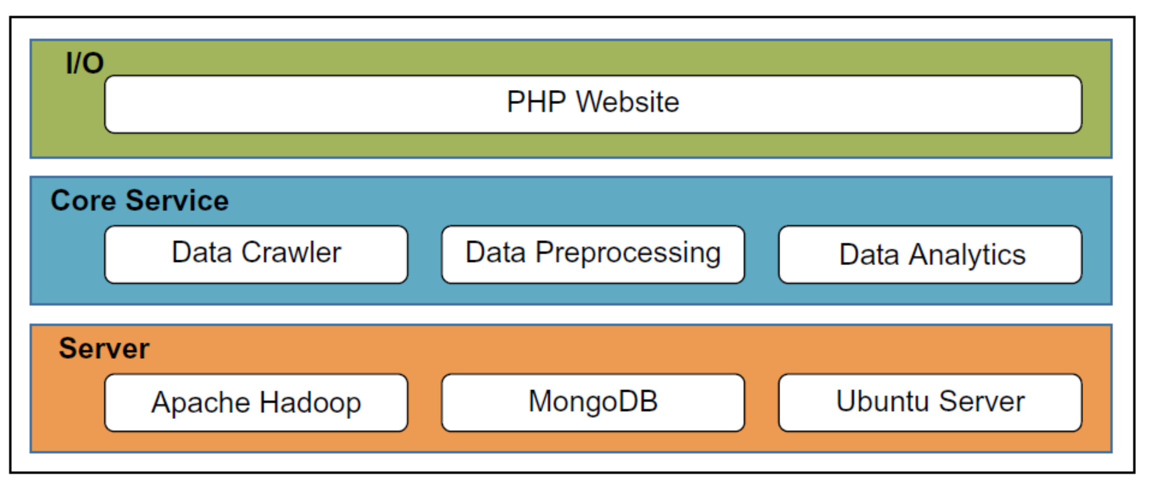
Detail:
Type: Efficiency

- Fig. 7.12. AskDOM mobile application.

7.1.3. Mining Public Opinions

- **IMPLEMENTATION**
- Fig. 7.13 shows the overall implementation architecture of the DOM engine.
- The structure has three main components which are Server, Core Service and I/O.

7.1.3. Mining Public Opinions



- Fig. 7.13. DOM engine architecture.

7.1.3. Mining Public Opinions

- **SERVER**
- The server section consists of three components: the Ubuntu server, MongoDB, and Apache Hadoop.

- The DOM engine was implemented based on Apache Hadoop MapReduce, which runs on the Ubuntu server. MongoDB, the famous NoSQL database, was also used in this framework.

- This type of database often includes highly optimized key value stores intended for simple retrieval and appending operations to improve the performance in terms of latency and throughput.

7.1.3. Mining Public Opinions

- **CORE SERVICE**
- Core Service, the main part of our framework, consists of three components.
- **(1)** *Data Crawler*: This module automatcially provides a raw data feed from social networks and stores it in the database, MongoDB.
- Each crawler code is specific for each social network or website.
- **(2)** *Data Preprocessing*: This component prepares raw data for analysis by tokenizing Thai and English words from sentences, removing outliers and reformatting data.
- Then the cleaned data will be sent to Data Analysis.

7.1.3. Mining Public Opinions

- **(3)** *Data Analysis*: There are three data analyzers in this component:
- **(3.1)** *Sentiment Analysis* evaluates sentiments in Twitter text and finds peoples' moods on a particular topic.
- For example, how people think about traffic in Bangkok.
- **(3.2)** *Clustering-based Summarization* organizes Twitter text into clusters and selects representative sentences from each one to form a text summary.
- The generated summary is presented as supporting evidence for the sentiment analysis results.
- **(3.3)** *Influencer Analysis* determines people's positions in network, which indicates how influential they are.
- The influential people are more likely to acquire connections and have more connections.

7.1.3. Mining Public Opinions

- **I/O**
- I/O, the web-service implemented using PHP, receives the result from Core Service and then sends them to the client-side to display in a JSON format.
- Since the amount of data in social networks is increasing every second, using the static resources (eg, static server) may not be practical.
- So, DOM was designed to run on the cloud.
- The cloud provides the ability to add blob storage depending on the size of data. Furthermore, DOM has the ability to scale the number of processers.
- In other words, DOM can increase or decrease the number of mappers and reducers for running a job.

7.1.3. Mining Public Opinions

- **VALIDATION**
- To validate the effectiveness of XDOM, a subjective experiment was conducted to assess the sentiment prediction accuracy.

- **VALIDATION**
- **PARAMETER**
- • 184,184 messages from Facebook, Twitter and Foursquare (both positive and negative messages) were divided into short and long messages, including 172,717 short messages (≤150 characters) and 11,467 long messages (>150 characters).
- • 12 subjects (6 males and 6 females) participated in the experiment. They were students at the Computer Engineering Department, King Mongkut’s University of Technology Thonburi, Thailand.

7.1.3. Mining Public Opinions

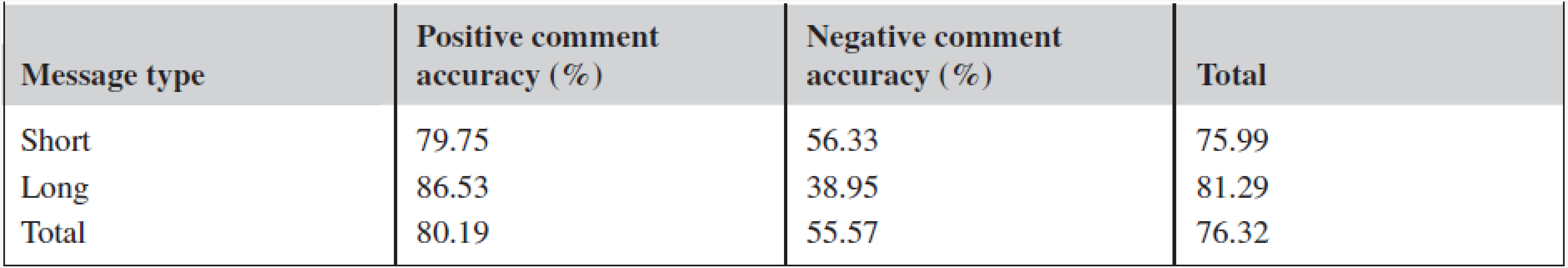
- **VALIDATION METHOD**
- **1.** For the human end, 184,184 messages were divided into 12 parts, each of which was assigned to each subject.
- They classified the messages into positive and negative classes.
- **2.** For the DOM engine, 184,184 messages were classified by the engine into positive and negative classes.
- **3.** The results of both human and DOM were compared and analyzed together to assess the system’s prediction accuracy.

7.1.3. Mining Public Opinions

- **VALIDATION RESULTS**
- Tables 7.4 and 7.5 show the comparison results of 12 students and the DOM engine.
- It was found that the DOM engine can classify messages and conduct sentiment analysis with an accuracy of over 75%.
- The accuracy of the DOM engine is in the standard of text classification, so the DOM engine is practical for use in social network analysis and can be applied to many dimensions in the real word.

7.1.3. Mining Public Opinions

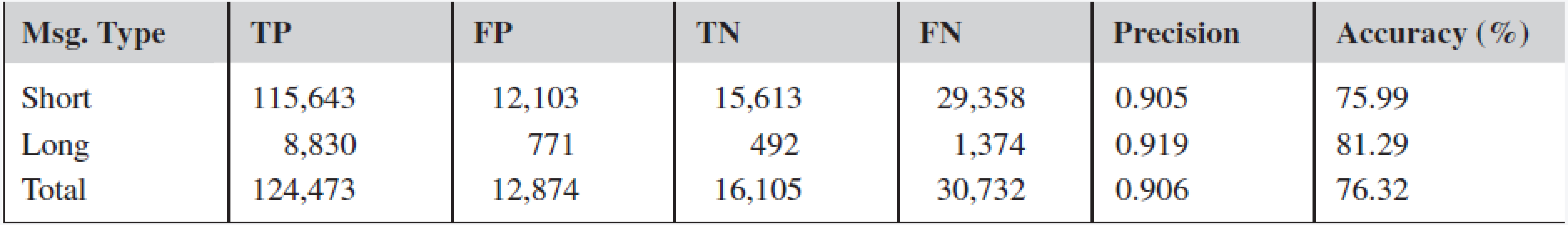
- Table 7.4. Summary of Prediction Accuracy.



Message type	Positive comment accuracy (%)	Negative comment accuracy (%)	Total
Short	79.75	56.33	75.99
Long	86.53	38.95	81.29
Total	80.19	55.57	76.32

7.1.3. Mining Public Opinions

- Table 7.5. Detail Analysis of the System Effectiveness.



Msg. Type	TP	FP	TN	FN	Precision	Accuracy (%)
Short	115,643	12,103	15,613	29,358	0.905	75.99
Long	8,830	771	492	1,374	0.919	81.29
Total	124,473	12,874	16,105	30,732	0.906	76.32

7.1.3. Mining Public Opinions

- **CASE STUDY**
- In addition to the evaluation of the system effectiveness, the XDOM engine was tested further on various case studies that were of interest to the Thai public during the time periods.
- Each case study aims to explore either a specific social or political issue that people were discussing widely on the Internet, thus it offers a summary of Internet public opinions on that issue.

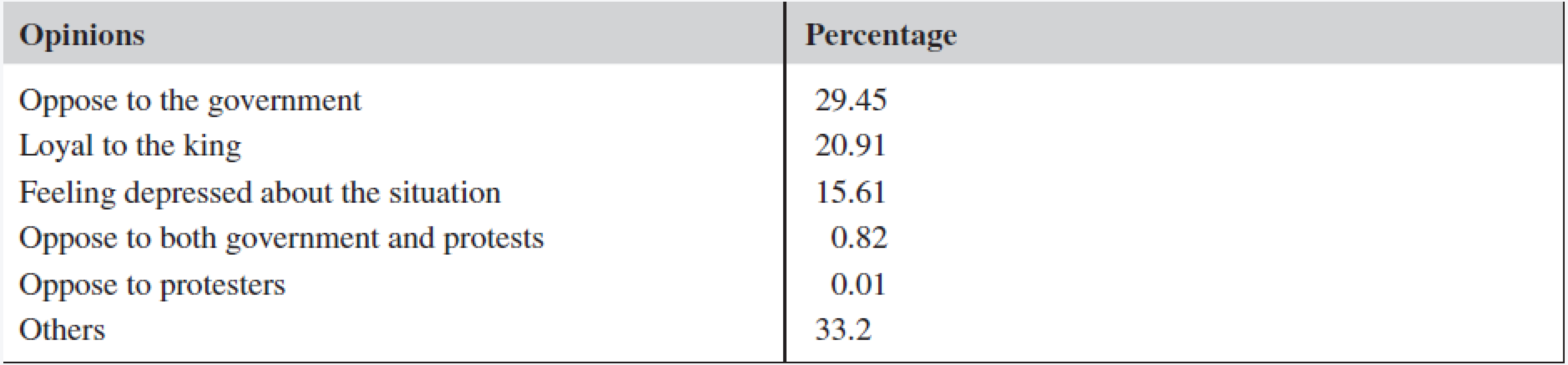
- **POLITICAL OPINION: #PRAYFORTHAILAND**
- Around the end of 2013, citizens of Bangkok were faced with multiple rounds of political protests, and violent acts toward both protesters and officers.

7.1.3. Mining Public Opinions

- Hashtag “#prayforthailand” is one that was frequently used in social media to express the concerns over the situation.
- Different opinions were expressed regarding this political issue.
- DOM was used to mine the general public opinions that were expressed in the social network to determine the political climate at that time.
- Tweets were collected around the Bangkok area that contain the hashtag “#PrayForThailand.”
- There were over 100 K tweets collected from 29 November to 7 December 2013.
- .The Naïve Bayes and Support Vector Machine (SVM) were implemented to the DOM engine to classify political opinions into six predefined categories as shown in Table 7.6.
- DOM can accurately put tweets into categories with more than 85% accuracy.

7.1.3. Mining Public Opinions

- Table 7.6. Summary of Opinions with “#prayforthailand”.



Opinions	Percentage
Oppose to the government	29.45
Loyal to the king	20.91
Feeling depressed about the situation	15.61
Oppose to both government and protests	0.82
Oppose to protesters	0.01
Others	33.2

7.1.4. Exploring Twitter Sentiment Analysis and the Weather

- **SYSTEM BACK-END ARCHITECTURE**
- The back-end architecture is realized through Python (version 2.7) and the noSQL database CouchDB.

- Python 2.7 is used to compile programs for fetching and filtering Twitter and weather data, for sentiment analysis and integrating data collection based on views.

- In contrast, CouchDB is responsible for data storage, view creation, and the return view result to the website.

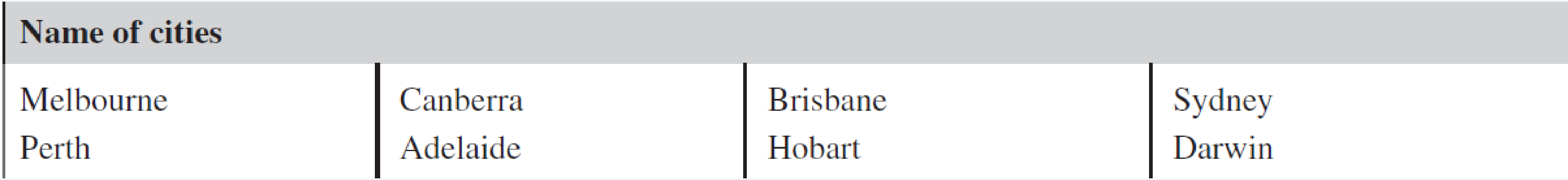
7.1.4. Exploring Twitter Sentiment Analysis and the Weather

- As shown in Fig. 7.14, the system includes two external sources of data and one internal source.
- The Twitter streaming API provides a targeted location's raw tweets, which are content-filtered and sentiment-analyzed before being stored in the CouchDB.
- We focus explicitly on those tweets including geo-location information.
- During this work, over 700,000 tweets were collected from eight Australian cities and over 33,000 tweets from 174 suburbs in Melbourne.

- A list of cities and suburbs follows:

7.1.4. Exploring Twitter Sentiment Analysis and the Weather

- **1.** Name of eight cities:



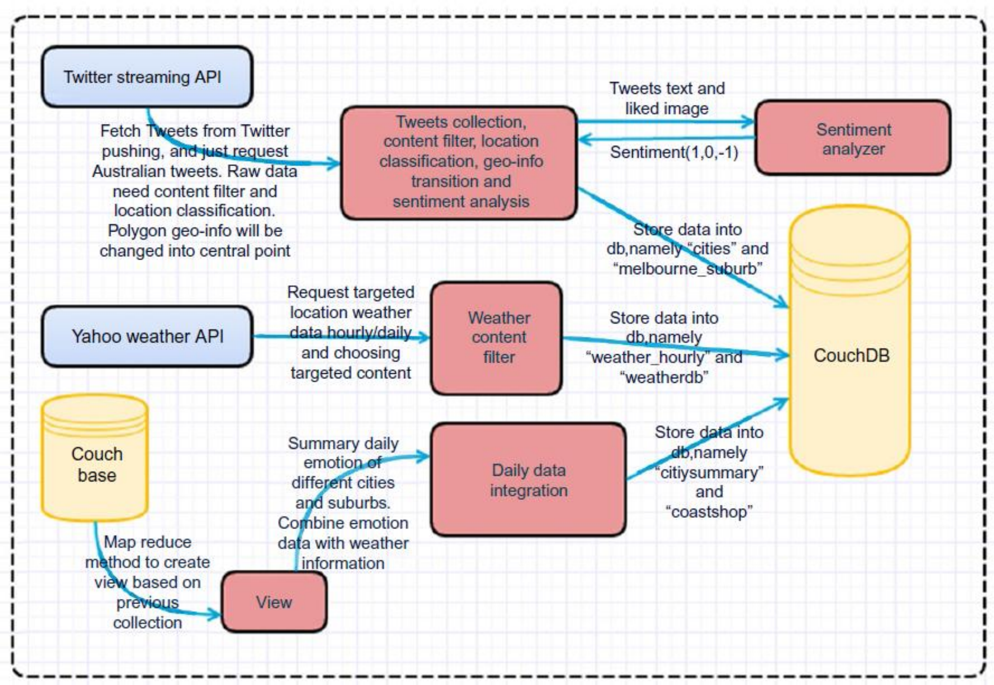
Name of cities			
Melbourne	Canberra	Brisbane	Sydney
Perth	Adelaide	Hobart	Darwin

- **2.** Name of 174 Melbourne suburbs:
- 174 suburbs in Melbourne were explored.
- The shopping mall and coast suburbs explored were:

7.1.4. Exploring Twitter Sentiment Analysis and the Weather

- • Shopping Malls
- HighPoint Shopping Centre, Emporium, Chadstone Shopping Centre, Essendon DFO, and
- South Wharf DFO.
- • Cxoast suburbs
- Altona, Altona Meadows, Seaholme, Williamstown, Port Melbourne, Newport, Albert Park,
- St. Kilda West, St, Kilda, Elwood, Brighton, and Middle Park.

7.1.4. Exploring Twitter Sentiment Analysis and the Weather



Twitter streaming API

Fetch Tweets from Twitter pushing, and just request Australian tweets. Raw data need content filter and location classification. Polygon geo-info will be changed into central point

Tweets collection, content filter, location classification, geo-info transition and sentiment analysis

Tweets text and liked image

Sentiment(1,0,-1)

Sentiment analyzer

Store data into db,namely "cities" and "melbourne_suburb"

Yahoo weather API

Request targeted location weather data hourly/daily and choosing targeted content

Weather content filter

Store data into db,namely "weather_hourly" and "weatherdb"

CouchDB

Couch base

Summary daily emotion of different cities and suburbs. Combine emotion data with weather information

Daily data integration

Store data into db,namely "citiysummary" and "coastshop"

Map reduce method to create view based on previous collection

View

- Fig. 7.14. System back-end architecture.

7.1.4. Exploring Twitter Sentiment Analysis and the Weather

- In terms of weather data, daily and hourly weather data was obtained from the Yahoo weather API for these eight cities.

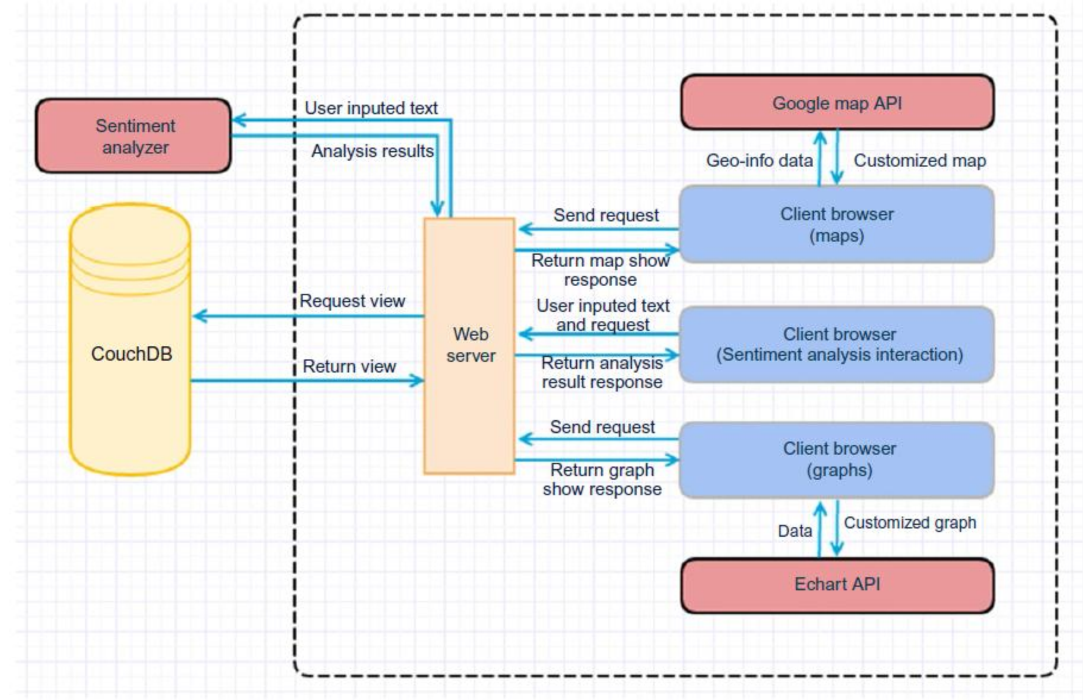
- Based on the weather and tweets collection, cloud-based MapReduce methods were utilized to create the views of the data.
- These views are used for web site requests and to provide data to daily integration programs.

- Two scheduled integration programs (for city and suburb levels) are used for calculating the summary of emotions and their relationship with daily weather.

7.1.4. Exploring Twitter Sentiment Analysis and the Weather

- **SYSTEM FRONT-END ARCHITECTURE**
- The web service architecture support capabilities to present data.
- The core technologies implemented include HTML5, CSS3, JavaScript, AJAX, Google Maps API (based on JQuery), eChart API, and Bootstrap for the responsive web design.
- The main presentation of information utilizes eChart and Google API, which support asynchronous communication with the server.
- As one can see from Fig. 7.15, some functions of the web-based front end require data directly from CouchDB.
- These data are created in the back end using MapReduce methods to generate views in CouchDB.

7.1.4. Exploring Twitter Sentiment Analysis and the Weather



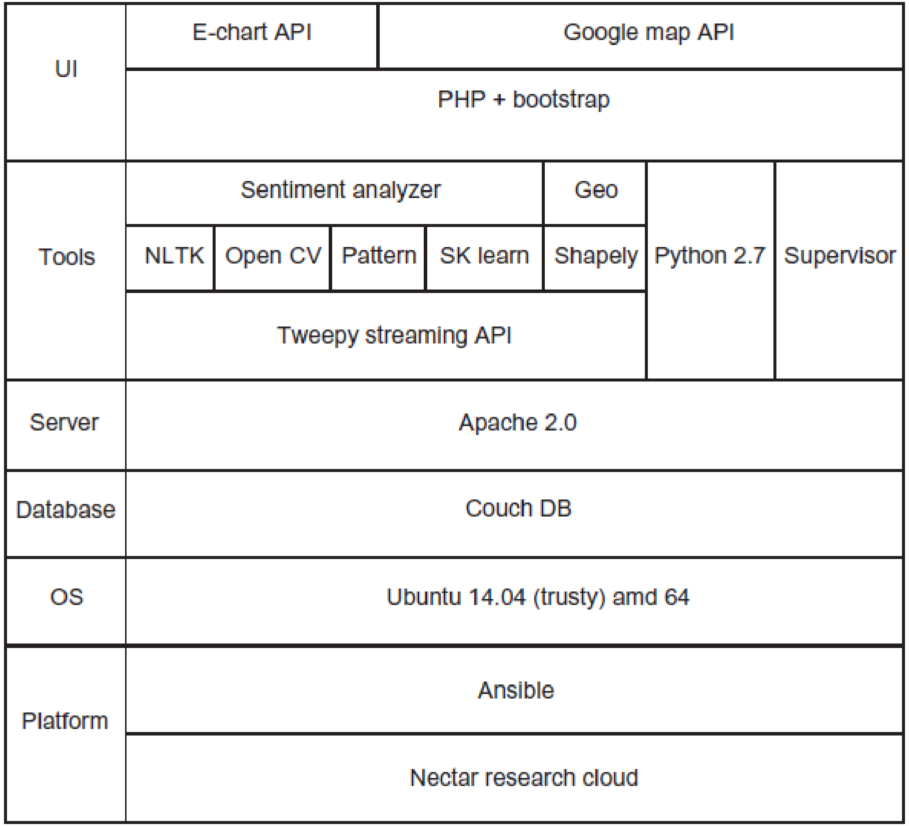
- Fig. 7.15. System front-end architecture.

7.1.4. Exploring Twitter Sentiment Analysis and the Weather

- **SOFTWARE STACK**
- Fig. 7.16 presents the software stack that was adopted in this work.
- To support sentiment analysis, various approaches were explored: toolkits such as the Natural Language Toolkit (NLTK), open CV, Pattern, and SK Learn packages NLTK support preprocessing of tweet text contents, and also offer the Naïve Bayes supervisor to implement frequency of terms analysis.
- Open CV is used for analysis of images associated with Twitter profiles.
- Pattern is a mature tweet sentiment analysis tool that is based on parts of speech (POS), which are used for computational linguistics analysis.
- Each of these tools is used to generate features for sentiment analysis.
- SK learning is subsequently used to apply these features to different machine-learning models and finally return the analyzed results.

7.1.4. Exploring Twitter Sentiment Analysis and the Weather

- Fig. 7.16. Software stack
- in the system.



UI

E-chart API

Google map API

PHP + bootstrap

Tools

Sentiment analyzer

Geo

NLTK

Open CV

Pattern

SK learn

Shapely

Python 2.7

Supervisor

Tweepy streaming API

Server

Apache 2.0

Database

Couch DB

OS

Ubuntu 14.04 (trusty) amd 64

Platform

Ansible

Nectar research cloud

7.1.4. Exploring Twitter Sentiment Analysis and the Weather

- **MACHINE-LEARNING METHODOLOGY**
- There are many ways to support machine learning and analysis of data.
- The Naïve Bayes machine-learning technique was used as our baseline approach.
- After that deep supervised machine-learning techniques was utilized, including random forest (RF), support vector machine (SVM), and logistic regression (LR), which are applied to classify the emotion (sentiment) of tweets.
- For correlation analysis, an unsupervised machine learning (density-based spatial clustering of applications with noise (DBSCAN) cluster algorithm) and time series are implemented to explore the relation between weather and tweets clusters and their emotions.

7.1.4. Exploring Twitter Sentiment Analysis and the Weather

- **TWEETS SENTIMENT ANALYSIS**
- Text-based sentiment analysis is a well-studied area. Go et al. (2009) point out that an accuracy of 82.9% can be achieved with SVM and a simple unigram model.
- They also mention several important attributes that differentiate twitter messages from other text-based resources.
- First, the length of a twitter message is restricted to 140 characters.
- This means that multiple-sentence-based sentiment analysis is not suitable.
- In addition, twitter users tend to use slang and abbreviations to express their opinions.
- There are already developed tools for supporting this task, for example, the Pattern module of Python, which focuses on word level sentiment classification.
- For this project, due to the limited length of tweets, the word becomes more significant to the sentiment of a sentence. In addition, it also includes slang such as: “lol,” which is commonplace in Twitter.

7.1.4. Exploring Twitter Sentiment Analysis and the Weather

- ***Naïve Bayes as a baseline***
- The most direct way to establish a user’s emotion in tweets is through the textual content.
- Traditionally, terms-frequency statistics features based on a labeled training set is the simplest way to predict a tweet’s text sentiment.
- Naïve Bayes supervised machine learning usually is used for this purpose.
- To this end, it is necessary to establish a training data set.
- In this work, 80% of our manually labeled tweets were used as a training set, and the rest of the tweets are treated as the actual test set.
- Regrettably, the accuracy of this approach was limited, achieving just 54%.

7.1.4. Exploring Twitter Sentiment Analysis and the Weather

- ***Tweet preprocessing***
- Tweet preprocessing is divided into two parts.

- For the tweet’s text content, the text needs to be filtered, for example, to remove Unicode for emojis, external links, and special strings (‘#’ and ‘@’).

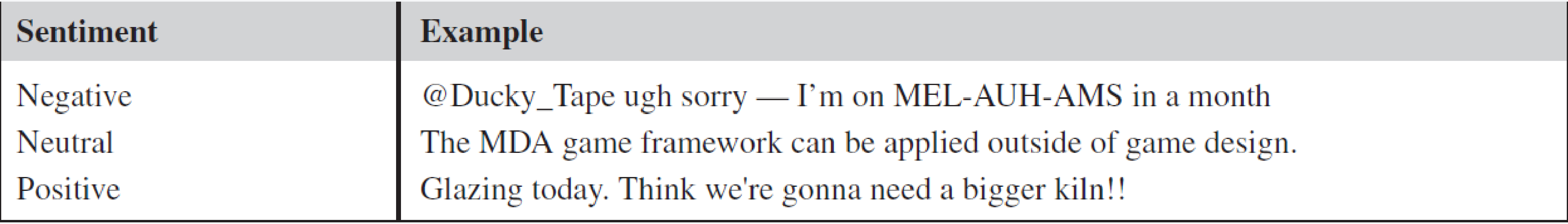
- Second, if there is picture or photo attached to the tweet, it will be downloaded via its source link.

7.1.4. Exploring Twitter Sentiment Analysis and the Weather

- ***Training set***
- The main training set contained 2613 tweets (data available at: http://115.146.86.188:5984/_utils/database.html?traindata), which are labeled manually.
- It includes 395 negative tweets, 1079 neutral tweets, and 1139 positive tweets. Examples of these tweets are presented in Table 7.7.

7.1.4. Exploring Twitter Sentiment Analysis and the Weather

- Table 7.7. Example of Labeled Tweets.



Sentiment	Example
Negative	@Ducky_Tape ugh sorry — I'm on MEL-AUH-AMS in a month
Neutral	The MDA game framework can be applied outside of game design.
Positive	Glazing today. Think we're gonna need a bigger kiln!!

7.1.4. Exploring Twitter Sentiment Analysis and the Weather

- ***Feature engineering***
- There are three kinds of features.
- The first one is the twitter sentiment score, which includes the Pattern Sentiment Parser and sentiment word-frequency classification.
- The second category is based on the color of images and the last one explores an analysis of the images, for example, for smile detection.

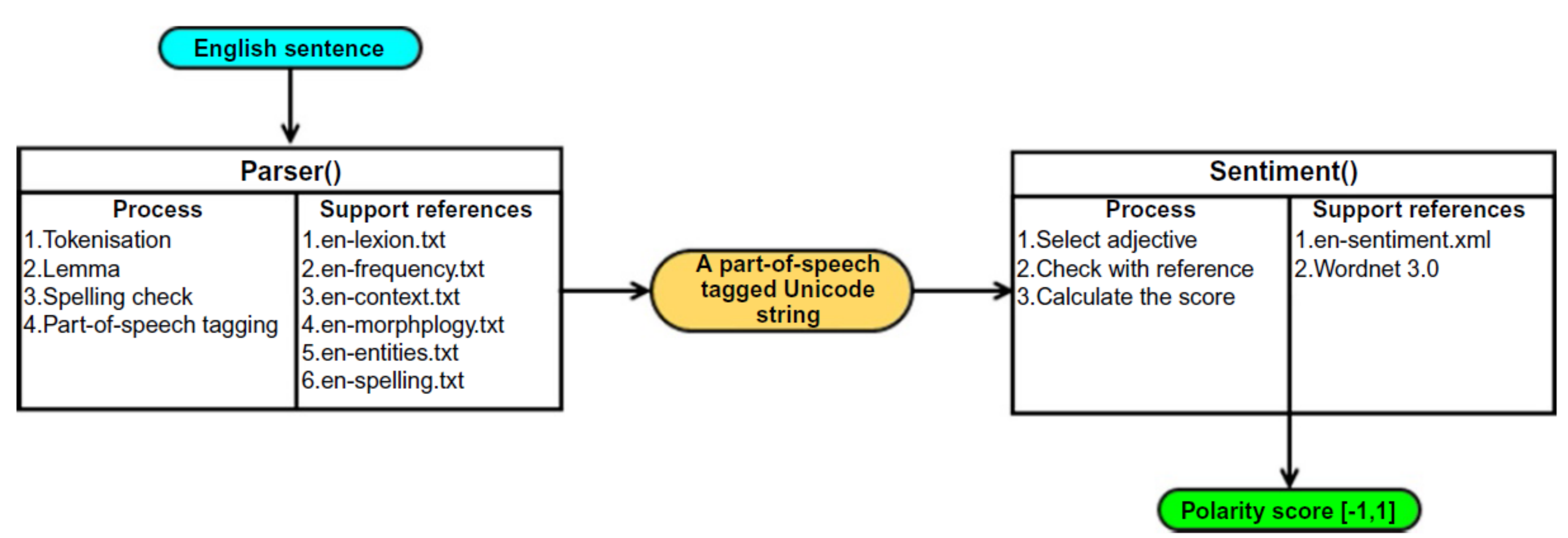
7.1.4. Exploring Twitter Sentiment Analysis and the Weather

- ***Twitter sentiment score feature***
- There are four features in this part (Pattern Sentiment Parser, the number of positive words divided by the length of sentence, the number of negative words divided by the length and sentiment word frequency classification).
- The second and third features are used to take the length of a sentence and the ratio of positive and negative words into consideration.
- They are also based on the Pattern Sentiment Parser.
- The last one is based on the labeled data.

7.1.4. Exploring Twitter Sentiment Analysis and the Weather

- The first feature can be illustrated as shown in Fig. 7.17.
- The main process of the Parser method is to get a POS tagged Unicode string.
- The POS is a method to remove the ambiguous content of an English sentence.
- Actually, Pattern tags the word according to the Brill's rule's based tagger v1.14 and Penn TreebankII tag set.
- In order to increase the accuracy of tagging, Pattern generates its own lexical references as a dictionary.
- For example, except for Eric Brill's tag dictionary (1992), the en-lexion.txt has introduced Twitter POS annotated data provided by Carnegie Mellon University (2011).
- Some of the support references have been trained under the Brown Corpus and Penn Treebank.
- Thus, the part-of speech tagging for tweets is much more accurate through the Pattern Parser method.

7.1.4. Exploring Twitter Sentiment Analysis and the Weather



English sentence

Parser()

Process	Support references
1.Tokenisation	1.en-lexion.txt
2.Lemma	2.en-frequency.txt
3.Spelling check	3.en-context.txt
4.Part-of-speech tagging	4.en-morphplogy.txt
	5.en-entities.txt
	6.en-spelling.txt

A part-of-speech tagged Unicode string

Sentiment()

Process	Support references
1.Select adjective	1.en-sentiment.xml
2.Check with reference	2.Wordnet 3.0
3.Calculate the score	

Polarity score [-1,1]

- Fig. 7.17. The process of sentiment analysis in pattern module.

7.1.4. Exploring Twitter Sentiment Analysis and the Weather

- Based on the result of the Parser process, the Sentiment method calculates the degree of sentiment according to its own sentimental lexical resource.

- This dictionary assigns to each synset of Wordnet 3.0 and collects about 2900 adjectives with sentiment polarity.

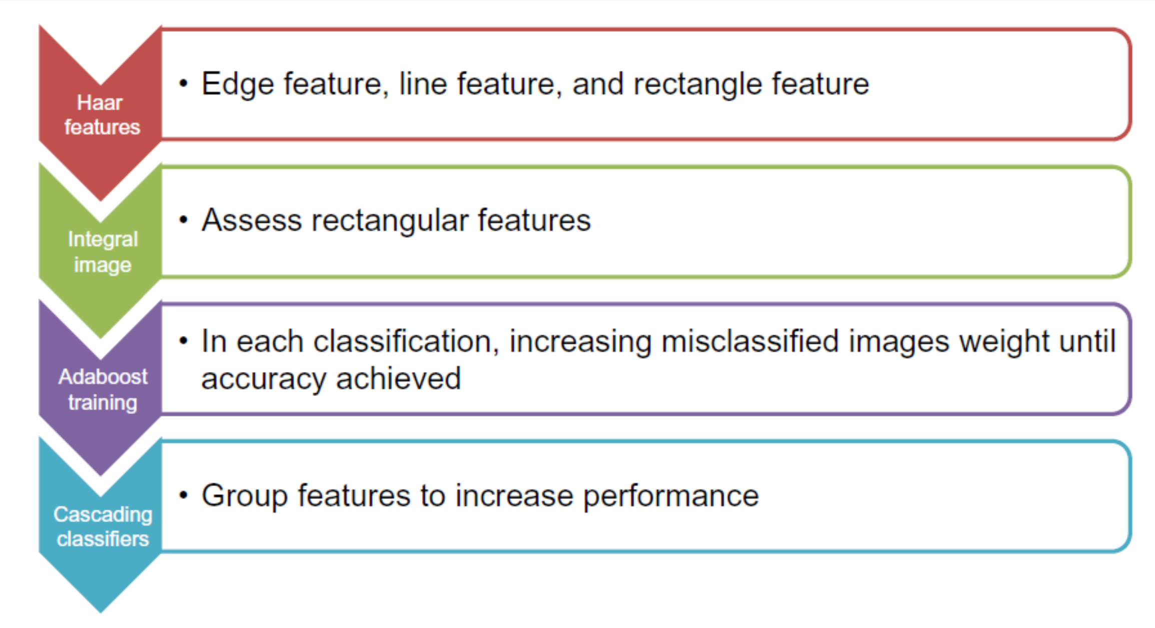
- Thus, this process of sentiment extracts the adjectives from the previous results and then establishes the score from the lexical resources.

- Finally, the score of sentiment is given as a figure from −1 to 1.

7.1.4. Exploring Twitter Sentiment Analysis and the Weather

- ***Smile detection feature***
- For smile detection, Haar feature-based cascade classifiers proposed by Viola and Jones have been applied.
- This algorithm includes four stages as shown in Fig. 7.18.
- The cascade function is the core part of this algorithm.
- It is trained with a considerable amount of right and wrong pictures.
- In the OpenCV package, there are pretrained classifiers for faces and smiles that are stored as XML files.
- In this project, the “haarcascade_frontalface_default.xml” is used to detect faces and “haarcascade_smile.xml” is applied to detect smiles.
- To increase the accuracy, a smile can only be detected when it is within a face.

7.1.4. Exploring Twitter Sentiment Analysis and the Weather



Haar features

- Edge feature, line feature, and rectangle feature

Integral image

- Assess rectangular features

Adaboost training

- In each classification, increasing misclassified images weight until accuracy achieved

Cascading classifiers

- Group features to increase performance

- Fig. 7.18. Haar cascade algorithm steps.

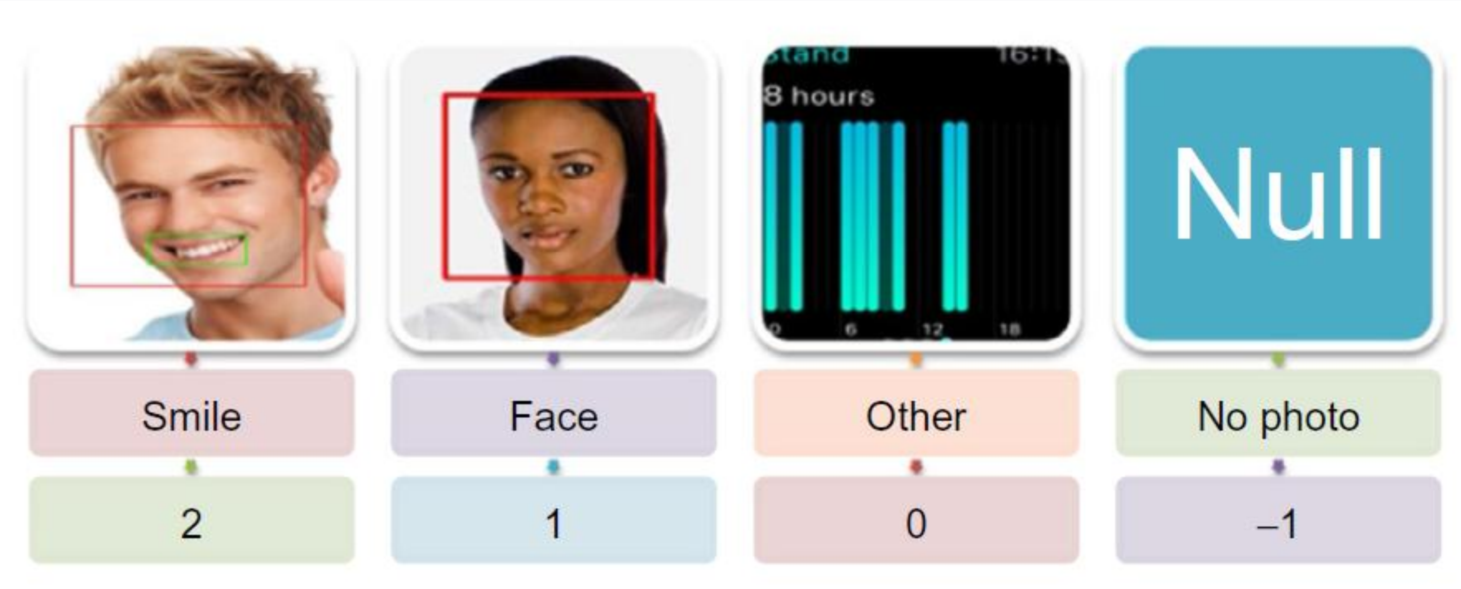
7.1.4. Exploring Twitter Sentiment Analysis and the Weather

- As Fig. 7.19 shows, a smile detected, a face detected, other photo, and no photo are converted to the values 2, 1, 0, and −1 respectively as feature values.

- In the case where multiple smiles are detected, this is also equal to 2.

- If someone posts a neutral tweet (based on their text) but with a photo including a smile, it would be recognized overall as a positive tweet.

7.1.4. Exploring Twitter Sentiment Analysis and the Weather



- Fig. 7.19. Smile and face detection to smile feature value.

7.1.4. Exploring Twitter Sentiment Analysis and the Weather

- ***Classifier models***
- A range of machine-learning techniques (SVM, RF, LR, and ensemble stacking) have been explored in this chapter.
- This section includes an evaluation of the system robustness through the learning curve and reliability in indicating sentiment through Precision, Recall, and F1-scores.

- In addition, the parameters for each model have been optimized with a grid search method.
- The grid search in this project is used to examine all the combinations of proposed parameters for each estimator.
- The grid search is based on fivefold cross validation and evaluation metrics for Precision.
- Due to this exhaustive search, there are too many combinations of results.

7.1.4. Exploring Twitter Sentiment Analysis and the Weather

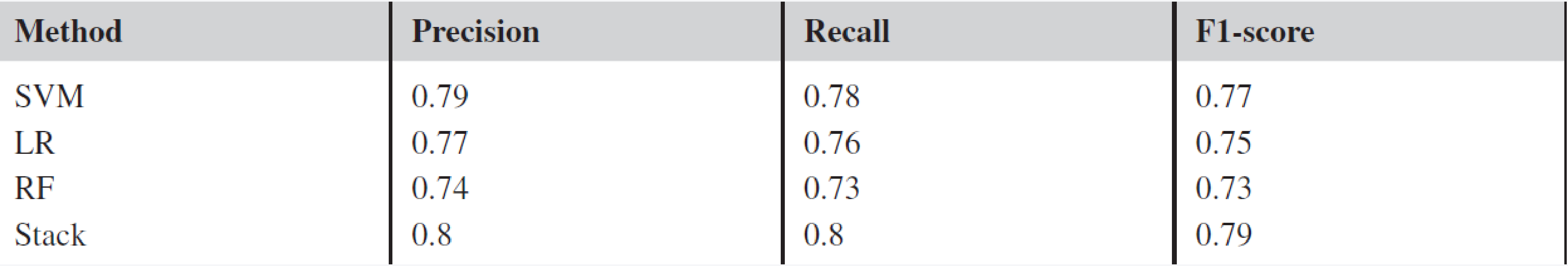
- To evaluate whether current classifiers are reliable, the learning curve graph is drawn.
- The horizontal axis corresponds to the number of samples (from 10%, 20% … to a 100% training set) and the vertical axis corresponds to the accuracy score.
- The score is the mean accuracy for threefold cross validation of that part of the dataset.
- A smaller value implies that fewer instances are misclassified.
- The difference between cross- validation scores and training scores is used to determine whether the model is robust enough.
- If the gap of these two scores is large, it means the model may overfit the training set.
- Otherwise, it means the training error and test error are similar, and hence, the model is robust.

7.1.4. Exploring Twitter Sentiment Analysis and the Weather

- ***Stacking***
- A stacking ensemble method is applied to achieve a more accurate prediction.
- This is based on a weighting system used for prediction of the final result.
- If all classifier predictions are different from each other, the prediction is dependent on the SVM classifier, due to its robustness and accuracy.
- After application of this method, overall performance is improved, as shown in Table 7.8.

7.1.4. Exploring Twitter Sentiment Analysis and the Weather

- Table 7.8. Comparison for Overall Performance.



Method	Precision	Recall	F1-score
SVM	0.79	0.78	0.77
LR	0.77	0.76	0.75
RF	0.74	0.73	0.73
Stack	0.8	0.8	0.79

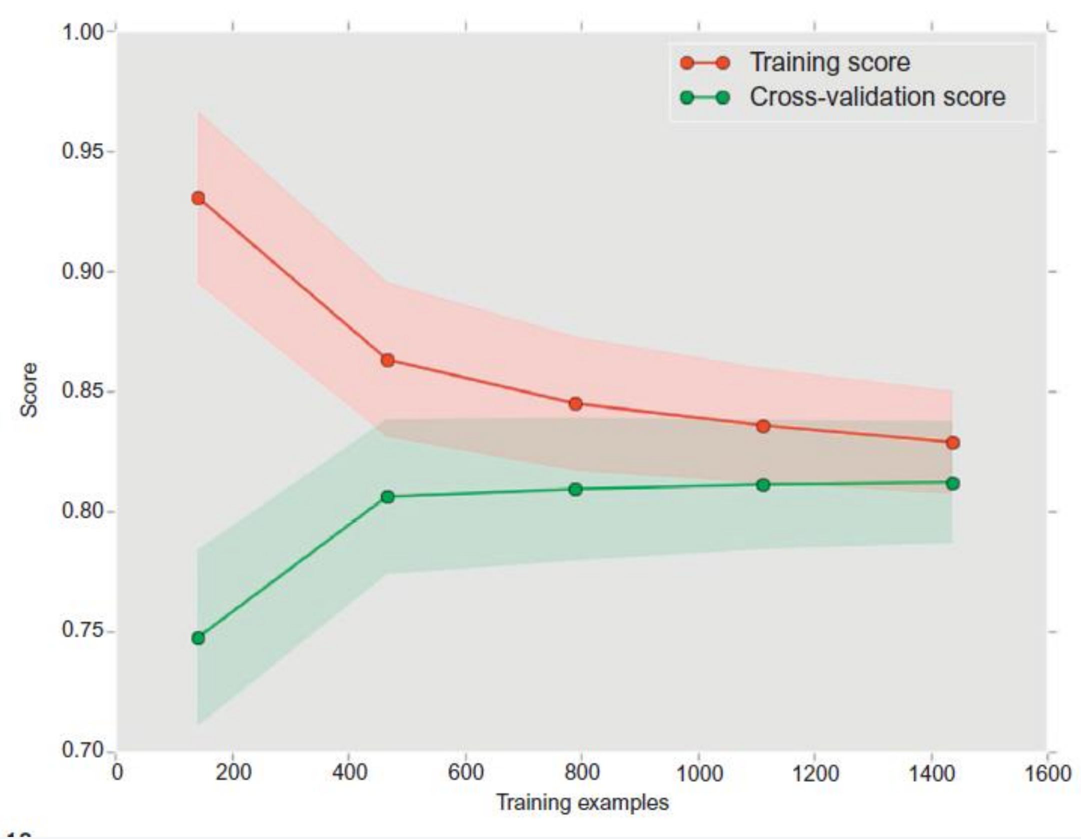
7.1.4. Exploring Twitter Sentiment Analysis and the Weather

- The resultant learning curve (see Fig. 7.20) is more robust than the single model and the accuracy achieved is around 0.8.

- This means that less misclassification occurs and that the model is not over/under-fit based on the training data set.

7.1.4. Exploring Twitter Sentiment Analysis and the Weather

- Fig. 7.20. Stacking
- method with three
- optimized learning
- algorithms.



7.1.4. Exploring Twitter Sentiment Analysis and the Weather

- **WEATHER AND EMOTION CORRELATION ANALYSIS**
- Two approaches are used to determine the correlation between weather and emotion.
- One is based on a time series, whilst the other is based on a cluster algorithm.
- For the time series, the covariance is used to evaluate whether two variables are independent, and the Pearson correlation coefficient is introduced to evaluate the degree of relation.

- For the clustering algorithm, a comparison using the DBSCAN cluster method is chosen, and its corresponding evaluation metrics are shown.

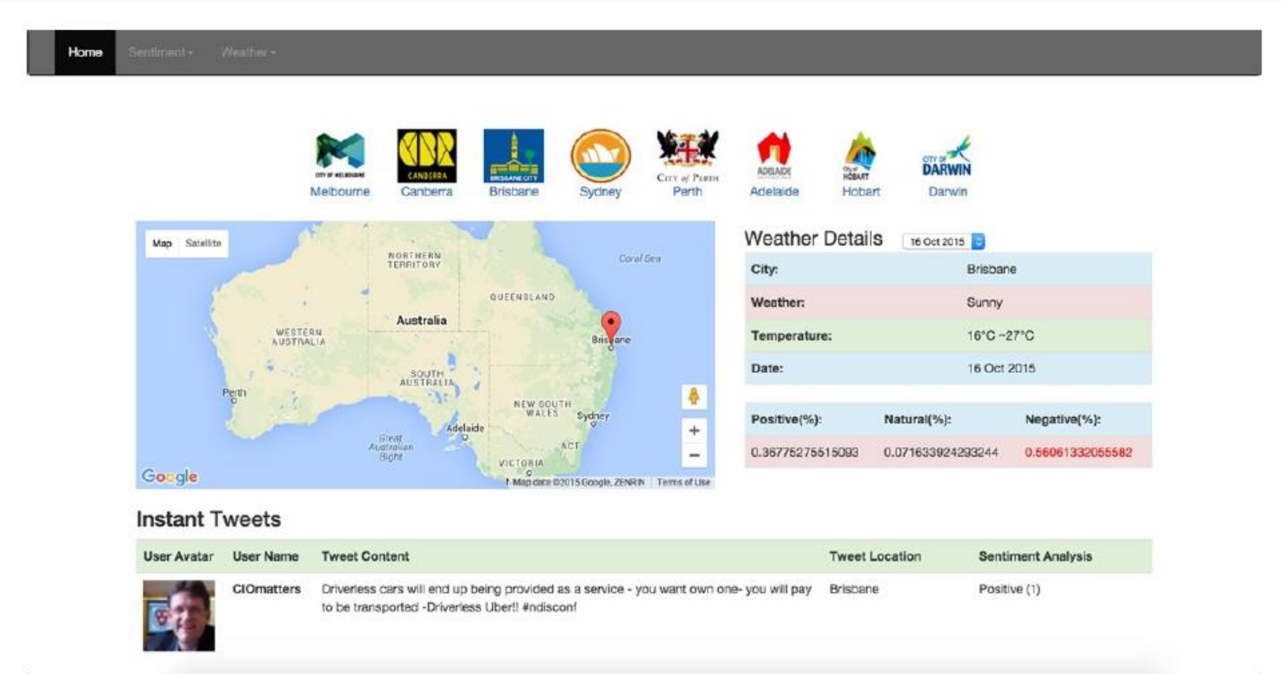
7.1.4. Exploring Twitter Sentiment Analysis and the Weather

- **SYSTEM IMPLEMENTATION**
- In this section, the Big Data processing and analysis functions implemented in the system are introduced.
- The web front end consists of three main parts: a home page, a sentiment page and a weather page.
- Their content and functions are described here.

7.1.4. Exploring Twitter Sentiment Analysis and the Weather

- **HOME PAGE**
- Fig. 7.21 shows the function used to present and analyze tweets.
- The sentiment analyzer program analyses tweets as they are harvested before they are stored into CouchDB.
- Additionally, the tweets’ geo-location has been marked in the map.
- Second, historical weather information and summarized emotion situations of eight cities has been provided for users to search according to the cities’ name and specific dates.

7.1.4. Exploring Twitter Sentiment Analysis and the Weather



Weather Details 16 Oct 2015

City:	Brisbane
Weather:	Sunny
Temperature:	16°C ~27°C
Date:	16 Oct 2015

Positive(%):	Natural(%):	Negative(%):
0.36775275515093	0.071633924293244	0.56061332055582

Instant Tweets

User Avatar	User Name	Tweet Content	Tweet Location	Sentiment Analysis
	ClOmatters	Driverless cars will end up being provided as a service - you want own one- you will pay to be transported -Driverless Uber!! #ndisconf	Brisbane	Positive (1)

- Fig. 7.21. Home page in the web site.

7.1.4. Exploring Twitter Sentiment Analysis and the Weather

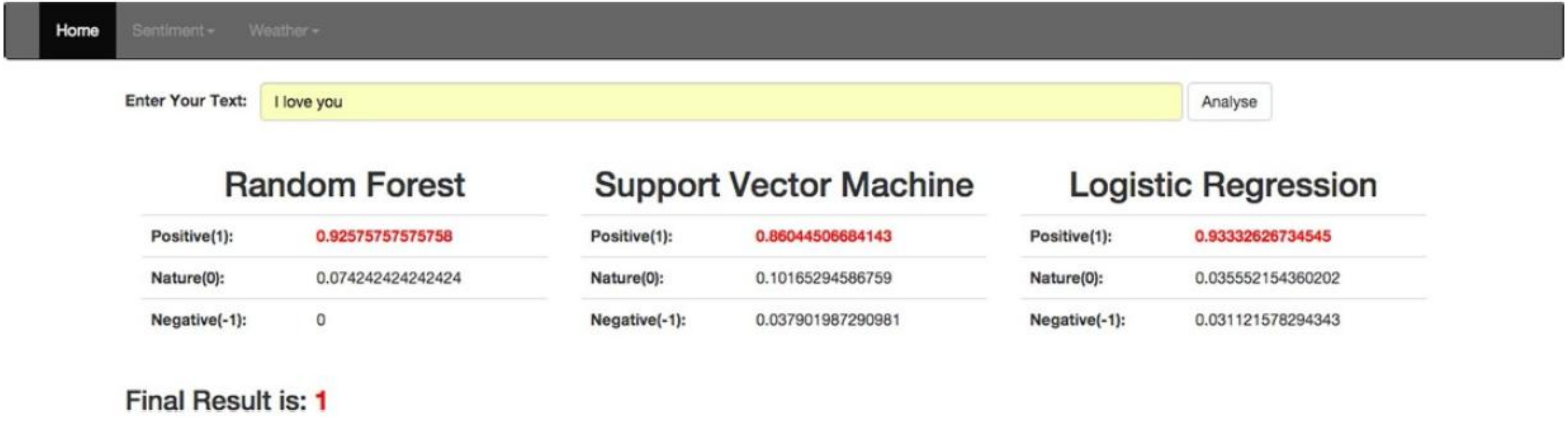
- **SENTIMENT PAGES**
- The sentiment pages consist of a features description page, a models description page and a test yourself page.

- The previous pages are used to introduce the features and models utilized in the sentiment analysis process.

- Fig. 7.22 illustrates an interaction page provided for users to interact with the sentiment analyzer program.

- After the user types the text of interest, the system will return and display the analysis results for each model and a final result.

7.1.4. Exploring Twitter Sentiment Analysis and the Weather



Enter Your Text: I love you

Random Forest	
Positive(1):	0.92575757575758
Nature(0):	0.074242424242424
Negative(-1):	0

Support Vector Machine	
Positive(1):	0.86044506684143
Nature(0):	0.10165294586759
Negative(-1):	0.037901987290981

Logistic Regression	
Positive(1):	0.93332626734545
Nature(0):	0.035552154360202
Negative(-1):	0.031121578294343

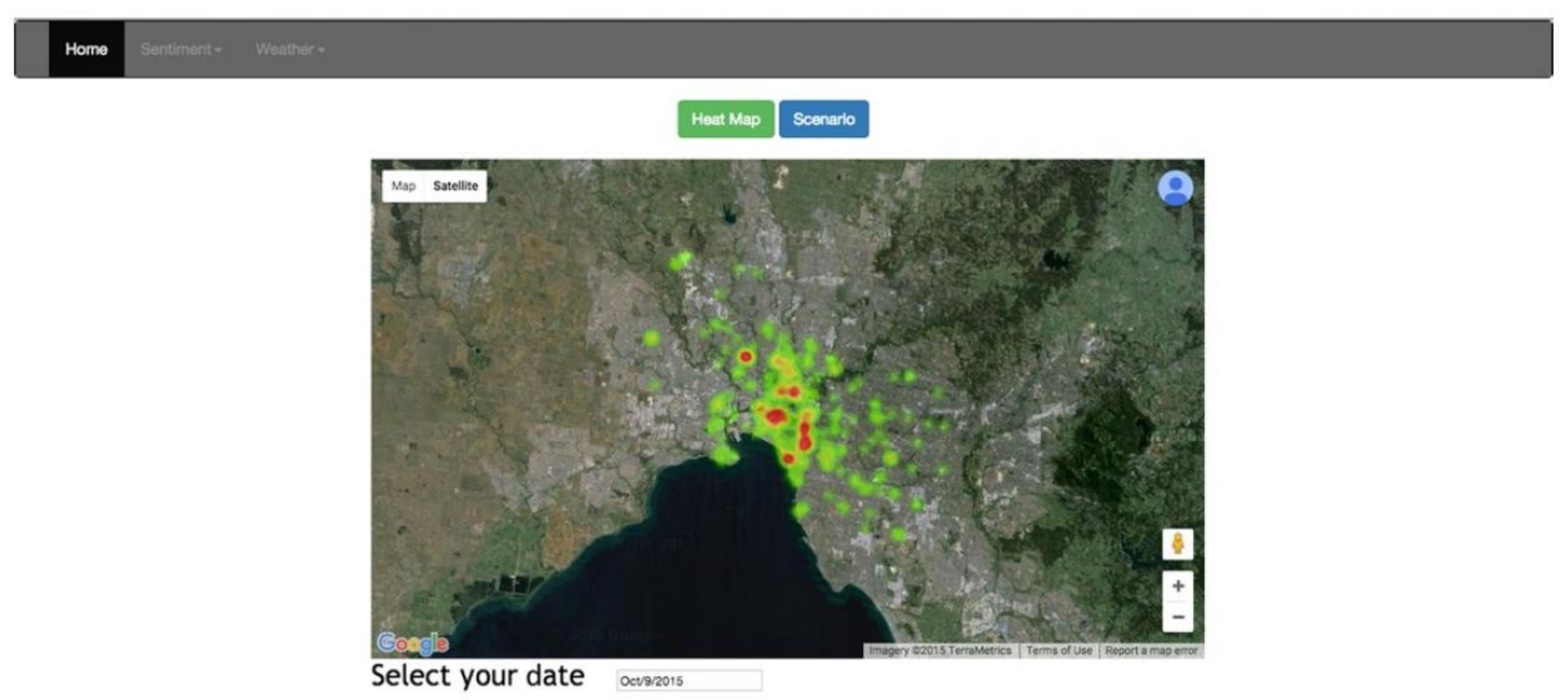
Final Result is: 1

- Fig. 7.22. Interaction functions in web site.

7.1.4. Exploring Twitter Sentiment Analysis and the Weather

- **WEATHER PAGES**
- The weather pages implement the front-end data visualization in the system.
- This allows exploring the correlation between emotion and weather from the whole city down to individual suburb levels.
- Fig. 7.23 shows a heat map illustrating the tweet distribution across Melbourne.
- Users can further view the tweet distribution based on temporal aspects, for example, a given date.

7.1.4. Exploring Twitter Sentiment Analysis and the Weather


Home
Sentiment
Weather
Heat Map
Scenario
Map
Satellite
Google
Imagery ©2015 TerraMetrics | Terms of Use | Report a map error

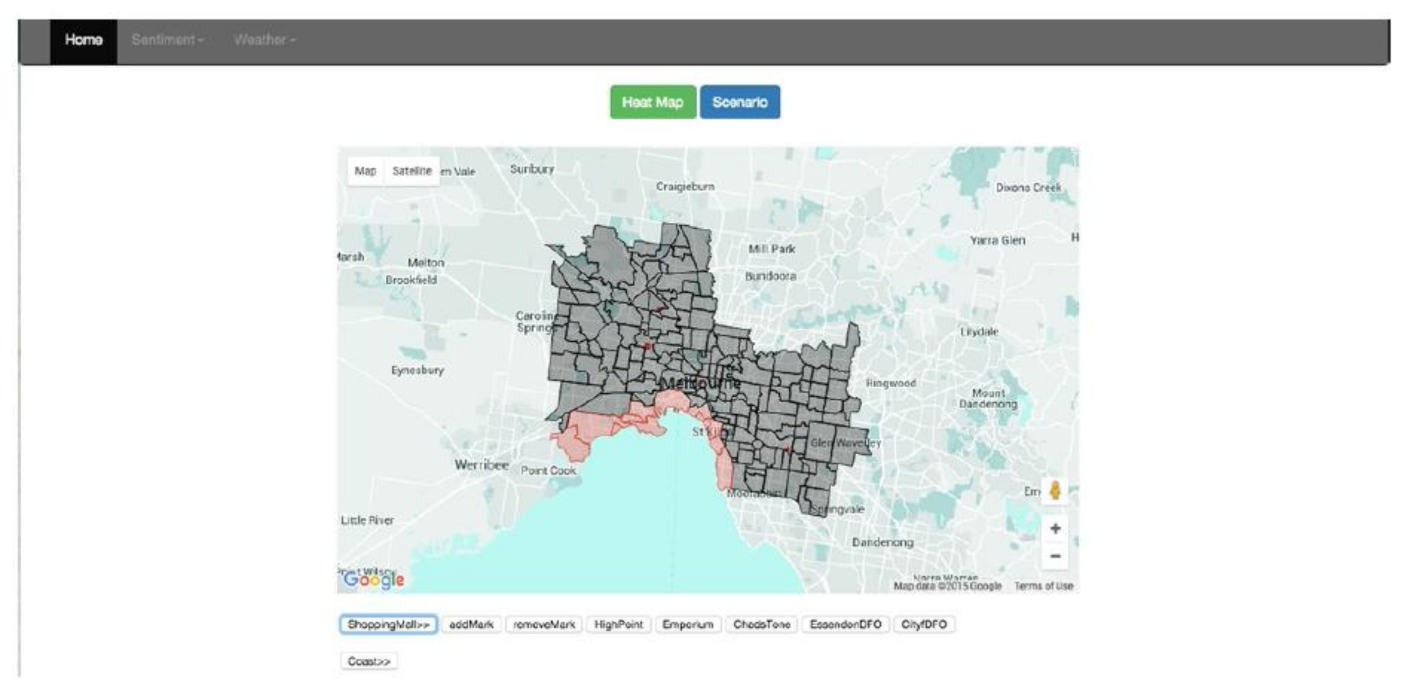
Select your date Oct/9/2015

- Fig. 7.23. Heat map for Melbourne suburbs.

7.1.4. Exploring Twitter Sentiment Analysis and the Weather

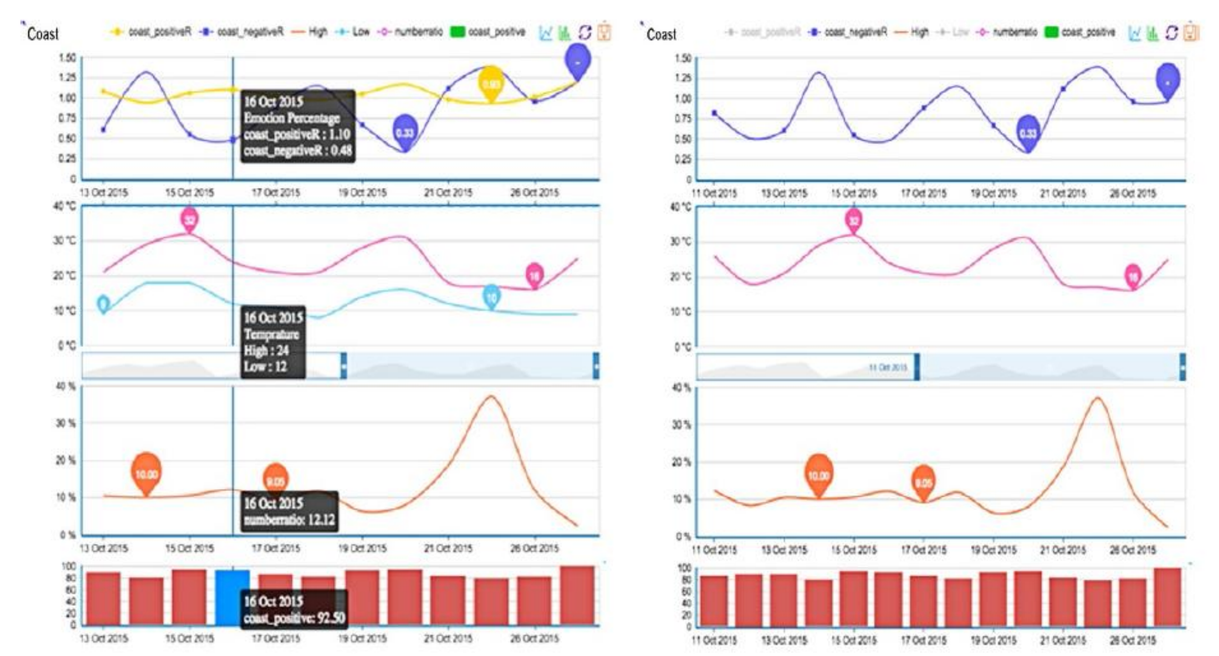
- Fig. 7.24 shows a more detailed scenario of Twitter sentiment across the Melbourne suburbs. Here the map in the middle displays the suburbs under consideration.
- The coastal suburbs area and shopping mall locations can also be highlighted and marked in the map.
- Thus, it is hypothesized that in extremely hot weather many people go to malls where it is cooler.
- The system allows exploration of such phenomenon.
- Additionally, users can visualize the distribution of emotions in different areas compared to the whole of Melbourne, factoring in the temperature and other phenomenon (Fig. 25, left).
- Further functions include allowing users to choose the data to be included in the graphs.
- For example, users can adjust the average emotion graph to show only negative emotions and adjust the period of time of interest (Fig. 7.25, right).

7.1.4. Exploring Twitter Sentiment Analysis and the Weather



- Fig. 7.24. Map for Melbourne suburbs.

7.1.4. Exploring Twitter Sentiment Analysis and the Weather



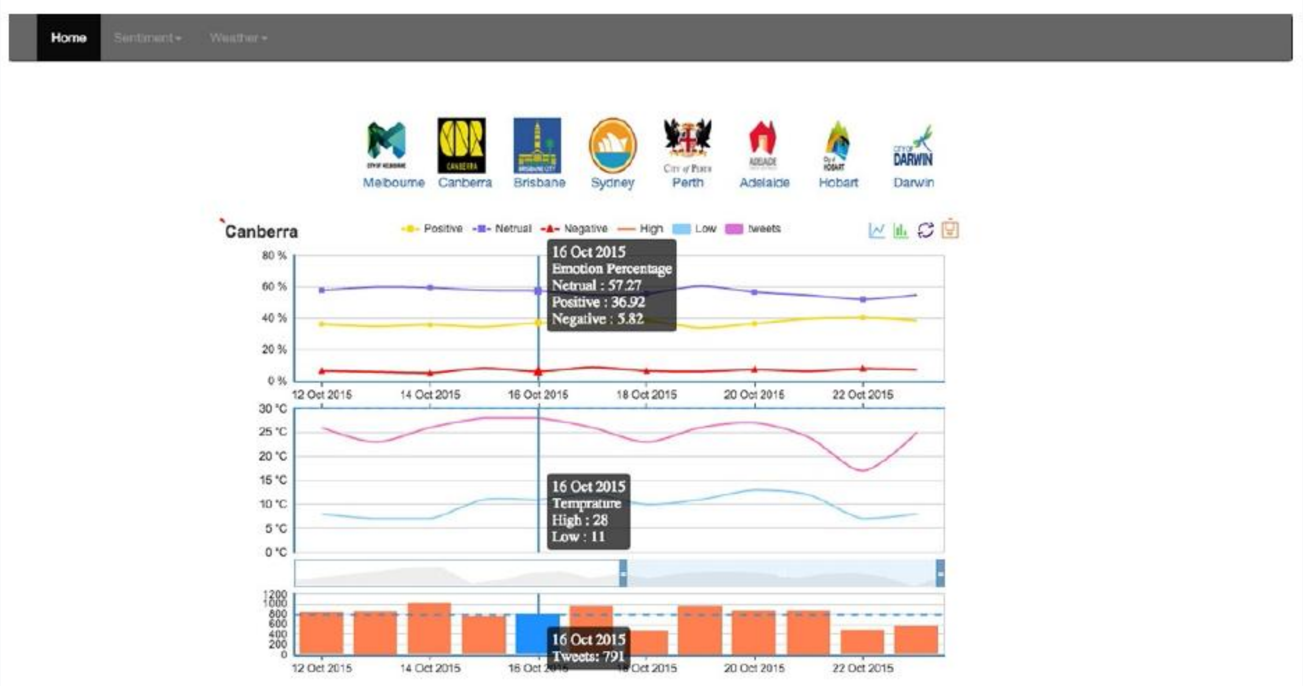
- Fig. 7.25. Original graphs for coast suburbs *(left)* and customized graphs *(right)*.

7.1.4. Exploring Twitter Sentiment Analysis and the Weather

- It is worth mentioning that in order to objectively present the trend of emotion change, the first curve graph in Fig. 7.25 is generated via calculating daily the ratio of percentage of positive/negative sentiment in shopping mall/coastal areas against the percentage of positive and negative sentiment for the whole Melbourne area.

- Fig. 7.26 shows the data for different Australian cities.
- As seen, three graphs are used to present the data (the average emotion, the temperature and the corresponding tweets numbers).
- Customization functions are offered to control the content and display of the graphs. Users can also access different cities' data via the navigation buttons on the graphs.

7.1.4. Exploring Twitter Sentiment Analysis and the Weather



- Fig. 7.26. Web pages for eight Australian cities.

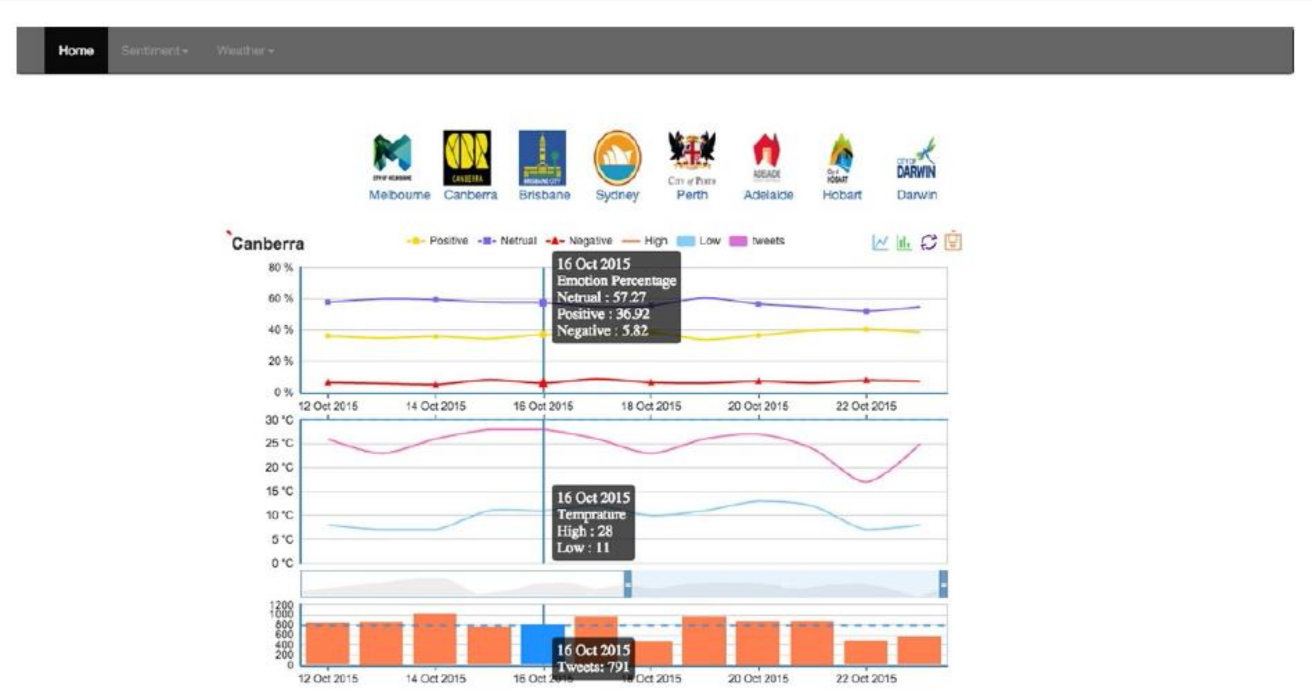
7.1.4. Exploring Twitter Sentiment Analysis and the Weather

- Finally, the weather condition of the web front end presents the positive and negative tweets percentages as well as the total tweets number based on different weather conditions (clear, cloudy, rainy, sunny, and windy).

- Fig. 7.27 shows the percentage of emotions displayed through a pie graph.
- Additionally, the bar graph is used to present the total number of tweets.

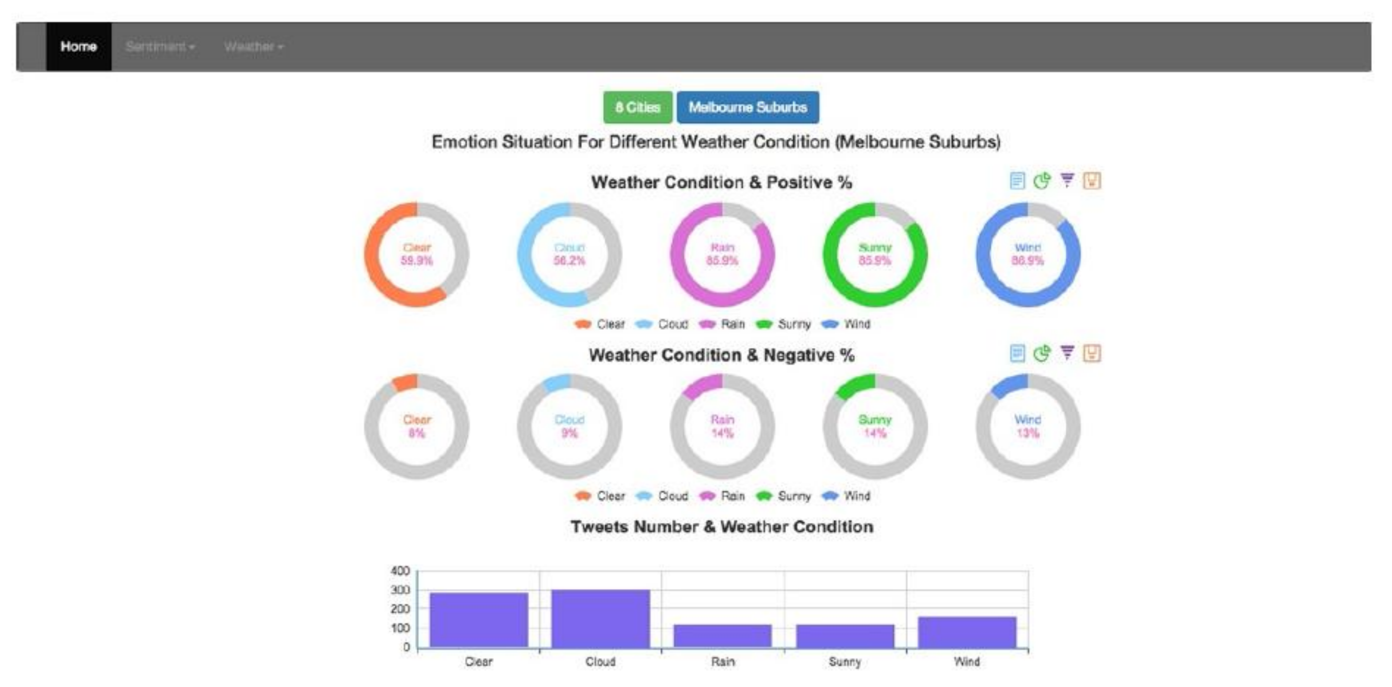
- Fig. 7.28 has been generated via calculating the average of daily positive emotion percentages based on specific weather conditions during a given period.
- The data in the bar graph in Fig. 7.28 shows the average number of daily tweets in different weather conditions during a given period.

7.1.4. Exploring Twitter Sentiment Analysis and the Weather



- Fig. 7.27. Web pages for eight Australian cities.

7.1.4. Exploring Twitter Sentiment Analysis and the Weather



- Fig. 7.28. Web page for presenting data for specific weather conditions.

7.1.4. Exploring Twitter Sentiment Analysis and the Weather

- **KEY FINDINGS**
- The correlation weather and twitter data was explored at hourly and daily intervals.
- Hourly data is especially informative because Melbourne can get through four seasons in one day.
- For daily data, the highest/lowest temperatures are typically taken into consideration.
- This is still useful since it allows for establishing emotional changes when the difference is large.

7.1.4. Exploring Twitter Sentiment Analysis and the Weather

- The correlation between weather and Twitter sentiment is presented in two parts: analysis with hourly weather data and analysis with daily weather data.

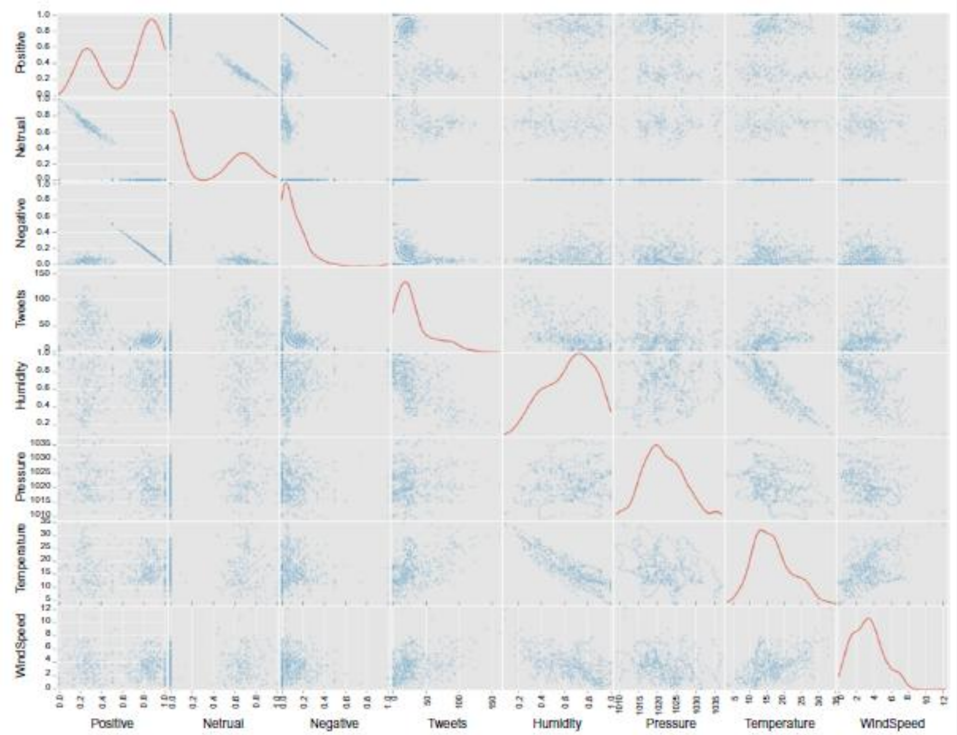
- The weather data used comprises variables including humidity, pressure, temperature, and wind speed.

- The twitter information includes positive, neutral, and negative emotion as well as the number of tweets.

- The red rectangle in Fig. 7.29 highlights these variables and their correlation scatter figures.

7.1.4. Exploring Twitter Sentiment Analysis and the Weather

- Fig. 7.29. Melbourne overall
- correlation hour variables
- scatter.



7.1.4. Exploring Twitter Sentiment Analysis and the Weather

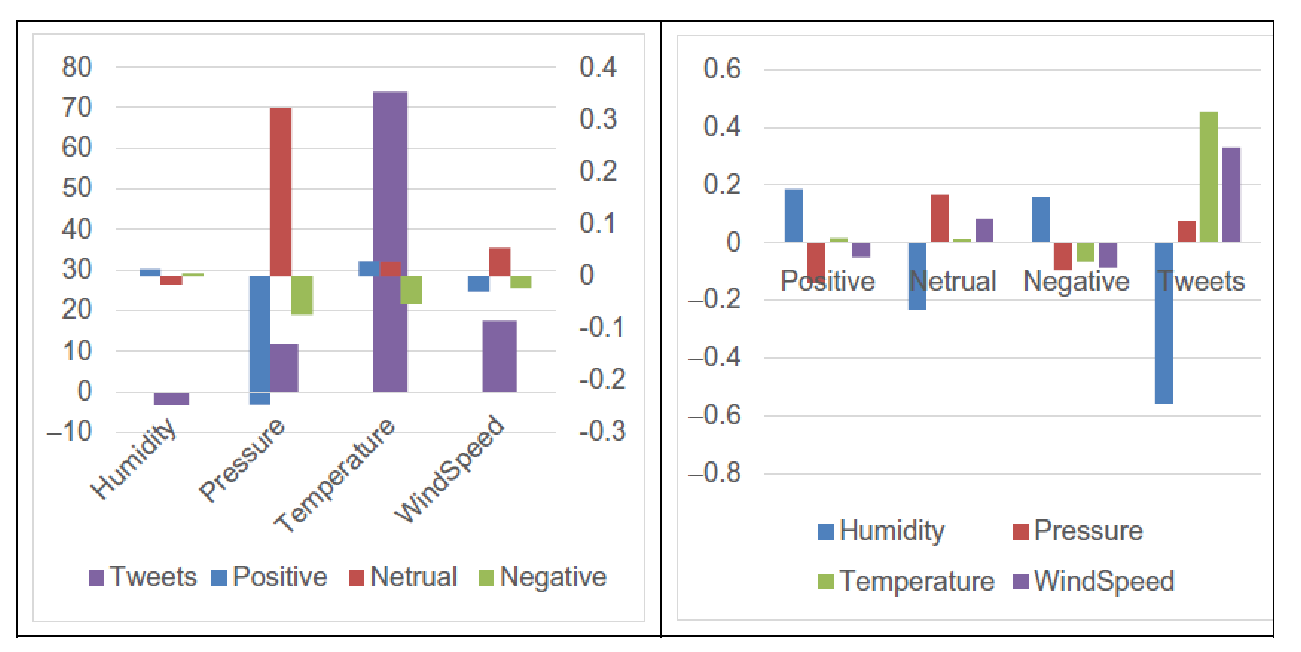
- **ANALYSIS**
- **WITH HOURLY WEATHER DATA**
- This section contains three scenarios.
- The first scenario is the overall Melbourne analysis.
- The second scenario is twitter information from air-conditioned shopping malls.
- The last one is based on Twitter analysis around the coastal regions of Melbourne.

7.1.4. Exploring Twitter Sentiment Analysis and the Weather

- As seen from Fig. 7.29, the red rectangle shows tweet numbers expressing a positive correlation with wind speed and temperature, and a negative correlation with humidity.

- Fig. 7.30 shows the detailed correlation coefficient and covariance for each variable.
- Outside the rectangle, there are some further correlations. For example, the temperature shows a negative correlation with humidity and a positive correlation with wind speed.
- The red curves in the diagonal area are the distribution for each variable.
- As one can see from those curves, most of them exhibit a Gaussian-like distribution.

7.1.4. Exploring Twitter Sentiment Analysis and the Weather



- Fig. 7.30. Melbourne overall covariance *(left)* and correlation *(right)*.

7.1.4. Exploring Twitter Sentiment Analysis and the Weather

- As seen from Fig. 7.30, twitter variables are not completely independent because covariance is not equal to zero.

- Furthermore, the tweet numbers show a strong positive correlation with temperature with around a 0.45 correlation coefficient and a negative correlation with humidity with around a −0.58 correlation coefficient.

- This means people tend to tweet less when the humidity is high or when the temperature is low.

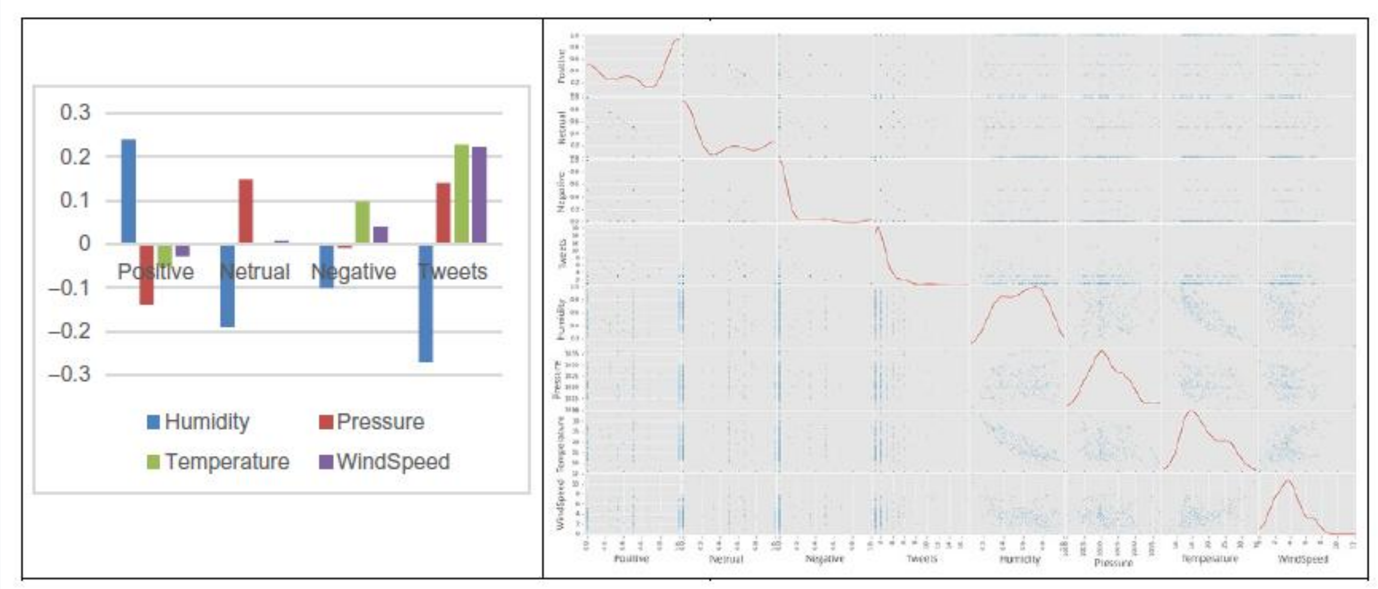
- However, the emotion shows a marginal correlation with weather attributes.

7.1.4. Exploring Twitter Sentiment Analysis and the Weather

- Fig. 7.31 shows that the correlation coefficient in shopping malls for each pair is similar to the overall area figure.

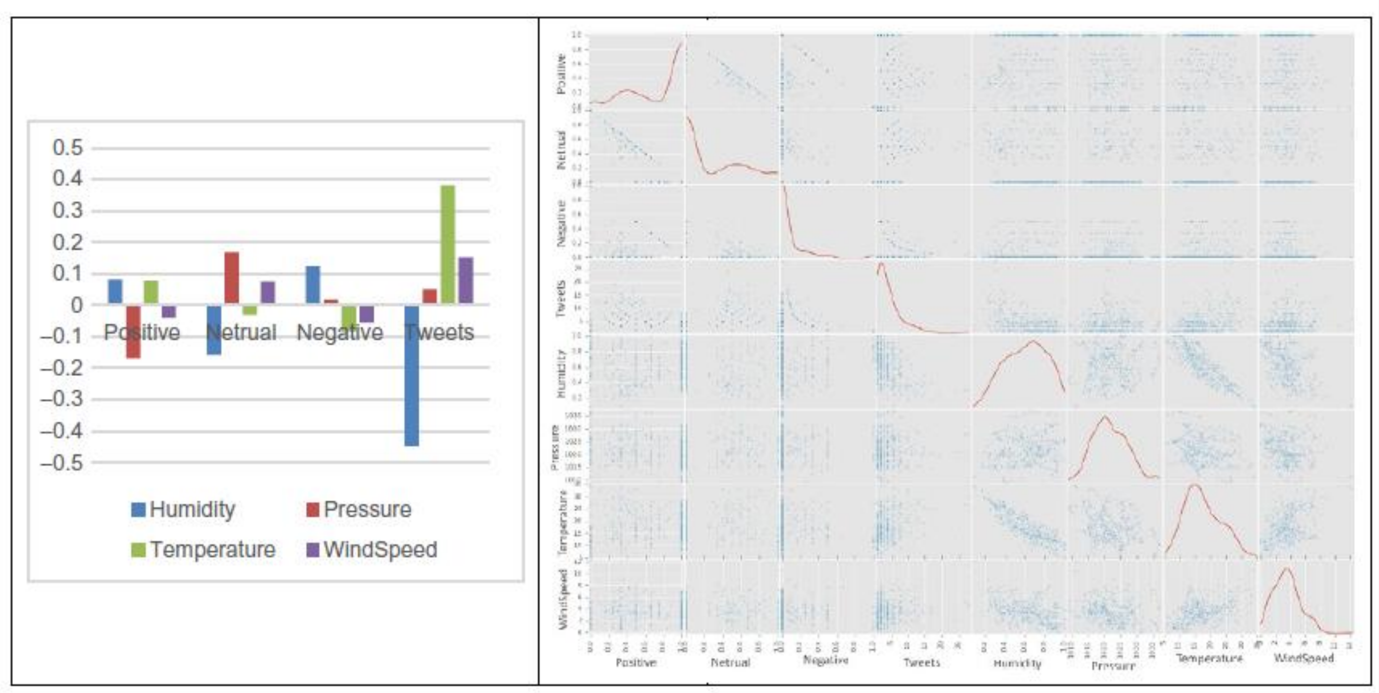
- However, the positive percentage becomes a little more positively correlated with humidity than the overall graph.
- In addition, the number of tweets is less related to weather attributes than the overall figure, even though both are negatively correlated with humidity, but positively correlated to other weather phenomenon (Fig. 7.32).
- Twitter data of people tweeting in nearby coastal regions shows less correlation with weather attributes, with all emotion attributes' correlation coefficients near to zero.

7.1.4. Exploring Twitter Sentiment Analysis and the Weather



- Fig. 7.31. Shop area attributes correlation coefficient *(left)* and scatter graph *(right)*.

7.1.4. Exploring Twitter Sentiment Analysis and the Weather



- Fig. 7.32. Coast area attributes correlation *(left)* and scatter graph *(right)*.

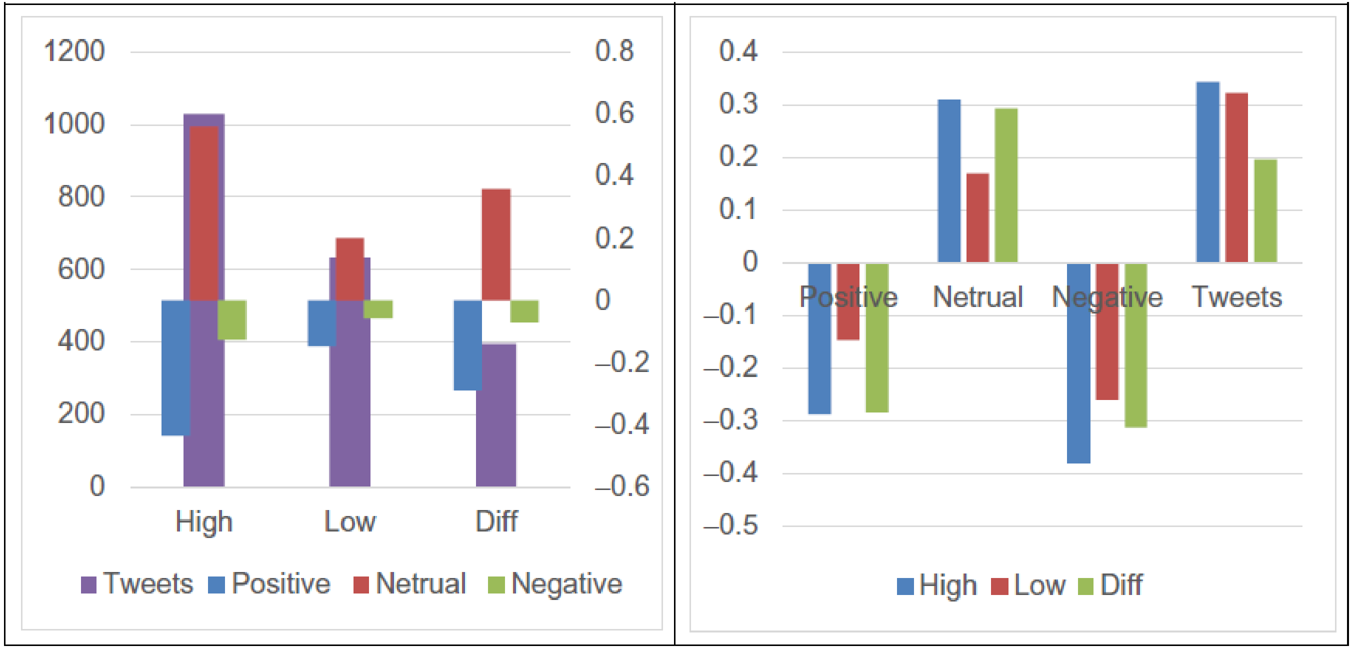
7.1.4. Exploring Twitter Sentiment Analysis and the Weather

- **ANALYSIS WITH DAILY WEATHER DATA**
- As seen, temperature shows a negative correlation with humidity and a positive correlation with wind speed.

- The results also obey this observation.

- As a result, temperature itself can represent these variables to some extent.
- Therefore, daily weather variables reflecting the highest and lowest temperature differences in one day are aligned with Twitter sentiment analysis as shown from Fig. 7.33.

7.1.4. Exploring Twitter Sentiment Analysis and the Weather



- Fig. 7.33. Melbourne overall covariance *(left)* and correlation *(right)*.

7.1.4. Exploring Twitter Sentiment Analysis and the Weather

- As seen in Fig. 7.33, Twitter and temperature variables are not independent of one other.
- However, the correlation coefficient (0.35) is not as significant as the overall correlation.
- Although the correlation is low, the correlation figure still shows that people tend to post more positive and negative tweets when the temperature is low, and they tend to post more tweets when the temperature is high or the difference is significant.

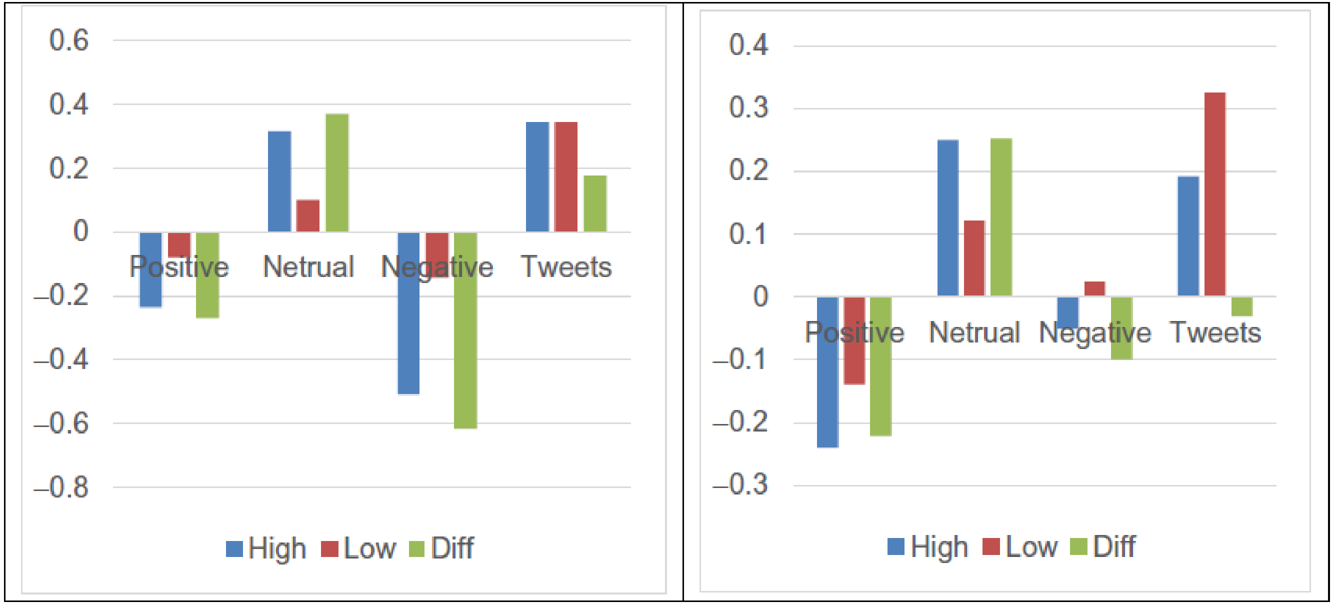
7.1.4. Exploring Twitter Sentiment Analysis and the Weather

- Fig. 7.34 shows the correlation in coastal and air-conditioned shopping areas.

- Surprisingly, the difference between the highest and lowest temperatures and the percentage of negative tweeters in coastal areas exhibit a strong negative correlation.

- One reason for this might be that when the difference becomes large, coastal areas tend to have unpredictable weather patterns.

7.1.4. Exploring Twitter Sentiment Analysis and the Weather



- Fig. 7.34. Coast *(left)* and shop *(right)* area correlation.

7.1.4. Exploring Twitter Sentiment Analysis and the Weather

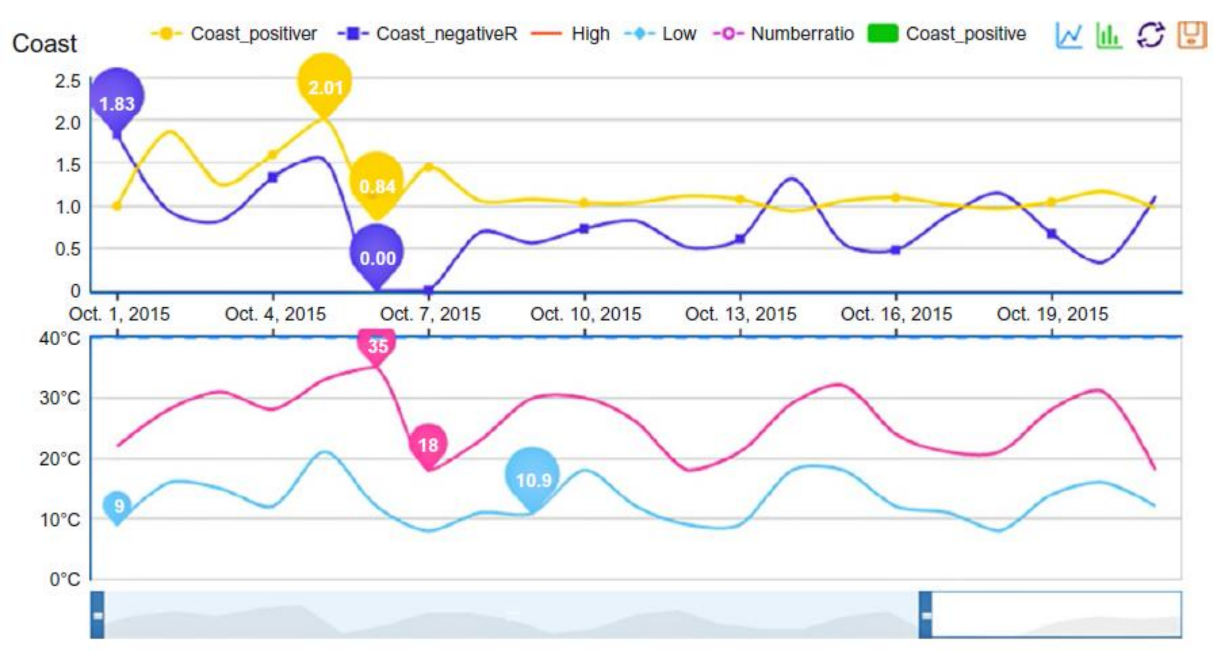
- **STRAIGHTFORWARD WEATHER IMPACT ON EMOTION**

- ***Higher temperature makes people happier in Melbourne coast***
- As seen from Fig. 7.35, people who live in coastal suburbs are the most affected by highest temperatures.

- There is a trend showing that when the temperature increases, the percentage of positive tweets increases.

- When the highest temperature reduces, the percentage of positive tweets follows a similar trend.

7.1.4. Exploring Twitter Sentiment Analysis and the Weather



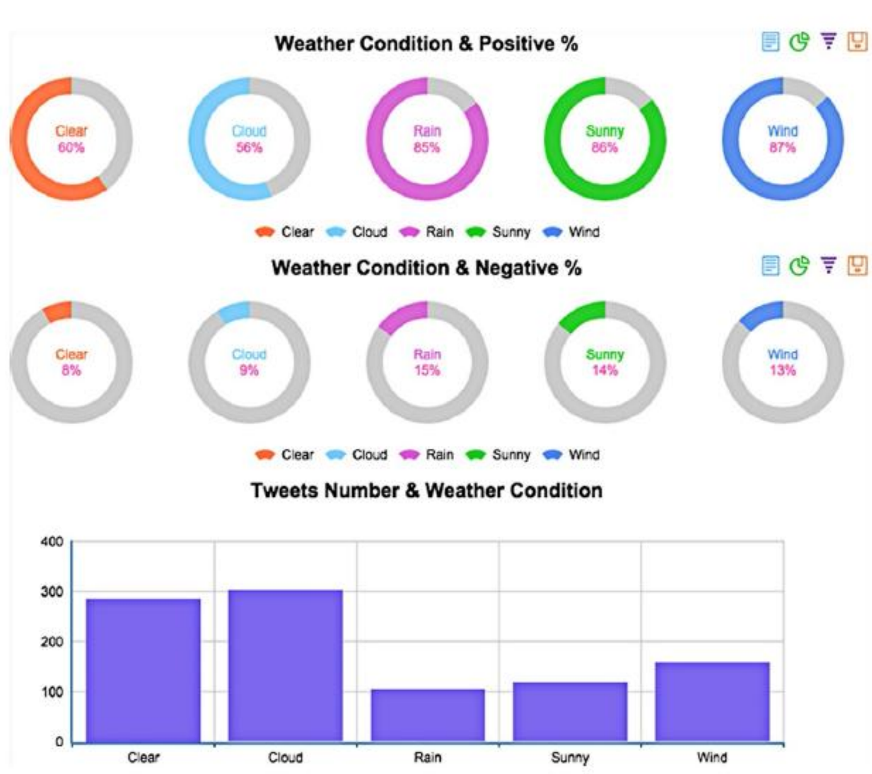
- Fig. 7.35. Emotion and weather data from Oct. 1 to Oct. 21 in Melbourne coast area.

7.1.4. Exploring Twitter Sentiment Analysis and the Weather

- ***People prefer windy days in Melbourne***
- Fig. 7.36 shows the highest factor affecting the percentage of positive and negative sentiment is the wind.
- It is also noted that 85% of people exhibit a positive emotion in rainy weather, while only 15% of people exhibit a negative emotion.
- Furthermore, it can be observed that people tend to post tweets in clear weather conditions.

7.1.4. Exploring Twitter Sentiment Analysis and the Weather

- Fig. 7.36. Emotion data
- for different weather condition
- from Oct. 1 to Oct. 26.


Weather Condition & Positive %
Clear 60%
Cloud 56%
Rain 85%
Sunny 86%
Wind 87%
Clear Cloud Rain Sunny Wind
Weather Condition & Negative %
Clear 8%
Cloud 9%
Rain 15%
Sunny 14%
Wind 13%
Clear Cloud Rain Sunny Wind
Tweets Number & Weather Condition
Clear
Cloud
Rain
Sunny
Wind