# Chapter 2
# Twitter Sentiment Tracking for Predicting Marketing Trends

**Cagdas Esiyok and Sahin Albayrak**

**Abstract** We present a web-based Twitter sentiment tracking tool for brands. The tweets about four companies, namely, Facebook, Twitter, Apple, and Microsoft are collected by this system. The collection is implemented in an hourly basis in 17 Anglophone cities from which these tweets are sent. After collecting the tweets, the system classifies them as positive or negative by using the Naïve Bayes and Maximum Entropy classification methods. Later on, the system determines the winner brand of each city according to the percentage of positive tweets sent by users located in the aforementioned cities. Lastly, the winner brands of the day can be monitored on a web page using Google Maps. To increase the performance of classification methods, the tweet texts are preprocessed, such as through converting all the letters to lower case, both for training hand-classified dataset and for the collected tweets. Furthermore, statistical tracking charts can be viewed via web page of the system. A dataset is produced by collecting 362,529 tweets in 9 days via Twitter API for the research, which is automatically classified by the system. Performance of the Naïve Bayes and Maximum Entropy classification methods is also evaluated with the hand-classified dataset.

## Carl Marks Is an Intern

Finally, holidays had started—no school for a couple of weeks. “Once we have holidays, I will do nothing but chill in the sun,” he promised himself only weeks before the last day of school. His parents weren’t too happy about this attitude though: “Carl, just one more year until you finish high school. I think you should spend this summer holiday working as an intern somewhere”, his mom told him. “Look at your sister. She didn’t do anything last year and now, she has to do an internship to find out what she is interested in”, his dad added. “She is losing a whole

C. Esiyok (✉) · S. Albayrak
Technische Universität Berlin, Berlin, Germany
e-mail: cagdas.esiyok@dai-labor.de

S. Albayrak
e-mail: sahin.albayrak@dai-labor.de



F. Hopfgartner (ed.), *Smart Information Systems*, Advances in Computer Vision and Pattern Recognition, DOI 10.1007/978-3-319-14178-7_2

year because of that. You go and find yourself an interesting internship now. It will definitely make it easier for you to decide next year what to do next." Eventually, Carl backed down. He actually already knew that he would like to become a computer scientist, but he also understood that his parents wouldn't let him enjoy his summer holidays this year. It took him two days only until he had his first offer. "Being a computer scientist is awesome", he thought. "The companies are just waiting out there to hire IT experts."

Indeed, his internship turned out to be pretty interesting. He was in particular pleased by the work environment that he found: Free soft drinks and the mandatory Foosball table in the corner guaranteed a deluxe start-up experience. While his mind was occupied with these thoughts, his project manager Sandra entered the room.

"Hello Carl, did you get a chance to have a look at my e-mail?" Sandra asked.

"Hi Sandra," Carl said, smiling. "Which one do you m–"

"The last one," Sandra said, cutting Carl off. "I have sent it just now," she added, smiling and blinking her eyes.

"Come on Sandra, how come I could get a chance to check it," he said, laughing. "I am not a superhero."

"Yeah, that's true," she acknowledged, smiling. "Please let me summarize it then . . ."

"Of course," Carl agreed, "please!"

"To sum up, the main task is tracking positive and negative comments about our company and the opponent companies in Twitter," she said, taking a deep breath.

"Hmmm," Carl pondered, "it seems that we need to develop a web-based tracking system for Twitter, don't we?" he asked.

"Absolutely right," she acknowledged. "Actually, we could separate the main task into sub tasks . . ." she added. "Firstly, the system is supposed to collect tweets about companies from several cities."

"It sounds we are going to employ the Twitter API," Carl mumbled.

"Yes, Carl, it seems that you are very familiar with Twitter due to daily online activities."

"So familiar!" he sighed, rolling her eyes.

"I remember Carl, you had told me that you wanted to work on different projects," she admitted. "But to be honest, I think you are one of the most competent interns who could achieve this task on time owing to your experiences," she asserted.

"Here I am," Carl bragged, smiling.

"Then, please stop to whine and keep on listening," she said, smiling. "The second task is to classify the tweets that you collected in the first task as positive or negative."

"Hmmmm, the second task is bipolar sentiment analysis," he said.

"Yes, it is," she told. "After sentiment analysis, for each city, your system should detect the company that has the highest percentage of positive tweets as the third task."

"It sounds as if it is a kind of competition," he said. "We are going to determine the winner company of each city according to ratio of positive tweets received."

"Kind of a competition," she agreed. "We need to pre–"

"How will we . . ." Carl interrupted. "How will we present the results?" he asked, "By means of a map or illustration . . ." he added.

"If you didn't interrupt me, I was about to say," Sandra said, smiling.

"Oops, sorry . . ." he said, looking up.

"As a last step, the winner brands of the day can be monitored on a web page using the Google Maps", she told.

"Let me conclude," he said. "The first step is collecting the tweets, the second one is applying bipolar sentiment analysis and the last step is developing an interface so as to present the results," he muttered.

"Good brief!" she told. "That's what we are going to do."

"Then, what is the main ambition of this project for our company?" he wondered.

"To detect any bad trend," she replied.

"What do you mean by detecting any bad trend?" he asked.

"I mean, by means of this system, our company can intervene in any bad trend," she told, fingering her pendant.

"Would you please give me an example?" he asked.

"For example," she answered, "reactions of users to a new product could be tracked and analyzed automatically in order to learn whether users liked or disliked it."

"That would be really nice for companies," he said, smiling.

"Definitely," she agreed. "Time is of the essence, I wish you luck Carl."

## 2.1 Introduction

As described in Carl's story, sentiments and opinions of customers might be very important for companies in our times. Social and micro-blogging platforms are mostly utilized to get this kind of information. Especially, Twitter, whose popularity

is incrementally increasing day by day, is one of the latest trends in the recent era. This 140-character-allowing micro-blogging social platform has a wide range of users varying from people to organizations, such as politicians, celebrities, and companies. According to Kwak et al. [16], the number of Twitter users was 40 million in the world in July 2009, but in August 2014, it is revealed on Twitter's official web page[1] that Twitter has 271 million monthly active users, although it is a young company established in 2006. This drastic change in the number of users sheds clear light on the growth of the company.

The recognition that such a growing company has a vital impact on everyday life has become an integral element of encouragement for researchers to conduct studies in regards to the reflections of tweets on the real world to make some predictions. For example, Asur and Huberman [3] were able to forecast box-office revenues for some movies by using the tweets. Another study showed that Twitter has a vital role in elections if used effectively. Tumasjan et al. [31] discovered that messages in favor of a candidate party can alter the election result. A similar study by Diakopoulos and Shamma [10] also demonstrated that Twitter is one of the best ways to predict the election results. In that study, the tweets, which were sent by the users during the 2008 USA presidential debate, were tracked. It was found that the number of negative tweets posted by the users was less than the number of negative tweets posted when McCain spoke. Afterward, Obama won the election against McCain. Jansen et al. [13] analyzed more than 150,000 micro-blog posts which contain brand comments, sentiments, and opinions. They showed that micro-blogging is a kind of electronic word-of-mouth of customers which are related to brands and products.

As it can be understood from the studies above, Twitter has become one of the best ways of getting customers' opinions and making predictions about the results of elections and events. Considering all the predictions made in this way, several precautions can be taken in case of an unfavorable outcome. For example, a company can decrease the prices by tracking the sentiment of tweets about a particular product. A negative trend on twitter can lead to a decrease in prices. In accordance with this kind of purposes, in this chapter, a web-based system was created in order to extract information from Twitter by tracking sentiment.

In this chapter, the primary objective is to present a web-based Twitter sentiment tracking tool. This tool collects the tweets about four brands namely, Facebook, Twitter, Apple, and Microsoft, in an hourly basis in 17 Anglophone cities from where these tweets were sent. The list of the cities used in this analysis can be observed in Fig. 2.1. After collecting tweets, the system analyzes sentiments of tweets and classifies them as positive or negative by using two classifier methods namely Naïve Bayes and Maximum Entropy. Later on, the system determines the winner brand of each city according to the percentage of positive tweets by using the information coming from the users located in selected cities. At the end, the winner brands can be seen using Google maps. For example, if the winner brand is Microsoft in New York on a selected day, the system used in this chapter shows the Microsoft logo

[1] https://about.twitter.com/company/.

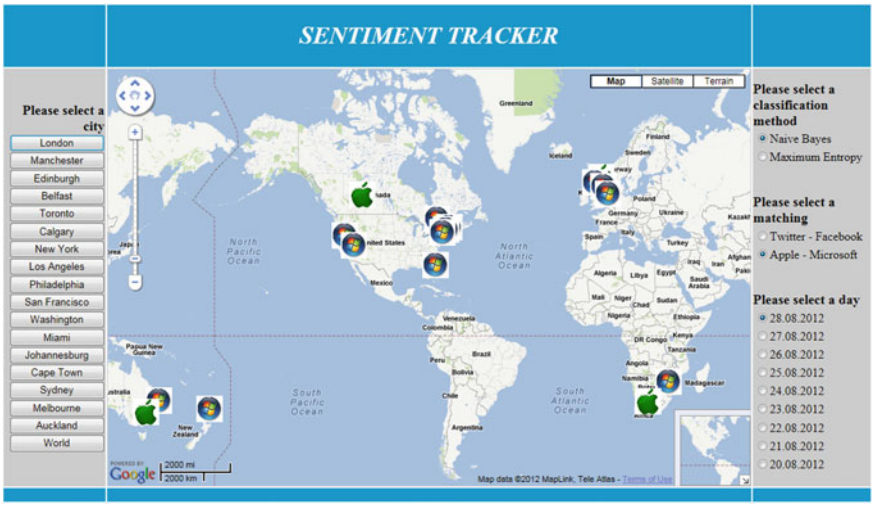


**Fig. 2.1** Screenshot of web interface

on New York in Google maps as it can be seen from the Fig. 2.1. Furthermore, the statistical indicators can be viewed by the system as well.

It is demonstrated that the Naïve Bayes classification and Maximum Entropy classification methods can be effectively used to perform sentiment analysis on tweets. To the best of our knowledge, this is the first geographic location-based sentiment tracking system for Twitter, which allows one to monitor the brands in line with the views of people in different cities.

Section 2.2 starts with the concepts of blogging and microblogging. Afterward, Twitter, Sentiment Analysis, Natural Language Processing, Maximum Entropy, and Naive Bayes classification methods are briefly described. Section 2.3 provides an informative introduction to the technologies used while developing the web-based project. Python and Natural Language Tool Kit (NLTK) are introduced, followed by Twitter API, Google Maps API, and hand-classified dataset. Section 2.3 also describes how the tracking sentiment for Twitter project was implemented. Section 2.4 covers the evaluation of the system. Section 2.5, finally, summarizes the chapter and gives a brief outlook for future studies.

## 2.2 Background

### *2.2.1 Blogging and Micro-Blogging*

Blogging can be described as a platform where people can share their hobbies and personal experiences on the World Wide Web. It has become one of the social

phenomena with Web 2.0. Also known as Weblogs, blogs are updated in a regular pattern in an attempt to incorporate most recent archived posts.

One of the most vital features of blogs is that a single author promotes and maintains each of them with the newest shares appear at the top. In general, blog posts include texts only; nevertheless, they may incorporate photos or some other multimedia content. Most of the blogs provide hypertext links that allow users to go to other websites just through a click, and many blogs make it possible for the users to leave comments. Recently, advances in technology facilitated the development of blogging and enhanced its accessibility. Blogs have been used intensively in the popular media; this has been evident with the intensive use of the blogging in political campaigns, new organizations and businesses. Blogs that are specifically allocated for politics, news, and the share of technological developments are the main blogs and websites in general that receive a great number of visitors a day.

In such a diverse environment consisting of various types of blogs, Herring et al. [12] categorized them under three types. The first one is the individually authored personal journals. The second is entitled as "filters" as they select and share commentary on information received from other websites. The last one is knowledge blogs. A vast majority of their sample consists of the personal journal type, which is responsible for 70.4 % of their sample. In this type, authors post their experiences in their lives and inner thoughts, opinions, and feelings.

One can define micro-blogging as the type of blogging that allows people to share their opinions and actions at the time of writing as short messages. In other words, it fills the gap between instant messaging and blogging. This relatively new type of blogging makes it possible for individuals to post laconic text updates, using a variety of communication channels ranging from text messages for mobile phones and instant messaging to e-mail and the Web.

The main difference between the regular blogging and micro-blogging is the text size restrictions appearing in the micro-blog posts. Micro-bloggers are permitted and confined to present their post in a limited size of text message. This feature enables micro-blogs to be amendable by sending text messages from mobile clients such as mobile phones. Appearing as an easily accessible system via mobile clients, micro-blogging has become very popular with the contributions of a wide range of users composed of average persons, celebrities, and commercial organizations. For distinct purposes, individual users such as politicians, actors, musicians, academics, and students use this blogging type regularly. Businessmen, institutions, and activists use this system intensively as well.

Micro-blogs may indicate what the micro-blogger is doing and thinking. Micro-blogs may also provide information about the news, entertainment sector, and good deals. The ones providing specific data, in general, provide reference to an external resource owing to their limited size, which makes it hard to convey the news by themselves. As broadcasting is briefly defined as spreading information over a large range of audience, micro-blogs can be used as a source of broadcasting information about anything the users want to learn about. There are various micro-blogging

services ranging from *Tumblr*[2] to *Plurk*,[3] and the most popular of all, as indicated above, is *Twitter*.[4]

### 2.2.2 Twitter

Twitter is a popular micro-blogging tool which has taken big steps since the date of establishment in October 2006. As the popularity of micro-blogs grows among Internet users, the interests of academics on micro-blogs grow accordingly. The high number of micro-blog posts enables researchers to specialize and focus on different research areas. Because micro-blogging is a new concept, the studies and research on the matter is new as well.

In Twitter, posts, or in Twitter jargon "tweets," are confined to 140 characters. The posts can consist of plain texts, links, and keywords that possess a special meaning in Twitter such as hashtags, mentions, and retweets. Hashtags are single word tokens, which follow the hash symbol, '#'. They can appear anywhere in a tweet, and they are used to tag a tweet, and a tweet can only be hash tagged by its author. Mentions, on the other hand, are the user names used in Twitter, which go after an "at" symbol, '@'. A user in Twitter can use the pattern '@<username>' in order to address another user. Retweets are used when a user wants to spread a tweet published by another user. To underline that a tweet is a repeat (re-tweet) of another tweet, users write RT in their tweets.

### 2.2.3 Natural Language Processing

Natural Language Processing (NLP) can be described as a subtitle of computer science, which deals with languages and uses Machine Learning techniques to process human language.

NLP incorporates many subfields and tasks some of which are automatic summarization, discourse analysis, machine translation, relationship extraction, and answering questions. The improvement of these fields has positive repercussion on the developments of many other areas in different fields.

The studies on NLP commenced as early as the 1940s. The very first application of NLP could be observed as a Machine Translation application developed during the World War II in order to break codes. In 1950, a criterion of intelligence was suggested by Alan Turing, which in present time is referred as the "Turing test" [32]. With this criterion, computers were rendered able to imitate a person in a conversation with a human judge.

---

[2] http://www.tumblr.com/.

[3] http://www.plurk.com/.

[4] http://www.twitter.com/.

After the 1960s, NLP studies were enriched with Artificial Intelligence (AI). With the impact of AI, NLP studies focused on world knowledge and tried to get better in the construction and manipulation of meaning representations. The first vital work shaped by AI was Green et al. [11] BASEBALL question-answering system. Starting in 1961, the system was working on the problems of addressing and constructing data and knowledge.

In 1966, the report published by the Automatic Language Processing Advisor Committee (ALPAC) asserted that the 10-year-long research could not satisfy expectations. In the light of the report, the research on NLP diminished considerably in international sphere.

Starting from the late 1980s, the inclusion of ML approaches to NLP paved the way for the resurrection of the studies. In that regard, the work of Rosenschein and Shieber [25] is of great importance. Their research handled a scheme for syntax-directed translation, reflecting upon compositional model-theoretic semantics.

The advances in computer science considerably paid off in making the 1990s the expansion period for NLP. Distinct approaches have become a source of examination with the contributions of improved computerized methods. A valuable study in the 1990s is the study of Berger et al. [5]. With an efficient implementation of the approach, their study provided a maximum-likelihood approach for automatic construction of maximum entropy models.

Joachims [14] studied on the text classifiers learning, using the Support Vector Machine (SVM). The work of Joachims is very crucial since examining certain features of learning with text data proved the suitability of SVM. Not only did it provide theoretical but it also created empirical evidence during the process of examination.

Another success story in this field comes from the study of Soderland [30]. In his research, Soderland presented a system, which was designed to cope with different text styles. The system strives to handle different sets of rules requirement problem of Information Extraction (IE) systems, by grasping the rules of text extraction automatically. It also targets to deal with various text styles in a wide range including high structured ones and free texts.

Starting from the early twenty-first century, NLP has turned out to be a rooted area incorporating distinct branches related to many areas. Numerous studies are carried out today owing to the contributions provided by NLP techniques.

### *2.2.4 Sentiment Analysis of Text*

Boiy et al. [6] define sentiments as "emotions, or as judgments, opinions or ideas prompted or colored by emotions."

Determining the attitudes, feelings, and opinions of a writer or speaker, which is in a text or video related to a topic, would be the definition of sentiment analysis. Pang and Lee [22] express this process as the computational examination of an opinion,

sentiment, and attitude. The combination of the work done for this sake is described in the literature as opinion mining, sentiment analysis, and/or subjectivity analysis.

Determining the attitude of a speaker or a writer is of great importance for the sentiment analysis as it is the main purpose of it. However, this can be very difficult at times. In Li and Wu's [17] own words: "The attitude can be any forms of judgment or evaluation, the emotional state of the author when writing, or the intended emotional communication."

In sentiment analysis, two primary approaches are used, namely, linguistic and machine learning. In linguistic approaches, studies are conducted by creating a set of rules, and then by comparing them with the analyzed text. An example of the linguistic approach could be the study of Benamara et al. [4] who proposed a sentiment analysis technique based on adverb–adjective combinations (AAC). The technique utilizes a linguistic analysis of adverbs of degree.

Devitt and Ahmad [9] put forward that sentiment analysis in computational linguistics has closely observed how textual features, such as lexical, syntactic, and punctuation, alter the emotional content of the text. Furthermore, the sentiment analysis considerably contributes to the automatic detection of these features so as to gather a sentiment metric for a word, sentence, or the whole text.

On the other hand, in machine learning approaches, methods rely on statistical evaluations and analyzes such as frequency of positive and negative entities in any text.

The reason behind the growing interest in this field stems from the benefits it can provide. The main advantages of this research area are observed in stock market. Predicting stock market behavior based on the sentiment results of Twitter posts, according to Bollen et al. [7], can result in favorable outcomes. Moreover, O'Conner et al. [21] underscore measuring public opinion poll in regards to presidential elections from blog data. Pang and Lee [22] mention the advantages of using the sentiment analysis in dealing with business intelligence tasks with respect to customer feedback.

### *2.2.5 Text Classification*

Text classification can be defined as assigning predefined category labels to documents such as e-mails to detect whether they are spam or nonspam, or web pages to detect whether they are in English, German, or Turkish.

In this chapter, a supervised learning method was used, which is to say, first a set of training documents were labeled, and then a machine learning algorithm was applied to the document for classification.

Chen et al. [8] clearly state the increasing importance text classification. They argue that the enhanced availability of digital texts and incremental increase in the need to access them rendered text classification as a vital task. For a long time until recently, various methods based on machine learning and statistical theory have been implemented in text classification.

The methods implemented in this chapter are Naïve Bayes and Maximum Entropy. These methods have been efficiently applied to text classification studies in the literature. There are many successful examples in the literature about Bayesian Probabilistic classifiers [1, 15, 28, 33] and Maximum Entropy classifiers [2, 18, 23].

#### 2.2.5.1 Naïve Bayes Method

The Binary Independence Model was developed by Yu and Salton [34] and Robertson and Jones [24] in the 1970s. The model held the status of being one of the first models utilized in probabilistic information retrieval. The Naïve Bayes Method can be briefly reviewed as follows:

Let $\overrightarrow{x}$ be a vector to be classified, and $c_k$ be a possible class. The information to be known is the probability that the vector $\overrightarrow{x}$ belongs to the class $c_k$. First, the probability $P(c_k|\overrightarrow{x})$ is transformed using Bayes' rule.

$$P(c_k|\overrightarrow{x}) = P(c_k) \times \frac{P(\overrightarrow{x}|c_k)}{P(\overrightarrow{x})} \tag{2.1}$$

$P(c_k)$, i.e., the class probability can be estimated from training data. Due to the sparsity of training data, in most cases direct estimation of $P(c_k|\overrightarrow{x})$ is impossible. $P(\overrightarrow{x}|c_k)$ is decomposed below,

$$P(\overrightarrow{x}|c_k) = \prod_{j=1}^{d} P(x_j|c_k) \tag{2.2}$$

where $x_j$ is the $j$th element of vector $\overrightarrow{x}$. So $P(c_k|\overrightarrow{x})$ becomes as follows:

$$P(c_k|\overrightarrow{x}) = P(c_k) \times \frac{\prod_{j=1}^{d} P(x_j|c_k)}{P(\overrightarrow{x})} \tag{2.3}$$

By using this equation, $P(c_k|\overrightarrow{x})$ can be calculated and $\overrightarrow{x}$ can be classified with the highest $P(c_k|\overrightarrow{x})$.

#### 2.2.5.2 Maximum Entropy Method

Nigam et al. [20] defined maximum entropy as a technique for estimating probability distributions using data. The most important rule in maximum entropy is that when nothing is known, the distribution should be kept uniform; in other words, distribution should have maximal entropy. In order to gather a set of constraints for the model, which describe class-specific expectations for the distribution, labeled training data

is utilized. The constraints are signified as expected values of "features," any real-valued function of an example.

It is noteworthy of highlighting that there is no conditional independence assumption between features, as the Naïve Bayes classifier does. Importantly, it makes no conditional independence assumption between features, as the Naïve Bayes classifier does.

## 2.3 Implementation

### *2.3.1 Technologies Used*

#### 2.3.1.1 Python

Python can be described as a programming language, which allows one to work fast and integrate the user systems efficiently. The gains in productivity and decrease in maintenance costs can be observed in the short-run once starting to use Python.

Sanner [27] defines python as "an interpreted, interactive, object-oriented programming language . . . [which] provides high-level data structures such as list and associative arrays (called dictionaries), dynamic typing and dynamic binding, modules, classes, exceptions, automatic memory management, etc." He adds that despite having a quite simple yet elegant syntax, it is a powerful programming purpose for general purpose. The language was developed in 1990 by Guido van Rossum. It is free as in the case of other scripting languages, including for commercial purposes. Another vital feature of the language is that it can be used in any modern computer.

Sanner also declared that an important resource for Python, apart from the available books, is the Python website.[5] The website generates access to code, documentation, articles, mailing lists, and packages.

The system described in this chapter mainly uses Python for collecting the tweets via Twitter API, making preprocesses on collected tweets, and writing, reading database.

#### 2.3.1.2 Natural Language Toolkit

Natural Language Toolkit (NLTK) is one of the best ways for studying natural language processing using Python. It is an open source toolkit and can be run on all platforms, which are supported by Python such as Linux, Windows, and Unix.

What NLTK means is clarified in detail on the website of the NLTK.[6] It is stated that NLTK is a platform in which Python programs are developed to work with human

[5] http://www.python.org/.

[6] http://www.nltk.org/.

language data. The system offers interfaces that are not difficult to use to more than 50 corpora and lexical resources including WordNet. Moreover, the system also provides a set of text processing libraries for classification, tokenization, stemming, tagging, parsing, and semantic reasoning.

Regarding further benefits of the system, Loper and Bird [19] stated that NLTK leads to a simple, extensible, and even framework for projects and assignments. They declared that NLTK is well documented, easy to understand how it works, and simple to use.

In this chapter, NLTK is used to classify the tweets by using the Naïve Bayes and Maximum Entropy classifier methods.

#### 2.3.1.3 Twitter API

Twitter API is described, by Sharifi et al. [29], as an API based completely on HTTP, and it is provided by Twitter. With Twitter API, users can accomplish nearly any task that can be achieved through Twitter's web interface. As for the nonwhite listed users, Twitter Rest API allocates 150 requests per hour to a user.

Fortunately, Twitter Search API, which is used in this chapter, does not have this kind of a restriction for developers. But frequency and complexity of requests is important to avoid being in blacklisted users.

Certain points are found crucial to be grasped before using the Search API. For example, the Search API is an index composed of the most recent tweets, not an index demonstrating all tweets. Currently, the index incorporates tweets of 6–9 days. Furthermore, the Search API cannot be used to search for tweets that are older than a week. Queries are subject to restrictions owing to complexity. In this case, the Search API will report an error as a response. All queries are made without identification to be provided; in other words, search does not require authentication. The search pays attention to relevance, not to completeness. This may result in some tweets and users' being missed from the search results. The Search API cannot use the near operator, so the geo-code parameter should be used. Queries are restricted to 1,000 characters, including any operators. During the process of geo-based searches with a radius, 1,000 different sub-regions will be taken into consideration when evaluating and processing the query.

In this chapter, Twitter Search API is used to collect tweets.

#### 2.3.1.4 Google Maps API

The Google Maps API, which is a free service provided by Google, allows developers to embed high-resolution maps into their web pages by using the JavaScript technology.

Furthermore, the API provides various functions that enable manipulation of the maps as well as making it possible to make additions to the content of the map via lots of services. Using this API, the users are enabled to design and create strong maps applications on their websites.

As stated in the study of Rousseaux and Lhoste [26], satellite views in high resolution are provided for certain zones as well. Apart from this, the street view that covers 360° panoramic street level views of plenty of cities is offered as well. In this chapter, Google Maps API is used to show the tracking results on the web page.

### 2.3.2 Hand-Classified Dataset

We use a hand-annotated dataset for training and testing the Twitter sentiment analysis algorithms purposes, it is composed of 1,035 hand-classified tweets as positive and negative.

In order to preprocess the raw tweets, Python script was written. This script mainly reads all tweets from our dataset and preprocesses all of them as can be seen in Fig. 2.2. Then, it writes these preprocessed tweets into a new dataset.

### 2.3.3 Background Processes

#### 2.3.3.1 Tweet Collecting

A Python script written for this chapter collects all tweets about the four brands based on 17 cities where the tweets are sent. Figure 2.3 shows the flowchart of this script.

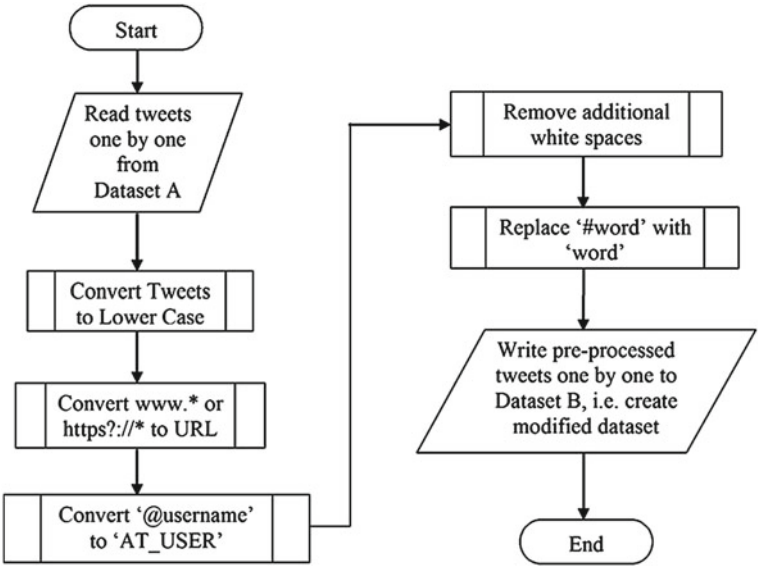


**Fig. 2.2** Flowchart of preprocess steps for dataset

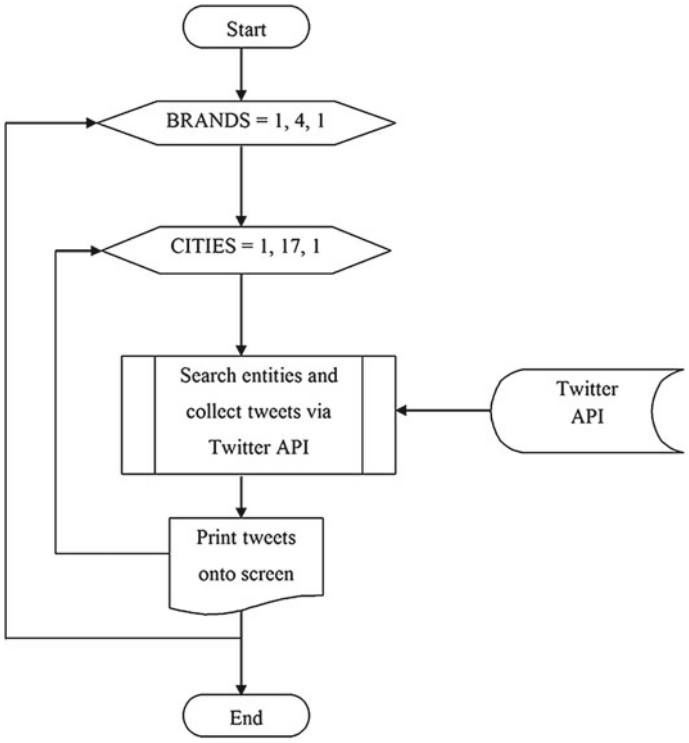


**Fig. 2.3** Flowchart of tweet collecting steps

Relevant tweets that match a query are returned in the JavaScript Object Notation (JSON) format by the Twitter API. As described on JSON web page,[7] JSON's properties, which make JSON an ideal data-interchange language, are listed as the followings: a very lightweight data-interchange format, easy for humans to read and write as well as easy for machines to parse and generate based on a subset of the JavaScript Programming Language, JSON is a text format that is not dependent upon language at all, but it uses conventions that are well-known to programmers of the C-family of languages.

#### 2.3.3.2 Sentiment Analysis

In this chapter, the Naïve Bayes classification method and Maximum Entropy classification methods are used to make the sentiment analysis. A Python script was written in order to classify the tweets collected. Basically, this script first trains the Maximum Entropy classifier and the Naïve Bayes classifier with training-modified

[7] http://www.json.org/.

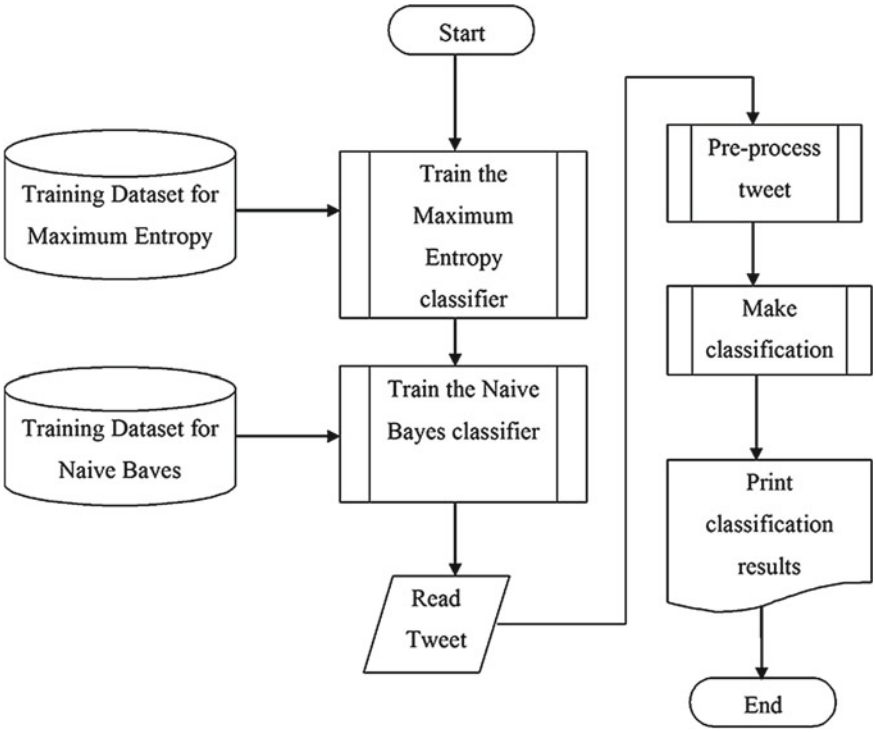


**Fig. 2.4** Flowchart of classifier Python script

datasets which are presented in the subsection of the hand-classified data set and then read the given tweets. Second, script preprocesses these tweets, and then, the tweets are classified as positive or negative (Fig. 2.4).

**Step by Step Training Process**

*Step* 1 To automatically classify a tweet, first the classifier needs to be trained. To do that, a list of hand-classified tweets is required. 512 hand-classified tweets are used to train the Maximum Entropy classifier. 1,035 hand-classified tweets are used to train the Naïve Bayes Classifier. The reason of using the 512 hand-classified tweets rather than 1,035 for Maximum Entropy classifier is to avoid the slow training process. Even when 512 tweets are used, the training process with 40 iterations takes unfeasible duration for an online system.

*Step* 2 A feature vector needs to be created. The feature vector is the most crucial item in employing a classifier. A good feature vector can foresee how successful the results of the classifier will be.

*Step* 3 After creating the feature vector, a sequenced feature list is produced. The most frequently used word is the first member of the feature list array. The feature list is used to train classifiers.

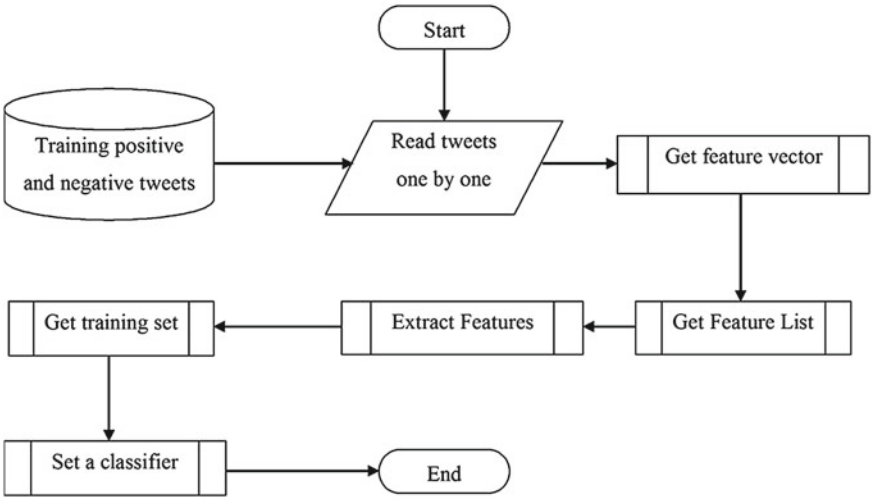


**Fig. 2.5** Flowchart of training process

*Step* 4 After producing the feature list, the next step is extracting features. In a sample tweet such as "He has changed in his bag," the feature words to be extracted are "changed', "bag," "has," and "he." Then, these feature words are examined whether they are included in the feature list words in order to extract features.

*Step* 5 Features are applied to the classifier. To sum up, a flowchart of the training process is set as it can be observed from Fig. 2.5.

#### 2.3.3.3 Automated Tweet Collecting and Classification

The tweet collector script presented in the subsection of tweet collecting and the classifier script presented in the subsection of sentiment analysis above are combined as a new Python script to produce an automated tweet collecting and classification system. First, the script trains the Maximum Entropy classifier and Naïve Bayes classifier with the training-modified datasets. Second, the system collects tweets about the four companies from the users located in several cities, and lastly, the script classifies tweets as positive or negative, and then it stores them into database. This script is converted into an executable file format to run it hourly as a background process on web server. Another reason why we converted it into the executable file format is to be able to run it without requiring a Python compiler installation. Below, Fig. 2.6 shows the flowchart of this executable file.

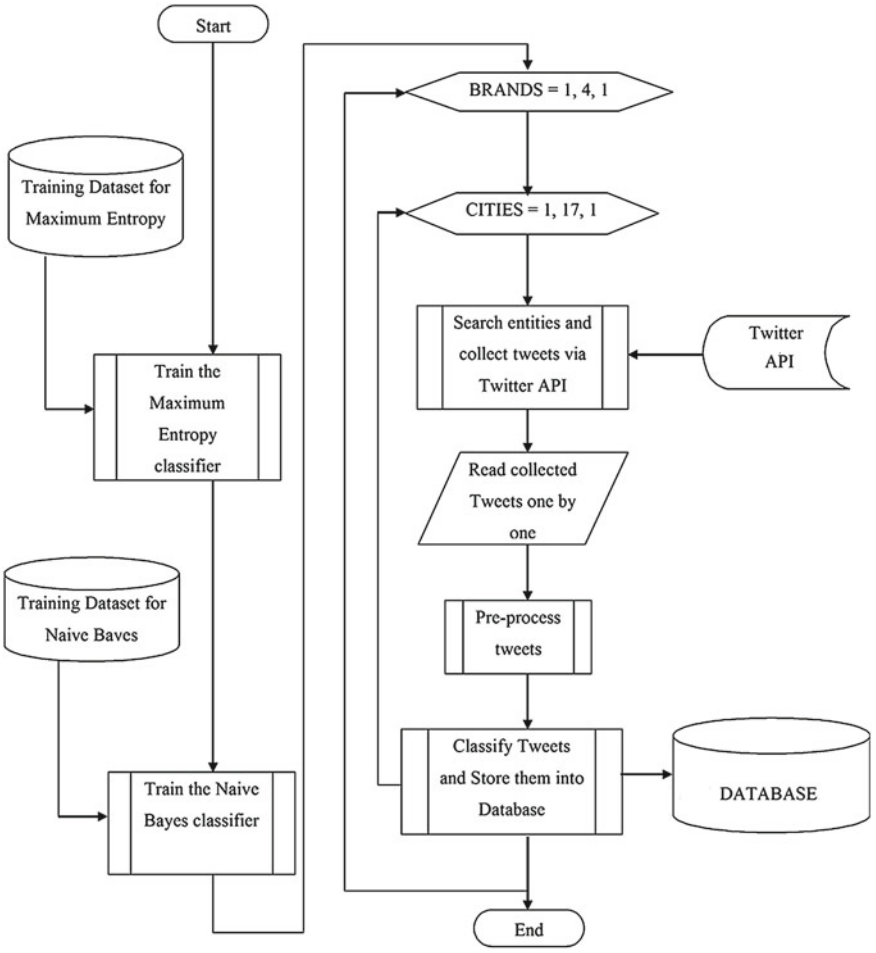


**Fig. 2.6** Flowchart of tweet collecting and classification executable file

### *2.3.4 Web Interface*

One of the main objectives of this chapter is to create a web-based sentiment tracking system for Twitter. This section briefly presents web interface of our tracking system.

#### 2.3.4.1 Main Page

The main page of web interface shows the winner brand of the day on Google Maps. In order to monitor the result, Google API and JavaScript are employed. Screen shots of the main page can be seen from Figs. 2.1 and 2.7. On the left side, users

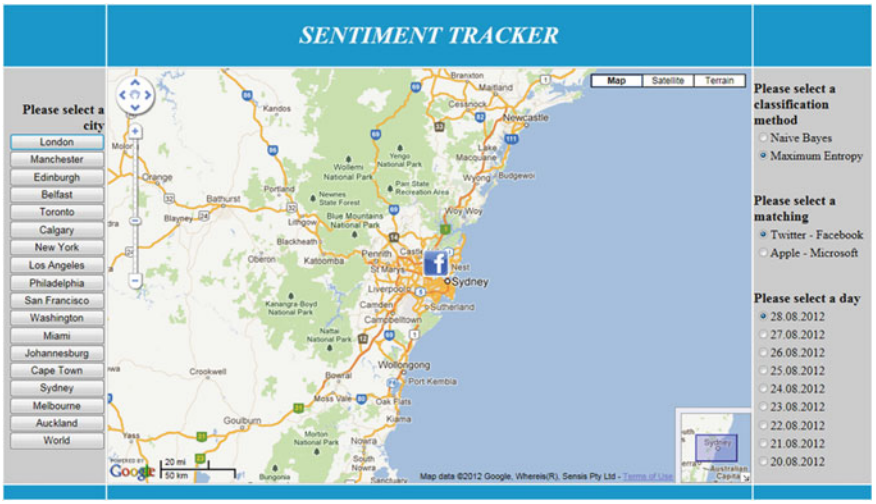


**Fig. 2.7** Screen shot of the main page after selecting the city of Sydney

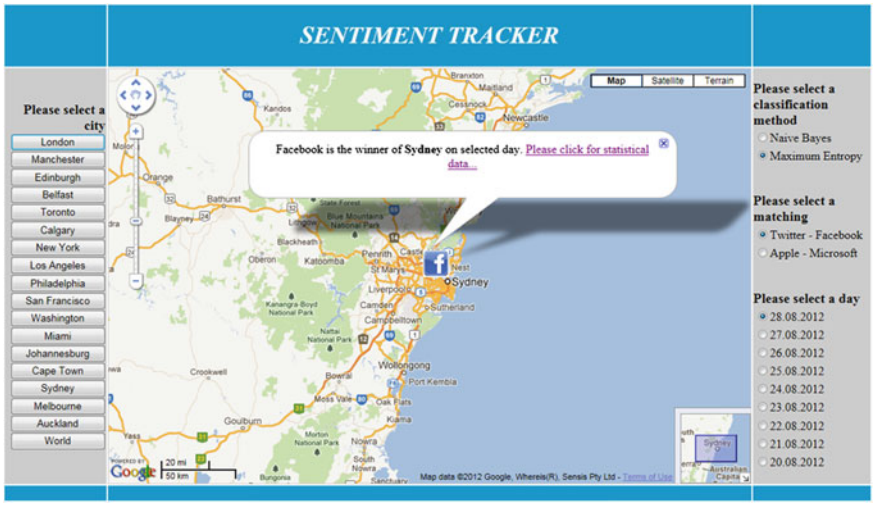


**Fig. 2.8** Screen shot of the main page after clicking on a brand logo

can select the cities in order to monitor the winner brand. Users can also choose the classification method, brand matching, and the day on the right side of the web page.

If user clicks on the brand logo, a bubble appears as it can be seen from the Fig. 2.8. By following the link, the user can monitor the statistical charts related to cities and brands.

#### 2.3.4.2 Chart Page

The aim of this page is to show the line charts of statistical data of brands according to days. If a user clicks on the "Please click for statistical data" link which can be seen from Fig. 2.8, the chart page is opened and shows the data as can be seen from the Fig. 2.9. First, it connects to database to collect the last 9 days results and then retrieves the following data to draw the charts.

- Percentage of positive tweets about Twitter classified by the Naïve Bayes method.
- Percentage of positive tweets about Facebook classified by the Naïve Bayes method.
- Percentage of positive tweets about Twitter classified by the Maximum Entropy method.
- Percentage of positive tweets about Facebook classified by the Maximum Entropy method.

#### 2.3.4.3 Trigger Page

The aim of this page is to run scheduled tasks using only ASP.NET without setting a Windows web service. By means of a Trigger page, the web interface of our system could be run on every web server which supports ASP.NET hosting. Trigger page is called hourly to do scheduled tasks which can be listed as follows. First, it calls the executable file which is responsible for tweet collecting and classification—described in detail in the sub section of automated tweet collecting and classification—so as to collect and classify the last tweets sent. Then, it connects to the database in order to draw the last count of positive and negative tweets, such as last count of tweets about Twitter by users located in London on a given date. Third, it amends the database after getting the last count of positive and negative tweets. Lastly, it deletes duplicate records because it is possible that Twitter API might collect the same tweets as collected in the previous call.

To sum up the whole system, Fig. 2.10 shows the flowchart of the Web Interface.

## 2.4 Evaluation

### *2.4.1 Performance of Classification Methods by Number of Data*

In this section, the performance of the Naïve Bayes and Maximum Entropy classification methods of the NLTK are evaluated according to the number of training data. In order to do this task, training datasets are produced based on hand-classified dataset which is described in Sect. 2.3. For all of the evaluations, the same testing

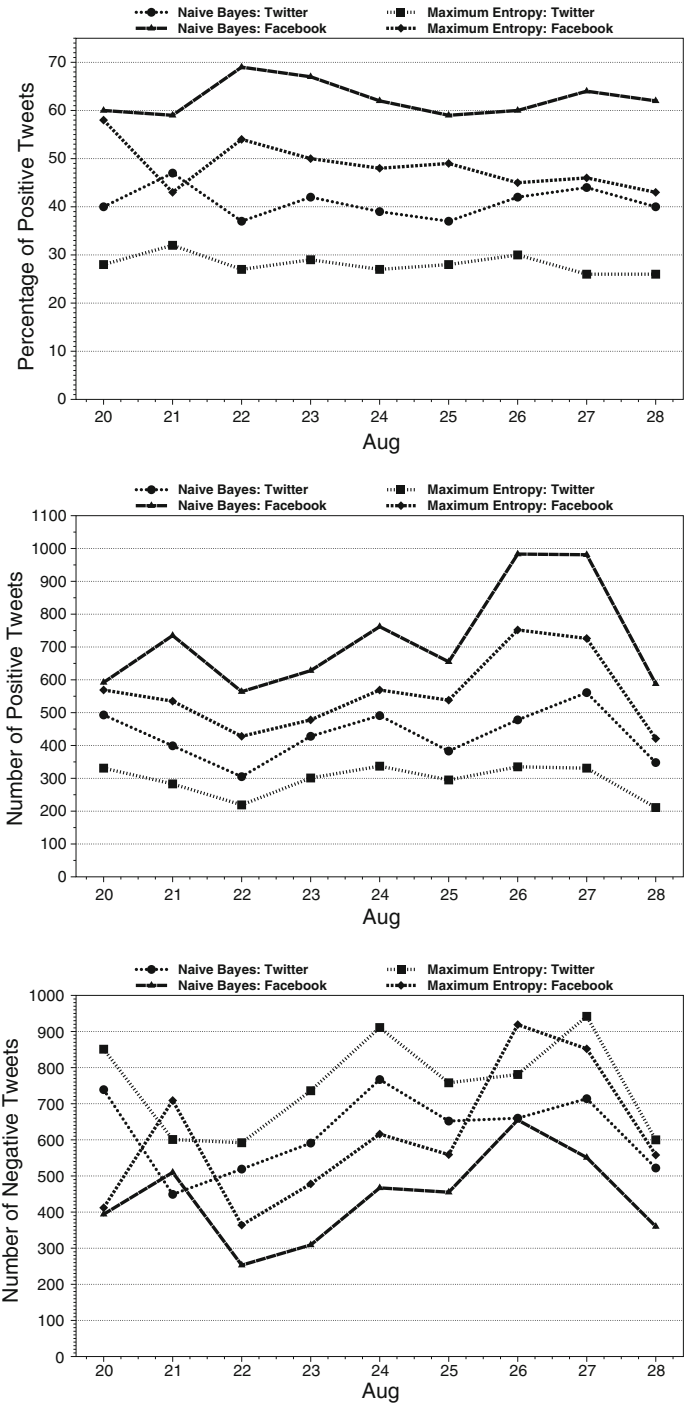


**Fig. 2.9** Charts produced by system

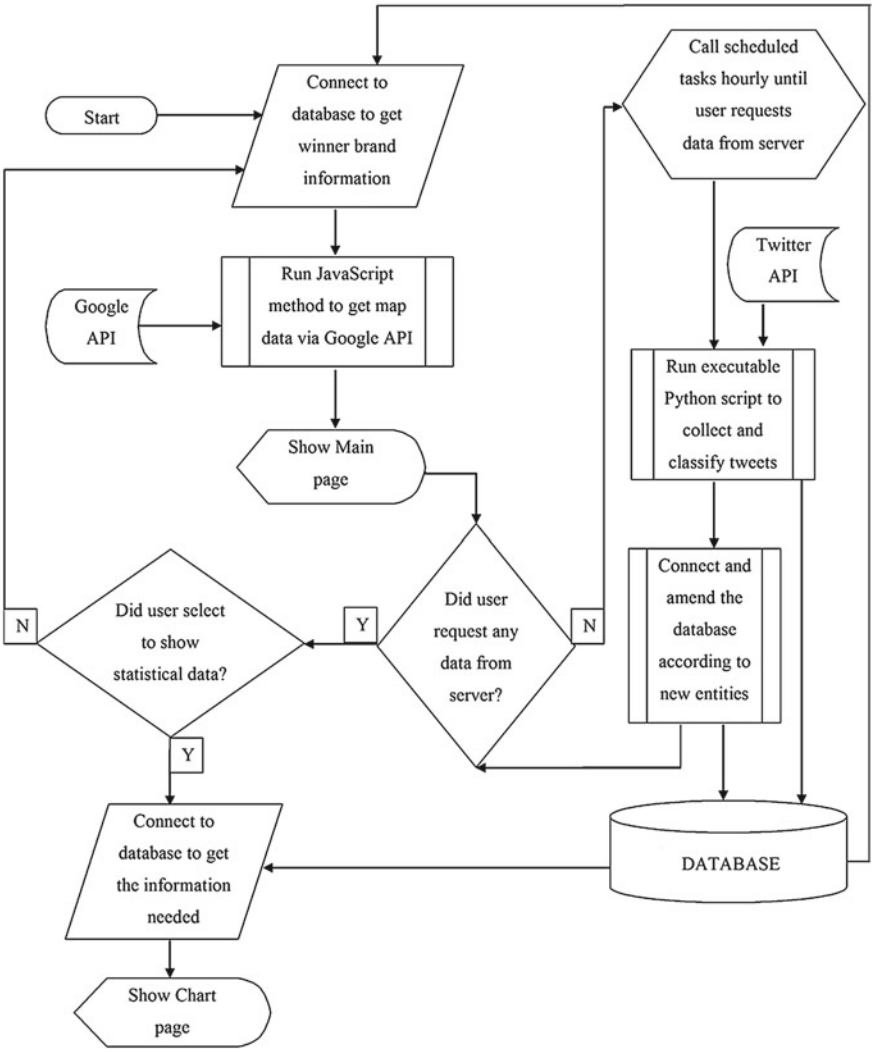


**Fig. 2.10** Flowchart of the web interface

dataset is used, and the number of data in the training set is increased continuously. Figure 2.11 shows the accuracy and the precision of classification methods.

As it can be observed from the Fig. 2.11, the precision of the Maximum Entropy reaches its optimum level at the point where the number of training data indicates 650. It reveals almost no change from 550 training data to 100 training data.

On the other hand, the accuracy level of the Maximum Entropy does not demonstrate a considerable change. It reaches its peak level at 700 training data point.

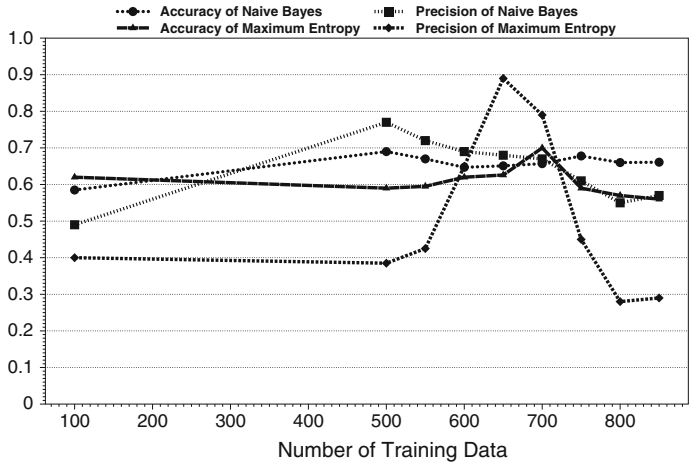


**Fig. 2.11** Accuracy and precision of classification methods through number of training data

The accuracy of the Naïve Bayes pursues nearly a straight line from 850 to 600 training data. It reaches its maximum level at 520 training data, and then follows a slow decrease.

The precision level of the Naïve Bayes increases from 850 to 520 training data, and reaches its optimum level at 520. Afterward, it demonstrates a decreasing trend.

### *2.4.2 Performance of Maximum Entropy by Iteration Number*

In this section, how the Maximum Entropy classification method of NLTK is affected by the change in the number of iterations is demonstrated. The number of iterations is set as 10 at first, and then increased by 5 until 75. The training dataset and the testing dataset are not changed, the only altered element is the iteration number of the Maximum Entropy method. The Fig. 2.12 shows the accuracy and precision values of the Maximum Entropy method.

As one can spot from the figure, the accuracy value of the Maximum Entropy demonstrates an increasing pattern until the point where the iteration number reaches 25, and then it goes through a slight decrease; however, it increases until the point where the iteration number indicates 45 where the accuracy value reaches its peak level. After the peak, the value pursues a slowly decreasing path. After the iteration number shows 50, the accuracy value of the Maximum Entropy does not change almost at all regardless of the increase in the number of iterations.

The precision value of the Maximum Entropy demonstrates a steady increase until the point where the number of iteration reaches 65 although its pace decreases for some time. It shows discontinuation for a short period of time between the iteration

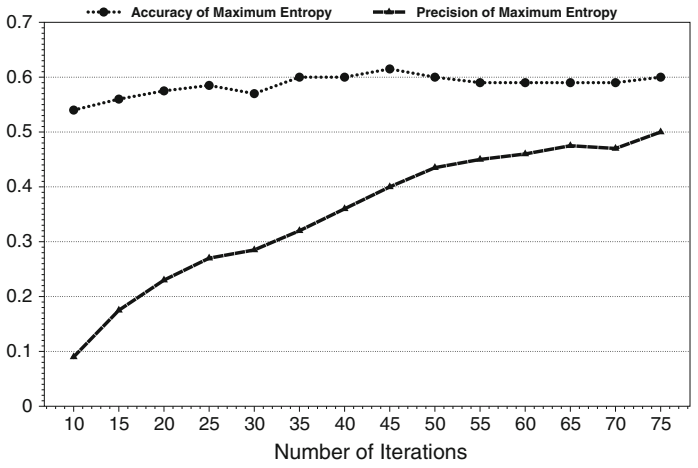


**Fig. 2.12** Accuracy and precision of Maximum Entropy

number levels 65 and 70, and then it follows an increasing trend. Considering this, it can be concluded that an increase in the number of iteration affects the precision value positively.

### *2.4.3 Sentiment Mining on Tracking Results*

A database is produced and classified by our Twitter sentiment tracking system. It has 362,529 automatically classified tweets collected by tracking in 9 days via the Twitter API.

The tracking process that commenced on August 20 was finalized on August 28. Throughout the 9 days period, recent news that might have an impact on the views of Twitter users on the four tracked brands (Facebook, Twitter, Apple, and Microsoft) are screened.

During this time frame, Microsoft announced on August 23, 2012 that it renewed its 25-year-old logo. A lot of tweets with regard to the new logo were collected. With the spread of the news, it was observed that in 11 cities among 17 cities, the percentage of positive tweets of Microsoft increased on August 24.

Particularly, the increase in four cities was remarkable. The increase in one of the cities, Miami, can be observed from Fig. 2.13.

### *2.4.4 Constancy of Tracking Results*

When the tracking result charts are monitored, it is seen that, in general, there are no big jumps or declines in the percentage of positive tweets. This stability case

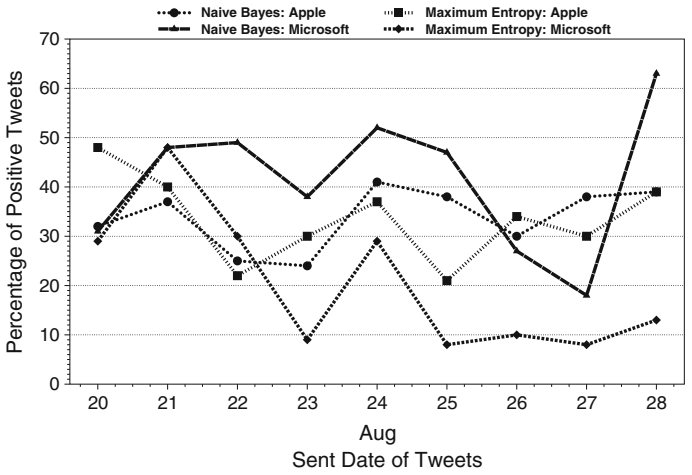


**Fig. 2.13** Increase of percentage of positive tweets of Miami on 23 August

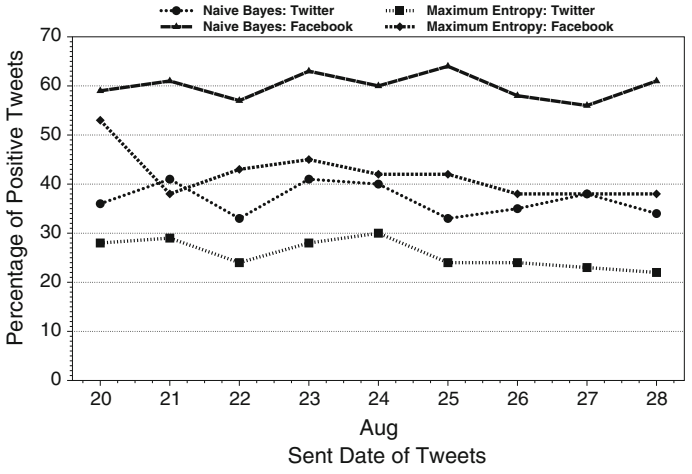


**Fig. 2.14** Percentage of positive tweets of London

can be better observed particularly in the cities where a higher number of tweets are collected such as London and New York. As it can be seen from Fig. 2.14, even though the classification method changes, the percentage of positive tweets does not change much.

As a result, the increase of data; in other words, the increase in the number of collected tweets, renders the results produced by the system more reliable. In this way, the system can generate more stable results.

## 2.5 Conclusion and Future Works

### *2.5.1 Conclusion*

Twitter is one of the most popular micro-blogging and social networking services. It allows visitors to read and post short messages limited to 140 characters. With its increasing popularity as a micro-blogging system, Twitter has become one of the best ways of monitoring the views of the users regarding certain products or things, in general. What is more, the enhanced use of the system and sharing of the users' views about specific matters before the actual date of their appearance as a concrete happening make it possible to make predictions by analyzing the current tweets. Twitter helps to identify the negative and positive opinions about a brand or a product. In order to manage the analysis of the users' posts about certain issues or things, a web-based tracking sentiment system for Twitter is developed which is able to satisfy the requirements of Carl described in the use case section.

In Sect. 2.2, first, blogging and micro-blogging are described. Afterward, Twitter and statistical data about Twitter are presented. Then, a short story of the Natural Language Processing is provided. The concept of sentiment analysis is also deeply analyzed and the Naïve Bayes and Maximum Entropy classification methods are briefly explained.

The technologies used in this chapter are laconically introduced in Sect. 2.3. First, Python and Natural Language Tool Kit are introduced. Afterward, the properties of Twitter and the Twitter API are elaborated, followed by Google API. The implementation section also covers the subsections: hand-classified dataset, background processes, and web interface. The hand-classified dataset includes 1,035 hand-classified tweets as positive and negative. To enhance the performance of our classification methods, tweet texts in the training dataset and the tweets collected via Twitter API are preprocessed. How our Python script collects and classifies tweets is represented in the subsection of background processes. In this section, basic properties of our Web Interface are described. How our system performs scheduled tasks, such as hourly tweet collection, is also expressed in this section.

Some evaluations are made in Sect. 2.4 to test and increase the performance of the Naïve Bayes and Maximum Entropy classifiers. For example, the performance of the Naïve Bayes and Maximum Entropy classification methods are evaluated by altering the number of training data. How the Maximum Entropy classification method is affected by the change in the number of iteration is shown as well.

It is also demonstrated that the Naïve Bayes classification and the Maximum Entropy classification methods can be used to conduct the sentiment analysis, but Maximum Entropy method is quite slow during the training process in comparison to the Naïve Bayes. Furthermore, 362,529 tweets are collected and automatically classified, which is described in Sect. 2.3. While collecting the tweets about the four companies in 9 days, we noticed that current news about brands have an impact on the views of Twitter users. During tracking, on August 23 Microsoft announced that the brand logo was changed. A lot of tweets were collected about that news and it was

observed that in 11 out of 17 cities, the percentage of positive tweets on Microsoft increased on August 24.

### *2.5.2 Future Works*

As future works, the number of training dataset could be increased. This will pave the way for performance-enhancing results, particularly for the Naïve Bayes classification. The preprocessing step can be further developed. To illustrate, emoticons can be used for sentiment analysis. For example, an emoticon for smiling could be placed with SMILE or "hahaha" can be altered to LAUGH. On the other hand, the Support Vector Machine method might be another method in addition to the Naïve Bayes and Maximum Entropy classifier methods. Furthermore, the tracking process may not be confined to in days. With a longer tracking period, a better and more effective data mining can be implemented. This system can be also modified in order to monitor the views of the electors on the candidates in the elections in each city. Or, how a person, institution, or opinion is perceived in different parts of the world can be tracked with a modified version of the system.

**Acknowledgments** The first author has been funded by the Ministry of National Education, Republic of Turkey.

## References

1. K.M. Al-Aidaroos, A.A. Bakar, Z. Othman, Medical data classification with Naive Bayes approach. Inf. Technol. J. **11**(9), 1166–1174 (2012)
2. D. Allard, D. D'Or, R. Froidevaux, An efficient maximum entropy approach for categorical variable prediction. Eur. J. Soil Sci. **62**(3), 381–393 (2011)
3. S. Asur, B.A. Huberman, Predicting the future with social media, in *2010 IEEE/WIC/ACM International Conference on Web Intelligence and Intelligent Agent Technology* (*WI-IAT*), vol. 1 pp.492–499 (2010)
4. F. Benamara, C. Cesarano, A. Picariello, D.R. Recupero, V.S. Subrahmanian, Sentiment analysis: adjectives and adverbs are better than adjectives alone, in *ICWSM* (2007)
5. A.L. Berger, V.J. Della Pietra, S.A. Della Pietra, A maximum entropy approach to natural language processing. Comput. Linguist. **22**(1), 39–71 (1996)
6. E. Boiy, P. Hens, K. Deschacht, M.-F. Moens, Automatic sentiment analysis in on-line text, in *ELPUB*, pp. 349–360 (2007)
7. J. Bollen, H. Mao, X. Zeng, Twitter mood predicts the stock market. J. Comput. Sci. **2**(1), 1–8 (2011)
8. J. Chen, H. Huang, S. Tian, Feature selection for text classification with Naive Bayes. Expert Syst. Appl. **36**(3), 5432–5435 (2009)
9. A. Devitt, K. Ahmad, Sentiment polarity identification in financial news: a cohesion-based approach, in *Proceedings of the 45th Annual Meeting of the Association of Computational Linguistics* (2007)

10. N.A. Diakopoulos, D.A. Shamma, Characterizing debate performance via aggregated Twitter sentiment, in *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems, CHI'10* (ACM, New York, 2010), pp. 1195–1198
11. B.F. Green, A.K. Wolf Jr, C. Chomsky, K. Laughery, Baseball: An automatic question-answerer, in *Papers Presented at the 9–11 May 1961, Western Joint IRE-AIEE-ACM Computer Conference, IRE-AIEE-ACM'61* (*Western*) (ACM, New York, 1961), pp. 219–224
12. S.C. Herring, L.A. Scheidt, S. Bonus, E. Wright, Bridging the gap: a genre analysis of weblogs, in *Proceedings of the 37th Annual Hawaii International Conference on System Sciences, January 2004*, pages 11 pp.- (2004)
13. B.J. Jansen, M. Zhang, K. Sobel, A. Chowdury, Twitter power: tweets as electronic word of mouth. J. Am. Soc. Inf. Sci. Technol. **60**(11), 2169–2188 (2009)
14. T. Joachims, *Text Categorization with Support Vector Machines: Learning with Many Relevant Features* (Springer, New York, 1998)
15. K. Sang-Bum, H. Kyoung-Soo, R. Hae-Chang, H. Myaeng, Some effective techniques for Naive Bayes text classification. IEEE Trans. Knowl. Data Eng. **18**(11), 1457–1466 (2006)
16. H. Kwak, C. Lee, H. Park, S. Moon, What is Twitter, a social network or a news media? in *Proceedings of the 19th International Conference on World Wide Web, WWW'10* (ACM, New York, 2010), pp. 591–600
17. N. Li, D.D. Wu, Using text mining and sentiment analysis for online forums hotspot detection and forecast. Decis. Support Syst. **48**(2), 354–368 (2010)
18. T. Liu, W. Che, S. Li, Y. Hu, H. Liu, Semantic role labeling system using maximum entropy classifier, in *Proceedings of the Ninth Conference on Computational Natural Language Learning, CONLL'05* (Association for Computational Linguistics, Stroudsburg, 2005), pp. 189–192
19. E. Loper, S. Bird, NLTK: the natural language toolkit, in *Proceedings of the ACL-02 Workshop on Effective Tools and Methodologies for Teaching Natural Language Processing and Computational Linguistics—Volume 1, ETMTNLP'02* (Association for Computational Linguistics, Stroudsburg, 2002), pp. 63–70
20. K. Nigam, J. Lafferty, A. McCallum, Using maximum entropy for text classification, in *IJCAI-99 Workshop on Machine Learning for Information Filtering*, vol. 1, pp. 61–67 (1999)
21. B. O'Connor, R. Balasubramanyan, B.R. Routledge, N.A. Smith, From tweets to polls: linking text sentiment to public opinion time series, in *Proceedings of the Fourth International AAAI Conference on Weblogs and Social Media* (2010)
22. B. Pang, L. Lee, Opinion mining and sentiment analysis. Found. Trends Inf. Retr. **2**(1–2), 1–135 (2008)
23. D. Quercia, J. Ellis, L. Capra, J. Crowcroft, Tracking "gross community happiness" from tweets, in *Proceedings of the ACM 2012 Conference on Computer Supported Cooperative Work, CSCW'12* (ACM, New York, 2012), pp. 965–968
24. E.S. Robertson, K.S. Jones, Relevance weighting of search terms. J. Am. Soc. Inf. Sci. **27**(3), 129–146 (1976)
25. S.J. Rosenschein, S.M. Shieber, Translating English into logical form, in *Proceedings of the 20th Annual Meeting on Association for Computational Linguistics, ACL'82* (Association for Computational Linguistics, Stroudsburg, 1982), pp. 1–8
26. F. Rousseaux, K. Lhoste, Rapid software prototyping using Ajax and Google map Api, in *IEEE Second International Conferences on Advances in Computer-Human Interactions, ACHI'09*, (IEEE, 2009), pp. 317-323
27. M.F. Sanner, Python: a programming language for software integration and development. J. Mol. Graph. Model. **17**(1), 57–61 (1999)
28. K.-M. Schneider, A comparison of event models for Naive Bayes anti-spam e-mail filtering, in *Proceedings of the Tenth Conference on European Chapter of the Association for Computational Linguistics—Volume 1, EACL'03* (Association for Computational Linguistics, Stroudsburg, 2003), pp. 307–314
29. B. Sharifi, M.-A. Hutton, J. Kalita, Summarizing microblogs automatically, in *Human Language Technologies: The 2010 Annual Conference of the North American Chapter of the Association for Computational Linguistics, HLT'10* (Association for Computational Linguistics, Stroudsburg, 2010), pp. 685–688

30. S. Soderland, Learning information extraction rules for semi-structured and free text. Mach. Learn. **34**(1–3), 233–272 (1999)
31. A. Tumasjan, T.O. Sprenger, P.G. Sandner, I.M. Welpe, Predicting elections with Twitter: what 140 characters reveal about political sentiment. ICWSM **10**, 178–185 (2010)
32. A.M. Turing, Computing machinery and intelligence. Mind **59**, 433–460 (1950)
33. K. Tzeras, S. Hartmann, Automatic indexing based on Bayesian inference networks, in *Proceedings of the 16th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval, SIGIR'93* (ACM, New York, 1993), pp. 22–35
34. C.T. Yu, G. Salton, Precision weighting; an effective automatic indexing method. J. ACM **23**(1), 76–88 (1976)

http://www.springer.com/978-3-319-14177-0

Smart Information Systems
Computational Intelligence for Real-Life Applications
Hopfgartner, F. (Ed.)
2015, XII, 372 p. 133 illus., 12 illus. in color., Hardcover
ISBN: 978-3-319-14177-0