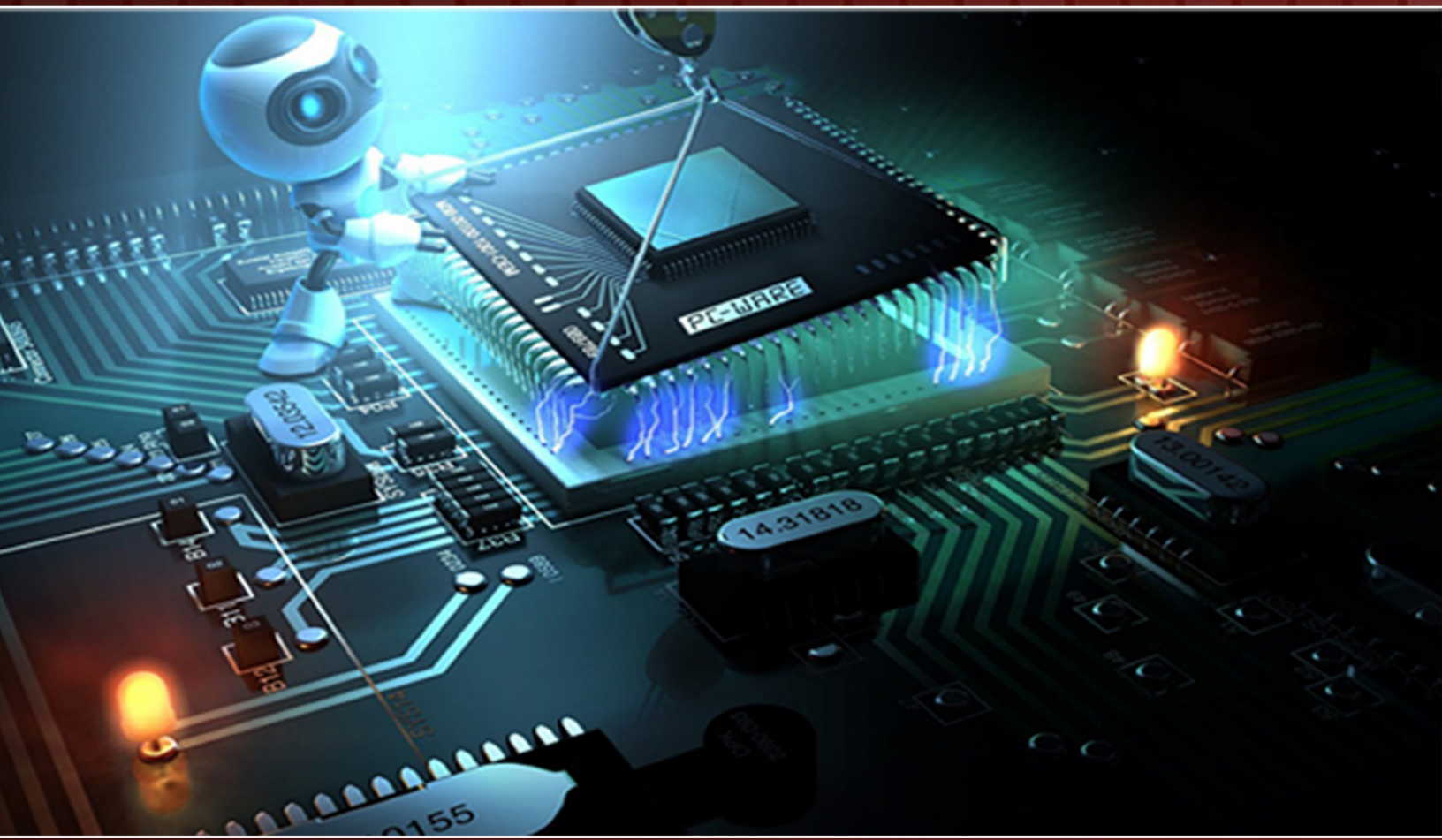
ISSN: 2054-7390

Volume 6 No 1, Feb 2018

Transactions on Machine Learning and Artificial Intelligence



Society for Science and Education
UNITED KINGDOM

TABLE OF CONTENTS

EDITORIAL ADVISORY BOARD	I
DISCLAIMER	II
SportsBuzzer: Detecting Events at Real Time in Twitter using Incremental Clustering Jeyakumar Kannan, AR. Mohamed Shanavas, Sridhar Swaminathan.	1
Extracting Sentiments and Summarizing Health Reviews from Social Media Using Machine Learning Techniques Mozibur Raheman Khan, Rajkumar Kannan	24
Temperature, Precipitation and Relative Humidity Fluctuation of Makkah Al Mukarramah, Kingdom of Saudi Arabia (1985-2016) Saifullah Khan, Yasser Alghafari	42
Publishing Student Graduation Projects Based on the Semantic Web Technologies Linh Trinh Thi Ngoc, Hung Hoang Bao, Hue Nguyen Thi Hoa; Dung Vo Hoang Phuong	59
Structural Optimization of Deep Belief Network by Evolutionary Computation Methods including Tabu Search Tomohiro Hayashida, Ichiro Nishizaki, Shinya Sekizaki, Masanori Nishida, Murman Dwi Prasetio	69
Comparison Performance Evaluation of Modified Genetic Algorithm and Modified Counter Propagation Network for Online Character recognition Adigun Oyeranmi. J, Fenwa Olusayo D, Babatunde. Ronke. S	81

EDITORIAL ADVISORY BOARD

DISCLAIMER

All the contributions are published in good faith and intentions to promote and encourage research activities around the globe. The contributions are property of their respective authors/owners and the journal is not responsible for any content that hurts someone's views or feelings etc.

SOCIETY FOR SCIENCE AND EDUCATION
UNITED KINGDOM

TMLAI TRANSACTIONS ON MACHINE LEARNING AND ARTIFICIAL INTELLIGENCE

Volume 6 No. 1
ISSN 2054-7390

SportsBuzzer: Detecting Events at Real Time in Twitter using Incremental Clustering

[1]Jeyakumar Kannan, [2]AR. Mohamed Shanavas, [3]Sridhar Swaminathan
[1,2]Department of Computer Science, Jamal Mohamed College, Tiruchirappalli, India;
[3]Department of Computer Science Engineering, Bennett University, Greater Noida, India;
meetjey@gmail.com; arms3375@gmail.com; sridhar.swaminathan@bennett.edu.in

ABSTRACT

In the recent past, twitter users are highly regarded as social sensors who can report events and Twitter has been widely used to detect social and physical events such as *earthquakes* and *traffic jam*. Real time event detection in Twitter is the process of detecting events at real time from live tweet stream as soon as an event has happened. Real time event detection from sports tweets, such as Cricket is an interesting, yet a complex problem. Because, an event detection system needs to collect live sports tweets and should rapidly detect key events such as *boundary* and *catch* at real-time when the game is ongoing. In this paper, a novel framework is proposed for detecting key events at real time from live tweets of the Cricket sports domain. Feature vectors of live tweets are created using TF-IDF representation and tweet clusters are discovered using Locality Sensitive Hashing (LSH) where the post rate of each cluster based on the volume of tweets is computed. If the post rate is above the predefined threshold, then a key event recognized from that cluster using our domain specific event lexicon for Cricket sports. The predefined threshold helps to filter out small spikes in the tweets volume. The proposed real-time event detection algorithm is extensively evaluated on 2017 IPL T20 Cricket live tweets using ROC evaluation measure. The experimental results on the performance of the proposed approach show that the LSH approach detects sports events with nearly 90% true positive rate and around 10% false positive rate. The results have also demonstrated the influence of different parameters on the accuracy of the event detection.

Keywords: Social media; Twitter; Sports event detection; locality sensitive hashing; incremental clustering.

1 Introduction

Communication between people in modern world is now happening digitally through online social media such as Facebook and Twitter, with the use of high speed internet, web and mobile technologies. Online social media have drastically changed the way of communication between people, groups, and communities [1]. Users of these social media often share information and express individual as well as collective opinions on different issues in the world. Microblogging services such as Twitter is one such form of famous and most widely used social media where diverse group of users share small digital content such as short texts, links, images, or videos [2]. Twitter allows people to share the content in the form of a short text called *Tweet* which is no longer than 140 characters, quickly and easily to the rest of the world. Tweets shared by the people contain diverse information such personal opinions, news, general

DOI: 10.14738/tmlai.61.3861
Publication Date: 26th November, 2017
URL: http://dx.doi.org/10.14738/tmlai.61.3861

information based on their individual behaviors and interests [3]. In a way, this makes the social media users as sensors which share information to the world.

Despite being a medium for users' personal information, Twitter also helps people, groups and organizations to be well informed with live information around the world. Precious knowledge can be gained by monitoring and analyzing the Twitter content continuously. Numerous organizations have now started exploiting Twitter to analyze customers' opinion on their products and services using sentiment analysis. Other than business related applications, Twitter has also become an easier source of information for societal related applications such as retrieval of real-life events, viral news content, prediction of election results and crimes. Recent studies have found that the information provided by human sensors in Twitter can be exploited for detection of real-life events. Due to a large volume of Twitter data and redundancy among Tweets representing the same events, an automation of event detection becomes inevitable.

Recent research has found that major social and environmental events such as earthquakes, deaths of celebrities, and elections can be detected using Twitter [4]. Unlike a regular Television and paper based news medium, reporting of events and news is rapid and quick in the Twitter social media where the information reaches out rest of the world within few seconds. Hence Twitter can be exploited for real time event detection. Real time detection of events has lot of real-life applications in catastrophic situations, politics, entertainment, etc. In addition to the challenges in event detection such as limited length of tweets, rumors, noises like grammar errors, typos and abbreviations, real time event detection is considered much more challenging and complex due to a difficulty in the collection and processing of large volume of Twitter data.

Whilst several event detection approaches based on strategies ranging from term interestingness to topic modeling have been proposed these years, these approaches suffer from high computational cost. Even though research on event detection has been studied well for over a decade, only a limited number of research work have been carried out in the domain of sports. Unlike detection of general events with high profile and global interests, sports event detection is targeted at the detection of events happening within a game by considering a burst of tweets in a smaller scale at a specific time, where traditional event detection approaches are slower and fails to cope with theses scalability issues. In addition, the research on event detection in sports domain focused mostly on NFL soccer game and used offline twitter datasets. Only a few recent research work have aimed to detect key events from NFL games at real-time. However, there is no research work on real-time event detection for *Cricket* sports. Other than the common challenges involved in generic real-time event detection, Cricket event detection poses additional challenges due to large volume of tweets representing Cricket events that happen frequently and rapidly almost every minute during the game.

In this research work, we study the problem of event detection at real time from live Cricket tweets. We propose a novel event detection approach by adopting LSH technique [5] for the domain of Cricket sports. For the incoming live tweets, feature vectors are computed using TF-IDF scores and clustered into different buckets that are indexed on tweet signatures. An event is detected from each active cluster leveraging post rate and is recognized utilizing our Cricket event lexicon. The architecture of the proposed event detection framework is depicted in figure 1.

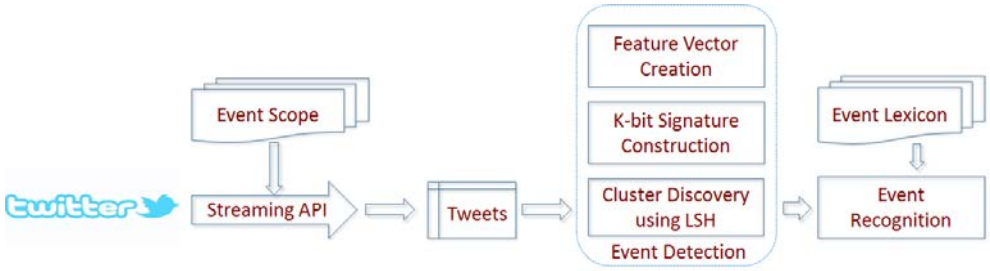


Figure 1. System architecture of *SportsBuzzer*

Our work is unique in a way that we apply LSH to the domain of Cricket sports where events are reported frequently for every minute. To the best of our knowledge, ours is the first of its kind that adopts LSH for sports with rapid events like Cricket, for discovering tweet clusters at real-time from huge volume of live tweets. The major contributions of this paper are summarized as follows:

1. Unlike previous approaches which used offline datasets and focused on NFL games, we present a novel approach which detects events rapidly in real-time from live Cricket tweets based on LSH and event lexicon. Our approach is computationally fast as it adopts LSH technique to discover tweet clusters. Since similar tweets of a particular event will fall into the same bucket, duplicate events are greatly reduced.
2. Similar to NFL soccer sports, Cricket has been one of the popular sports and attract a lot of viewers to the game. Since many viewers post tweets of key events, a widely agreed event lexicon for Cricket sports will help future research. Therefore, we propose an event lexicon that has not been reported before in any previous literature. The event lexicon represents 37 key events for the Cricket sports.

The rest of the paper is organized as follows. In section 2, we survey the related work on Twitter event detection. We introduce our data collection method and tweet preprocessing steps for creating feature vectors in section 3. In section 4, we construct signatures to represent tweet feature vectors in LSH. We describe our LSH based real-time event detection method in section 5 and examine the performance of the proposed approach in section 6. Finally in section 7, we conclude the paper and describe our future work.

2 Related Work

A considerable number of recent research work can be found in Twitter event detection. Based on the domain of which events occur, event detection approaches can be generally classified into different categories such as social, political, environmental, and sports event detections. Recent approaches on event detection can be categorized based on the different classes of solution such as term-interestingness based, incremental clustering based, topic modeling based and frequency based approaches [6]. The readers are recommended to refer to some recent surveys [2, 6] for more detailed comparison of approaches on Twitter event detection. However, recent research on twitter event detection is discussed in this section.

Previous work in twitter event detection focused mostly on detecting physical and social events. Sakaki et al [7] and Qu et al [8] aimed to detect earthquake incidents using Twitter. In another work, Vieweg et al [9] studied detection of natural events such as *grassfire* and *floods* using microblogs. TwitterStand [10] clustered tweets to discover news topics from the Twitter data. Sakaki et al [11] proposed an approach

for detecting events such as *earthquakes* and *typhoons* where the authors exploited SVM which is trained on manually annotated Twitter data containing positive and negative tweet samples. Popescu and Pennacchiotti [12] analyzed public discussions happening in Twitter for detecting controversial events related to celebrities. In a related work [13], the authors proposed concert event detection based on factor graph model where the clusters representing events are formed automatically and a canonical value is produced for each event. One of the main drawbacks of the approaches discussed so far is that they detect events from Twitter only several minutes after the actual event.

Becker et al [14] proposed an approach for detecting planned events from twitter data which is filtered using precise queries representing the events where the queries are formed by combining simple rule-based query building strategies. Becker et al. [15] proposed a centrality based approach for extracting high-quality and useful Tweets that are related to different events. The authors in [16] proposed an event detection approach based on generative language modeling using quality indicators of microblogs where query expansion is used to collect messages from microblogs. Weerkamp and de Rijke [17] extracted quality indicators which are useful for event detection such as different emoticons, tweet post length, expressions, word capitalization and URLs. Gu et al. [18] proposed an N-gram based event modeling approach called ETree which uses content analysis approaches for grouping large volume of tweets.

Some approaches exploit geo-locations associated with the tweets for event detection. Valkanas and Gunopoulos [19] proposed an event detection system which clustered users based on their geo-locations where the event detection is achieved by monitoring a sudden change in the emotional state of the user groups. Lee and Sumiya [20] proposed a geo-social event detection system which detects local festivals based on modelling and monitoring of crowd behavior in Twitter. The approach analyzes the regularity in geographical locations using geo-tags of the twitter data.

Since words and their frequencies in tweets are highly correlated to the specific events in general, several term interestingness based approaches have been proposed. Twevent [21] detects nontrivial word segments using statistical information of continuous and non-overlapping word segments in tweets. Bursty event segments are extracted using a fixed window based frequency detection approach where relevant event segments are clustered to filter events based on newsworthiness score using Wikipedia sources. TwitInfo [22] detects spikes in the twitter data and labels them automatically using meaningful and most frequently occurring terms. Gathering of initial tweets are achieved using input keywords from the user where the relevant tweets are crawled. Finally the peaks in the large twitter volume are labeled as sub-events by the system.

TwitterMonitor [23] system detects emerging topics by considering whether high frequency terms co-occur within a small time window that has tweets with bursty terms, and finally applies a greedy strategy to generate groups to reduce computational costs. In a similar system enBlogue [24], emerging topics are detected by measuring window based tag pair statistics where tag correlation shifts in unusual manner are marked as emerging topics. Weng and Lee [25] developed an event detection system EDCoW, exploited analysis of wavelet on word frequencies by calculating new features of words. The authors in [26] aimed to detect emerging topics in Twitter by comparing the frequency of current words and previous words using a directed graph comprising terms that belong to emerging topics.

Some previous work exploit the concept of topic modelling for event detection in Twitter. In an event detection system TwiCal [27], structured representation of important events are extracted from the twitter stream using a latent variable based model for open-domain event detection. A similar system called LECM (Latent Event and Category Model) [28] utilizes semantic concepts to classify different types of events. Hannon et al. [29] exploited post rate of tweets to produce highlights of a World Cup game in an offline mode. However their system was not able to detect the specific events from the game. Chakrabarti and Punera [30] utilized Hidden Markov Models which are trained for representing events in a game. It should be noted that most of the previous work in the domain of sports have not focused on real-time event detection.

Incremental clustering algorithm [31] is utilized for detecting events from Twitter stream where the similarities between a tweet and event clusters are computed for identifying newsworthy events using SVM. Named Entity Recognition was exploited [32] for event detection and tracking where bursty events are detected using named entities in tweets. Few approaches [33], [34], [35] exploited Locality Sensitive Hashing to measure the novelty of a tweet by comparing with previous tweets. Based on the novelty of a tweet, it is further processed for new event detection.

It can be seen that the event detection is accomplished using different strategies. The earlier approaches focused more on detecting physical and social events such as celebrity events, natural disasters, elections etc. Few approaches exploited the network features such as geographical locations of the tweets while others used strategies such as measuring term interestingness and exploiting topic modeling for event detection. It should be noted that most of these previous work were performing event detection on offline datasets. Even though a few work have been carried out in the domain of sports mainly focusing on NFL soccer games, there is no previous work on event detection for Cricket sports, to the best of our knowledge.

3 Preprocessing of sports tweets

With a length of just 140 characters, Twitter has the shortest delay in delivering user comments to citizens, compared to other social media platforms such as *blogs*. As tweets are highly noisy, which contain URLs, mentions, replies and others, preprocessing has been a fundamental step in detecting key events from live tweet streams.

3.1 Collecting live tweets

Cricket fans and audience of a live game post tweets about interesting moments throughout game time. So, these twitter users can be considered as sensors who can deliver current updates about key events (e.g. Boundary, Sixer, Catch) in a game at real-time. *SportsBuzzer* relies on and leverages these sensors to collect data and perform robust detection and recognition of key events at real-time.

SportsBuzzer requires live tweets which can be filtered based on some relevant keywords and without any maximum limit for streaming. Hence, *Streaming API* of Twitter (www.twitter.com) is the most suitable type of data collection for our real time event detection task. With the help of Streaming API, we will be able to collect live tweets continuously, based on the scope of events such as hashtags. For instance, we have used a keyword *RCBvRPS* to stream all live tweets at real-time when the game was ongoing. The keyword *RCBvRPS* denotes an IPL T20 cricket game between the teams *Royal Challengers Bangalore* (RCB) and *Rising Pune Supergiant* (RPS) that was held during April 2017 in India. Our *SportsBuzzer* runs

continuously collecting tweets without any break during the entire game time, detects events from tweets at real-time and also archives all gathered tweets in JSON format for later offline analysis.

3.2 Removing noise from tweets

Figure 2 depicts the work flow of converting a raw cricket tweet into a feature vector. The feature vector will be used as input for the clustering process. The creation of a feature vector consists of two steps - preprocessing a given raw tweet and preparation of the feature vector from the preprocessed tweet. In the preprocessing step, a set of features (i.e. unigrams) are extracted from each tweet. For simplicity, only unigram features from a tweet are considered in this work. Later, each preprocessed tweet is converted to its equivalent feature vector. Both steps are explained below in greater detail.

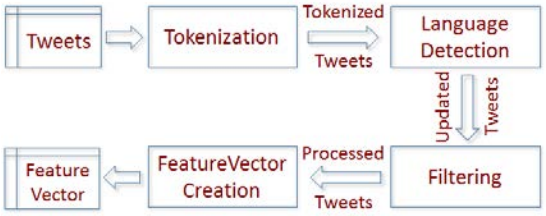


Figure 2. Work flow for building tweet feature vector

The preprocessing step eliminates noise which reduces the accuracy of an event detector. Here, each tweet is first tokenized into a sequence of terms. Only tweets posted in English are considered for further processing. In Information Retrieval (IR), the most commonly occurring words in a text document are called *stop words*. The tokenized tweet is checked against a standard stop word list. The list of stop words from Python's Natural Language Tool Kit (NLTK) is used to eliminate all terms that are least meaningful and less contributing to event detection. For instance, articles in English such as *a*, *an* and *the* are removed from tweets. The URLs and all token starting with '@' (i.e. a mention or reply) are also removed from the raw tweets. Tokens that contain only alphanumeric characters are treated as valid tokens. At the end of this process, each tweet has only features that will be included in the vector space model.

3.3 Creating tweet feature vector

Mathematically, a vector is represented by its direction and magnitude. Many term weighting schemes use values or magnitudes to represent the features of a text document such as tweet. Term Frequency - Inverse Document Frequency (TF-IDF) has been one of the most fundamental term weighting schemes in IR, which is used to represent the features of a tweet as numerical values. TF-IDF is formally defined as a product of term frequency (TF) and inverse document frequency (IDF). Here, TF represents the importance of a term in a document and IDF represents the importance of the term in the entire document corpus. The TF-IDF weighing scheme assigns weight to term *w* in document *d* using equation 1 and 2 [36]:

$$tf - idf(w,d) = tf(w,d) * idf(w) \quad (1)$$

$$idf(w) = log \frac{N}{df(w)} \quad (2)$$

Here, $tf - idf(w,d)$ is known as TF-IDF weight of a term *w* in document *d*. The term frequency $tf(w,d)$ represents the number of times the term *w* occurs in a document *d*. The inverse document frequency $idf(w)$ helps to scale down the term frequency of w, if the term w occurs in almost all documents in a

data corpus. Here, *N* is the number of documents in a data corpus and $df(w)$ denotes the number of documents in which the term *w* occurs at least once. Then, the features of a tweet are mapped to the vocabulary in order to generate the tweet feature vector where TF-IDF weight is assigned to a vocabulary term that appears in the tweet. For simplicity, we keep our vocabulary as a static dictionary.

4 Incremental Clustering of sports tweets

In this section, we first describe the problem of finding nearest neighbors and approximate nearest neighbors. Then, we examine the suitability of locality sensitive hashing for the incremental clustering of tweets. Incoming live tweets have to be represented as feature vectors so that they can be clustered in LSH. We finally explain the process of construction of signatures for tweet feature vectors which are the indices for the LSH approach.

4.1 Approximate nearest neighbors

Given a set of N points P = {P_1,P_2,P_3,…,P_N} represented as a matrix M and a query point Q, a nearest neighbor search finds a point in P that is closest to Q ∈ M [5]. In case of vector space model, points are documents and a query point is the document to be searched. The nearest point to a query point can be found by simply computing the distance between all points in *P* to *Q* and selecting the one point $P_i \in P$ which is the closest to *Q*. The nearest neighbour approach is computationally expensive for high dimensional data such as documents represented in vector space model. This problem is known as *Curse of Dimensionality* [5].

To alleviate this problem, a variation known as Approximate Nearest Neighbour (ANN) search was proposed. The ANN search finds an approximate nearest neighbor point P' in P that is the closest to Q within a radius r, as shown in equation 3.

$$\forall P' \in P, d(P', Q) < (1 + \varepsilon) d(P', Q) \tag{3}$$

Here, $d(P', Q)$ is the distance between P' and Q and $(1 + \varepsilon)$ is a constant factor [5]. Locality Sensitive Hashing (LSH) [5] is a popular approach to address the problem of ANN search. In this paper, we have adopted LSH technique to find tweet clusters from which events are detected by computing the post rate of a cluster. Once an event is detected, then it is recognized using the event lexicon of Cricket sports.

4.2 LSH for incremental clustering

The LSH technique has been applied to information retrieval, pattern recognition, dynamic closest pairs and fast clustering problems [5]. The key idea of LSH is to apply hash functions in such a way that the probability of collision is much higher for similar objects than for dissimilar objects [5], i.e. the objects close to each other will most likely fall into the same bucket. Intuitively, a hash function is locality sensitive if two points that are close under the similarity distance measure are more likely to collide.

In LSH, initially a set of points will be preprocessed and stored into L number of buckets. Each point is hashed using *k* hash functions and stored in L buckets. The concept of bucket can be implemented in several ways. Hash table is an obvious choice for representing a bucket. The hash value of data points acts as an index of a hash table. With the hash value of a query point *q*, all buckets are searched to retrieve the points that are similar to *q*. The similarity distances between *q* and each of the retrieved similar points are computed and the one point that is close to *q* is selected as a nearest neighbor for the query point *q*. The LSH scheme proposed by Charikar [37] applied cosine similarity metric to compute the similarity between two document vectors, where cosine similarity is a dot product of feature vectors normalized by

their norms. The cosine similarity will become 1 if the document vectors are parallel and 0 if the document vectors are orthogonal to each other.

4.3 Signature for tweet feature vector

Traditional event detection methods do not provide sufficient solutions to handle the exponentially growing social media streams where LSH has become a successful solution for processing these large data streams. The LSH approach applies hash functions such that the probability of collision, i.e., falling into the same bucket, is much higher for similar tweets than that of dissimilar tweets. The gap between two dissimilar tweets should be larger enough so as to prevent the collision of dissimilar tweets into the same bucket. In this research, our proposed methodology for discovering clusters of tweets uses hash table based LSH for computing the nearest neighbours.

We adopt the LSH approach proposed by Charikar [37] that defines a hash function *h* to generate a *k*-bit signature for the tweet feature vector. The hash function (equation 4) computes the dot product between the tweet feature vector *u* and *m*-dimensional random unit vector *r* and retains the sign of the resulting product. Each dimension in *r* is drawn from Gaussian distribution with mean 0 and variance 1.

$$h(u) = \begin{cases} 1, if\ r.u\ \geq 0 \\ 0, if\ r.u < 0 \end{cases} \quad (4)$$

The *k*-bit signature reduces the dimension of the original tweet feature vector. As it is a low dimensional vector, LSH approach clusters large number of tweet vectors very fast. Charikar applied cosine similarity metric to compute the similarity between two document vectors and is defined in equation 5.

$$\cos\big(\theta(u,v)\big) = \cos\big((1 - \Pr[h(u) = h(v)])\pi\big) \quad (5)$$

Here, $\theta(u,v)$ is the cosine angle between the vectors u and v and is proportional to the hamming distance of their signature vectors while preserving the cosine similarity in high dimensional space. $\Pr[h(u) = h(v)]$ is the probability that a random hyper plane separates two vectors, which is proportional to the cosine angle between them. The hamming distance is the number of bits that differ between two binary vectors.

5 Proposed Approach

In this section, we first describe our proposed approach for real-time Cricket event detection. We utilize locality sensitive hashing method for implementing the online incremental clustering of sports tweets. With the detected clusters, key events are recognized by leveraging our event lexicon. While game spectators would continue to tweet about the same event for a long time, our detector would alert it repeatedly, assuming a new event. Hence, we also elaborate on handling duplicate event alerts.

5.1 Event detection using LSH

Our event detection framework *SportsBuzzer*, strives to analyze cricket tweet streams to detect events, such as *boundary* and *sixer*, accurately and as early as possible. We utilize locality sensitive hashing method to cluster event related tweets at real time. Post rate of a cluster will be computed by considering the volume of tweets at a given time. If it is above a predefined threshold, an event is declared detected. We delete the cluster once an event is detected from it. The particular event will be recognized by analyzing the representative tweets from a cluster. An event that has occurred most number of times in

those selected tweets, is the recognized event. The proposed algorithm for LSH based event detection and recognition is depicted in figure 3.

Input: Live tweets, similarity threshold ST, buckets L, signature length K, post rate T
Output: Event *name* and its *tweets*

```
create event lexicon for pre-determined event types
build TF-IDF based dictionary D using lexicon
for each bucket i ∈ L do
    create hash table ht[i]
    create random vector rv using Gaussian distribution
end for
repeat
    for each incoming tweet t do
        construct tweet feature vector tv for t using D
        create k-bit signature ts for tv
        for each bucket i ∈ L do
            get collision for ts
            add tv with key ts in ht[i]
        end for
        get nearest neighbor NN for tv from collisions
        if similarity(tv, NN) < ST then
            create new cluster c
            addTweetVectorToCluster(tv, c)
        else
            if tv not in NN's cluster c_NN then
                addTweetVectorToCluster(tv, c_NN)
            end if
        end if
    end for
until connection closed

for each cluster c ∈ C do
    if postRate(c) > T then
        get text of all tweets in cluster c
        select an event with highest document frequency using lexicon
        display event name and its tweets using lexicon
        delete cluster c
    end if
end for
```

Function: addTweetVectorToCluster(*tv*, *c*)

1: *c*.tweetFrequency += 1
2: *c*.tweetVector += *tv*

Function: postRate(*c*)

1: sort timestamps of tweets based on *c*.tweetFrequency
2: cFirst ← *count*(tweets) in first half timestamps
3: cSecond ← *count*(tweets) in second half timestamps
4: **return** cSecond / cFirst

Figure 3. Proposed algorithm for LSH based key events detection

To improve the fidelity of the detector, we do not consider clusters that contain a single tweet. Similarly, we delete all clusters whose life span is more than five minutes, because we expect an event might occur within five minutes itself in Cricket sports. An important requirement for a real time event detection system is that it should detect and report events in near real time to the needy people. Our online incremental clustering approach clusters similar tweets together so as to detect and recognize key events quickly. To obtain the optimal post rate, similarity threshold, number of buckets and number of hash functions, we iterate our incremental clustering method with different parameter values. Because, the choice of these values impact the accuracy of the event detection. The evaluation results with different parameter setup will be explained in detail in section 6.

5.2 Lexicon-based event recognition

Once the post rate of a cluster is above the predefined threshold, *SportsBuzzer* assumes that some event has occurred in that cluster. The event recognizer then identifies the specific event in that cluster based on document frequency measure. The document frequency of each key event is computed in a case insensitive way, considering all tweets in the middle timestamp of the particular cluster. The event recognizer selects an event which has the highest document frequency and declare that event a winner. It should be noted that the document is characterized by the representative tweets of a cluster.

For an accurate event recognition, it is crucial that we design a domain specific lexicon describing all game terminologies for Cricket sports. Two important requirements should be considered while designing the lexicon. The event names should be more descriptive, as every game viewer tweets about the same event in different ways, using different words. Every game viewer just uses the event name to describe the happening of an event, because of the limited size of a tweet. Our event lexicon (a portion of which is shown in figure 4) describes 37 Cricket sports events, such as *bowled out*, *run-out*, *lbw* and *leg bye*. The lexicon is populated with different event terminologies collected from ESPNCricInfo (www.espncricinfo.com/ ci/content/story/ 239756.html) website.

```
BOUNDARY = ['boundary', 'four', 'fours', '4']
SIXER = ['sixer', 'six', '6']
CATCH = ['catch', 'c']
```

Figure 4. Lexicon for few events of *Cricket* sports

Our event lexicon is easy to implement and a better choice for a real time event detection from live tweets. Because, it does not require any training for the event recognition as required by other statistical event recognition models. Furthermore, there are real time applications which do not have training data such as celebrity deaths and terrorist attacks. Therefore, it is very practical to adopt lexicon based approach for event recognition.

5.3 Preventing duplicate event alerts

Duplicate event is an event that is repeatedly reported as a new event for a long time, even after the actual event has occurred a long back. The LSH algorithm groups tweets of an event such as boundary, into the same cluster until that event is detected (i.e., the post rate of that cluster is above the threshold). New cluster will not be created until a new tweet is sufficiently dissimilar from existing clusters. Thereby, all similar tweets will go into the same cluster and thus an event alert happens only once.

Nevertheless, there is still an issue with the duplicate event reporting. After an event is detected and reported for a cluster, the cluster will be deleted. Sometimes, because of the intense discussion of the current event among the audience of the game, a new cluster will be created once again for the same event based on the new set of tweets describing the same event. Hence, the event detector will report this duplicate event as a new event. We solve this problem by comparing the timestamp of an event that is already reported with the timestamp of the new cluster. If the time difference of these two timestamps is less than 60 seconds, then we ignore the new cluster and do not alert this event as a new event. Our assumption is that an event of the same type cannot happen once again within 60 seconds. Because, in cricket sports, an over containing six balls should be delivered within 5 to 6 minutes. So, the process of bowling and batting should be finished within an average time frame of 60 seconds.

6 Experimental Results

In this section, we will present the experimental results of the proposed Cricket key events detection approach. We evaluate the *SportsBuzzer* approach using tweets of IPL T20 2017 cricket sports. The proposed approach has been implemented in Python and Pika. The evaluation proves that LSH based event detection method can detect events with more than 90% true positives and less than 10% false positives. We will now present the data set, evaluation criteria and parameter setup and evaluation results.

6.1 Dataset

Twitter's Streaming API was used to crawl live tweets at real time using official hashtags of games provided by Indian Premier League (*www.iplt20.com*) in 2017 IPL T20 season held during April 2017 in India. Our dataset collection includes tweets of 44 games with a file size of over 6GB. Out of these games, we selected a game RCBvsRPS held on 16 April 2017 as it was considered an interesting and most anticipated match. Table 1 shows the description of RCBvRPS game.

Table 1. Game statistics of RCBvRPS

RCBvRPS game	Total	Total min	Mean (re)tweets per min	Min (re)tweets per min	Max (re)tweets per min	Standard deviation
Tweets	34967	232	150.72	38	354	67.4779909455
Retweets	16162	232	69.664	13	176	34.4786150528

Figure 5 depicts the tweet post rate of RCBvRPS game. It shows that the volume of tweets posted during the end of the game is high. Also, it contains several exciting moments throughout the game time.

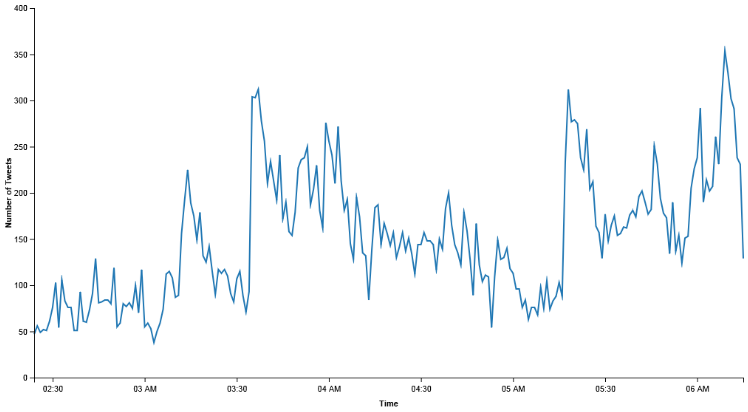


Figure 5. Post rate of tweets of game RCBvRPS

We have collected ground truth of all events from IPL live commentary site (*www.iplt20.com*). We have also cross-verified the time of each event with other live commentary websites. Table 2 shows the description of ground truth events for the RCBvRPS game.

Table 2. Summary of events in ground truth

Game	No. of ground truth events	No. of Boundaries	No. of Catches	No. of Sixers	Other events
RCBvRPS	81	24	6	9	42

6.2 Evaluation Criteria and Parameter Setup

Using all 24 boundaries, 9 sixes and 6 catches happened in the RCBvRPS game of 2017 IPL T20 cricket season, we illustrate the effectiveness of our LSH-based event detection method using Receiver Operating Characteristics (ROC) curves.

The results generated by our event detector are compared against the ground truth of RCBvRPS game. We define four evaluation windows with different times namely 1min, 5min, 10min and 15mins for comparison. Accordingly we compute the number of hits and misses for each evaluation window. A detection is considered a *hit* if the detected event is reported within a particular evaluation window, otherwise it is a *miss*.

Like any binary classifier, our detector can make two types of errors: reporting an event when nothing happens (i.e., false positive) and reporting nothing when an event happens (i.e., false negative). True Positive Rate and False Positive Rate (equation 6 and 7) are computed as follows:

$$TPR = \frac{TP}{TP+FN} \tag{6}$$

$$FPR = \frac{FP}{FP+TN} \tag{7}$$

For a particular study, different set of TPRs and FPRs are computed with various parameter settings. There are four parameters to be adjusted namely Post Rate (PR = 0.2, 0.5, 0.8), nearest neighbor Similarity Threshold (SIM = 0.2, 0.5, 0.8), number of hash tables (L = 5, 10, 15) and number of projections (K = 5, 11,

13, 19). For an experiment with a particular parameter setup with different Post Rates results in a set of TPRs and FPRs. The RoC curves are plotted using these rates and Area Under ROC curves (AUROC) are calculated. The AUROC curve represents the accuracy of the event detector. A high AUROC denotes a high true positive rate and low false positive rate while a low AUROC denotes a low true positive rate and high false positive rate

6.3 Results

We evaluate *SportsBuzzer* system using 2017 IPL T20 cricket games and present the accuracy of the event detection for key events such as *boundary*, *catch*, *sixer*, *boundary+catch+sixer, boundary+catch, boundary+sixer, catch+sixer*. We also show the influence of various parameters such as similarity threshold in finding nearest neighbor (SIM), number of hash tables (L) and number of projections (K), besides the effect of retweets in real time event detection. Finally, we compare the average computation times to find a nearest neighbor from buckets.

6.3.1 Performance on detecting events

Figure 6 shows the ROC curves that illustrate the performance of our LSH approach in detecting different Cricket events such as *boundary*, *catch*, *sixer*, and major events (*boundary+catch+sixer, boundary+catch, boundary+sixer, catch+sixer*) in the RCBvRPS game. The evaluation is conducted for different evaluation window sizes such as 1, 5, 10 and 15 minutes, with fixed parameter values SIM=0.5, L=10 and K=13.

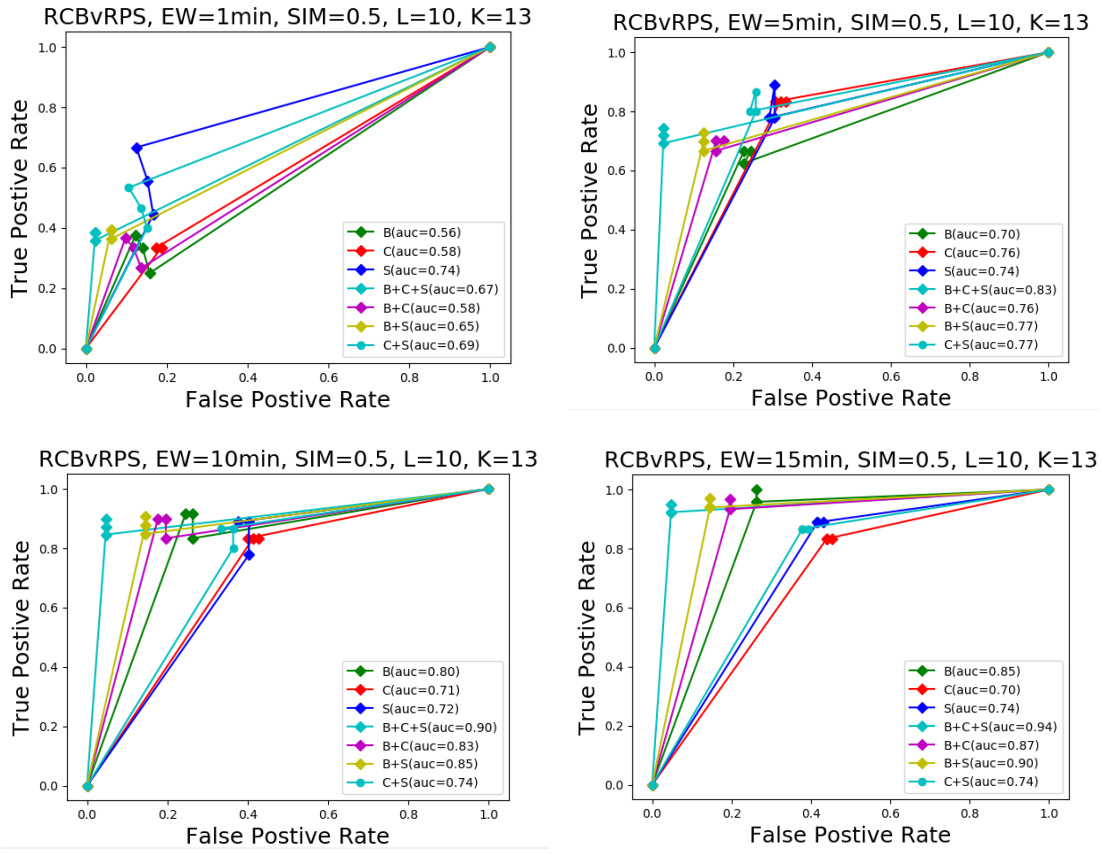


Figure 6. Detection performance of individual events

Results show that our LSH based approach delivers a decent performance in detecting the key game events. The event *sixer* is detected fast within 1 min evaluation window than other events, because sixer

is considered a highly exciting event by the viewers of this game. When size of the evaluation window increases, boundary is detected well, as twitter users reported this event with smaller delay. Due to a high initial excitement among twitter users, catch is detected well within 5 minutes and the performance decreases for 10 and 15 min windows. Major events (*boundary+catch+sixer* combination) of the game are also detected well with our LSH approach. Major events (*boundary+sixer* combination) performs better than other combinations. From these graphs, we can observe that almost all key events are detected with a decent accuracy (with 80 percent true positives in AUROC) within an evaluation window of 10 minutes. Also, the performance for evaluation window of 15 minutes is highly similar to that of 10 minutes window. So, we can conclude that most of the key events are detected and reported well even within 10 minutes from the actual happening of those events.

6.3.2 Influence of different similarity thresholds

Figure 7 shows the ROC curves that illustrate the performance of our LSH approach in detecting key events for different evaluation windows, under various similarity thresholds. The influence of various similarity thresholds (0.2, 0.5 and 0.8) on event detection is evaluated for key events and a combination of key events under a fixed L and different evaluation window size of 1, 5, 10 and 15mins.

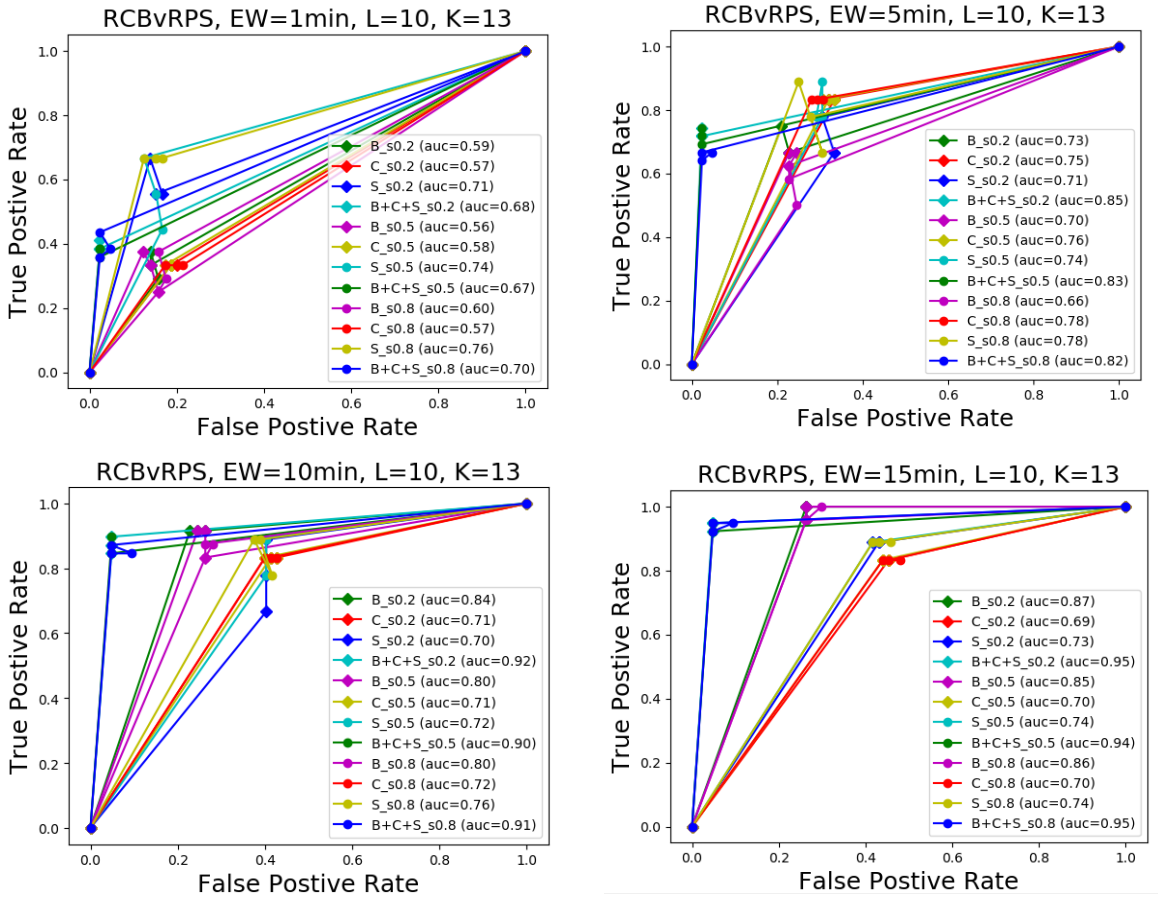


Figure 7. Detection performance for different similarity thresholds

From these graphs, we can observe that the overall performance is improved as the window size increases. Even though individual events present slightly different performances, the effect of similarity thresholds

can be assessed using a combination of major events such as B+C+S. When the evaluation window is small (i.e. 1 min), high similarity threshold is preferable, whereas for higher evaluation windows (like 10 or 15mins), lower similarity threshold is better. However, no significant difference can be noticed in the performances with any of these thresholds, because of the consistency in the representation of tweets using LSH based projections. Therefore, a reasonable threshold of 0.5 greatly strikes a balance for our approach in detecting all key events.

6.3.3 Influence of hash table size

Figure 8 shows the ROC curves that illustrate the performance of our LSH approach in detecting key events for different evaluation windows, under various hash table sizes. The influence of various hash table sizes (5, 10 and 15) on event detection is evaluated for key events and a combination of key events under a fixed similarity threshold and K and different evaluation window size of 1, 5, 10 and 15mins.

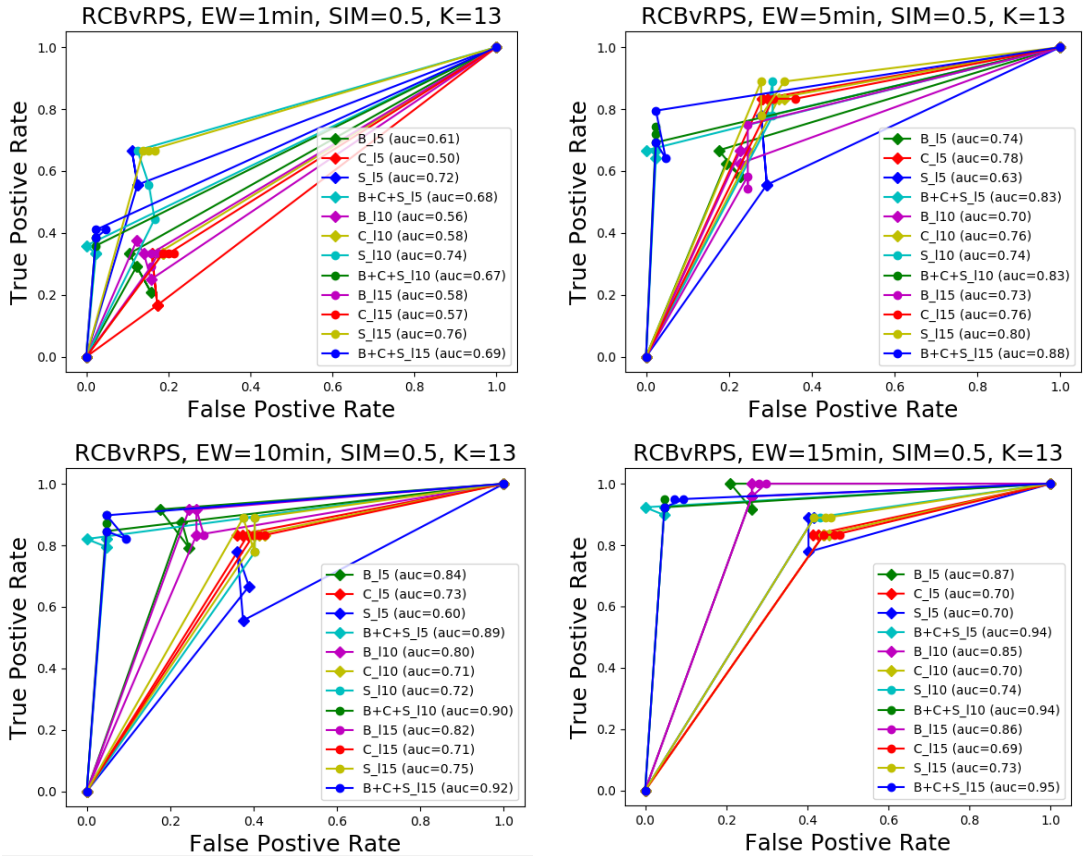


Figure 8. Detection performance for different sizes of hash tables

From the results shown in the ROC curves, it is evident that the accuracy of event detection improves in many cases when the number of hash tables increases, especially in major events such as B+C+S. For higher number of hash tables, the true positive rate reaches nearly 90% and the false positive rate is only 10%. Obviously, higher number of hash tables increases the chance of detecting the appropriate nearest neighbor. In a way that if one hash table misses out the correct nearest neighbor, other hash tables with different projections are more likely to consider it as a nearest neighbor. It is recommended that the number of hash tables should be selected carefully as it might increase the search time of the nearest neighbor. For a better tradeoff between speed and accuracy, it is preferable to keep L to a medium value.

6.3.4 Influence of different number of projections

Figure 9 shows the ROC curves that illustrate the performance of our LSH approach in detecting key events for different evaluation windows, under various projections. The influence of the number of projections (5, 11, 13 and 19) on event detection is evaluated for key events and major events under a fixed similarity threshold and L values and different evaluation window size of 1, 5, 10 and 15mins.

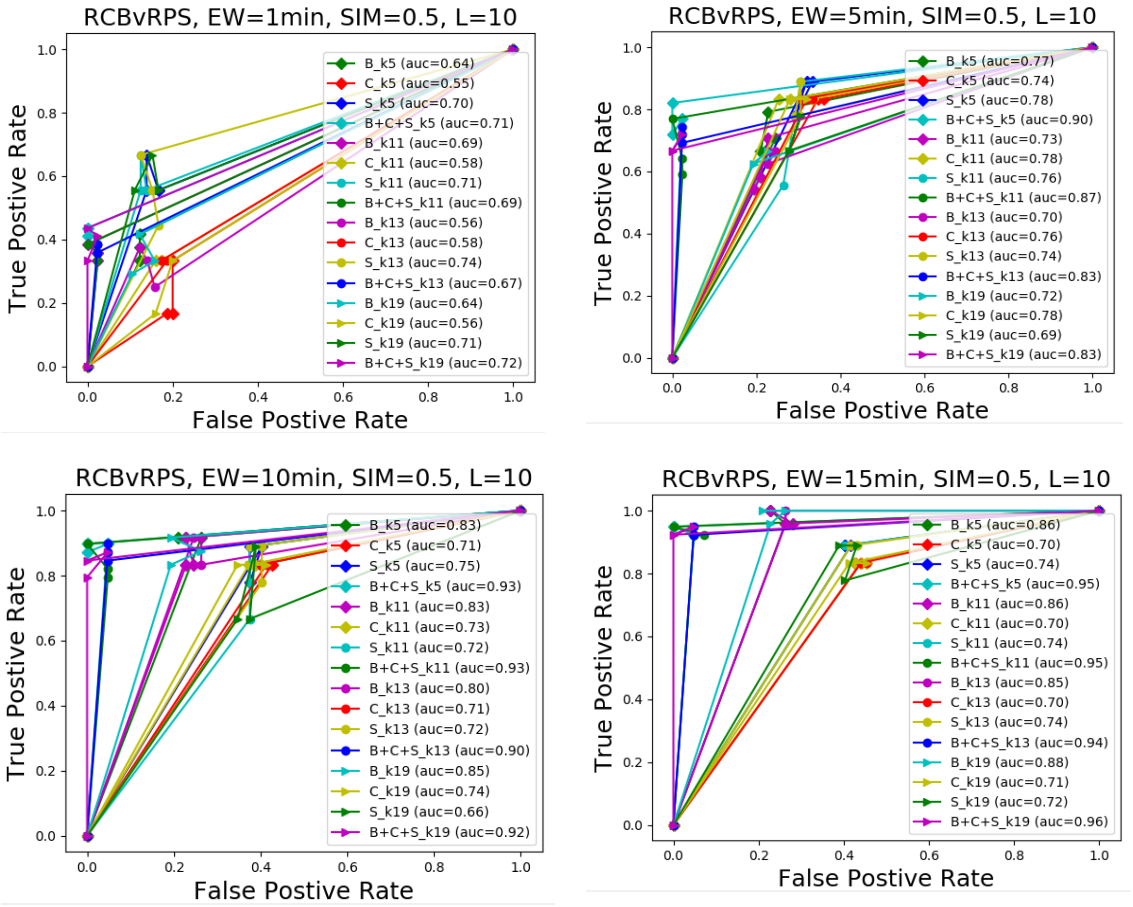


Figure 9. Detection performance for different number of projections

From the results shown in the ROC curves, our proposed approach achieves better accuracy for smaller number of projections, which can be noticed easily in the case of major events (B+C+S). The intuition is that when the number of projections are low, two relevant tweets might be indexed with the same signature. When the number of projections increases, location sensitivity of a tweet increases, thereby two relevant tweets might get different signatures. However, it is likely that the signature for both relevant and irrelevant tweet might be same because of lower number of projections, which results in both tweets falling into the same bucket. Similar to parameter L, number of projections is directly proportional to the search time of a nearest neighbor. Therefore, value for K should be chosen accordingly. Based on our experiments, we can conclude that K = 11 is a reasonable value to balance speed and accuracy.

6.3.5 Analysis on hash table size and number of projections

Figure 10 shows the ROC curves that illustrate the performance of our LSH approach in detecting key events for different evaluation windows, under three combinations of hash tables and projections. The influence of a number of hash tables (5, 10 and 15) and projections (5, 11 and 19) on event detection is evaluated under a fixed similarity threshold and different evaluation window size of 1, 5, 10 and 15mins.

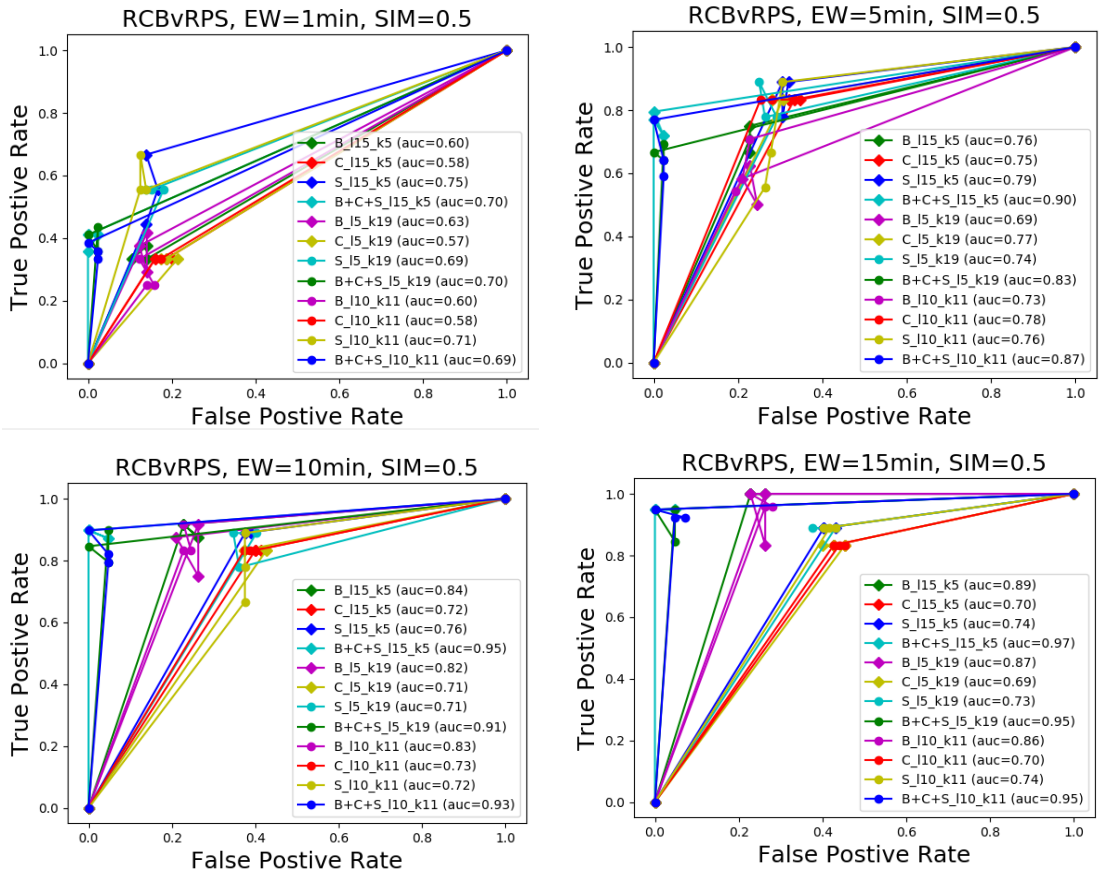
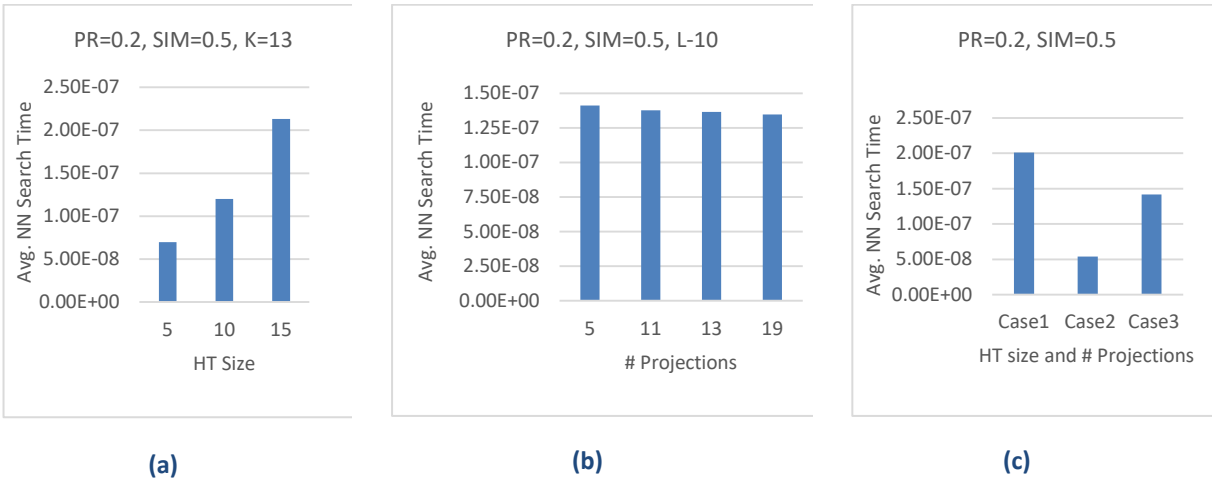


Figure 10. Detection performance for different number of hash tables and projections

From the results shown in the ROC curves, it is clear that true positive rate improves for larger evaluation windows, as seen in the previous sections. Our LSH based proposed approach achieves a better accuracy for higher number of hash tables (L=15) and smaller number of projections (K=5). As discussed in the previous sections, higher number of hash tables increases the chance of getting correct nearest neighbours, while lower number of projections gives the same signature to relevant and similar tweets. Similarly, medium number of hash tables (L=10) and projections (K=11) achieves the second best performance, whereas lower number of hash tables (L=5) and higher number of projections (K=19) decreases the accuracy. Therefore, a higher number of hash tables and lower number of projections is preferred.

We have also computed the average search time of finding nearest neighbours from 35000 tweets of the game RCBvRPS. The search time increases linearly when the number of hash tables (HT) increase (figure 11a). The number of projections (P) almost remains a constant (figure 11b). While considering both (HT and P), search time is high for case1 (HT=15, P=5), low for case2 (HT=5, P=19) and medium for case3

(HT=10, P=11) (figure 11c). We can see that the hash table size has more influence in deciding the speed of the nearest neighbor search.



(a) **(b)** **(c)**

Figure 11a, 11b & 11c. Nearest neighbor search time

6.3.6 Performance under different evaluation windows

Figure 12 shows the ROC curves that illustrate the performance of our LSH approach in detecting key events under different evaluation window of size 1, 5, 10 and 15mins. Since there is a significant delay between the actual event time and the time twitter users post tweets, the influence of an evaluation window greatly impacts the accuracy of our LSH approach.

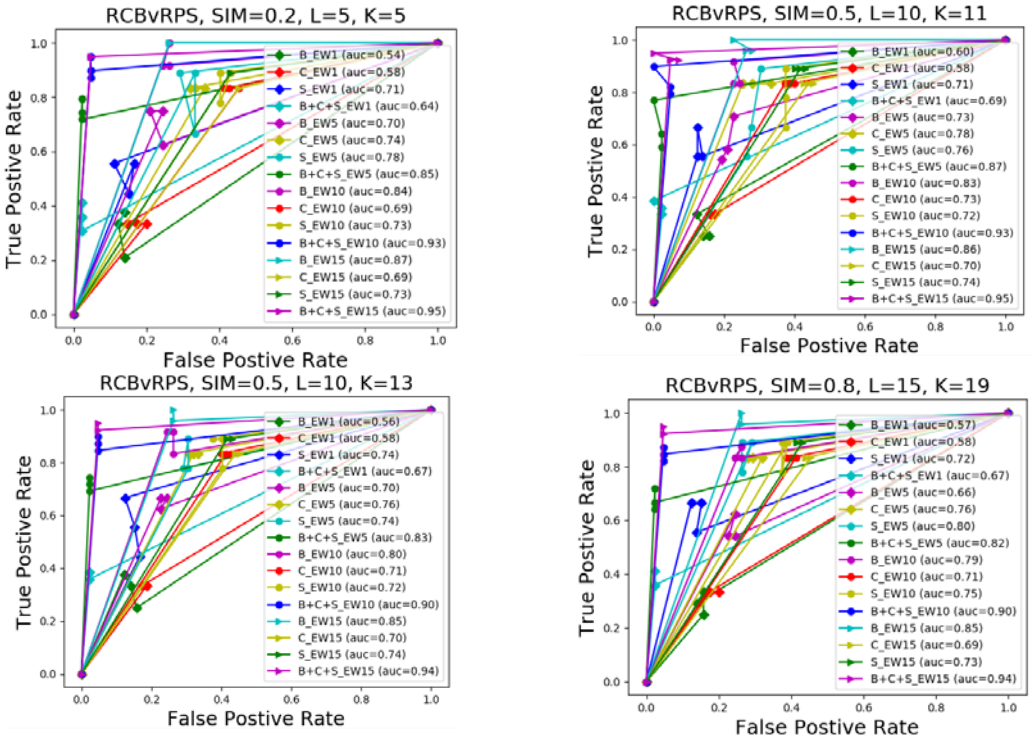


Figure 12. Detection performance for different evaluation windows

From the results shown in the ROC curves, our LSH approach detects most of the events within a window of 5 minutes, after the event has happened. Since a larger evaluation window allows some delay in detecting events, true positive rates reaches around 80% from nearly 60%. Hence, accuracy can be improved if considerable time delay in detection is permissible.

6.3.7 Performance of all tweets vs. no retweets

Figure 13 shows the ROC curves that illustrate the performance of our LSH approach in detecting key events with fixed SIM, L and K values. Since incoming tweets may also include their retweets, influence of retweets in detecting events is analyzed by evaluating the LSH approach with *all tweets* and with *no retweets*. The evaluation is conducted for different evaluation window of 1, 5, 10 and 15mins.

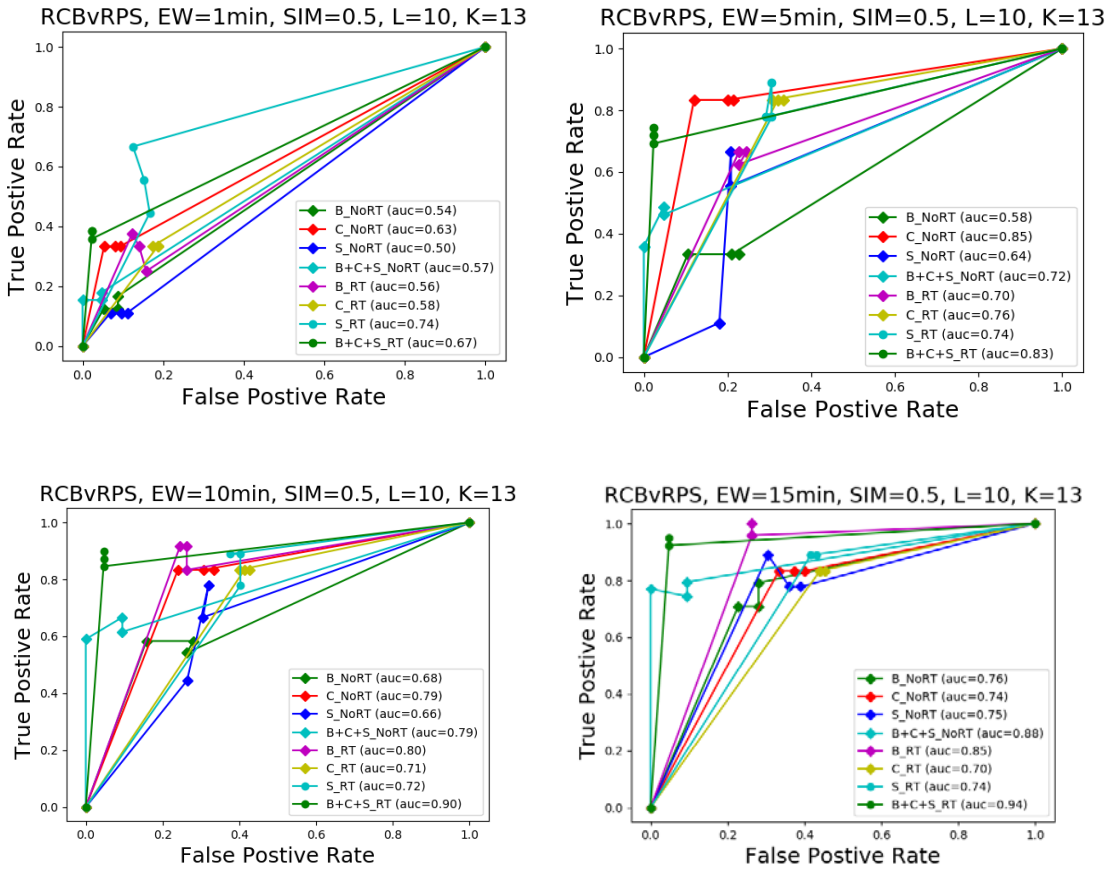


Figure 13. Detection performance for *all tweets* and *no retweets*

From the results shown in the ROC curves, it is apparent that the inclusion of retweets generally improves the accuracy of event detection, which we can observe easily based on the performance in major events detection (B+C+S). Because many twitter users have the habit of retweeting the tweet that is already posted by others. Also, they just want to approve that an event has actually happened, by way of retweeting. Therefore a large number of retweets creates a burst in the tweet volume which makes the event detection easy for the detector. Nevertheless, retweets do not help in detecting catch events. This may be due to a delayed retweeting of catch event long time after it has occurred. Whether retweets to be included or not for event detection can be sometimes decided based on the event we want to detect.

6.4 Limitations of SportsBuzzer

The performance of our real time event detection framework, *SportsBuzzer*, is impacted by various factors including latency or delay in the flow of signals from twitter users. There are three types of delays encountered in Twitter social media namely human delay, Twitter delay, and processing delay [3]. In this research work, we have addressed the issue of the processing delay. Processing delay occurs due to the processing time involved in data collection and analysis of a large volume of data. Our SportsBuzzer significantly reduces the processing time for the analysis of data by adopting LSH for implementing incremental clustering concepts. In our real time system that collects live tweets, data collection time is very minimal because of the recent fast processors.

7 Conclusion

In contrast to existing event detection approaches for sports domain, *SportsBuzzer*, a novel real-time event detection approach is presented in this paper. SportsBuzzer adopts LSH for discovering tweet clusters. A new cluster is created when an incoming live tweet is sufficiently dissimilar from existing clusters. An event is declared detected when the post rate of an active cluster exceeds the pre-defined threshold. Then, the event represented within the cluster is recognized utilizing our event lexicon for Cricket sports. Also a cluster is considered active if it contains at least two tweets. A cluster will be deleted once an event is detected from it or its life span is more than five minutes. We fix this time based on the assumption that at least one event would happen in one over which will last nearly five minutes.

Results of the extensive experiments demonstrated the efficacy of the LSH approach for event detection. As many twitter users take few minutes to post their tweets after an event has occurred, an evaluation window of five minutes was sufficient to detect most of the events. LSH effectively discovered tweet clusters with appropriate values for threshold, number of hash tables and number of projections. Influence of these parameters were also analyzed in the experiments. For a better tradeoff between speed and accuracy, a medium value for number of hash tables and number of projections is recommended.

In future, we will investigate whether we can improve the event detection by characterizing event lexicon as a dynamic lexicon instead of the present static lexicon. Similarly, we will study whether slope of the tweet rate curve can be exploited, instead of choosing the middle time for recognizing the detected event. Further, it would be interesting to explore other data structures that may speed up the clustering process, besides comparing our LSH approach with other widely popular clustering algorithms.

REFERENCES

[1]. Boyd, D. M and N. B. Ellison. Social network sites: Definition, history, and scholarship. Journal of Computer-Mediated Communication, 2007. 13(1): p. 210–230.

[2]. Atefeh, F and Khreich, W. A survey of techniques for event detection in twitter. Computational Intelligence, 2015. 31(1): p. 132-164.

[3]. Zhao, D and M. B. Rosson. How and why people Twitter: The role that micro-blogging plays in informal communication at work. In Proc. ACM International Conference on Supporting Group Work, GROUP '09, ACM, New York, NY, 2009. p. 243–252.

[4]. Zhao, S., Zhong, L., Wickramasuriya, J and Vasudevan, V. Human as real-time sensors of social and physical events: A case study of twitter and sports games. ArXiv preprint, 2011. arXiv:1106.4300.

[5]. P. Indyk and R. Motwani. Approximate nearest neighbors: Towards removing the curse of dimensionality. In Proc. Thirtieth Annual ACM Symposium on Theory of Computing, Dallas, Texas, USA, 1998. p. 604–613.

[6]. Hasan, M., Orgun, M. A and Schwitter, R. A survey on real-time event detection from the Twitter data stream. Journal of Information Science, 2017. 0165551517698564.

[7]. T. Sakaki., M. Okazaki and Y. Matsuo. Earthquake shakes Twitter users: real-time event detection by social sensors. In Proc. ACM WWW '10, 2010.

[8]. Y. Qu., C. Huang., P. Zhang and J. Zhang. Microblogging after a major disaster in China: a case study of the 2010 Yushu earthquake. In Proc. ACM 2011 conference on Computer supported cooperative work, 2011.

[9]. S. Vieweg., A. L. Hughes., K. Starbird and L. Palen. Microblogging during two natural hazards events: what twitter may contribute to situational awareness. In Proc. ACM CHI '10, 2010.

[10]. J. Sankaranarayanan., H. Samet., B. E. Teitler., M. D. Lieberman and J. Sperling. TwitterStand: news in tweets. In Proc. ACM SIGSPATIAL, 2009.

[11]. Sakaki, T., M. Okazaki and Y. Matsuo. Earthquake shakes Twitter users: Real-time event detection by social sensors. In Proc. 19th International Conference on World Wide Web, WWW '10, ACM, New York, NY, 2010. p. 851–860.

[12]. Popescu, A. M and M. Pennacchiotti. Detecting controversial events from Twitter. In Proc. 19th ACM International Conference on Information and Knowledge Management, CIKM '10, ACM, New York, NY, 2010. p. 1873–1876.

[13]. Benson, E., A. Haghighi and R. Barzilay. Event discovery in social media feeds. In Proc. 49th Annual Meeting of the Association for Computational Linguistics: Human Language Technologies, vol. 1, HLT '11, Association for Computational Linguistics, Stroudsburg, PA, 2011. p. 389–398.

[14]. Becker, H., F. Chen., D. Iter., M. Naaman and L. Gravano. Automatic identification and presentation of Twitter content for planned events. In Proc. International AAAI Conference on Weblogs and Social Media, Barcelona, Spain, 2011.

[15]. Becker, H., M. Naaman and L. Gravano. Selecting quality Twitter content for events. In Proc. International AAAI Conference on Weblogs and Social Media, Barcelona, Spain, 2011b.

[16]. Massoudi, K., M. Tsagkias, M. De Rijke and W. Weerkamp. Incorporating query expansion and quality indicators in searching microblog posts. In Proc. 33rd European Conference on Advances in Information Retrieval, ECIR'11. Springer-Verlag: Berlin, Heidelberg, 2011. p. 362–367.

[17]. Weerkamp, W and M. De Rijke. Credibility improves topical blog post retrieval. In Proc. ACL, Columbus, OH, 2008. p. 923–931.

[18]. Gu, H., X. Xie, Q. Lv, Y. Ruan and L. Shang. ETree: Effective and efficient event modeling for real-time online social media. In Proc. Web Intelligence and Intelligent Agent Technology, WI-IAT 2011, IEEE/WIC/ACM International Conference, 2011. 1: p. 300–307.

[19]. Valkanas, G and Gunopulos, D. How the Live Web Feels About Events. In Proc. In Proc. 22nd ACM International Conference on Information and Knowledge Management CIKM, 2013. p. 639–648.

[20]. Lee, R and K. Sumiya. Measuring geographical regularities of crowd behaviors for Twitter-based geo-social event detection. In Proc. 2nd ACM SIGSPATIAL International Workshop on Location Based Social Networks, LBSN '10, ACM, New York, NY, 2010. p. 1–10.

[21]. C. Li., A. Sun., and A. Datta. Twevent: Segment-based event detection from tweets. In Proc. ACM International Conference on Information and Knowledge Management, ser. CIKM '12. ACM, 2012. p. 155–164.

[22]. Marcus, A., Bernstein, M. S., Badar, O., Karger, D. R., Madden, S and Miller, R. C. TwitInfo: Aggregating and Visualizing Microblogs for Event Exploration. In Proc. CHI, 2011. p. 227–236.

[23]. Mathioudakis, M and Koudas, N. TwitterMonitor: Trend Detection over the Twitter Stream. In Proc. SIGMOD/ PODS, 2010. p. 1155–1158.

[24]. F. Alvanaki., M. Sebastian., K. Ramamritham and G. Weikum. Enblogue: Emergent topic detection in web 2.0 streams. In Proc. ACM SIGMOD, SIGMOD '11, New York, USA, 2011. p. 1271–1274.

[25]. Weng, J and Lee, B.-S. Event Detection in Twitter. In Proc. ICWSM, 2011. p. 401–408.

[26]. Shane Fitzpatrick. Improving new event detection in social streams. 2014. Master Thesis.

[27]. A. Ritter., Mausam., O. Etzioni and S. Clark. Open domain event extraction from Twitter. In Proc. 18th ACM SIGKDD, KDD '12, New York, USA, 2012. p. 1104–1112.

[28]. D. Zhou., L. Chen and Y. He. An unsupervised framework of exploring events on Twitter: Filtering, extraction and categorization. In Proc. AAAI Conference on Artificial Intelligence, 2015. p. 2468–2475.

[29]. J. Hannon., K. McCarthy., J. Lynch and B. Smyth. Personalized and automatic social summarization of events in video. In Proc. ACM IUI, 2011.

[30]. D. Chakrabarti and K. Punera. Event Summarization using Tweets. In Proc. AAAI ICWSM, 2011.

[31]. Becker, H., Naaman, M and Gravano, L. Beyond Trending Topics: Real-Wrold Event Identification on Twitter. In Proc. ICWSM, 2011. 11: p. 438–441.

[32]. A. J. McMinn and J. M. Jose. Real-time entity-based event detection for Twitter. In Proc. Experimental IR Meets Multilinguality, Multimodality, and Interaction. CLEF '15, Springer, 2015. p. 65–77.

[33]. Cataldi, M., Di Caro, L and Schifanella, C. Emerging Topic Detection on Twitter Based on Temporal and Social Terms Evaluation. In Proc. MDM/KDD, 2010. p. 4:1–10.

[34]. Petrovi´c, S., Osborne, M and Lavrenko, V. Streaming First Story Detection with Application to Twitter. In Proc. NAACL HLT, 2010. p. 181–189.

[35]. M. Hasan., M.A. Orgun and R. Schwitter. TwitterNews: real time event detection from the Twitter data stream. PeerJ PrePrints, 2016.

[36]. M. A. Russell. Mining the Social Web: Analyzing Data from Facebook, Twitter, LinkedIn, and Other Social Media Sites. O'Reilly Media Inc, 2011.

[37]. M. S. Charikar. Similarity estimation techniques from rounding algorithms. In Proc. 34th Annual ACM Symposium on Theory of Computing, Montreal, Quebec, Canada, 2002. p. 380-388.

SOCIETY FOR SCIENCE AND EDUCATION
UNITED KINGDOM

TMLAI TRANSACTIONS ON MACHINE LEARNING AND ARTIFICIAL INTELLIGENCE

Volume 6 No. 1
ISSN 2054-7390

Extracting Sentiments and Summarizing Health Reviews from Social Media Using Machine Learning Techniques

[1]Mozibur Raheman Khan , [2]Rajkumar Kannan
Department of Computer Science, Bishop Heber College (Autonomous), Tiruchirappalli, India.
[1]mozibmsc@gmail.com, [2]rajkumar@bhc.edu.in

ABSTRACT

Most of the health organizations provide an array of medical services and request their beneficiaries to provide their experience's in the form of opinion/reviews for which they are associated. Doctors of national and international repute have hundreds and even thousands of reviews authored by the health consumers around the globe. For an individual it is difficult and time consuming process to look all the reviews before taking an appropriate decision. Thus it is necessary to summarize the reviews to make an individual to take prompt decision. For a doctor it is also difficult to keep track of patient's reviews given by the patients in different time intervals, but he may have the summary of his entire patient's reviews to understand what is the best can be done to the patient's community. This research paper aims to mine and summarize the medical reviews authored by the health consumers. This article is performed by summarization of text in three steps, the first step is to identify the health features that have been commented by health consumers, the next one is to identify opinions of each review sentence and deciding whether each opinion sentence is positive or negative and finally summarizing the results.

Keywords—national and international repute, medical reviews, health consumer, summarization of text and health features

1 Introduction

The growth and expansion of internet, more and more services are provided on the Web, and more and more people are also deriving the benefits of the offered services. To provide better online services and to make prompt decision it has become a common practice for online health service provider to enable their health consumer to provide reviews or to express their opinions about the services they are enjoying. Reputed service organization gets large number of reviews from across the different region by different people, for an individual it is time consuming process to read the entire reviews.

In some cases reviews may be long and in some cases opinions are reflected for a particular feature. If a person wants to visit a hospital to access the health related services, he/she may not take an appropriate decision by reading few reviews. Even if he/she takes the decision, the decision may be biased. The suffering of public and in general health consumers due to poor medical facilities and less expertise of health consultant has increased their suffering by many folds. Hence it is advised to look the summary of the large reviews before making a final decision. Thus the need arises to collect the reviews that express

DOI: 10.14738/tmlai.61.3595
Publication Date: 16th, December 2017
URL: http://dx.doi.org/10.14738/tmlai.61.3595

their views for particular services they are willing to access or before willing to buy any product. Then these reviews are summarized to help the health consumers to approach the appropriate medical consultant. 'Opinions' mainly include opinionated text data such as blog/review articles, and associated numerical data like aspect rating is also included [1].some of the websites provides the very useful information related to health domain. Figure1 is screen shot taken from www.ratemds.com which provides numerical ratings as well as corresponding reviews of the different health experts.

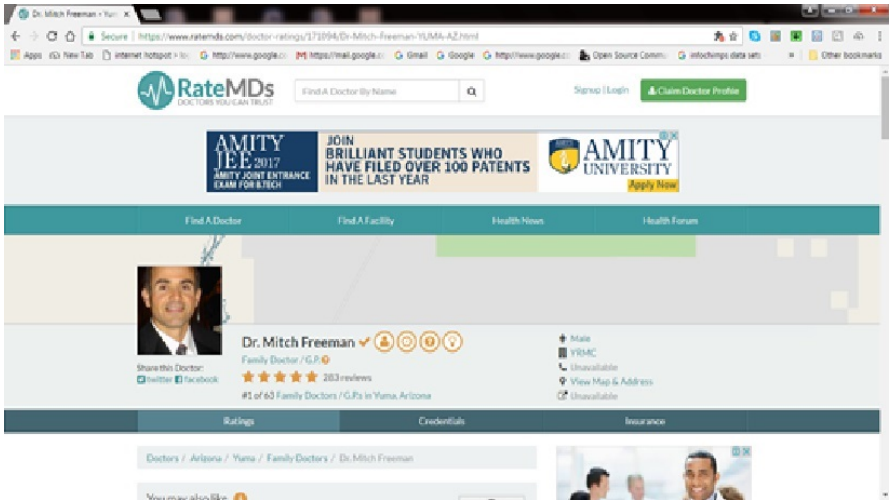


Figure 1: Screen shot with its reviews obtained from www.ratemds.com

The focus of this research is to study the problem of generating *feature-based summaries* of medical reviews given by different health consumers at various time intervals. Here, *features* broadly mean the doctor features (or attributes) and the attribute of its supporting teams members.

Given a set of medical reviews of a particular health consultant, the task involves three subtasks, the first step is identifying health features of the doctor that health consumers have expressed their opinions, the next for each feature, identifying review sentences that give positive or negative opinions; and finally producing a summary using the discovered information. Below is an example of feature-based summary. Consider the reviews of a particular doctor say, health expert. The summary looks like the following:

Summary of a Health Expert

Feature: staff
Positive:
Dr. Freeman's staff is very friendly and his knowledge alone is worth the extra money. I have always left with a good experience and highly recommend him to others looking for a doctor in YumaSubmitted Oct. 27, 2015
Negative:

Feature: Punctuality
Positive:
He is good
Negative:

Feature: Recommend
Positive: "Dope doctor! I recommend him. And i don't usually do this but i mean it. 👍"
Negative:

Feature: Knowledge
Positive:
Dr. Freeman's staff is very friendly and his knowledge alone is worth the extra money. I have always left with a good experience and highly recommend him to others looking for a doctor in YumaSubmitted Oct. 27, 2015
Best Dr. I've ever seen. He is Knowledgeable and cares.
Negative:
This Dr. Dose not except Medicare without charging 100$ a month extra

Figure 2: An example of summary

In Figure 2, staff and Punctuality are the doctor features. We have one medical review that express positive opinions about the staff, and knowledge and one review that express negative opinions for knowledge. With such a feature-based summary, one may understand the general opinion for a particular doctor. If he/she is very interested in a particular feature, he/she can drill down by following the individual review sentences to understand the level of satisfaction of health consumer or what may be the complaint. For a doctor of high repute/hospital of high repute may look the summary to understand what actually they are doing? And what supposed to be done, so that they can provide the services to suite the requirement of the health consumer.

Our task is different from traditional text summarization [9-11] in a number of ways. This health review summary in our case is *structured* rather than another (but shorter) free text document as produced by most text summarization systems. Second, we are only interested in features of the doctor that patients have opinions on and also whether the opinions are positive or negative. Traditional text summarization captures all the original text and important points but we follow the different techniques to summarize the health reviews.

As indicated above, our task is performed in three main steps; the first step is to capture health features that has been commented by patients. Data mining and natural language processing techniques are used to perform our task. This part of the study has been reported in [19].However, for completeness, we will summarize its techniques in this paper and also present a comparative evaluation.

The next step is to Select the reviews consists of opinion sentences and determine whether each opinion sentence is positive or negative. Note that these opinion sentences must contain one or more health

features identified above. *Opinion orientation* of each sentence is determined (whether the opinion expressed in the sentence is positive or negative), by performing three subtasks. First, a set of adjective words (which are normally used to express opinions) is identified using a natural language processing method. For the selected features we have corresponding opinion and these opinions are called as *opinion words.*

Summarizing the results. This step aggregates the results of previous steps and presents them in the format of Figure 2.Section 3 presents the detailed techniques for performing these tasks. A system, called Health Review *S*ummarization has also been implemented. Our experimental results with a large number of medical reviews of doctor available online show that health review summarization system (HRS) and its techniques are highly effectiveness.

Rest of the paper is organized as follows. In Section 2, related works are presented. In Section 3, feature Based Opinion Summarization approach is introduced. In Section 4, experiments and results are presented. In Section 5, the conclusion is presented.

2 Related Work

This work is related to on Mining and Summarizing Customer Reviews [2]. The system performs the summarization in three main steps (as discussed before),the first step is mining product features that have been commented on by customers, the next one is identifying opinion sentences in each review and deciding whether each opinion sentence is positive or negative, and finally summarizing the results. These steps are performed in multiple sub-steps.

Given the inputs, the system first downloads all the reviews, and put them in the review database. We then look very “hot” (or frequent) features that most of the people have expressed their opinions on. Collect the opinion words and determined its semantic orientations of the opinion words. Once opinion words are extracted, then the system then finds those infrequent features. In the last two steps, they found the orientation of each opinion sentence is identified and a final summary is produced. Note that POS tagging [28] is used from natural language processing, which helps us to find the features.

Another work is related to on semantic classification of reviews [3]. Using available training corpus from some Web sites, where each reviews already has a class (e.g., thumbs-up and thumbs-downs, or some other quantitative or binary ratings), they designed and sentiment classifier is built after experimenting a number of methods. They have shown that such classifiers perform quite well with test reviews and classifiers is used to classify sentences obtained from Web search results, which are obtained by a search engine using a product name as the search query.

Collecting individual sentences/opinion from the web searches, performance is limited due to noise and ambiguity. But in the context of a complete web-based tool and aided by a simple method for grouping sentences into attributes, the results are qualitatively quite useful.

The reputation of the target product by compare reviews of different products in one category is discussed in [4]. However, it does not summarize reviews, and it does not mine product features on which the reviewers have expressed their opinions. Although they do find some frequent phrases indicating reputations, these phrases may not be product features (e.g., “doesn’t work”, “benchmark result” and “no problem(s)”).Knowing the reputations of your own and/or competitors' products is important for marketing and customer relationship management. It is, however, very expensive to collect and analyze

survey data manually. This paper express the reputations of doctors globally on the Internet and health reviews are downloaded or can be crawled automatically those express health consumer opinions for the concerned expert or the health service provider.

In [5], the discussion on opinion-oriented information extraction. Their aim is to create summary representations of opinions to perform question answering.They propose to use of opinion-oriented "scenario templates" to act as summary representations of the opinions expressed in a document or a set of documents.

Our approach is different. We initially interested to identify the doctor features and user opinions on these features to automatically produce a summary. This work is also partially related but different from subjective genre classification, sentiment classification, text summarization and finding the terminology. It is discussed by each of them below.

2.1 Subjective Genre Classification

Genre classification classifies texts into different styles, e.g., "editorial", "novel", "news", "poem" etc. Although some techniques for genre classification can recognize documents that express opinions [6-8], they do not tell whether the opinions are positive or negative. In this work, we have to determine the opinion polarity and to perform opinion classification at the sentence level rather than at the document level.

A more closely related work is [12], in which the authors investigate sentence subjectivity classification and concludes that the presence and type of adjectives in a sentence is indicative of whether the sentence is subjective or objective. However, their work does not relate to our task of determining the semantic orientations of those subjective sentences. Even they neither find the features nor interested on which features opinions have been expressed.

2.2 Sentiment Classification

The phrase *sentiment analysis* is closely resembles with that of "opinion mining" in certain respects**.** The term "sentiment" is used in reference to the automatic analysis of evaluative text and tracking of the predictive judgments there in appears in 2001 papers [13, 14] because of these authors' interest in analyzing market sentiment. They use a manually crafted lexicon in conjunction with several scoring methods to classify stock postings on an investor bulletin. It subsequently occurred within 2002 papers [32,33] which were published in the proceedings of the annual meeting of the Association for Computational Linguistics (ACL) and the annual conference on Empirical Methods in Natural Language Processing (EMNLP).

Works of [15,16] on sentiment-based classification of entire documents use models inspired by cognitive linguistics. The author work in [43] also manually constructs a discriminant-word lexicon and use fuzzy logic to classify sentiments. Generates sentiment timelines is reported in [14]. The author captures online discussions about movies and displays a plot shown with the number of positive and negative sentiment messages over time. Messages are classified by looking for specific phrases that indicate the author's sentiment towards the movie (e.g., "great acting", "wonderful visuals", "uneven editing"). Each phrase are taken manually and added to a special lexicon and manually tagged as indicating positive or negative

sentiment. The lexicon is domain dependent (e.g., movies) and must be rebuilt for each new domain. In our work, this research is concerned with most frequent features and applied classification techniques.

Applies a specific unsupervised learning technique based on the mutual information between document phrases and the words "excellent" and "poor", where the mutual information is computed using statistics gathered by a search engine[17].Examine several supervised machine learning methods for sentiment classification of movie reviews and conclude that machine learning techniques performs well with other method that is based on human-tagged features although none of existing methods could handle the sentiment classification with a reasonable accuracy[18].

A sizeable number of papers mentioning "sentiment analysis" focus on the specific application of classifying reviews as to their polarity (either positive or negative), a fact that appears to have caused some authors to suggest that the phrase refers specifically to this narrowly defined task. However, the term "sentiment analysis" more broadly used by researchers to mean the computational analysis of opinion, sentiment, and subjectivity in text**.** "Sentiment analysis" and "opinion mining" denote the same field of study (which itself can be considered a sub-area of subjectivity analysis).They have attempted to use these terms more or less interchangeably in their work.

A novel approach is proposed based on latent semantic analysis (LSA) to identify product features [46]. Furthermore, they found a way to reduce the size of summary based on the product features obtained from LSA. They consider both sentiment-classification accuracy and system response time to design the system. This system can be extended to other product-review domains easily. This research paper is mainly based on movie reviews collected from Internet Blogs that do not consist of any rating information. Sentiment analysis is performed to determine the semantic orientation of the reviews and movie-rating score is based on the sentiment-analysis result. In addition to the accuracy of the classification, system response time is also taken into account in our system design. Although they have focused on movie review, the whole design is not only for movie-review domain. They have performed sentiment classification on movie review dataset, which is available[1] .The dataset includes 1000 positive and 1000 negative movie reviews. Similarly, SVM is used to perform the classification task. The kernel function used in the system is RBF and *K*-fold cross validation (i.e., $K = 5$) is used in the experiment.

2.3 Text Summarization

Summarization technique can be of two types. The first one Extractive Summary is a summary that represent by selecting representative text segments, usually sentences, from the original documents and another one is Abstractive Summary does not use the existing sentences in representing the summary rather it analyzes documents and directly generates sentences.

[1]http://www.cs.cornell.edu/People/pabo/movie-review-data

Because it is very tedious to produce readable and complete sentences, studies on extractive summary are more popular than that on abstractive summary. Extracting salient sentences from text and coherently organizing them to build a summary of the entire text is the key area of summarizing documents that focused on proposing paradigms**.** The relevant works in this regard includes [27, 28 and 47]. While traditional works focused on summarizing a single document, later, researchers shifted the idea on summarizing multiple documents originated from multiple sources.

The definition of a summary as a text that is generated from one or more texts, that conveys what the original text conveys, and that is lesser than that of the actual text(s) and must be less than that of [53]. This simple definition provides three aspects of automatic text summarization:

Single document or multiple documents can be summarized

Summaries should preserve the essence of the original text or paragraph

The length of the Summaries should be short.

Even if we agree unanimously on these points, it seems from the literature that any attempt to provide a more elaborate definition for the task would result in disagreement within the community. In fact, many approaches differ on the manner of their problem formulations. Some common terms introduced in the summarization dialect: extraction is the procedure of identifying important sections of the text and producing them verbatim; abstraction aims to produce important material in a new way; fusion makes an attempt to combines extracted parts coherently; and compression aims to throw out unimportant sections of the text [53].

The authors in [23, 24] emphasize on identification and extraction of certain core entities and facts in a document, which are packaged in a template. This framework requires background knowledge in order to instantiate a template to a suitable level of detail. Therefore, it is not domain or genre independent [25, 26]. This is different from our work as our techniques do not fill any template and are domain independent. The passage extraction framework [e.g., 27-29] identifies certain segments of the text (typically sentences) that are the most representative of the document's content. Our work is different in that we do not extract representative sentences, but identify and extract those specific product features 1 and the opinions related to them. An idea is proposed to find a few very prominent expressions, objects or events in a document and use them to help summarize the document proposed [31]. This work is again different as we find all health related features from a set of health consumer review regardless whether they are prominent or not. Thus, our summary is not a traditional text summary.

Lots of works have been done on text summarization focusing on a single document. Recently few of researchers also studied on summarization of multiple documents covering similar information. The authors in[27] have Summarized the similarities and differences in the information content is the focus of their work**.**Our work is related but quite different because we take interest to find the key features that are discussed by multiple reviews. Summarizing the similarities and differences of reviews is not the key focus.

Opinion summarization has different aspects from the classic text summarization problem because the nature and structure of the data**.** While summarizing the opinion, usually the polarities of input opinions are crucial. Sometimes, those reviews are provided with additional information such as rating scores. The formats of the summary is proposed by the most of the researcher of the opinion summarization are more structured in nature with the segmentation by topics and polarities. However, techniques of text summarization still can be useful in opinion summarization when text selection and generation step. Once separating input data by its polarities and topics, classic text summarization techniques can be used to find/generate the most representative text snippet from each category.

In health-review summarization, generally health consumer is more interested to know the expertise of a particular doctor in a specific field. To understand the expertise of a doctor, we must look what opinion the expert receives for a specific feature from the reliable/trusted sources. Hence it is necessary to understand the important features of doctor and the corresponding opinion. Thus, feature-based summarization is used in health-review summarization. The feature-based summarization will focus on the doctor features on which the patients or public have expressed their opinions. In addition to doctor features, the summarization should include opinion information about the doctor or concerned hospitals; therefore, doctor features and opinion words are both important in feature-based summarization. As a result, doctor's features and opinion-word Identification are essential in feature-based summarization.

In case of feature-based summarization we are very much interested to find out the aspects and these salient aspects is given as an input, which is also called as features and subtopics, and generates summaries of each feature. For example, for the summary of 'doctor', there can be aspects such as 'punctuality', 'knowledge', 'care', 'cost', etc. By further splitting the input texts into smaller units, aspect-based summarization can show more details in a structured way. Further splitting of feature can be even more useful when overall opinions are different from opinions of each aspect because aspect-based summary distribute the opinion of each aspect separately. The feature-based approaches are very popular and have been heavily explored over the last few years [44].

2.4 Summary Generation

Using the results of feature discovery and sentiment prediction, it is then critical to generate and present the final opinion summaries in an effective and easy to understand format. This typically involves aggregating the results of the first two steps and generating a concise summary. The following techniques describe various generation methods for opinion summarization**.** Each technique has its own advantages and disadvantages and some techniques can be combined with others. For example, we may add a timeline to text selection methods**.**

Statistical Summary. The most popular format and commonly adopted is a summary showing statistics introduced in [44].Statistical summary directly uses the processed results from the previous two steps - a list of aspects and results of sentiment prediction. All positive and negative opinions for each aspect can be displayed, so that the readers can easily understand the overall sentiments of users at large. Along with the positive and negative occurrences, all sentences with sentiment prediction in each aspect is shown (Figure 1).

The author has showed statistics in a graph format [49]. With the graph based representation, they collect people's overall opinions about the target more intuitively. Opinion observer is software developed by Liu et al. in 2005 clearly shows the statistics of opinion orientation in each aspect and it allows the users to compare opinion statistics of several products. An example result is shown in Figure 2, which gives the summary of different doctor.This format of summary has been widely adopted even in the commercial world.

Text Selection. While statistical summaries help users understand the overall idea of people's opinion, sometimes reading actual text is necessary to understand specifics. Due to the large volume of opinions on one topic, showing a complete list of sentences is not very useful.

Aggregated Ratings. Proposed the advanced summary is reported in [50], *aggregated ratings*, which combine statistical summary and text selection. Based on the discovered aspects using clustering and topic modeling, they average the sentiment prediction results of phrases for each aspect as the final sentiment rating for that aspect. Aspect ratings are shown with representative phrases.

Summary with a Timeline. Opinion trends over a timeline reflected in [51, 52]. General opinion summarization focuses on finding statistics of the 'current' data. In reality, opinions change as time goes by. Opinion summary with a timeline helps us to see the trend of opinions on the target easily, and it also tells ideas to further analysis. To figure out what changes people's opinions, we can analyze the events that happened at the drastic opinion change.

2.5 Terminology Finding

In terminology finding, there are basically two techniques for is discovering terms in corpora: symbolic approaches that rely on syntactic description of terms, namely noun phrases, and statistical approaches that exploit the fact that the words composing a term tend to be found close to each other and reoccurring [19-22]. However, using noun phrases tends to produce too many non-terms (low precision), while using reoccurring phrases misses many low frequency terms, terms with variations, and terms with only one word.

3 Feature Based Reviews Summarization

Figure3 provides the architectural overview of our health reviews summarization system. The inputs for the system are, a doctor's name and the salient features from the corresponding reviews. The output is the summary of the reviews as the one shown in the introduction section. The system performs the summarization in three main steps (as discussed before), the first step is Mining health features features that have been commented on by health consumers; the second is identifying opinion sentences in each review and deciding whether each opinion sentence is positive or negative, and finally Summarizing the results. These steps are performed in multiple sub-steps.

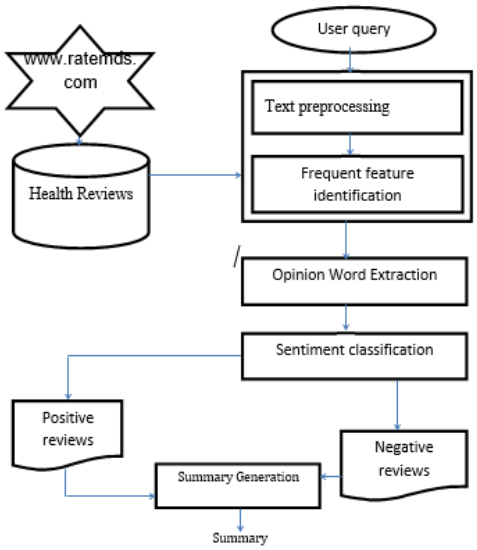


Figure 3: Architecture of Feature-based Reviews Summarization System

3.1 Data Collection and Preprocessing

We have collected a corpus of reviews from www.**ratemds.com** websites and these downloaded reviews were placed in the review database. As a preprocessing step the portions containing the reviews were extracted from html pages and these reviews were tokenized and separated into individual sentences. We then find those frequent features that many health consumers have expressed their opinions on. In the last two steps, the orientation of each opinion sentence is identified and a final summary is produced. Note that POS tagging is the part-of-speech tagging [28] from natural language processing, which helps us to find features and opinion. Below, we discuss each of the sub-steps in turn. Here is the review from the ratemds.com

3.2 Part-of-Speech Tagging (POS)

Identifying the interesting features from the health review are usually nouns or noun phrases. Thus the part-of-speech tagging is crucial. The process also identifies simple noun and verb groups (syntactic chunking). The following figure shows a sentence with POS tags shown in fIgure4.

<S> <NG><W C='PRP' L='SS' T='w' S='Y'> ***I*** </W> </NG>
<VG> <W C='VBP'> ***am*** </W><W C='RB'> ***absolutely***
</W></VG> <W C='IN'> ***in*** </W> <NG> <W C='NN'> ***awe***
</W> </NG> <W C='IN'> ***of*** </W> <NG> <W C='DT'> ***this***

Figure 4: POS tagging

3.3 Frequent Features Identification

Before identification of an individuals persons features on which many people and patients have expressed their opinions, lets us explain what patients like to provide their opinion for respective features before discussing frequent feature identifications. Here is an example of a medical review written by the patients

Great Doctor! For the one of the few times ever, I can say I felt like a doctor was genuinely interested in listening to me and trying to solve my problems!

Figure 5: An example of review authored by health consumer

This sentence expresses the satisfaction of user with the standard service which is provided by a medical consultant. Here the patient's talks about the doctor attribute such as listening and solving the patient's problems. Sometime some features are implicit and hard to find. For example *i like this office because I ve got good service and great Dr.*

Here, the health consumer is talking about the staff *of* the hospital and other feature, but the word staff does not appear in the sentence. In this work, we focus on finding features that appear explicitly as nouns or noun phrases in the reviews. Here, we find the frequent features, i.e., those features that are talked by many health consumers. In our context, an item set is simply a set of words or a phrase that occurs together in some sentences.

The main reason for them to use association mining is because of the following observation. It is common that a customer review contains many things that are not directly related to product features. Different customers usually have different stories. Thus using association mining to find frequent item sets, is

appropriate because those frequent item sets are likely to be the product features. When no one talks for a product or product feature, those noun/noun phrases are said to be infrequent are likely to be non-product features.

3.4 Opinion Words Extraction

We now identify opinion words and these words are primarily used to express subjective opinions. Clearly, this is related to existing work on distinguishing sentences used to express subjective opinions from sentences used to objectively describe some factual information [36]. Subjective and objective categories are potentially important for many text processing applications and Work on subjective opinion [37, 38] has established a positive significance correlation with the presence of adjectives. Thus the presence of adjectives is useful for predicting whether a sentence is subjective, i.e., expressing an opinion. This paper uses adjectives as opinion words. Opinion words extraction for those sentences that contain one or more health features, as we are only interested in health consumer's opinions on these health providers**.** Let us first define an opinion sentence.

Definition: *opinion sentence*
If a sentence contains one or more product features and one or more opinion words, then the sentence is called an *opinion sentence.* We extract opinion words in the following manner (Figure 6)

for each sentence in the review database
if (it contains a frequent feature, extract all the adjective Words as opinion words)
nearby adjective is recorded as its effective opinion for each feature in the sentence .
for each feature in the sentence
the nearby adjective is recorded as its *effective opinion.*
/ A nearby adjective refers to the adjacent adjective that modifies the noun/noun phrase that is a frequent feature. */*

Figure 6: Opinion word extraction

3.5 Infrequent Feature Identification

Finding Frequent features are very easy that people normally exchange their comment for given entity. However, there are some features that only a small number of people talked about. These features can also be interesting to some patients/persons willing to derive health benefits and also to the service providers. The question is how to extract these infrequent features (association mining is unable to identify such features)? Considering the following sentences**:**

"The facility of the hospital is good."

"The location of the hospital is good."

for each sentence in the review database
*if (it has no frequent feature but one or more opinion words**)***
{
Find the nearest noun/noun phrase around the opinion word. These nearest
Noun/noun phrases is said to be infrequent feature.
}

Figure 7: Infrequent feature extraction

Most of the time the nearest noun/noun phrase modifies opinion word**.** This simple heuristic seems to work well in practice. A simple problem exit with the infrequent feature identification using opinion words

is that it could find some feature that are irrelevant**.** There is the reason to use common adjectives to describe a lot of objects, including both interesting features that we want and irrelevant ones**.** This is not a serious problem because the number of infrequent features, compared with the number of frequent features, is small**.** They account for around 15-20% of the total number of features as obtained in our experimental results. Infrequent features are generated for completeness. Frequent features are more important than infrequent ones because we need to display the summary of the frequent feature first and then low ranked feature and thus will not affect most of the users.

3.6 Sentiment Classification

Sentiment classification is similar to traditional binary-classification problem. There are many classification techniques are exit for different domains. We used three classification techniques namely Logistic Regression (LR), Support Vector Machine (SVM) and Gaussian Naive Bayes (GNV). Logistic Regression widely used in disciplines ranging from credit and finance to medicine to criminology and other social sciences. Logistic regression is considered to be very effective.

The second one is SVM is a supervised machine learning algorithm which works well with exiting text categorization[46].The goal of this machine learning algorithm is to find a decision boundary between two classes that is maximally far from any point in the training data. There is one interesting property of SVM is that their ability to learn can be independent of the dimensionality of the feature space. The third one we have used is Gaussian Naive Bayes classifiers. All these techniques are not involved in finding the features that is commented by different user.

Figure 8: Summarization screen shot

Figure 9: Screen shots with reviews

Figure 8 shows the empty screen receives an input that is a doctor name and the corresponding features then rating is calculated and Figure 9 explains rating and summarizing the particular doctor reviews and we can read all the reviews.

It is easier to some type of probability models that naive Bayes classifiers can be trained very efficiently in a supervised learning setting. In many practical applications, parameter estimation for naive Bayes models uses the method of maximum likelihood; in other words, one can work with the naive Bayes model without accepting Bayesian probability or using any Bayesian methods. With small number of training data we can estimate the parameters necessary for classification.

3.7 Summary Generation

Now we are ready to generate the final feature-based review summary, which is straightforward and consists of the following steps:

• For each discovered feature, related opinion sentences are put into positive and negative categories according to the opinion sentences' orientations.

• Ranking of all features is done according to the frequency of their appearances in the reviews. Feature phrases appear before single word features as phrases normally are more interesting to users. Different types of rankings are also possible. For an example, we can choose the rank of features based on the number of reviews that express positive or negative opinions.

The following shows an example summary for the feature "Recommend" of a doctor. It is not necessary that the individual opinion sentences (and their corresponding reviews, which are not shown here) can be hidden using a hyperlink to enable the user to have a quick look of global view of the summary.

Dr. Gretchen
Feature Recommend
Rating *

Positive:
<We highly recommend him>
<Dr. Liddell is wonderful and I recommend him highly to my friends and family>
<She even remembers past conservations we've had! Appointments are readily available but am sure once word gets out how good she is, it will get harder! Highly recommend Dr. Bortolotti.>
<I finally found my Doctor! Took 20 years!!!!She never rushes you out of her office, and if you call to speak to her, SHE calls you back. (instead of a nurse) I would highly recommend!!I highly recommend her. I highly recommend her to everyone.>
….
Negative :

<The only complain is long wait to see her.>
<With that said I highly recommend her…She doesn't just go "by the book.>
<" I highly recommend her!I really like him and his staff, but have had some trouble with getting prescriptions filled in a timely manner, which I found frustrating, but was only an issue because I was in and out of town (and may have been>
…

Figure 10: Review summarization of a health service provider

4 Experiments and results

4.1 Data Sets

We have collected health reviews of one fifty doctor from ratemds.com and these collected reviews have been placed in reviews database. This site provides hundreds of reviews for thousands of doctor from across the globe. Each of the reviews includes a text review and other numeric ratings are available for various other features. We have received all these from family doctor/GP. The site provides numerical rating of four aspects namely staff, punctuality, helpfulness and knowledge. Textual comments are written by the health consumers with an average of three sentences. For each doctor, we first downloaded the first available reviews. Looking at the sites nearly we can understand that there are ten important specialty available. They are Internist, Gynecologist,Family/general, peddiatrist ,Dentist,Psychiatrist, Orthopedist,Cardiologist, Gastroenterologist, Dermatologist and so on. For each specialty there are top reputed doctors are available and each doctor is receiving hundreds of reviews. For hundred and fifty doctor, we have collected 1745 reviews and these reviews are summarized.

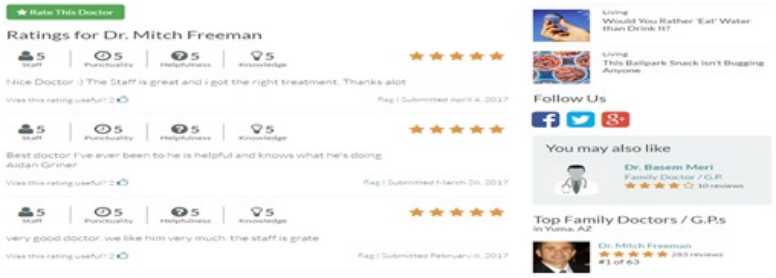


Figure 11: Screen shot of health review for a doctor Mitch Freeman

This proposed technique has been implemented in python and we now evaluate our sentiment's extraction and health reviews summarization system from the classification perspectives. We have a corpus of 1745 reviews and these review documents were then cleaned to remove HTML tags. After that, NLP preprocessing techniques is used to generate part of-speech tags. Our system is then applied to perform summarization.

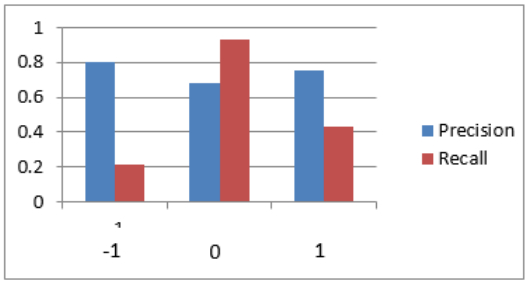
We must identify the orientation of the opinion, is positive or negative. If the user gives no opinion in a sentence, the sentence is not tagged as we are only interested in sentences with opinions in this work. There is a small complication in feature tagging is that features can be explicit or implicit in a sentence. Most features appear explicitly in opinion sentences, e.g., *punctuality in "The wait is a little long but well worth the time spent".* Some features may not appear in sentences. We call such features implicit features, e.g., *punctuality in "The doctor manages his appointment time properly".* Both explicit and implicit features are easy to identify by the human tagger.

Another issue is that judging for evaluation, we manually read all the reviews. For each sentence in a review, if it shows user's opinions, all the features on which the reviewer has expressed his/her opinion are tagged and the opinions in reviews can be somewhat subjective. It is not difficult to judge the opinion is whether it is positive or negative, even in a sentence which expresses its opinion clearly. However, deciding whether a sentence offers an opinion or not can be debatable. For some extreme cases, we reached a consensus between the primary human tagger (the first author of the paper) and the secondary tagger (the second author of the paper).

4.2 Results and Discussion

Bar chart provides the precision and recall results of the feature generation function of Feature Based Summarization. We evaluated the results at each step of our algorithm. The figure10, 11 and 12 gives the recall and precision of frequent feature generation for each doctor using three different classifier. The results indicate that the frequent features contain a lot of errors especially in Gaussian Naïve Bayes, i.e., low precision and moderate recall.

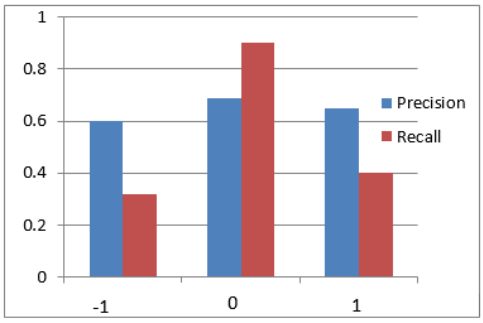
The results can be improved by applying SVM which shows better results than previous one. We can see that the precision is improved marginally and recall is improved drastically. There is another dramatic improvement in the precision by applying logistic regression techniques. The recall level almost does not change comparing to the previous step.



-1 stands for negative sentiments
0 stands for neutral sentiments
1 stands for positive sentiments

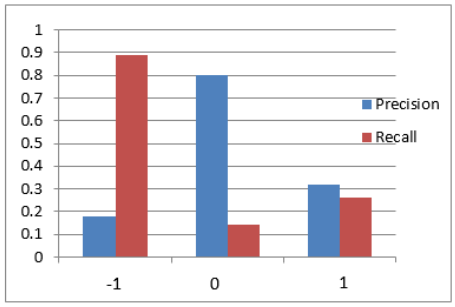
Sentiments

Figure 12: Precision and Recall using LR for Medical reviews data sets



Sentiments

Figure 13: Precision and Recall using SVM for medical reviews data sets



Sentiments

Figure 14: Precision and Recall using GNV for Medical Reviews data sets

5 Conclusion

In this paper, we proposed a set of techniques for extracting sentiments and summarizing the health reviews based on data mining and natural language processing methods. The objective is to provide a feature-based summary of a large number of health reviews of various doctors available online. Our experimental results indicate that the proposed techniques are very promising in performing these tasks. We sincerely believe that this problem will become increasingly important as more people are expressed their opinions on the Web. Summarizing of all will be useful to health consumers and also crucial to health service providers.

In future, we can extend this work to provide the aggregated summary of the information provided with large number of reviews available for given, respective provider. We plan to further improve and refine our techniques, and to deal with the outstanding problems such as the intensity of opinions, opinion changes over a period and investigating opinions expressed with adverbs, verbs and nouns.

REFERENCES

[1]. Kim, Hyun Duk, et al. "Comprehensive review of opinion summarization." (2011).

[2]. Minqing Hu and Bing Liu, Mining and Summarizing Customer Reviews.

[3]. Dave, K., Lawrence, S., and Pennock, D., 2003. Mining the Peanut Gallery: Opinion Extraction and Semantic Classification of Product Reviews. *WWW'03*.

[4]. Morinaga, S., Ya Yamanishi, K., Tateishi, K, andFukushima, T. 2002. Mining Product Reputations on the Web.*KDD'02*. [18].

[5]. Cardie, C., Wiebe, J., Wilson, T. and Litman, D. 2003.Combining Low-Level and Summary Representations of Opinions for Multi-Perspective Question Answering. *2003 AAAI Spring Symposium on New Directions in Question Answering.*

[6]. Karlgren, J. and Cutting, D. 1994. Recognizing Text Genres with Simple Metrics using Discriminant Analysis.*COLING'94*.

[7]. Kessler, B., Nunberg, G., and Schutze, H. 1997. Automatic Detection of Text Genre. In *Proc. of 35th ACL/8th EACL*.

[8]. Finn, A., Kushmerick, N., and Smyth, B. 2002. Genre Classification and Domain Transfer for Information Filtering. In *Proc. of European Colloquium on Information Retrieval Research*, pages 353-362.

[9]. Goldstein, J., Kantrowitz, M., Mittal, V., and Carbonell, J.1999. Summarizing Text Documents: Sentence Selection and Evaluation Metrics. *SIGIR'99*.

[10]. Salton, G. Singhal, A. Buckley, C. and Mitra, M. 1996.Automatic Text Decomposition using Text Segments and Text Themes*. ACM Conference on Hypertext.*

[11]. Tait, J. 1983. *Automatic Summarizing of English Texts.* Ph.D. Dissertation, University of Cambridge

[12]. Hatzivassiloglou, V. and Wiebe, 2000. J. Effects of Adjective Orientation and Gradability on Sentence Subjectivity. *COLING'00*.

[13]. S. Das and M. Chen, "Yahoo! for Amazon: Extracting market sentiment from stock message boards," in *Proceedings of the Asia Pacific Finance Association Annual Conference (APFA)*, 2001.

[14]. Tong, R., 2001. An Operational System for Detecting and Tracking Opinions in on-line discussion. *SIGIR 2001 Workshop on Operational Text Classification*.

[15]. Hearst, M, 1992. Direction-based Text Interpretation as an Information Access Refinement. In Paul Jacobs, editor,*Text-Based Intelligent Systems*. Lawrence Erlbaum Associates.

[16]. Sack, W., 1994. On the Computation of Point of View.*AAAI'94*, Student abstract.

[17]. Turney, P. 2002. Thumbs Up or Thumbs Down? Semantic Orientation Applied to Unsupervised Classification of Reviews. *ACL'02*.

[18]. Pang, B., Lee, L., and Vaithyanathan, S., 2002. Thumbs up? Sentiment Classification Using Machine Learning Techniques. In *Proc. of EMNLP 2002*

[19]. Jacquemin, C., and Bourigault, D. 2001. Term extraction and automatic indexing. In R. Mitkov, editor, *Handbook of Computational Linguistics*. Oxford University Press.

[20]. Justeson, J. S., and Katz, S.M. 1995. Technical Terminology: some linguistic properties and an algorithm for identification in text. *Natural Language engineering*1(1):9-27.

[21]. Daille, B. 1996. Study and Implementation of Combined Techniques for Automatic Extraction of Terminology. *The Balancing Act: Combining Symbolic and Statistical Approaches to Language*. MIT Press, Cambridge

[22]. Church, K.W. and Hanks, P. 1990. Word Association Norms, Mutual Information and Lexicography *Computational Linguistics*, 16(1):22-29

[23]. DeJong, G. 1982. An Overview of the FRUMP System. *Strategies for Natural Language Parsing*. 149-176

[24]. Tait, J. 1983. *Automatic Summarizing of English Texts.* Ph.D. Dissertation, University of Cambridge

[25]. Sparck J. 1993a. Discourse Modeling for Automatic Text Summarizing. *Technical Report 290,* University of Cambridge Computer Laboratory

[26]. Sparck J. 1993b.What might be in a summary? *Information Retrieval* 93:9-26.

[27]. Paice, C. D. 1990. Constructing Literature Abstracts by Computer: Techniques and Prospects. *Information Processing and Management* 26:171-186.

[28]. Kupiec, J., Pedersen, J., and Chen, F. 1995. A Trainable Document summarizer. *SIGIR'199*

[29]. Salton, G. Singhal, A. Buckley, C. and Mitra, M. 1996.Automatic Text Decomposition using Text Segments and Text Themes. *ACM Conference on Hypertext*

[30]. Mani, I., and Bloedorn, E., 1997. Multi-document Summarization by Graph Search and Matching. *AAAI'97*.

[31]. Boguraev, B., and Kennedy, C. 1997. Salience-Based Content characterization of Text Documents. In *Proc. Of the ACL'97/EACL'97 Workshop on Intelligent Scalable Text Summarization.*

[32]. P. Turney, "Thumbs up or thumbs down? Semantic orientation applied to unsupervised classification of reviews," in *Proceedings of the Association for Computational Linguistics (ACL)*, pp. 417–424, 2002.

[33]. B. Pang, L. Lee, and S. Vaithyanathan, "Thumbs up? Sentiment classification using machine learning techniques," in *Proceedings of the Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pp. 79–86,2002.

[34]. Wiebe, J. 2000. Learning Subjective Adjectives from Corpora. *AAAI'00*.

[35]. Wiebe, J., Bruce, R., and O'Hara, T. 1999. Development and Use of a Gold Standard Data Set for Subjectivity Classifications. In *Proc. of ACL'99*.

[36]. Bruce, R., and Wiebe, J. 2000. Recognizing Subjectivity: A Case Study of Manual Tagging. *Natural Language Engineering*.

[37]. Miller, G., Beckwith, R, Fellbaum, C., Gross, D., and Miller, K. 1990. Introduction to WordNet: An on-line lexical database. *International Journal of Lexicography (special issue)*, 3(4):235-312.

[38]. Hatzivassiloglou, V. and Mckeown, K., 1997. Predicting the Semantic Orientation of Adjectives. In *Proc. of 35th ACL/8th EACL*.[41]. Fellbaum, C. 1998. *WordNet: an Electronic Lexical Database*, MIT Press.

[39]. NLProcessor–*Text AnalysisToolkit*.2000. http://www.infogistics.com/textanalysis.html

[40]. Huettner, A. and Subasic, P., 2000. Fuzzy Typing for Document management. In *ACL'00 Companion Volume:Tutorial Abstracts and Demonstration Notes*.

[41]. Manning, C. and Schütze, H. 1999. *Foundations of Statistical Natural Language Processing*, MIT Press.Cambridge, MA: May 1999.

[42]. Hu, M., and Liu, B. 2004. Mining Opinion Features in Customer Reviews. To appear in *AAAI'04*, 2004.

[43]. Liu, B., Hsu, W., Ma, Y. 1998. Integrating Classification and Association Rule Mining. *KDD'98*, 1998.

[44]. Chien-Liang Liu, Wen-Hoar Hsaio, Chia-Hoang Lee, Gen-Chi Lu, and Emery Jou Movie Rating and Review Summarization in Mobile Environment IEEE transactions on systems, man, and cybernetics—part c: applications and reviews, vol. 42, no. 3, may 2012 397

[45]. Radev, D. R., Hovy, E., and McKeown, K. (2002). Introduction to the special issue on summarization. Computational Linguistics., 28(4):399{408. [1, 2]

[46]. BALAHUR, A. AND MONTOYO, A. 2008. Multilingual feature-driven opinion extraction and summarization from customer reviews. In *NLDB '08: Proceedings of the 13th international conference on Natural Language and Information Systems*. Springer-Verlag, Berlin,Heidelberg, 345–346.

[47]. HU, M. AND LIU, B. 2006. Opinion extraction and summarization on the web. In *AAAI'06: proceedings of the 21st national conference on Artificial intelligence*. AAAI Press, 1621–1624.

[48]. LU, Y., ZHAI, C., AND SUNDARESAN, N. 2009. Rated aspect summarization of short comments. In *WWW '09:Proceedings of the 18th international conference on World wide web*. ACM, New York, NY, USA, 131–140.

[49]. KU, L.-W., LIANG, Y.-T., AND CHEN, H.-H. 2006. Opinion extraction, summarization and tracking in news and blog corpora. In *AAAI Symposium on Computational Approaches to Analysing Weblogs (AAAI-CAAW)*.100–107.

[50]. MEI, Q., LING, X., WONDRA, M., SU, H., AND ZHAI, C. 2007. Topic sentiment mixture: modeling facets and opinions in weblogs. In *WWW '07:* Proceedings *of the 16th international conference on World Wide Web*.ACM, New York, NY, USA, 171–180.

[51]. Radev, D. R., Hovy, E., and McKeown, K. (2002). Introduction to the special issue on summarization. Computational Linguistics. 28(4):399{408. [1, 2]

SOCIETY FOR SCIENCE AND EDUCATION
UNITED KINGDOM

TMLAI TRANSACTIONS ON MACHINE LEARNING AND ARTIFICIAL INTELLIGENCE

Volume 6 No. 1
ISSN 2054-7390

Temperature, Precipitation and Relative Humidity Fluctuation of Makkah Al Mukarramah, Kingdom of Saudi Arabia (1985-2016)

Saifullah Khan[1], Yasser Alghafari[2]
[1]*Institute of Social Sciences Bahauddin Zakariya University, Multan, Pakistan*
[2]*Presidency of Meteorology and Environment (PME), Riyadh, Saudi Arabia*
saifullahkhan33@gmail.com; ayasser@pme.gov.sa

ABSTRACT

The study presents the temperature, rainfall and relative humidity fluctuation of Makkah Al Mukarramah, Saudi Arabia for a time period of 1985-2016 in terms of general climatology, climate change, seasonal pattern and extreme weather condition. This is a city in the Tihamah plain of western Saudi Arabia, the capital of Makkah Province, birth place of Prophet Muhammad (PBUH) and holistic place in Islam. The factors that control the climate are surface water, coastal management, overgrazing, forestry, livestock, drought, desertification, industrialization, landuse change, tourism, altitude, location and marine influence etc. The mean monthly temperature of the city is 32˚C having mean maximum of 38˚C and mean minimum of 25 degree Celsius. The mean monthly temperature of the city shows an increase of one degree Celsius having -0.1˚C decrease in maximum and -1.1˚C in minimum temperature. Generally, there is a rise and fall in the temperature condition and shows periodic pattern after each ten years throughout the period. The total precipitation of the city is 189 millimeters (7.4inches); having an increase of 36millimeters (1.4inches) and shows an increasing trend. The relative humidity of the area is 46percent with an increase of 0.7percent. January and February are the wettest, while June and July are the hottest months of the city. The area shows an arid continental climate having two main seasons that is winter (5 months) and summer (7 months), which can further be sub-divided into four rainy seasons namely winter, post-winter, summer, and post-summer seasons. Annually, the temperature condition of the area rises from January to June, remains stable till September and slackens upto December. The heaviest rainfall of Makkah Al Mukarramah recorded in January, February and September and constitutes as wettest months of the year. The lowest rainfall of the city seems in June and July (driest months). To overcome the issue of climate change at Makkah Al Mukarramah, it is recommended to control wars, air pollution, improve forests, and to establish well canal system in the area.

Keywords: Climatology, Climate Change, Seasonal Change, Hajj and Ummrah, Summer Season, Winter Season

1 Introduction

The current study explains the general climatology, climate change and variation of different weather elements comprises of temperature, precipitation, and relative humidity of Makkah Al Mukarramah, Saudi Arabia. As climate change particularly the global warming, cooling and its impacts on the environmental

DOI: 10.14738/tmlai.61.3978
Publication Date: 28th December, 2017
URL: http://dx.doi.org/10.14738/tmlai.61.3978

condition have attracted a great deal of interest to their theoretical and applied value. However, recently the change in weather elements in Saudi Arabia particularly at Makkah Al Mukarramah shows sweeping shifts as compared to the historical observations.

The climate of Makka Al Mukarramah province is marked by desert climate at the east and maritime climate at the west having hot long summers and moderate short winters. The distribution of weather elements in Makkah Al Mukarramah generally is due to the altitudinal and latitudinal zones of horizontal atmospheric convergence and divergence, the maritime or continental origin of prevailing air masses and the seasonal shifting of the zonal pressure and wind system. The local convectional system, which results from diurnal surface heating, also causes variation in the annual distribution of the weather elements.

The issues and changes of climate and weather is not a new phenomenon at Makkah Al Mukarramah but a number of workers discussed it in the past and recently. In which the utmost are summarized as; Alghafari and Khan (2016) has discussed the temperature and precipitation of Madinah Al Munawarah, Kingdom of Saudi Arabia for a time period of 1959 to 2011. Abdou (2014) has analyzed the temperature trends on Makkah Al Mukarramah, Saudi Arabia. Alharbi (2015) has described his views about the native settlements in Makkah Al Mukarramah area and factors affecting its distribution. Alrowaily *etal* (2016) have described the impact analysis of flooding area in Saudi Arabia. Determann (2012) has submitted a thesis on globalization, the state, and narrative plurality; Historiography in Saudi Arabia. Hussein, Bassam, and Zaidi (2014) have presented a book on extreme natural hazards, disaster risks and societal implications, a natural hazard in Saudi Arabia.

Makkah Al Mukarramah is one of the holiest places of Islam, situated at the eastern bank of Red sea in Kingdom of Saudi Arabia. It is the capital of Makkah Province and a birth place of Prophet Muhammad (Peace Be upon Him), the last prophet of God, the almighty. The city is located 70 kilometers (43 miles) inland from Jeddah in a narrow valley at an altitude of 277 meters (909 ft) above mean sea level. Its resident population is roughly two million, (2012), although visitors more than triple this number every year during Hajj period held in the twelfth Muslim lunar month of Zul Hijjah. The Makkah Al Mukarramah province located from 18^0-14` to $23^0$-23`North latitudes and 21^0-14` to $22^0$-23`East longitudes. The Holy city of Makkah Al Mukarramah situated at 39^0-25` East longitude and $21^0$-24` North latitude. The Major settlements that cover the entire Makkah Al Mukarramah province comprises of Rabigh, Tuwwal, Jiddah, Taif, Zalim, and Kiyat (Figure-1).

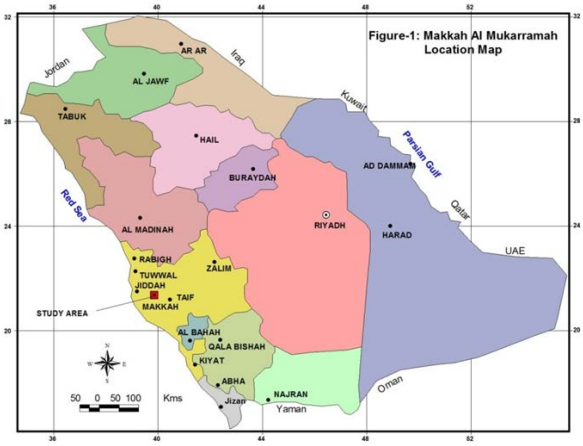


Figure 1

Historically, Makkah has also been known as Bakkah. In ancient times, Makkah was chiefly notable as a staging post on the trade route linking the spice producers of the east with Mesopotamia and the Mediterranean. Makkah lay about midway between Marib, one of the main cities, perhaps the capital, of the kingdom of Sheba (Yemen) and Petra (in Jordan), a city founded by Nabatean Arabs around the 6th century CE and which became a thriving center with commercial interests spreading into Syria. The religious significance of Makkah was established long before Islamic times. It was in Makkah that Allah commanded Ibrahim to leave Haajar and his young son Ishmael; it was in Makkah that Allah brought forth water from the Well of Zamzam which saved the life of Ismael and his mother and then allowed Makkah to develop as a habitable place. It was in Makkah that Allah instructed Ibrahim to build "the House of God" (the Holy Kaaba). As a result, from earliest times, Makkah became a place of pilgrimage and, although as centuries passed the pure faith of the Prophet Ibrahim became corrupted by idolatry and paganism, Makkah retained its hold on the minds of men as a place where men should worship. When Makkah came under the control of the Quraysh tribe, it was a noted trading center, a place for pilgrimage and the site of festivals chiefly remarkable for intensely fought poetry competitions and the excessive behavior of the idolaters.

2 Methodology

The work discusses the weather condition and climate change of Makkah-Al-Mukarramah taking into account the mean monthly and mean annual temperature, precipitation, relative humidity, and extreme events from 1985 to 2016 (31 years). The weather data obtained from the Meteorological Department, Makkah-Al-Mukarramah, Saudi Arabia. The monthly and annual averages and deviation from the mean have calculated for each weather element and tabulated for the analysis. The monthly and annual data have been further processed into seasonal means and deviation that led to the fluctuation of hot/dry or moderate/wet period.

For seasonal variation, the year has been divided into two main seasons that is summer and winter, so that months of the year having positive deviation from the mean considered as summer months and otherwise winter (Figure-2). Based on total annual precipitation, these two main seasons are further sub-divided into four sub-rainy seasons that is winter season (November to February), post winter season (March to April), summer season (May to July), and post summer season (August to October). For all variables, the data has been subjected to various statistical techniques like deviation from the mean, averages; sum, time series etc. and the results are shown on tables, charts, and graphs.

3 Findings and Discussions

As Makkah Al Mukarramah is the Holy place of the Muslims Ummah and majority of them are travelling in this city to perform the Holly Huj and Ummrah every year. It is therefore, the work is of a prime importance for the guidance of Muslim Ummah so, they are able to know about the climate of the Makkah Al Mukarrama, Kingdom of Saudi Arabia.

3.1.1 Climatology of Makkah-Al-Mukarramah:

The temperature, rainfall and relative humidity are generally, the utmost weather elements that represent climate of a particular area. No narrative of the weather and climate can be inclusive devoid of a notation of the prevailing temperature, rainfall and relative humidity inclination more than ever of its

distribution in place and time. The temperature, rainfall and relative humidity stipulation of a location endow with a working condition for all physical, physiological and ecological phenomenons. Resultantly, majority of the bio-climate indices are based on temperature, rainfall and relative humidity.

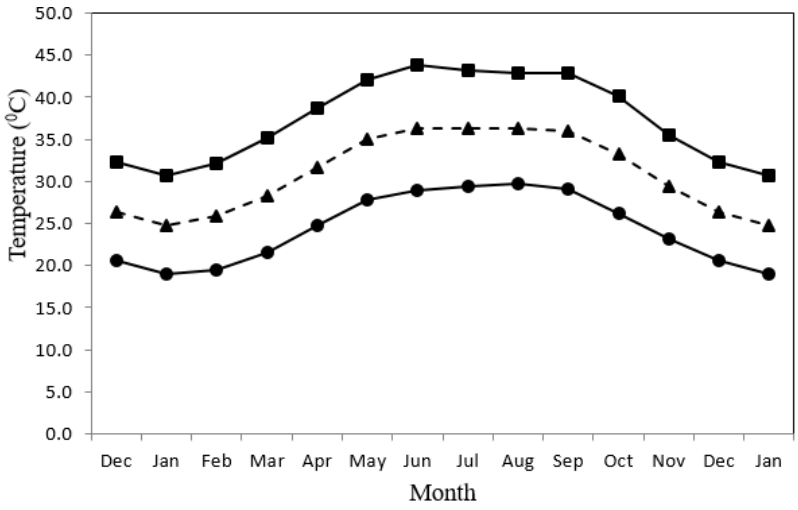


Figure-2: Makkah Al Mukarramah Mean Monthly, Mean Monthly Maximum and Minimum Temperature (1985-2016)

3.1.2 Temperature Distribution

The temperature of Makkah-Al-Mukarramah is fairly representative of the plain strip of land at the eastern coast of Red Sea with continental plain at the observatory and marine climates at Jeddah having hot long summers and warm short winters (Figure-2). The mean monthly temperature of Makkah-Al-Mukarramah is 31.6˚C having maximum temperature of about 38.3˚C and minimum of 25 degree Celsius. The area has moderate temperature in winters, when the mean monthly temperature drops to 26.9˚C (December to March) and hot in summers, when it rises up to or above 35 degree Celsius (April to October). The highest mean monthly temperature of the area is above 36˚C from June to September with a maximum of 43˚C in July and minimum of 29˚C in July to September, respectively and being hottest months of the Observatory. The lowest mean monthly temperature of 24.8˚C with 30.7˚C maximum and 19˚C minimum temperature recorded in January and constitutes as a coldest month of the area (Table-1). However, the temperature condition at Makkah-Al-Mukarramah is more severe as compared to Madinah Al Munawarah.

The annual cycle of temperature reveals that the temperature condition of the area rises up from January to June and slackens till December (Figure-3). This variation in the annual temperature of the area is a result of revolution of the earth, angle of the sunrays, precipitation and topography of Makkah-Al-Mukarramah. The deviation of mean monthly maximum and minimum temperature reveals that it is below the mean condition from November to March and placed in winter months, whereas it is above the average line from April to October and considered as summer months of the area. This shows that there are two main seasons in the area that is summer, which lasts from April to October (7 months), and winter that lasts from November to March (5 months) and fall in continental climates. The extreme average maximum temperature of the city is 45.42˚C recorded in June 2009 and 2012 as well as in July 2002 thrice times with a lowest maximum of 27.2˚C recorded in 1992 during the period of 1985-2016 and being the hottest and warmest years of the area. The lowest mean monthly temperature is 21.8˚C (1992) with mean

monthly highest temperature of 38.2˚C (1990 and 2012) and mean monthly minimum of 16˚C recorded in 1992 and highest mean monthly minimum temperature of 31.5˚C in July 2012, and August 2015 respectively and being coldest and hottest years of the series (Table-2).

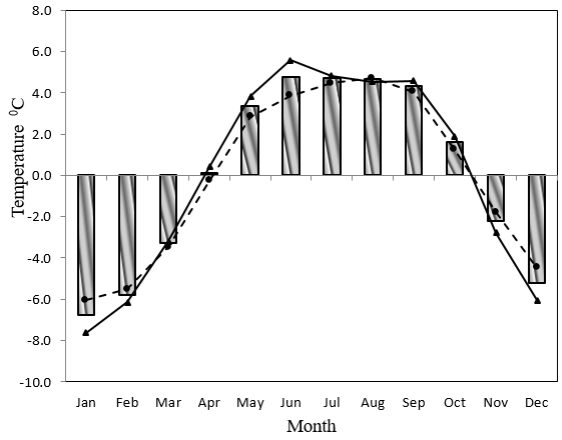


Figure-3: Makka Al Mukarramah Deviation of Mean Monthly, Maximum, Minimum Temperature (1985-2015)

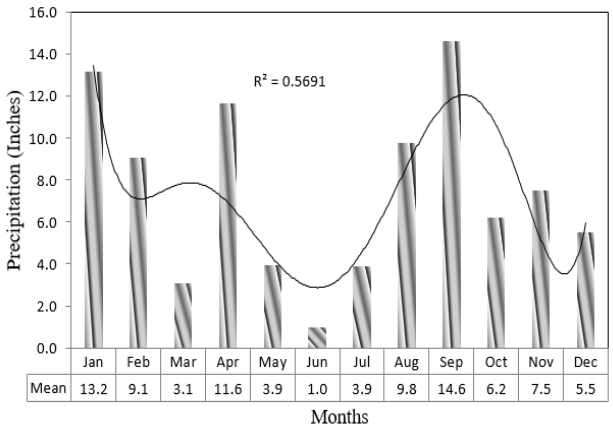


Figure-4: Makkah-Al-Mukkaramah Mean Monthly Precipitation (1981-2015)

Table-1: Makkah Al Mukarramah Temperature, Rainfall, Relative Humidity and Wind Speed (1985-2016)

Variable	Jan	Feb	Mar	Apr	May	Jun	Jul	Aug	Sep	Oct	Nov	Dec
Mean_T	24.8	25.8	28.3	31.7	35.0	36.4	36.3	36.3	36.0	33.2	29.4	26.4
Maxi_T	30.7	32.1	35.1	38.7	42.1	43.9	43.1	42.8	42.9	40.2	35.5	32.3
Mini_T	19.0	19.5	21.5	24.7	27.8	28.9	29.5	29.7	29.0	26.2	23.2	20.5
Rainfall	13.2	9.1	3.1	11.6	3.9	1.0	3.9	9.8	14.6	6.2	7.5	5.5
R_Humidityy	57.9	54.3	48.0	42.7	36.4	32.4	33.7	39.4	44.8	49.9	57.3	59.0

Source: Meteorological Department, Kingdom of Saudi Arabia

3.1.3 Rainfall Distribution

Succeeding to temperature, rainfall is the basic climate and weather factor. The agriculture activities on the terrain are the only real and lasting source of wealth, and it is in a large measure dependent on rainfall. The crop production fluctuates with the rainfall in such a way as to leave no doubt that the rainfall has been the real component, and the carrying capacity of grazing land in head of stock per square kilometer emphasis the same truth. It also influences the rate of evapotranspiration from vegetation and soil, which not only, affects the production of crops but also increases the ratio of moisture in atmosphere. The size and growth of vegetation is also closely related to the amount of rainfall. Sometimes, it also plays vital role in the industrial location and diseases controlling of an area.

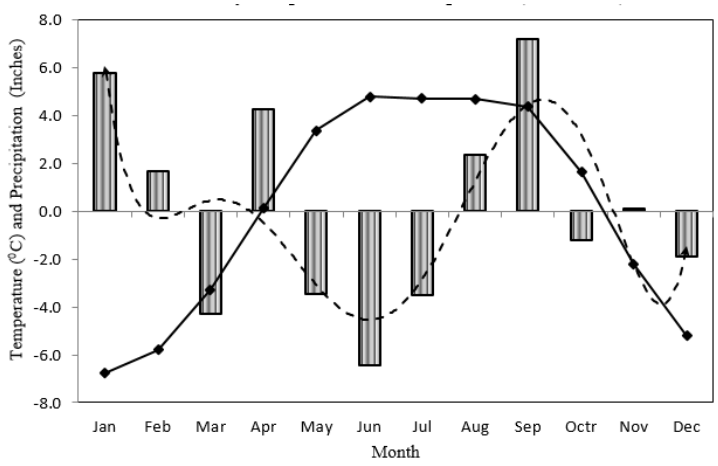


Figure-5: Makka Al Mukaramah Mean Monthly Temperature and Precipitation (1985-2015)

Annual rainfall is probably the most important climate indicator of productivity. The annual rainfall of the city is 189.1mm (7.4 inches), which is insufficient for plants growth and the city fall in the arid continental climate. However, during some years, the rainfall of the area rose to above 10 inches (254 mm) and showing semi-arid climates. The heaviest rainfall of 370.9 mm (14.6 inches) recorded in September (moistest) and lowest of 24.3 mm (One inches) in March (driest). The total rainfall of the area is 38.3 inches (972.8 mm) in winters that rose to 51 inches (1295.4 mm) in summers (Table-1). The heaviest rainfall of the area ever been recorded is 2362.2 mm (93 inches) in September 1999 (Table-3). In general, the higher rainfall occurred in the months of January, February, April, August and September and constitutes as the wettest months of the year and more suitable for the Holly Umrah.

The mean monthly rainfall indicates that it increases with decrease in temperature from November to February, while it decreases from April to October excluding April, August and September with the rise in temperature (Figure-4, Figure-5 and Table-1).

3.1.4 Relative Humidity

The relative humidity is defined as the amount of water vapor in the air relative to what the air can hold (Critchfield, 1978). An air should be called saturated, when its relative humidity reaches to 100 percent. It is also specific determinant of the amount and rate of evapotranspiration and critical climate factor in the rate of moisture loss by plants and animals, including human beings. It also expresses the average condition of water vapors in atmosphere. Relative humidity has also a physiological significance, as it determines the efficiency of the water-evaporating phase of our body cooling mechanism. It is also an important source of all forms of condensation and precipitation, where there is little moisture in the air; the precipitation will be low as compared to high atmospheric moisture. As the air is a principal absorber of solar energy and earth radiant energy as well, water vapor operates as a heat regulator and so have great effects on air temperature.

Generally, the average relative humidity of Makkah Al Mukkaramah recorded during 1981-2015 is 46.3% and represents a dry climate. The highest relative humidity of about 59% observed in December, whereas the lowest of 32.4% in June. The relative humidity is remaining high (above 50%) from November to February and constitutes as the pleasant months of the year. During March, April, September and October, it is between 40-50% and represents moderate atmospheric condition and 30 to 40% in the

excluding months and declared as a driest months of the area (Figure-6). The lowest ever recorded relative humidity of Makkah Al Mukkaramah is 24.6% (2013) with chill and calm condition. The highest of 67.2% relative humidity has recorded in 1997 and constitutes as the most humid year of the series. The trend of relative humidity shows that it decreases from December to June and increases onward till November (Table-1).

3.2 The Characteristics Seasons

For the seasonal characteristics of weather element of Makkah Al Mukarramah, the year has divided into summer and winter season. The inter-relation of factors affecting climate of Makkah Al Mukarramah reveals that the summer month in coastal areas may not be the summer month inland, and a summer month in plain may not be that of the mountains. Therefore, months of the year having positive deviation from the mean temperature condition are considered as summer months, otherwise winter (Figure-3). Generally, in Makkah-Al-Mukarramah, the summer lasts from April to October (7 months) and winters from November to March (5 moths).

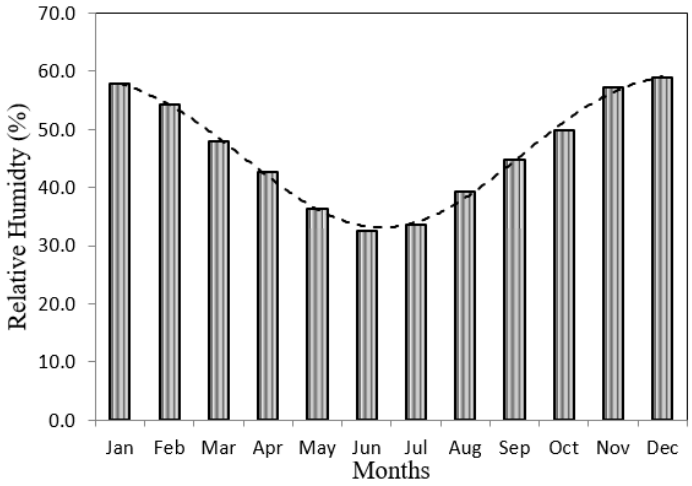


Figure-6: Makkah Al Mukkarmah Mean Monthly Relative Humidity (1981-2015)

On the basis of precipitation, these two main seasons of the city are further divided into four sub-rainy seasons (Figure-4). The winter season that lasts from November to February (Moist), post winter season from March to April (Moderate), Summer season from May to July (Hottest), and Post Summer season from August to October (Warmer). The summary of each season are presented as follow.

3.2.1 Winter Season

Obviously, the winter season of Makkah Al Mukarramah varies from November to February (Four months). During this season, the climate of Makkah Al Mukarramah is controlled by the interaction of Siberian high pressure, the Mediterranean lows (Western Depressions) and the Sudan trough (Jet Stream). The Mediterranean cyclones, which travel from west to east in association with upper troughs and active phases of subtropical and polar jets, are the main sources, which cause rainfall during these months in the area. Their potential generally decreases from north to south except for the mountainous areas, where uplift motion acts as a trigger factor for the uplifting of winds into the condensation level. During winter season, the ridge of the Siberian high extends into the northeastern regions of Saudi Arabia. The trough

of the Sudan low brings warm humid air in the lower atmospheric layer to the southwestern parts of the country. When the cold air, which is associated with the Siberian ridge, extends especially far into the southern region and the humid air from the Sudan trough is concentrated along the Red Sea, the rainfall that occurs in this case is due to both instability and orography.

The rainfall generated by these western disturbances of the cooler season is usually fairly widespread and light to moderate. These disturbances provide a total of 35.2inches (895.1mm) rainfall in the city and count as the moist season of the area with moderate temperature and maximum relative humidity. The heaviest fall is observed in the month of January 13.2 inches (334.5 mm), while the lowest is 140mm or 5.5inches in December. The general pattern of the rainfall during these months shows an increasing trend from November to January and decreases in February. In winter, the mean monthly temperature of the city is dropped to 26.8˚C, mean maximum temperature of 32.7˚C, mean minimum temperature of 20.6˚C, and relative humidity of 57.1 percent with clear skies (Table-2). During winter, the deviation of maximum temperature is -0.2^{0}C (Decrease), whereas the minimum temperature reveals an increase of 1.4^{0}C. There is no change in the rainfall condition; however the relative humidity shows an increase of 0.1 percent.

Table-2: Makkah Al Mukarramah: Mean Monthly Seasonal Temperature (^{0}C), Rainfall (Inches) and Relative Humidity (%)

	Maximum Temperature		Minimum Temperature	
Season	Average/Sum	Deviation	Average/Sum	Deviation
Winter	32.7	-0.2	20.6	1.4
Post Winter	36.9	0.1	23.1	0.0
Summer	43.1	-0.1	28.7	0.0
Post Summer	42.0	-0.1	28.3	0.1
	Rainfall		Relative Humidity	
Season	Average/Sum	Deviation	Average/Sum	Deviation
Winter	35.2	0.0	57.1	0.1
Post Winter	14.7	-0.1	45.3	0.0
Summer	8.8	0.1	34.2	0.0
Post Summer	30.6	0.0	44.7	0.0

Source: Meteorological Department, Kingdom of Saudi Arabia

3.2.2 Post Winter Season

The season varies from March to April (Two months) and is characterized by moderate temperature and rainfall. It is some time called as the cool moderate season of post winter and the summer season. From March to April, the anticyclones subsidence and clear skies, characteristic of the winter months still prevails and this in combination with a much stronger solar radiation sets the weather pattern for the season. Temperatures are high and a heavy, dry haze envelops in the interior, but drought still grips most of the city. The post winter season, in general, is characterized by violent weather, in the form of thunderstorms and squalls. The rainfall accompanying this vigorous convective system is low, but occasionally well-developed cumulonimbus clouds, are generated with strong squall wind, and violent dust storms.

In post winter season, the mean monthly temperature exceeds 30˚C, mean maximum temperature of 26.9˚C, mean minimum temperature of 23.1˚C with moderate rains of about 14.7 inches (374.6mm) and relative humidity of 57.1 percent (Table-2). These are the specific determinants, which caused parching

of leaves in plants and evaporation of sweats from human bodies. In post winter season, the area of Makkah Al Mukarramah is still under the influence of western depression associated with convection caused by the local heating that causes thunderstorms and rains. During post winter season, the deviation of mean temperature shows an increase of 1.7˚C, mean maximum temperature 0.1˚C and minimum temperature of 0.01˚C. The mean monthly rainfall reveals a decrease of -0.1inches (2.54mm), while the relative humidity remains stable.

3.2.3 Summer Season

The summer season of Makkah Al Mukarramah varies from May to July (Figure-2) having extreme temperature, low rainfall and relative humidity with chill condition. The deflected monsoon currents, generally, travel northwards towards Saudi Arabia on the shores of the Arabian Sea. This branch of monsoon reaches to Saudi Arabia at mid-June and reaches to its climax in July. However, it is of low vertical extant and generally, produces stratus clouds in the coastal areas and cumulus clouds in plains and causes heavy rains in the month of July with low pressure and high temperature on the continental area. The monsoon currents remain steady till it begins retreating towards the beginning of August. The variation in precipitation intensity from monsoon is due to its long trajectory decreasing the moisture index of these depressions as they travel over continental areas. These winds are the only source of summer rains in Makkah Al Mukarramah, which keep temperature low in the month of July. These winds give torrential rains with showers and cause damage in different sectors of the human life.

As the sub-continent is heated intensively in April and May, the zonal westerly start to move northward and it changes its direction towards southwest. As a result, the jet stream, which had been at about 30˚N during winter and post winter, tends to disappear. Disappearance becomes more frequent as the season advances and each disappearance is associated with a northward surge of the summer monsoon. Finally, in late May or early June, the jet disappears completely over Saudi Arabia and takes up a position at about 40˚N. Simultaneously, there occurs a shift of the low latitude trough and ridge positions, and the upper trough which previously was located at about 85˚E quickly moves westward some 10˚ and takes up a position over western Indo-Pakistan sub-continent at approximately 75˚East. The heating of the Middle East and the development of a surface pressure trough are unable to produce a northward advance of the ITC until large scale dynamic features of the circulation aloft become favorable (Trewartha, 1961). When the jet stream reappears at Saudi Arabia, again in fall, the summer monsoon again retreat southward and is called reversible monsoon.

During this season, the total rainfall of Makkah Al Mukarramah is 8.8 inches (222.5mm) having heaviest of 3.9 inches (98.3mm) in July and lowest of one inch (24.3mm) in June and being the moist and driest months of the season. The mean monthly temperature of the city is raise to 35.9˚C with mean monthly maximum temperature of 43.1˚C and mean monthly minimum temperature of 28.7˚C with hot gusts and sandy storms. The relative humidity of the area falls to 34.2 percent with chill condition. The area is extremely hot during these months due to sun burning overhead rays and low moisture (Table-2).

During summer season, the deviation of mean monthly temperature of the area shows a decreasing trend of about -0.1˚C, while the mean monthly temperature remain stable with a minor increase. There is a rise of inches 0.1(2.54mm) in the rainfall and 0.01 percent in relative humidity (Table-2).

3.2.4 Post Summer Season:

The season varies from August to October and is characterized by pleasant weather with moderate temperature and low rainfall. During this season, the low pressure shifts to the Arabian Sea from continent and give way to high pressure on land areas. This change in the pressure system causes change in the direction of the monsoon from southeast towards southwest. Consequently, the monsoon winds start blow from land towards ocean and designated by the name reversible monsoon. By the late fall, the trough of low pressure, separating the easterly and westerly air currents establishes over the southern part of Saudi Arabia. Along the discontinuity between the equatorial westerly and the zonal easterly, various kinds of perturbations develop ranging all the way from weak monsoon depressions to hurricanes. The depressions follow less well definite tracks than in summer, but in general, their progress is westward, so that their rainfall effects are concentrated in coastal region of Saudi Arabia.

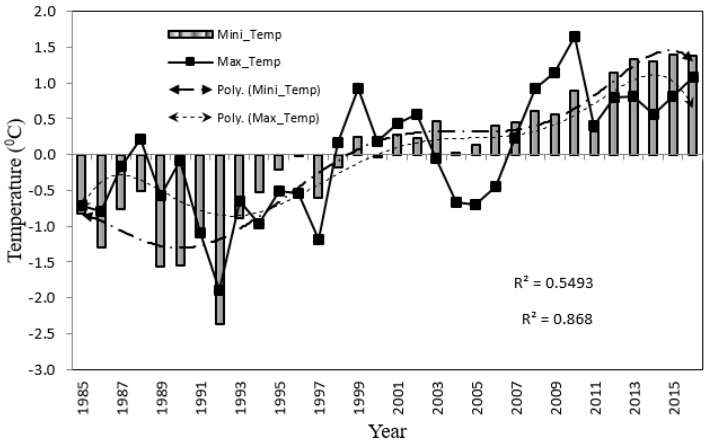


Figure-7: Makkah Al Mukarramah Deviation of Mean Monthly Maximum and Minimum Temperature (1985-2016)

During the fall months, the dynamic features of the circulation aloft, including jet stream and the orographically imposed troughs and ridges, begin to approach their cool season positions, with the reappearance at Saudi Arabia in September and October of the middle latitude westerly and the jet stream, and the re-establishment of the Polar front over the Mediterranean Sea/Europe. The western disturbances once more become an important control of weather in Saudi Arabia (Trewartha, 1961).

In post summer season, the mean monthly temperature of the Makkah Al Mukarramah falls to below 35.2˚C with mean monthly maximum temperature of 42˚C, and mean monthly minimum temperature of 28.3˚C. The total rainfall of Makkah Al Mukarramah remains low that is 30.6 inches (796.9mm) during these three months having relative humidity of about 44.7 percent. The monsoon lows give way to those of winter currents (Western Depression) and most of the rains are caused by local thunderstorms or reversible monsoon winds, which develop due to local high pressure and heating on the inland areas. The retreat of monsoon from north Arabian Sea in marked by disappearance of the stratus clouds with a gradual increase in daytime temperature (Table-2). During post summer season, the mean monthly maximum temperature shows a decrease of -0.1˚C, whereas there is an increase of 0.1˚C in mean monthly minimum temperature. The mean monthly rainfall and relative humidity of the area remains stable throughout the season.

3.3 Climate Change

The climate is not a static phenomenon but its change from time to time and place to place in term of variation as well as fluctuation. Sometime, these changes are made on seasonal basis or daily basis and show linear or periodic trends. The climate change of Makkah Al Mukarramah in term of temperature, rainfall and relative humidity is presented as follow.

3.3.1 Temperature Change

The natural, physical, religious and socio-economical ecosystems of Makkah Al Mukarramah are not only dependent on how much temperature rises or falls but also on how it varies from year to year and time to time. Generally, the mean monthly, mean maximum and minimum temperature of the area show an increasing trend during 1985-2016. The temperature of the city remains low during the period, where the concentration of rainfall remains high and converse condition during dry years. During 1985-98, the trend of mean maximum temperature is below the mean condition and turned to positive deviation onward till 2002. From 2003 to 2006, there is a decrease in the maximum temperature of the city, whereas it remained high from 2007 to 2016 (Figure-7). The deviation of mean monthly maximum and minimum temperature is almost same with little alter in few years of the series. The minimum temperature of the Makkah Al Mukarramah remained below the average condition from 1985 to 1998 and rise to positive deviation till 2016.

The mean monthly maximum temperature of the city shows an increase of one degree Celsius having -0.1˚C in mean monthly maximum and -1.1˚C in mean monthly minimum temperature during 1985-2016 (Table-3). The highest positive deviation of 1.3˚C in mean monthly temperature is recorded in 2010 and 2016; 1.7˚C in mean monthly maximum temperature (2010) and 1.4˚C mean monthly minimum temperature in 2015 and 2016 respectively. The lowest of negative deviation of -2.1˚C of mean monthly temperature has noted in 1992, -1.9˚C mean monthly maximum (1992) and -2.4˚C mean monthly minimum temperature in the same year and considered as hottest and coldest years of the series. Generally, the deviation from the mean temperature condition of the city shows an increase in mean monthly and mean monthly maximum temperature, while it indicates a decreasing trend in the mean monthly minimum temperature.

The work is further condensed into the deviation of five and ten years for the purpose to understand the time period of positive and negative deviation. The figure-9 shows that the mean monthly maximum temperature of the area was remain negative during the time period of 1985-90, 1991-95, 1996-2000 and 2001-2005 and turn to positive in the excluding years 2006-2016 and shows a periodic pattern of 20 years time interval.

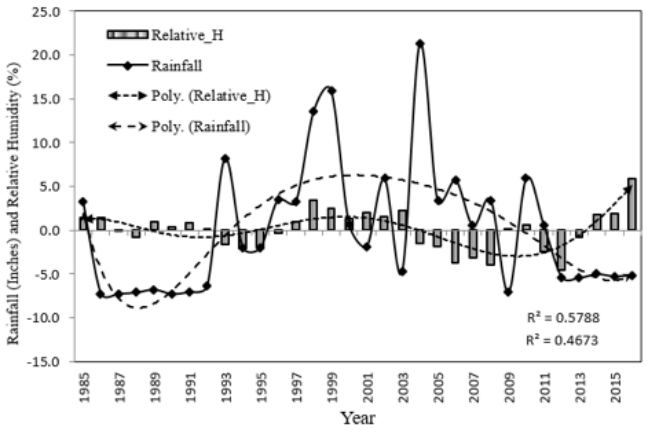


Figure-8: Makkah Al Mukarramah Deviation of Mean Monthly Rainfall and Relative Humidity (1985-2016)

But in general, the trend shows an increase throughout the time period of 1991-2016. As for as the mean monthly minimum temperature has concerned, it remained below the mean condition during the first fifteen years (1985-2000) and turned to positive deviation from 2001-2016. However, the increase in the minimum temperature of the area seems throughout the series. The mean monthly temperature of the area also shows the same pattern. It is concluded that the temperature condition at Makkah Al Mukarramah shows an increasing trend since 1985 till to date with little ups and downs in few years of the series having a linear trend (Figure-9).

As for as, the deviation from the mean of ten years has concerned, the mean monthly, mean monthly maximum and minimum temperature of the Makkah Al Mukarramah shows a negative deviation from 1985 to 2005 (20 years) and then change into positive deviation during the last decadal period of 2006-2016 and it is expected that it will further increase till 2025 (Figure-9).

3.3.2 Rainfall Change

After temperature, rainfall is the most important weather element that maintains stability in the water resources and agriculture activities in the area. Besides, it also plays a vital role as a controlling factor of temperature and other weather elements. The annual trend of rainfall shows that the area remains dry during 1985-95, wet during 1996-2011 and reversed during 2012-2016 having ups and downs for a short time interval (Figure-7). The area falls in arid continental climate and drought prevails in the city for most of the years. The longer and most severe drought of the city was observed during 1985-95, and moderate drought since 2012 till to date.

Generally, the sum of deviation from the mean condition of rainfall shows a total increase of 1.4 inches (36mm) 1985-2016. Overall, the trend of rainfall at Makkah-Al-Mukarramah shows an increasing trend throughout the series (Figure-8). The maximum of positive deviation of 21.3 inches (540.7mm) of the city was recorded in 2004, while the lowest of-7.3 inches (184.6mm) observed in 1986, 1987 and 1990 and constitutes as a wettest and driest years of the series with drought and chill condition (Table-3).

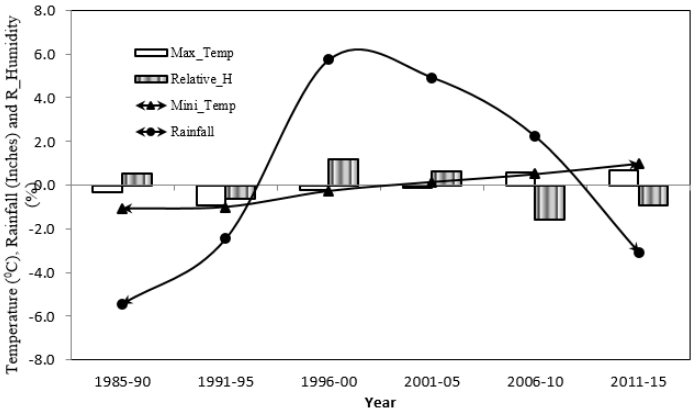


Figure-9: Makkah Al Mukarramah Five Years Deviation of Maximum and Minimum Temperature, Rainfall and Relative Humidity (1985-2015)

As for as the five years deviation of Makkah Al Mukarramah has concerned, during 1985-90 and 1991-95, the deviation from the mean shows a negative trend and the area remained under a long drought condition. From 1996-2010, the trend shows a positive deviation having heavy rains in some years and count as wet period of the Makkah Al Mukarramah. The trend in precipitation takes a negative deviation during 2011 to 2016 and expected that it will further decrease with passage of time. However, the overall trend of rainfall at Makkah Al Mukarramah reveals an increasing trend with passage of time (Figure-9 and Figure-10).

The ten years deviation of the Makkah Al Mukarramah has calculated for the purpose to understand the regular or periodic trend of rainfall in the area (Figure-10). The figure shows that the rainfall condition of Makkah Al Mukarramah has remained below the mean condition in the first decadal period of 1985-1995, turned into positive deviation from 1996 to 2005 and taken negative trend again during 2006-2016 and reveals a periodic trend after each ten years. It is expected that during 2016-2025 the rainfall condition of the area will remain positive with gradual increase in floods.

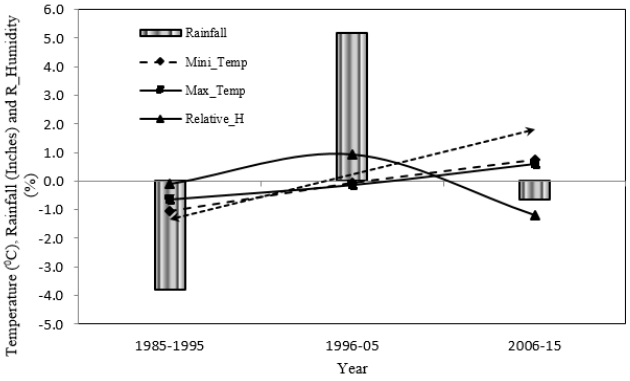


Figure-10: Makkah Al Mukarramah Ten Years Deviation of Maximum, Minimum Tempreature, Rainfall and Relative Humidity (1985-2015)

3.3.3 Relative Humidity

After temperature and rainfall, the relative humidity is the utmost variable of weather and climate that control the percentage of moisture in atmosphere and save the area from chill condition. Besides, it also

plays a vital role in the temperature inversion of atmosphere, hydrological cycle, conduction and evapotranspiration.

Obviously, the sum of deviation of mean monthly relative humidity of Makkah Al Mukarramah is 0.7% and shows an increase during 1985-2016. The highest positive deviation of 5.9% has recorded during 2016 and the lowest negative deviation of -4.6% in 2012 and considered the years of maximum and minimum relative humidity throughout the series (Table-3). The annual trend of deviation of relative humidity from the mean condition shows ups and downs after each five years having an increasing trend throughout the series (Figure-8). The five years deviation of relative humidity indicates that it has above the mean condition during 1985-90 and took negative trend from 1991-95. Onward, 1996 to 2005, it remained above the mean condition and then slackens from 2006 to 2016 (Figure-9). The ten years deviation of the relative humidity from the mean reveals that it is below the mean condition during 1985-1995, positive trend from 1996-2005 and then inverse during 2006-2016 (Figure-10). Generally, the trend of the relative humidity of Makkah Al Mukarramah shows a periodic change after each ten years.

Table-3: Makkah Al Mukarramah Deviation of Mean Monthly Temperature, Rainfall and Relative Humidity (1985-2016)

	Mean Monthly				Deviation from the Mean			
Year	Mini_T	Maxi_T	Rainfall	R_H	Mini_T	Max_T	Rainfall	R_H
1985	24.2	37.6	10.6	47.8	-0.8	-0.7	3.2	1.5
1986	23.7	37.5	0.1	47.8	-1.3	-0.8	-7.3	1.5
1987	24.2	38.1	0.1	46.2	-0.8	-0.2	-7.3	-0.1
1988	24.5	38.5	0.2	45.5	-0.5	0.2	-7.2	-0.8
1989	23.4	37.7	0.6	47.2	-1.6	-0.6	-6.8	0.9
1990	23.5	38.2	0.1	46.6	-1.5	-0.1	-7.3	0.3
1991	24.0	37.2	0.3	47.1	-1.0	-1.1	-7.1	0.8
1992	22.6	36.4	1.0	46.5	-2.4	-1.9	-6.4	0.2
1993	24.1	37.6	15.6	44.7	-0.9	-0.7	8.2	-1.6
1994	24.5	37.3	5.4	44.3	-0.5	-1.0	-2.0	-2.0
1995	24.8	37.8	5.3	44.5	-0.2	-0.5	-2.1	-1.8
1996	25.0	37.8	10.9	45.9	0.0	-0.5	3.5	-0.4
1997	24.4	37.1	10.6	47.3	-0.6	-1.2	3.2	1.0
1998	24.8	38.5	20.9	49.8	-0.2	0.2	13.5	3.5
1999	25.2	39.2	23.3	48.8	0.2	0.9	15.9	2.5
2000	25.0	38.5	8.1	47.6	0.0	0.2	0.7	1.3
2001	25.3	38.7	5.5	48.3	0.3	0.4	-1.9	2.0
2002	25.2	38.9	13.3	47.9	0.2	0.6	5.9	1.6
2003	25.5	38.2	2.6	48.6	0.5	-0.1	-4.8	2.3
2004	25.0	37.6	28.7	44.8	0.0	-0.7	21.3	-1.5
2005	25.1	37.6	10.7	44.5	0.1	-0.7	3.3	-1.8
2006	25.4	37.9	13.2	42.6	0.4	-0.4	5.8	-3.7
2007	25.5	38.5	7.9	43.2	0.5	0.2	0.5	-3.1
2008	25.6	39.2	10.8	42.3	0.6	0.9	3.4	-4.0
2009	25.6	39.5	0.3	46.3	0.6	1.2	-7.1	0.0
2010	25.9	40.0	13.3	46.9	0.9	1.7	5.9	0.6
2011	25.3	38.7	7.9	43.9	0.3	0.4	0.5	-2.4
2012	26.1	39.1	2.0	41.7	1.1	0.8	-5.4	-4.6
2013	26.3	39.1	1.9	45.5	1.3	0.8	-5.5	-0.8
2014	26.3	38.9	2.3	48.1	1.3	0.6	-5.1	1.8
2015	26.4	39.1	2.1	48.2	1.4	0.8	-5.3	1.9
2016	26.4	39.4	2.3	52.2	1.4	1.1	-5.1	5.9
Annual	25.0	38.3	7.4	46.3	-1.1	-0.1	1.4	0.7

Source: Meteorological Department, Kingdom of Saudi Arabia

4 Conclusion

- Makkah Al Mukarramah receives 7.4 inches (189.1 mm) rainfall annually and considered as arid continental climate. However, the coastal region of the province (Jeddah) is characterized by land and ocean breezes and shows a maritime climate. The average relative humidity of Makkah Al Mukarramah is 46.3 percent with an increase of 0.7percent.
- The annual march of temperature reveals that the temperature condition of the area rises up from January to June, remains stable till September and slackens upto December. The mean monthly temperature of the city recorded during 1985-2016 is 31.6˚C having maximum temperature of about 38.3˚C and minimum of 25˚C. The area has moderate temperature in winters, when the mean monthly temperature is dropped to 27˚C and hot in summers when it rises up to above 35 degree Celsius.
- The extreme average maximum temperature of the city is 45.4˚C recorded in June 2009 and 2012 with a minimum of 31.5˚C recorded in July 2012 having 38.2˚C mean monthly temperature in June 2009 and 2012 and being the hottest months/years of the series. The lowest mean monthly temperature of 20.9˚C (February, 1992) with mean monthly lowest maximum of 27.3˚C (February, 1992), and mean monthly minimum of 14.6˚C recorded in February 1992 and being a coldest months/years of the series.
- Makkah Al Mukarramah characterized by two main seasons that is summer season that lasts for seven months (April to October) and winter season for five months (November to March). The summers of the city are extremely hot, while the winters are warm. The city is extremely hot in July and August and cool from December to January. On the basis of rainfall and relative humidity, these two main seasons are further sub-divided into four seasons that is winter (November to February), post winter (March to April), summer (May to July) and post summer seasons (August to October). The area remains moist during winter seasons, moderate in post winter seasons, hot in summer season and warm in post summer seasons.
- The total precipitation of the city is 189.1mm (7.4 inches), which is insufficient for plants growth having arid continental climate with hot long dry summers and short warm winters. The heaviest precipitation of 3160.8mm (124.4inches) recorded in January and February 2004 and constitute as the moistest months/year of the series. In general, the heaviest rainfall occurred in the months of January, February and September. The average relative humidity of the area is 46.3% having highest of 67.2% in December 1997 and lowest of 24.6% in June 2013. During winter months the relative humidity remains high and more suitable for Holly Umrah, whereas there is a chill condition during summer months in the area that evaporates water from the human bodies and turns the skin dry. The mean monthly rainfall indicates that it increases with decrease in temperature from November to February, while it is decreases from April to October excluding April, August and September with the rise of temperature
- The mean monthly temperature of the city shows an increase of one degree Celsius having decrease of -0.1˚C in mean monthly maximum and -1.1˚C in mean monthly minimum temperature during 1985-2016. In general, the deviation from the mean shows that there is a periodic change in the temperature condition of the precipitation as well as temperature after each ten years.

- The area falls in arid climate, however during some years it rose up to above 10 inches (Semi-Arid) and drought prevails in the city for most of the years. The longer and most severe drought of the city observed during 1986-92 (7 years) and moderate drought in 2011-2016 (6 years). Generally, the sum of deviation from the mean condition of rainfall shows a total increase of 1.4 inches (36mm) during 1985-2016. The driest years of the series having precipitation 0.1inches (2.54mm) are 1986, 1987, and 1990 respectively, when the area passed through a severe drought and chill condition. Overall, the trend of rainfall at Makkah Al Mukarramah shows an increasing trend throughout the series and it is expected that the precipitation condition of the city will increase with passage of time. However, at Jeddah there is marine climate and completely different from Makkah Al Mukarramah.
- The heaviest rainfall of Makkah Al Mukarramah recorded in the month of January, February and September and constitutes wettest months of the year. The lowest rainfall of the city seems in June and July and marked as driest months of the year having high temperature and lowest relative humidity.
- The climate of Makkah Al Mukarramah is more severe as compared to Madina Al Munawarrah in term of Temperature, rainfall, relative humidity, wind speed, atmospheric pressure and dust storms. So the Muslims who visited these cities for Holly Huj and Umrah should be aware about these environmental differences in both Holly cities of Hejaz.

RECOMMENDATIONS

- The explosive materials used in wars and burning of oil wells particularly in Gulf war, Syria, and Iraq are hazardous to the green house of the nature. Therefore, it is required to control wars in the Middle East and to resolve the issues with cooperation instead of using supremacy and to provide harmony to the physical environment in the study area. These wars not only caused climate change in the Middle East but also increased the rate of fog and smog in the Southwest Asia and affected the winter rains.
- Vehicles and Industrialization are the utmost factors that tire out green house gases into the ocean atmosphere. Therefore, it is indispensable to perk up automobile engines, industrial machineries with new trend and techniques, and to design policy for the controlling of chlorofluorocarbons, and awareness of people about the ongoing environmental issues in the area.
- The meager forests of Middle East are basic element that exposed earth surface to irradiant solar energy and enhancing the evapotranspiration and pollution. Therefore, legislation is required to put off deforestation and also to ensure the immediate replanting of trees particularly on the mountains slopes and barren lands round the city of Makkah AL Mukarramah, and to make a policy for its protection and preservation. The suitable tree for the plantation in desert area is Acacia having low water requirements.
- The famous conference about, “how to combat global warming” are: Vienna convention for the protection of the ozone layer (1985), Montreal Protocol (1987), UN Earth Summit (1992), Kyoto protocol (1997), Agenda-21 (2002), and IPCC (2007) etc, but the conditions recommended in the summary of these summits has not implemented properly on national as well as international level by the member countries. It is therefore, recommended that the countries of Middle East

must plan a criteria to collect funds and to fight for the green revolution in the entire region together particularly at Medina Al Munawarah and Makkah Al Mukarramah (The Holly Places of Muslim Ummah).

- For intensive forestation, it is substantial to make available facilities to the occupants of the area and to hearten the community reforestation. Also, to initiate programs for the awareness of the locals about the roles of forests in green revolution and climate change.
- The ozone layer can be protected by replacement of chlorofluorocarbons by hydro-fluorocarbons released to atmosphere from aerosol, refrigeration and air conditioning, and foam Industries. Besides, a forestation, which releases oxygen in abundance to the atmosphere in turn plays a significant role in increasing concentration of ozone in the stratosphere, is highly appreciable in this regard.
- Further research is required to study the causes and impacts of climate change and global warming at Saudi Arabia so, that by proper planning, the problem arising due to the increase in temperature and decrease in precipitation shall be control for the performing of Holly Huj and Umrah.

REFERENCES

[1]. Alghafari, Y. Khan, S. (2016). Temperature and Precipitation of Madinah Al Munawarah, Kingdom of Saudi Arabia (1959-2015), Atmospheric and Climate Sciences, Vol. 6, pp. 402-414. Online. http://www.scirp.org/journal/acs; http://dx.doi.org/10.4236/acs.2016.63033.

[2]. Abdou, A. E. A. (2014). Temperature Trend on Makka, Saudi Arabia. Atmospheric and Climate Sciences, Vol. 4, pp. 457-481. http://www.scirp.org/journal/acs; http://dx.doi.org/10.4236/acs.2014.43044.

[3]. Alharbi, A. B. (2015). Native Settlements in Makkah Al Mukarramah Area and Factors Affecting its Distribution, Advances in Anthropology, Vol. 5, pp. 267-273. Online. http://www.scirp.org/journal/aa; http://dx.doi.org/10.4236/aa.2015.54020.

[4]. Alrowaily, A. *etal*. (2016). Impact Analysis of Flooding Area in Saudi Arabia, International journal of scientific and technical research in engineering (IJSTRE), Vol. 1. (2). pp. 1-7.Online. www.ijstre.com.

[5]. Determann, J. M. (2012). Globalization, the State, and Narrative Plurality; Historiography in Saudi Arabia, The Thesis submitted for the Degree of Ph.D in History, Department of History, School of Oriental and African Studies, University of London, UK. P. 349.

[6]. Hussein, M. T. Bassam, A. M. A. Zaidi, F. K. (2014). Extreme Natural Hazards, Disaster Risks and Societal Implications, Chapter-12. Natural Hazard in Saudi Arabia. Cambridge University Press. © Cambridge University Press, pp. 243-251.

[7]. CRITCHFIELD H.J., 1987., General Climatology., 4th Edition., Prentice' Hall of India New Delhi-110001., p. 429.

[8]. Khan J.A., May, 1993., The Climate of Pakistan., Rahber Publishers Karachi., p. 79.

[9]. Blair, T.A., 1942., Climatology., *General and Regional.,* Prentice-Hall, INC. New York., p. 478.

[10]. Trewartha, G.T., 1968., An Introduction to Climate., 4th Edition., Mc Grawhill Kogakusha, LTD., p. 408.

SOCIETY FOR SCIENCE AND EDUCATION
UNITED KINGDOM

TMLAI TRANSACTIONS ON MACHINE LEARNING AND ARTIFICIAL INTELLIGENCE

Volume 6 No. 1
ISSN 2054-7390

Publishing Student Graduation Projects Based on the Semantic Web Technologies

[1]Linh Trinh Thi Ngoc, [2]Hung Hoang Bao, [1]Hue Nguyen Thi Hoa; [1]Dung Vo Hoang Phuong
[1]Department of Infomation Technology
Korea-Vietnam Friendship IT College, Vietnam
[2]Management of the Ministry of Information and Communications
Korea-Vietnam Friendship IT College, Vietnam
linhttn@viethanit.edu.vn;hunghb@viethanit.edu.vn;huenth@viethanit.edu.vn;
dungvhp@viethanit.edu.vn

ABSTRACT

Linked Open Data is already widely available in several industries, such as libraries, biomedicine and Linked government data, etc. for data sharing, promoting semantic web development and maintaining a global culture of information exchange. In this paper, we focus on developling a linked data-based application for the management and publication of student graduation projects at ViethanIT library. Students can refer to these resources for further developing future research directions. In addition, the published student graduation projects are useful to detect plagiarism in other researches (projects). Therefore, we first introduce an ontology defined classes and properties to describe student graduation projects in the Korea-Vietnam Friendship IT College. We then outline how to apply the ontology along with SPARQL queries and RDFa to publish the student graduation projects on a semantic website.

Index Terms — Student graduation project, Digital library, Semantic web, Linked data, RDF, SPARQL, RDFa.

1 Introduction

Libraries play an important role in providing research and education resources for students and lecturers in a higher education institution (universities for short). Recently, with the development and widespread use of information and communication technology, going to libraries for finding documents is gradually being replaced by the search for information through the Internet. It is, thus, crucial for developing digital libraries. Resources from traditional libraries are transformed into digital documents based on the digital document descriptors, such as Marc [5], Dublin Core [6], BibTex [7], etc.

At a higher education institution's library, student graduation projects are the third important kind of resources besides books and textbooks, which need to be stored. This storage can help readers to look up documents, search for future research directions, and also help universities and lecturers better check plagiarism in other researches, particularly other student graduation projects. Student graduation projects have been digitized in several libraries. However, their contents are quite sketchy [8, 9]. Projects that have the same research area do not any a semantic link. Therefore, the storage, reference and search of the project could not meet the requirements of readers. We can see the frequency of finding the

DOI: 10.14738/tmlai.61.3945
Publication Date: 30th December, 2017
URL: http://dx.doi.org/10.14738/tmlai.61.3945

projects, the digital documents on the sites such as luanvan.net.vn, doan.edu.vn, luanvan365.com, thuvienluanvan.net, etc. that shows a high number of access times and very high access capacity compared to digital libraries of universities. Showing above comparison, we do not discuss the quality of digital documents, copyright, and other objective factors, but emphasize the ease of access and the ability to provide standard database of the sites.

Digital data at a university library often has many sources that may relate to libraries and other non-library organizations. But a difficulty in data exchange is the inconsistency in data formatting and data standards. This problem is a major obstacle to the interconnection and exchange of data between information systems, in which library information is of great interest. The semantic web and especially linked data encourage organizations to publish, share and link their data by using web pages. Data display can be improved through linking to other sources of information. Become part of the data-link site, or semantic cloud [1], it also means that libraries can better meet user expectations, such as the availability of information in a format that can be understood by readers and computers. In addition, participating in the semantic cloud can help with many complex library tasks while maintaining and optimizing, detecting duplication of local data sets.

Based on the above reasons, within this paper, we propose a solution for managing and publishing student graduation projects for digital libraries, using linked open data on the basis of semantic web. The scope of this paper is to describe the objects, to set the schema in the ontology forms for the identifiers of the digital objects, then use SPARQL to query the data and use RDFa to publish student graduation projects on the website. The actual data is the student graduation projects of Korea-Vietnam Friendship IT College (ViethanIT).

The rest of this paper is structured as follows: Section 2 introduces our ontology for creating linked data at ViethanIT library. In Section 3, we outline how to represent student graduation projects in RDF format The stored data is then published by using SPARQL queries and RDFa, which is presened in Section 4. Section 5 shows some experimental results. Finally, we conclude the paper in Section 6.

2 OntVIETHANIT - An Ontology for Creating Linked Data for Student Graduation Projects

We define semantic metadata to describe library data by linked data to the library of Viethanit. The main purpose is the information on textbook, reference material, graduation project in the library will become part of the web by publishing, sharing and cross-linking data on the web.

In Vietnam, the application of semantic technology in digital libraries has started to be interested in the last few years. Some results of the authors [2, 3, 4] focus on building a personal ontology to manage the library information that they are interested in. These ontologies are not built on a common rule, so reusing or integrating related data with other libraries or non-library organizations is difficult, it does not solve the problem of sharing data before.

Therefore, we develop applications based on the ontology developed by the semantic web community in the world. However, when applied in practice, specifically to the data of the Viethanit library, we found that no existing ontology was completely matched. Therefore, we must analyze the data modeling requirements for reuse of existing ontologies, for example:

- In order to model the data to represent people and organizations, we use some ontology as the set of RDA elements [10], the FOAF vocabulary [11]. In particular, the RDA covered the relevant functional requirements for unit logs (FRBR) very well.

- For topic headings, data modeling is based on the use of Simple Knowledge Organization System (SKOS) [12] and Dublin Core components [6].

In addition to reusing the appropriate classes and attributes, we have to build new classes and attributes. The ontology core OntVIETHANIT is shown in Figure 1.

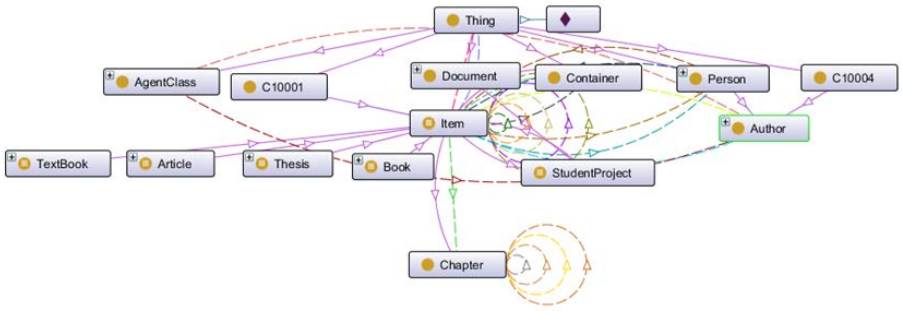


Figure. 1. OntVIETHANIT ontology

In this section, we describe the classes and attributes defined for storing and publishing student graduation projects, which are a very important part of our ontology and also the focus of the paper.

To describe a student graduation project (StudentProject), we define the #StudentProject class as follows:

<owl:Class rdf:about="http://viethanit.edu.vn/ns#StudentProject">

…

<rdfs:subClassOf rdf:resource="http://xmlns.com/foaf/0.1/Document"/>

<owl:equivalentClass rdf:resource="http://viethanit.edu.vn/ns#Item"/>

…

</owl:Class>

The class #StudentProject is the child (subClassOf) of the class http://xmlns.com/foaf/0.1/Document (defined by FOAF) and is equivalent to the class #Item, where class #Item is the subclass of class http: //rdaregistry.info/Elements/c/C1000 (defined by the RDA). Each the student graduation project are performed by a (or a group) of student, with a title and a field. To describe the author, we define the class #Student, the subclass of the class http://xmlns.com/foaf/0.1/Person (defined by FOAF) and the class #Author:

<owl:Class rdf:about="http://viethanit.edu.vn/ns#Student">

…

<rdfs:subClassOf rdf:resource="http://xmlns.com/foaf/0.1/Person"/>

<rdfs:subClassOf rdf:resource="http://viethanit.edu.vn/ns#Author"/>

…

</owl:Class>

Each student will belong to a Faculty and belong to a School class. So we also define the classes #Faculty and #SchoolClass.

To indicate the student or student group as the author of a graduation project, we define the attribute #hasAuthor, which is the inverse attribute of the property #isAuthorOf :

<owl:ObjectProperty rdf:about="http://viethanit.edu.vn/ns#hasAuthor">

…

<rdfs:domain rdf:resource="http://viethanit.edu.vn/ns#Item"/>

<rdfs:range rdf:resource="http://xmlns.com/foaf/0.1/Person"/>

<owl:inverseOf rdf:resource="http://viethanit.edu.vn/ns#isAuthorOf"/>

<rdfs:subPropertyOf rdf:resource="http://rdaregistry.info/Elements/w/P10061"/>

…

</owl:ObjectProperty>

The abtract of the graduation project introduces a general description of the project is also of interest, it help the reader know the basic content of the project quickly, then decide whether to choose the project to read or borrow? Therefore, the abtract information of the project should be managed. In addition, some information such as graduation year, page number, table of contents, and references, etc. are the information elements needed to describe the project in more detail, as reference information for the reader, can be used to refer to the author and extract the material from the reference information.We define or reuse attributes to describe this information. Other information such as surname, firstname, title, description, subject, abstract, date, etc. from the ontology provided by FOAF or Dublin Core, etc. Other information will be described using the properties (ObjectProperty or DatatypeProperty) defined by us, such as the ObjectProperty property: #hasAuthor.

3 Representation of student graduation projects using RDF

In this section, we outline how to represent student gradution projects at ViethanIT library in RDF format based on our ontology.

The RDF (Resource Description Framework) is a standard Web data exchange published by the World Wide Web. RDF is a set of rules for markup language, providing data model and simple syntax so that standalone systems can be exchanged and used. It is designed so that computer systems can understand and read information, not simply to present data to users.

The following is an example of a student graduation project imported on an RDF file (the code of the project is standardized and according to the existing digital library of ViethanIT).

Example 1:

```
<rdf:Description rdf:about="&ins;DALT141703">
    <rdf:type rdf:resource="&vh;StudentProject"/>
    <vh:hasContainer rdf:resource="&ins;StudentProjectIT"/>
    <dc:title>Building Spider game with OpenGL ES and C++</dc:title>
    <dc:description>Student graduation project</dc:description>
   <dc:subject>Mobile program, Game program, C++, OpenGL, Unity</dc:subject>
        <dc:language>Vietnamese</dc:language>
        <dcterms:abstract>…</dcterms:abstract>
        <vh:pagenums>38</vh:pagenums>
        <dc:date>01-06-2017</dc:date>
        <dcterms:publisher rdf:resource="http://viethanit.edu.vn/"/>
        <vh:numAuthors>1</vh:numAuthors>
        <vh:hasAuthor rdf:resource="&ins;CCLT08A014"/>
        <vh:isSupervisedBy rdf:resource="&ins;T04-15.111-060"/>
</rdf:Description>
```

We then publish graduation student projects on a semantic web using SPARQL queries and RDFa attributes. Therefore, in the next section, we introduce the SPARQL language and RDFa.

4 Implementation

4.1 Quering data using the SPARQL query language

SPARQL [14] is the standard query language and protocol for Linked Open Data on the web or in a semantic graph database (RDF triplestore). The SPARQL standard is designed and endorsed by the W3C, it allows users and developers to focus on what they want to know instead of how a database is organized. In addition, a SPARQL query can also be executed on any database that can be viewed as RDF via middleware. This is what makes SPARQL such a powerful language for computation, filtering, aggregation and subquery functionality. The power of SPARQL together with the flexibility of RDF can lead to lower development costs as merging results from multiple data sources is easier. Its goal is to assist people to enrich their data by linking it to other global semantic resources, thus sharing, merging, and reusing data in a more meaningful way. SPARQL has four types of queries: ASK, SELECT, CONSTRUCT, DESCRIBE.

In the following, we present some examples using SPARQL queries for managing and publishing student graduation projects on the web.

Example 2: To get a list of student graduation projects in 2016 in Information Technology Faculty, we use the following query:

```
SELECT ?studentProject  WHERE

{ ?studentProject rdf:type vh:StudentProject;

    vh:hasContainer ins:StudentProjectIT;

   dc:date ?d.

filter (YEAR(xsd:dateTime(?d))=2016).}
```

For each graduation project, to get all the information and post it on the web, we built a graph of the information related to that project using the SPARQL statements. The following query is an example for graphing information for the project named http://viethanit.edu.vn/instances#DALT131601:

Example 3:

```
CONSTRUCT  { ?studentProject ?x ?y}

WHERE

{?studentProject ?x ?y

        filter (?studentProject=<http://viethanit.edu.vn/instances#DALT131601>)

}
```

Note, the results returned by a SPARQL query is an XML file containing the nodes including the information requested. Based on this result, we export data to the semantic web by combining it with RDFa.

In addition to publishing information about student graduation projects, allowing users to search information is an indispensable function in the system. Therefore, we provide users with the search function based on the attributes: project name, keyword, specialty, project author, etc. For example, the following query is used to search game programming projects:

Example 4:

```
SELECT distinct ?name ?container ?title ?description ?subject ?language ?abstract ?page ?publisher
?date ?author ?supervisor  WHERE

 {

        ?name rdf:type vh:StudentProject;

        vh:hasAuthor ?author;

        vh:isSupervisedBy ?supervisor;

        vh:hasContainer ?container;
```

```
        dc:title ?title;
        dc:description ?description;
        dc:subject ?subject;
        dc:language ?language;
        dcterms:abstract ?abstract;
        vh:pagenums ?page;
        dc:date ?date;
        dcterms:publisher ?publisher.
    FILTER regex( lcase(str(?title)), "Building game" )}
```

With the use of the SPARQL query language, data queries on our ontology OntVIETHANIT and RDF files become easier. Thus, the publication, search and statistics of graduation projects of students are very convenient for the user.

```
<div typeof="vh:StudentProject" about="http://viethanit.edu.vn/instances#DALT131601">
    <h2>Project: <span style="font-style: italic" property="dc:title" content="Building Spider game with OpenGL ES and C++">Building Spider game with OpenGL ES and C++</span></h2>
    <table>
      <tr>
        <td> Author: </td>
        <td> <a property="vh:hasAuthor" href="http://viethanit.edu.vn/instances#CCLT08A014">&instances;CCLT08A014</a>
        </td>
    </tr>
    <tr>
        <td> Superviser: </td>
        <td> <a property="vh:isSupervisedBy" href="&instances;T04-15.111-023">http://viethanit.edu.vn/instances#T04-15.111-060</a>
        </td>
    </tr>…
```

Figure. 2. An example of using RDFa attributes to publish student graduation projects

4.2 Publishing data using RDFa

RDFa (RDF in attributes) [15] was first proposed by Mark Birbeck in the form of a W3C note entitled XHTML and RDF. RDFa is a way to express RDF data within XHTML, by enriching the existing human-readable data with RDF attributes. RDFa then enables us to bridge the gap between what humans see when viewing a document and what machines "see" when they process the same document.

According to [15, 16], RDFa alleviates the pressure on markup language designers to anticipate all the structural requirements users of their language might have, by outlining a new syntax for RDF that relies only on attributes. By adhering to the concepts and rules in this specification, language designers can import RDFa into their environment with a minimum of hassle and be confident that semantic data will be extractable from their documents by conforming processors. The RDFa attributes play different roles in a semantically rich document. Briefly, those roles are:

- Syntax attributes: @prefix, @vocab.
- Subject attributes: @about.
- Predicate attributes: @property, @rel, @rev.
- Resource attributes: @resource, @href, @src.
- Literal attributes: @datatype, @content, @xml:lang or @lang.
- Macro attributes: @typeof, @inlist.

Figure 2 is an example of some RDFa attributes that publish student graduation projects.

5 Results

Digital library of Vietnam-Korea Friendship Information technology College is using Koha Open Source. This Koha system has been updated with a huge amount of data about students, books, textbooks, reference materials, student graduation projects, etc. so, the data sources can be reused. Figure 3 shows the diagram of our system.

From KOHA system, which is now used at our library, we export data related to student graduation projects from the KOHA's database and save as a CSV file. We retrieve 1048 records. Using our system, which is developed based on Jena - a java framework [13], the records are then transferred into an RDF file. However, the data taken from the KOHA system is not as complete as we expected, more information need to be added to each resource, such as abstract, keywords, language, page number, etc. In addition, our system allows users to add and upload their resources via a web interface. The new data is added to the system if and only if it is well checked by at least one system administrator.

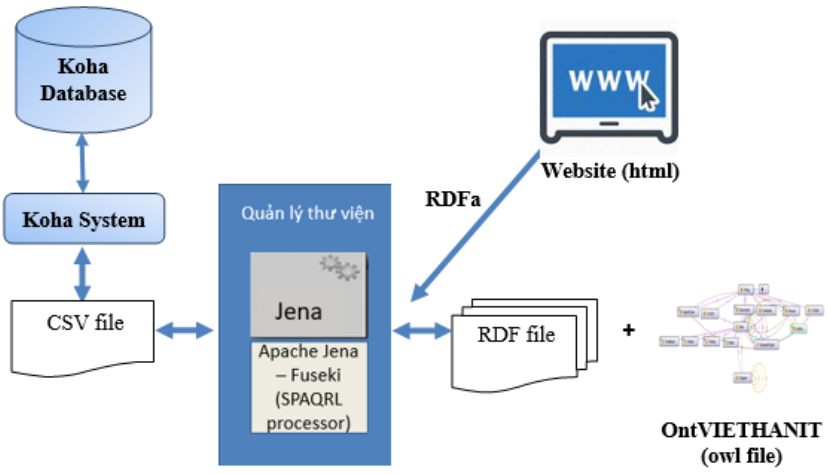


Figure. 3. The diagram of our system

In our system, the data is stored in RDF format with the desired information. They are used along with the OntVIETHANIT ontology to extract data by SPARQL queries to publish on the web. As a result of the execution of each SPARQL query, we achieve a XML file which results in nodes including required information. The information is then published on the web by adding RDFa attributes to XHTML. Figure 4 is an example of a student graduation project published.

Student graduation project

Project name: Building Spider game with OpenGL ES and C++

Author:	Kong Hung Luong http://viethanit.edu.vn/instances#CCLT08A014
Superviser:	Dung Vo Hoang Phuong http://viethanit.edu.vn/instances#T04-15.111-060
Description:	Student graduation project
Type:	http://viethanit.edu.vn/instances#StudentProjectIT
Keywords:	Mobile program, Game program, C++, OpenGL, Unity
Language:	Vietnamese
Abstract:	Researching OpenGL and C ++, building Spider game. Include the contents: - Overview of OpenGL and C++ - Introduction to Unity, Unity Engine - Analysis and Design of Spider game - Building Spider game
The total number of pages:	68
Publisher:	http://viethanit.edu.vn/
Date (online):	2017-06-01

References:

1. https://tfetimes.com/wp-content/uploads/2015/06/0735714347_Creating_Games_in_C-_A_Step-by-Step_Guide.pdf
2. http://canvas.projekti.info/ebooks/Game%20Coding%20Complete%20-%204th%20Edition.pdf

Figure. 4. A student graduation project published on the web

Using RDFa and OntVIETHANIT ontology, our website has become a website that has data linked and connected to the "semantic cloud" in the world. Information about the student graduation projects published on the web are always available and understandable for people and computers. Figure 5, for example, is the graph of a student graduation project published on the web depicted in Figure 4.

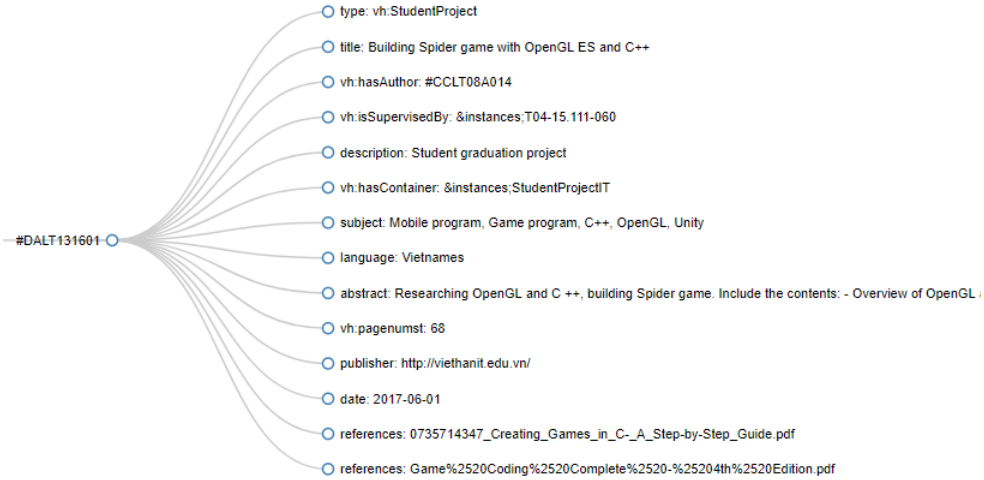


Figure 5. The semantic graphics of a student graduation project

6 Conclusion

In this paper, we first present the OntVIETHANIT ontology used to describe student graduation projects. Student graduation projects are then stored semantically in RDF format that allows machines to understand. Using the stored data, we created a semantic website for publishing of the student graduation projects in ViethanIT, in which we use the language SPAQRL to query database and add RDF statements directly into XHTML via RDFa. Our system is being tested and will be ready for use in the near future.

REFERENCES

[1] Jan Hannemann, Jürgen Kett, "Linked Data for Libraries", World library and information congress: 76th IFLA general conference and assembly, Swenden, (2010).

[2] Phu Nguyen Ngoc, "Applying semantic web technology and data mining to develop the system of searching and reporting science studying projects", Da Nang university, (2013).

[3] Hanh Pham Thi Hong, "Applying semantic web to search scientific information for Vietnam-Korea Friendship IT College", Da Nang university, (2016).

[4] Khanh Nguyen Le Tung, "Applying Semantic Web to build information retrieval system at the library of the Information Technology College", Da Nang university, (2013).

[5] https://www.loc.gov/marc/

[6] http://dublincore.org/documents/dces

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

[8] http://thuvienso.hau.edu.vn:8888/dspace/handle/123456789/3

[9] http://thuvienso.lhu.edu.vn/tailieuvn/luan-van-bao-cao/tat-ca-tai-lieu-luan-van-bao-cao-227-0.html

[10] http://www.rdaregistry.info/

[11] http://xmlns.com/foaf/spec/

[12] https://www.w3.org/2004/02/skos/references

[13] https://jena.apache.org/

[14] https://ontotext.com/knowledgehub/fundamentals/what-is-sparql/

[15] https://groups.drupal.org/node/22231

[16] https://www.w3.org/TR/rdfa-core/

SOCIETY FOR SCIENCE AND EDUCATION
UNITED KINGDOM

TMLAI TRANSACTIONS ON MACHINE LEARNING AND ARTIFICIAL INTELLIGENCE

Volume 6 No. 1
ISSN 2054-7390

Structural Optimization of Deep Belief Network by Evolutionary Computation Methods including Tabu Search

[1]Tomohiro Hayashida, [1]Ichiro Nishizaki, [1]Shinya Sekizaki, [1]Masanori Nishida, [1]Murman Dwi Prasetio
[1]Graduate School of Engineering, Hiroshima University, Hiroshima, JAPAN;
hayashida@hiroshima-u.ac.jp

ABSTRACT

This paper proposes structural optimization method of a Deep Belief Network (DBN) which consists of multiple Restricted Boltzmann Machines (RBMs) and a single Feedforward Neural Network (FNN) using several kinds of evolutionary computation methods and modularization. The performance, accuracy of data classification or data prediction, should strongly depend on the structure of the network. Concretely, the number of RMBs, the number of nodes in the hidden layer of RMB. The result of the experiments using some benchmarks for image data classification problems by DBN optimized by the proposed method, DBN without any structural optimization, and some other data classification methods indicate that our proposed method defeats other existing classification methods.

Keywords: Structural optimization; Deep Belief Network; Tabu search; Modularization; Evolutionary Computation.

1 Introduction

A neural network consists of a number of units which have simple nonlinear transfer functions and approximation capability for a number of kinds of complex problems comparative small number of calculation. Therefore, neural networks are applied to data analysis, data mining and data classification. A sufficient learning cannot be performed if the size of the network is too small. Adversely, overfitting occurs to the learning data and it loses generalization ability if the size is too large. Therefore, the appropriate structure of the neural network is required to be determined for each target problem for higher performance of a neural network. Traditionally, structure of a neural networks are determined through a trial and error procedure based on the experiences of a designer of the neural networks. However, a huge computation time is required by such determination process.

In recent years, several structural optimization methods of the neural networks simultaneously with learning are proposed. Delgado et al. [4] proposed a simultaneous optimization method of learning and structure using an evolutionary multi-objective optimization methods, SPEA 2 and NSGA II with the learning error and the number of units of the hidden layers of neural networks as the objective functions. The target is a recurrent neural network (RNN), and the structural information of RNNs are encoded in gene format, such as the number of intermediate layer and the presence or absence of connection between the units of neighboring layers. Katagiri et al. [15] improve the procedure of Delgado et al. for extended Multi-Context Recurrent Neural Network (exMCRNN) which include eliminate unnecessary connection between nodes and the elite preserving strategy. Hayashida et al. [8] propose a structural

DOI: 10.14738/tmlai.61.4048
Publication Date: 30th December, 2017
URL: http://dx.doi.org/10.14738/tmlai.61.4048

optimization method for Recurrent Neural Network (RNN) by introducing two stage taboo search, one of the meta-heuristic solutions. Here, they define that the structure of a RNN is determined by the number of inputs, the number of intermediate layer units, the number of feedback layers. Hayashida et al. [7] proposed a structural optimization for a combined neural network model of a Feedforward Neural Network (FNN) and an Auto Encoder (AE) which performs dimension compression to remove extra data and redundant data. Their procedure optimize the number of input data, the number of units of the middle layer of AE, and the number of units of the hidden layer of FNN by using tabu search.

Because of the performance of a neural network should be measured based not only on the learning accuracy but also the generalization capability, a neural network is evaluated based on the degree of error between both the training data and data for verification of the generalization capability in above mentioned optimization method. In order to evaluate a neural network, it is necessary to divide the known data into training data and for verification of the generalization capability. However, even in a method such as cross validation, the data may be biased and the network cannot be evaluated well. Nishida et al. [20] has improved the method of Hayashida et al. [7], they use the Self Organization Map (SOM) to convert the data mapped onto a 2 D plane by k-means method and divide them into training data and data for verification of the generalization capability. Though such data generation method, they succeed in reducing the bias of features between divided data, and improving learning accuracy and improving generalization capability.

Deep Neural Network (DNN) consists of a lot of multiple layers, and the data analysis performance such as data prediction, data classification, or data mining is dramatically improved compared with conventional neural networks such as Feedforward Neural Network (FNN). Therefore, a lot of applications of DNN are reported in the various study fields. A neural network composed of many layers is difficult to learn properly by back propagation, however, the learning procedure of DNN is constructed for appropriate learning by applying apply pretraining [9], drop out [27] and so forth. Various models of DNN such as Convolutional Neural Network (CNN) [13], Deep Belief Network (DBN) [25], are proposed. This paper focuses on DBN which has a structure with multiple layers of Restricted Boltzmann Machine (RBM). DBN has succeeded in acquiring higher data analysis capability by effectively incorporating a feature extraction process which is conventionally performed by trial and error. In DBN, multiple RBMs were incorporated into the learning process as feature extractors. DBN performs feature extraction with unsupervised learning called Pre-training and supervised learning called Fine-tuning are performed based on the extracted features.

From the structural characteristics of DBNs, it can be considered that there exists a great relationship between the structure of DBM, the number of hidden layers and units constituting each layer, and the performance in data classification or prediction. Performance improvement is expected by giving an appropriate structure corresponding to input data. This paper proposes a new method for highly accuracy and efficient structure optimization for DBNs. Additionally, this paper compares the proposed method and the conventional methods by the numerical experiments, and verifies the effectiveness of the proposed method.

The rest of this paper is constructed as follows: Section 2 introduces the explanation of the neural networks and the conventional methods. Section 3 outlines the proposed method and Section 4 describes the design and result of the numerical experiments. Finally, Chapter 5 summarizes this paper.

2 Neural Networks and Structural Optimization

2.1 Neural Networks

This section introduces several models of neural networks.

FNN, Recurrent Neural Network (RNN) are neural networks with a transfer function as a sigmoid function: $f(s) = 1/\{1 + \exp(-s)\}$. An FNN has a layered structured units, and it consists of an input layer, an output layer, and a layer disposed between them called a hidden layer. Signals from the input layer to the output layer are transmitted in only one direction. In general, the number of hidden layers is not necessarily 1. A neural network with a signal feedback structure added to FNN is called a RNN. There are types of RNN such as Jordan Network [14] and Elman Network [5] depending on its structure. RNN is applied to prediction of time series data analysis.

Generally, NN with a number of layers is called Deep Neural Network (DNN). Appropriate learning of DNN based on back propagation (BP) is difficult. Several learning methods corresponding to DNN such as pretraining [10], drop out [27] have been proposed. There exist various models of DNN such as Deep Belief Network (DBN) [9], Convolutional Neural Network (CNN) [13], and so forth.

2.1.1 Deep Belief Network (DBN)

DBN [9] is a type of DNN that performs data classification or data prediction with high-precision by performing feature extraction by using a network in which a plurality of Restricted Boltzmann Machines (RBM) are concatenated. In the feature extractor part, unsupervised learning called Pre-training is performed, and Fine-tuning which is supervised learning is performed in the remaining process of it [9, 16]. Figure 1 shows DBN consisting of three layers of RBM and one layer of FNN as an example of DBN.

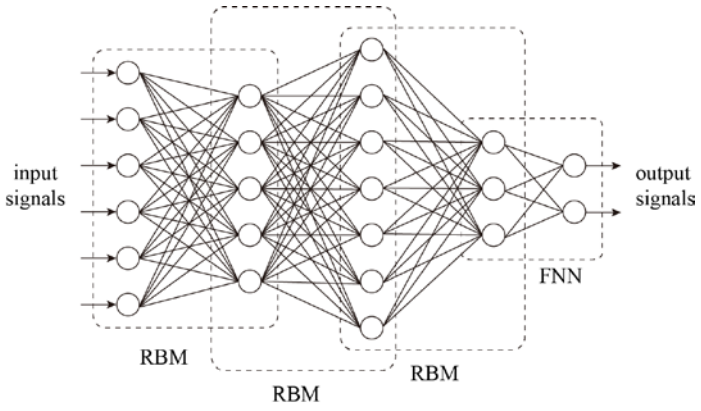


Figure 1. Deep Belief Network (DBN).

2.1.2 Restricted Boltzmann Machine (RBM)

A RBM is a Boltzmann machine with an undirected bipartite graph consisting of a visible layer and a hidden layer, and there is no connection between the units in a same layer. The connecting weights and the thresholds are updated so that the hidden layer extracts the feature amount of the input data of the

visible layer. A RBM can compress dimensionality of input data, feature learning, or collaborative filtering. Figure 2 shows an example of RBM.

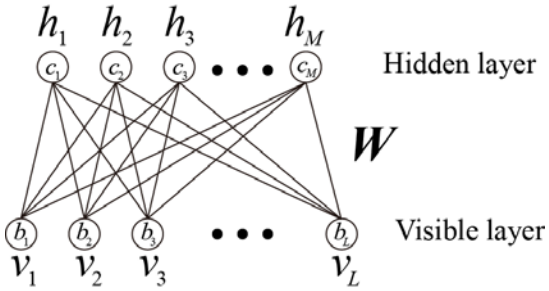


Figure 2. Restricted Boltzmann Machine (RBM).

Let L, M be numbers of units of a visible layer and a hidden layer shown in Figure 2, respectively. Let $\boldsymbol{v} = (v_1, v_2, \dots, v_L), v_i \in \{0,1\}, i = 1,2, \dots, L$ be a set of units of a visible layer, and $\boldsymbol{h} = (h_1, h_2, \dots, h_M), h_j \in \{0,1\}, j = 1,2, \dots, M$ be a set of units of a hidden layer, $b_i, i = 1,2, \dots, L$ be a bias of a unit i in a visible layer, $c_j, j = 1,2, \dots, M$ be a bias of a unit j in a hidden layer, and $\boldsymbol{W} = \{W_{ij}, i = 1,2, \dots, L, j = 1,2, \dots, M\}$ be the connecting weights between visible layer and hidden layer.

The conditional probability of hidden elements conditioned with visible elements is

$$p(h_j = 1|\boldsymbol{v}) = sigmoid(c_j + \textstyle\sum_i v_i W_{ij}). \quad (1)$$

And the conditional probability of visible elements conditioned with hidden elements is

$$p(v_i = 1|\boldsymbol{h}) = sigmoid(b_i + \textstyle\sum_j W_{ij} h_j), \quad (2)$$

where, $sigmoid(s) = 1/(1 + \exp(-s))$. The weight parameters W_{ij} and the biases $b_i, c_j, i = 1,2, \dots, L$, $j = 1,2, \dots, M$ are updated based on the maximum likelihood estimation method.

2.2 Structural Optimization of the neural networks

A structure of a neural network is characterized by number of hidden layers and number of units of each layer, and the performance is greatly affected by these characteristic parameters. That is, in order to obtain the appropriate property for the data classification of or data prediction, it is necessary to find the optimum characteristic parameters.

Delgado et al. [4] focus on a RNN called Elman Network with a feedback layer which only connects to the hidden layer, and they propose a structural optimization method based on genetic algorithm (GA), and they demonstrate the usefulness of structure optimization by numerical experiments. Katagiri et al. [15] focus on a Multi Context RNN (MCRNN) [11, 12] which consists of multiple feedback layers. And they propose an improved structural optimization method of Delgado et al. [4] for extended MCRNN by adding a structure of Time Delay NN (TDNN) to MCRNN.

Hayashida et al. [7] indicate the performance for data analysis of a neural networks with similar structures to be similar, and they propose a structural optimization method based on Tabu Search (TS) [6] for a combined neural network of a Sandgrass Type neural network and FNN named ST-FNN. Numerical experiments show that TS is more effective for a structural optimization method of NN than GA. Additionally, Hayashida et al. [8] proposed a structural optimization method of MCRNN by applying TS. In

order to deal with the problem that the number of structural parameters of MCRNN is many and the search space for structure evaluation becomes enormous. Therefore, by dividing the search space into a plurality of small regions, the searching process is performed by introducing a short term memory and a long term memory for the purpose of improving efficiency of comprehensive search of each small region and extensive search of the entire search space.

3 Structural Optimization of Deep Belief Networks (DBN)

This paper proposes a structure optimization method with parameters of each hidden layer and unit numbers of each layer of RBMs constituting DBN. The proposed method includes local search based on tabu search for structural optimization, modularization for improving of RBMs the learning efficiency which is required for structural evaluation of DBN and enormous calculation time. Furthermore, number of hidden layers and the number of units are optimized separately to reduce the search space.

3.1 Outline of the Structural Optimization

The outline of the DBN structure optimization method proposed in this paper is shown below.

Step 1. Optimize number of hidden layers**.**

Step 1-1. Let $n \in [\underline{n}, \bar{n}]$ be the number of hidden layers, and let $n = \underline{n}$ as the initial value. Let $n^* = n$ and $E_n^* = 0$.

Step 1-2. A DBN with n hidden layers is evaluated based on the training and generalization capability. Here, the number of units of each layer is 500. (The structural evaluation procedure is explained in the following subsection.)

Step 1-3. If $E_n > E_n^*$, then update the best solution as $E_n^* = E_n, n^* = n$.

Step 1-4. If $\bar{n} > n$, then let $n = n + 1$ and return Step 1-2. Otherwise let n^* be number of hidden layers.

Step 2. Optimize number of units of each layer (Rough search)

Step 2-1. Let m_i be number of units of i-th layer, $i = 1,2, \dots, n^*$, and let $t = 0$.

Step 2-2. Divide search range of number of units of i-th hidden layer, $[\underline{x}_i, \bar{x}_i]$ into k_i subranges. Let $d_i^{j_i} \equiv [\underline{x}_i^{j_i}, \bar{x}_i^{j_i}]$ be j_i-th subrange, $j = 1,2, \dots, k_i$, $\underline{x}_i^1 = \underline{x}_i, \bar{x}_i^{k_i} = \bar{x}_i$. n^*-dimensional subrange is represented as $D_j \equiv \Pi_{i=1}^{n^*} d_i^{j_i}$. Here, $j \in [1, \Pi_{i=1}^{n^*} k_i]$.

Step 2-3. Let $(\dot{x}_1^j, \dot{x}_2^j, \dots, \dot{x}_{n^*}^j)$ be the center of gravity of the subrange j, and let $\boldsymbol{x^j} \equiv (\hat{x}_1^j, \hat{x}_2^j, \dots, \hat{x}_{n^*}^j) = (\lfloor \dot{x}_1^j + 0.5 \rfloor, \lfloor \dot{x}_2^j + 0.5 \rfloor, \dots, \lfloor \dot{x}_{n^*}^j + 0.5 \rfloor)$ be the representative point of the subrange j.

Step 2-4. Evaluate the structure of the neural network corresponding to the representative point $\boldsymbol{x^j}$. Let E^j be the evaluate value of the subspace j.

Step 2-5. Choose a subspace with highest evaluation value, and let the selected subspace θ.

Step 2-6. If $\bar{x}_i^\theta - \underline{x}_i^\theta \leq k_i, \forall i$ or $t = T_l$, then go to Step 3. Otherwise, Generate next search range with vertices at the representative points $[x_i^{j-1}, x_i^{j+1}]$ in the neighboring subspace centered on θ, and go to Step 2-2 with update the number of iteration as $t = t + 1$.

Step 3. Optimize number of units of each layer (Detailed search using Tabu search)

Step 3-1. Randomly generate an initial solution $\boldsymbol{x^0} = (x_1, x_2, \ldots, x_{n^*})$ from the subspace θ, let $\boldsymbol{x}^* = \boldsymbol{x^0}$ be the current best solution, and E^* be the evaluation value for $\boldsymbol{x}^*$. The solution $\boldsymbol{x^0}$ is recorded in the tabu list. Set the number of iteration as $t = 0$.

Step 3-2. Evaluate each neighbor of the current solution $\boldsymbol{x}^t$ which are not included in the tabu list, and let $\hat{\boldsymbol{x}}^t$ solution with highest evaluation value in the evaluated solutions, and $\hat{E}^t$ be the evaluation value of $\hat{\boldsymbol{x}}^t$.

Step 3-3. Add the solution $\hat{\boldsymbol{x}}^t$ in the tabu list. If the number of solutions recorded in the tabu list is larger than the predetermined size of tabu list, the earliest recorded solution in the list is deleted instead of $\hat{\boldsymbol{x}}^t$.

Step 3-4. If $\hat{E}^t > E^*$, then let $E^* = \hat{E}^t$, and $\boldsymbol{x}^* = \hat{\boldsymbol{x}}^t$.

Step 3-5. If $t = T_{tb}$, then let $\boldsymbol{x}^*$ be the best solution and terminate the structural optimization procedure. Otherwise, let $t = t + 1$ and go to Step 3-2.

The rest of this section describes the procedure of structure evaluation, optimization of number of layers, number of units of each layer, and modularization of RBM for effective learning of DBN.

To optimize the structure of a DBN, considering the number of hidden layers and the number of units simultaneously is required. However, in this paper, after optimizing the number of hidden layers of DBN first, optimize the number of units of each hidden layer. Even when structural optimization is conducted in such order, verification experiments on the relation between DBN structure and data prediction accuracy are conducted to verify whether same structure are obtained or not, compared to a optimization procedure such that both are taken into consideration simultaneously. In the experiment, the accuracy for unknown data D_3 is calculated by using DBN where the number of units of each n hidden layer is fixed to 500. Additionally, the accuracy for D_3 is calculated by using DBN where the number of units of each n hidden layer is optimized by the method described in the next section. The 3-category image data is used for the experiment, and the classification accuracy for the data D_3 is set as the verification result. The experimental results are shown in Figure 3.

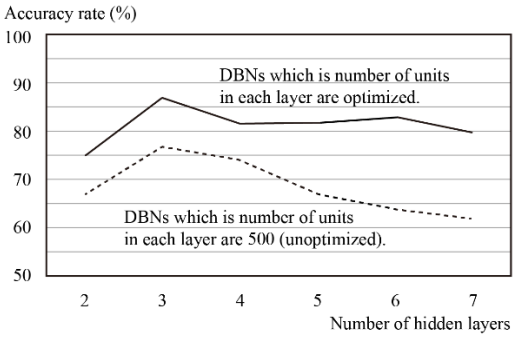


Figure 3. Relation between optimized and unoptimized number of units of each hidden layer.

From Figure 3, the data classification accuracy is highest when the number of hidden layers is 3 in both of two types of variation experiments indicated by a broken line and a solid line. In other words, DBN showed

that the superiority and inferiority relationship of performance based on the number of hidden layers is the same irrespective of whether the number of units of each hidden layer is optimized. In the experiment corresponding to Figure 3, only experiments using 3-category image data are shown. However, similar relationships are observed in other prior experiments. Therefore, the optimal number of hidden layers of DBN is determined that is with the highest performance fixing the number of each layer as 500 first. Subsequently, the numbers of units of all hidden layers are optimized.

3.2 Structure Evaluation

As the number of components of the neural network increases, the accuracy for learning data increases. However, the generalization ability for unknown data decreases. On the contrary, if the constituent elements of a neural network are small, the features of the target data cannot be properly learned. Therefore, it is desirable that the structure of a neural network is not only the prediction error with respect to the learning data, but also the prediction error with respect to the data not used for learning. In this paper, the target data is divided into three and evaluate the network structure by the following procedure.

At first, divide the target data D into three dataset (D_1, D_2, D_3) for learning, generalization verification, and test. Error back propagation is conducted using data D_1 and the verify generalization capability is evaluated based on output error when input data of D_2 is given to the learned neural network. Let e_{tr} be training error for data D_1 and e_{ve} be the output error for data D_2, i.e., e_{ve} represents the generalization capability. Let T be the number of data, $M\ be$ the number of units of the output layer, $O_j(t)$ be the output value from the j-th unit in the output layer of the neural network at the period t, and $Y_j(t)$ be the j-th factor of the target value. The error is defined by the mean square error between the target value and the output of neural network output as

$$e_A = \frac{1}{T}\sum_{j=1}^{M}\sum_{i=1}^{T}\left(O_j(i) - Y_j(i)\right)^2, A = tr, ve. \tag{3}$$

Based on these criteria of error, structure of a neural network A^k is evaluated by

$$E(A^k) = \frac{1}{e_{tr} + e_{ve}}. \tag{4}$$

3.3 Optimization of Number of Hidden Layers

Let $\underline{n}$ and $\overline{n}$ be minimum and maximum number of hidden layers, respectively. In the related literature [16], 500 is employed for the number of units of each hidden layer of DBN. Similarly, this paper employs 500 for the number of units of each layer in Step 1. Set the initial number of hidden layers be $\underline{n}$ and the number of hidden layers is added one by one up to $\overline{n}$. DBNs with n $(n = \underline{n}, \underline{n}+1, \ldots, \overline{n}-1, \overline{n})$ hidden layers such that the number of units of each hidden layer is 500 is evaluated based on the evaluation function (4). A DBN with highest evaluation value is selected and let n^* be the corresponding number of hidden layers.

3.3.1 Optimization of Number of Units of Each Layer

After the number of hidden layers n^* of DBN is determined, the number of units of each hidden layer should be determined. Let $(x_1, x_2, \ldots, x_{n^*})$ be a n^* dimensional solution in the solution space $X = \prod_{i=1}^{n^*}[\underline{x}_i, \overline{x}_i]$, where $x_i \in [\underline{x}_i, \overline{x}_i]$ is the number of the hidden layer. There exist numerous solutions in the space X. Therefore, the search space is divided and generate small areas to search and realize efficient

search by the following procedure. Here, let k_i be the number of division of the dimension of the solution space corresponding to the i-th hidden layer.

Let $d_i^{j_i} \equiv \left[\underline{x}_i^{j_i}, \overline{x}_i^{j_i}\right], j_i = 1,2,\dots,k_i, \underline{x}_i^1 = \underline{x}_i, \overline{x}_i^{k_i} = \overline{x}_i$ be the j-th interval of a subspace of i-th dimension x_i of the divided solution space, and $D_{\boldsymbol{j}} \equiv \Pi_{i=1}^{n^*} d_i^{j_i}$ be the divided subspace, where $\boldsymbol{j} = (j_1, j_2, \dots, j_{n^*})$. Select the center of gravity $(x_1^{\boldsymbol{j}}, x_2^{\boldsymbol{j}}, \dots, x_{n^*}^{\boldsymbol{j}})$ as a representative point of subspace $D_{\boldsymbol{j}}$. A subspace with the highest evaluation value (4) among the representative points of $\prod_i k_i$ subspaces is defined as $\boldsymbol{j}^* = (j_1^*, j_2^*, \dots, j_{n^*}^*)$.

Let a superior rectangular parallelepiped whose vertices are $x_1^{j^*-1}, x_1^{j^*+1}, x_2^{j^*-1}, x_2^{j^*+1}, \dots, x_{n^*}^{j^*-1}, x_{n^*}^{j^*+1}$ be new search space, and repeat the above steps until $\overline{x}_i^{j_i} - \underline{x}_i^{j_i} \leq k_i, \forall i$ is satisfied. If the condition $\overline{x}_i^{j_i} - \underline{x}_i^{j_i} \leq k_i, \forall i$ is satisfied, in other words, the division of the solution space is completed, the optimal solution of the network structure is searched by using taboo search described in the next section. As an example, the procedure of division of the solution space with $n^* = 2$ is shown in Figure 4.

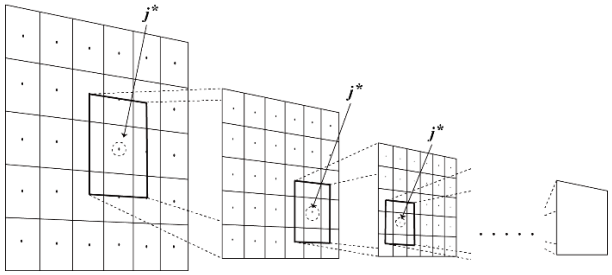


Figure 4. Division of the Solution Space ($n^* = 2$)

3.3.2 Structural Optimization by Tabu Search

In general, a pair of neural networks which have similar structures have similar performance to each other. In this paper, the optimal network structure is chosen by tabu search [6] which is one of evolutionary computation methods based on the neighbor search, from the solution subspace divided according to the above mentioned procedure.

3.4 Modularization of Structure of RBM

The DBN consists of connected plurality of RBMs and FNN. The learning process of the parameters such as connection weight and biases are performed in order from the RBMs closer to the input layer. In this paper, in order to avoid the redundancy of the structure evaluation in the structure optimization procedure of the DBN, a RBM utilizes the past learning information of another RBM which has common structure partially, input information to the RBM, number of units of a hidden layer, and the number of hidden layers, by following procedure.

For example, consider DBN1 including η_1 hidden layers which is the learning process is completed and DBN2 including η_1 hidden layers which is not completed. Let $\eta \geq \min\{\eta_1, \eta_2\}$, assume that the number of units of each hidden layer from the first layer to $(\eta - 1)$-th layer of DBN1 and DBN2 are all the same, and the number of units of the hidden layer of the η-th layer is different. Learning from the remaining the η-th to the η_2-th layer of DBN2 is performed by using the connection weights and biases of each hidden

layer from the first to the $(\eta-1)$-th layer of DBN1 in the learning process of DBN2. This mechanism improves learning efficiency of DBNs with different network structures.

4 Numerical Experiments

Caltech101 [17] are used as benchmark of image classification in many related literature. The image data of the Caltech 101 are grayscale 20×20 grids images, and each grid is scaled in the range of $[0,1]$. Images are classified into 4 categories "airplane", "cat", "face", "dolphin". There are 65 images per a category. This paper performs the following 2 kinds of experiments using Caltech101.

- 3-category classification experiment using 3 categories, “airplane”, “cat”, and “face”.
- 4-category classification experiment using 4 categories.

As a comparative methods, DBN without structural optimization, FNN, structural optimized FNN [20], k-means method are employed. Here, the number of hidden layers of a DBN without structural optimization is set to 3, and the number of each hidden layer unit is 500, 500, 2000. The number of hidden layer of a FNN without structural optimization is set to 3, and the number of hidden layer units was set to 200, 200 and 800, respectively. For a data classification problem with m categories, the number of units of an output layer is set to m, and that data is classified into a certain category corresponding to the unit such that output value is maximum in all output units. The experiments are conducted 10 trials, and the average value of classification accuracy is shown in Table 1 as experimental results.

Table 1. Image Classification Test: Result (accuracy (%))

Method	3−category	4-category
DBN with structural optimization (Proposed method)	**85.0**	**74.7**
DBN without structural optimization	77.1	62.2
FNN with structural optimization [20]	75.2	61.8
FNN without structural optimization	59.8	40.1
k-means method	58.6	37.4

From Table 1, the proposed method has the highest performance, and this experimental result indicates that the proposed method succeed to discover the appropriate structure of DBNs to increase the data classification accuracy. In the case of 3-category classification, structure of all DBNs obtained by the proposed method have 3-layer structure in all 10 trials. The average value of the number of units of the hidden layers are 454.7, 1834.5, and 2935.9 from the closer to the input layer, respectively. In each trial, numbers of units of hidden layers are similar to each other. Also, in the case of 4-category classification, DBNs with a 5-layer structure are obtained in all 10 trials. The average value of the number of units of the hidden layers are 457.0, 212.9, 2046.9, 1109.5, and 5974.9 from the closer to the input layer, respectively. Same as 3-category classification, almost same structure are obtained in all trials.

In the structure optimization of DBN by the proposed method, the solution space are divided into multiple subspaces first and solution search procedure are performed intensively in the promising regions. Such searching process can realize both diverse and intensive solution search and stably discover appropriate

structure. Additionally, it is also successful to improve the computational efficiency by modularization focusing on that the DBN has a structure in which a plurality of RBMs are superimposed.

5 Conclusion

This paper proposes structure optimization method for a DBN (Deep Belief Network) which consists of multiple RMBs (Restricted Boltzmann Machine) and a FNN (Feedforward Neural Network). The features of the proposed method are that it realizes searching both in wide range of solution areas by division of solution space and intensive search by tabu search, introduces modularization of RBMs to improve the calculation efficiency drastically by reducing the calculation amount in solution search. Numerical experiments using multiple categories image data indicates that it succeed in obtaining appropriate structure of DBN with high data classification accuracy by the proposed structural optimization method for DBNs. To develop a network structure optimization method that supports data analysis for high dimensional time series data such as voice data can be one of the future works.

REFERENCES

[1] [AgrawalEtal08] S. Agrawal, Y. Dahora, M.K. Tiwari and Y.-J. Son, "Interactive particle swarm: a Pareto-adaptive metaheuristic to multiobjective optimization," *IEEE Transactions on Systems, Man, and Cybernetics-Part A: Systems and Humans*, 38, pp. 258-277, 2008.

[2] [Bengio09] Y. Bengio, "Learning deep architectures for AI", M. Jordan eds*., Foudations and Trends in Machine Learning*, 2, Berkley, CA, USA, pp. 1-127, 2009.

[3] [Chen08] E. Chen, X. Yang, H. Zha, R. Zhang and W. Zhang, "Learning object classes from image thumbnails through deep neural networks", *International Conference on Acoustics, Speech and Signal Processing*, pp. 829-832, 2008.

[4] M. Delgado, M.P. Cuellar and M.C. Pegalajar, "Multiobjective hybrid optimization and training of recurrent neural networks," *IEEE Transactions on Systems, Man, And Cybernetics-Part B: Cybernetics*, 38, pp. 381-403, 2008.

[5] J.L. Elman, "Finding structure in time," *Cognitive Science*, 14, pp. 179-211, 1999.

[6] F. Glover and M. Laguna, *Tabu Search*, Kluwer Academic Publishers, 1997.

[7] T. Hayashida, I. Nishizaki and T. Matsumoto, "Structural optimization of neural network for data prediction using dimensional compression and tabu search," 2013 IEEE 6th International Workshop on Computational Intelligence & Applications Proceedings (IWCIA 2013), Hiroshima, Japan, pp. 85-88, 2013.

[8] T. Hayashida, I. Nishizaki and A. Suemune, "Structural optimization of recurrent neural networks using tabu search," *Journal of Japan Society for Fuzzy Theory and Intelligent Informatics*, 27, pp. 638-349, 2015.

[9] G.E. Hinton, S. Osindero and Y.-W. Teh, "A fast lerning algorithm for deep belief nets," *Neural Computation*, 18, pp. 1527-1554, 2006.

[10] G.E. Hinton and R.R. Salakhutdinov, "Reducing the dimensionality of data with neural networks," *Science*, 313, pp. 504-507, 2006.

[11] B.Q. Huang, T. Rashid and M.-T. Kechadi, "A new modified network based on the elman network," *Proceedings of IASTED International Conference on Artificial Intelligence and Application*, 1, pp. 379-384, 2004.

[12] B.Q. Huang, T. Rashid and M.-T. Kechadi., "Multi-context recurrent neural network for time series application", *International Journal of Computer, Electrical, Automation, Control and Information Engineering*, 1, pp. 3073-3082, 2007.

[13] Y. LeCun, L. Bottou, L. Bengio and P. Haffner, "Gradient-based learning applied to document recognition," Proceedings of the IEEE, pp. 1-46, 1998.

[14] M.I. Jordan, "Attractor dynamics and parallelism in a connectionist sequential machine," *Proceedings of the 8th Annual Conference of the Cognitive Science Society*, E. Cliffs, NJ: Erlbaum, pp. 531-546, Reprinted in IEEE Tutorials Series, New York: IEEE Publishing Services, 1990.

[15] H. Katagiri, I. Nishizaki, T. Hayashida and T. Kadoma, "Multiobjective evolutionary optimization of training and topology of recurrent neural networks for time-series prediction," *The Computer Journal*, 55, pp. 325-336, 2011.

[16] Y. Liu, S. Zhou and Q. Chen,``Discriminative deep belief networks for visual data classification'', *Pattern Recognition*, 44, pp. 2287-2296, 2011.

[17] F.-F. Li, R. Fergus and P. Perona, "Learning generative visual models from few training examples: An incremental Bayesian approach tested on 101 object categories", *Computer Vision and Image Understanding*, 106, pp. 59-70, 2007.

[18] [Martens10] J. Martens, "Deep learning via Hessian-free optimization," *Proceedings of the 27th International Conference on Machine Learning*, Haifa, Israel, 2010.

[19] [McCullochPits43] W.S. McCulloch and W. Pitts, "A logical calculus of the ideas immanent in nervous activity," *Bulletin of Mathematical Biophysics*, 5, pp. 115-133, 1943.

[20] M. Nishida, T. Hayashida, I. Nishizaki, and S. Sekizaki, "Structual optimization of neural network and training data selection based on SOM," *2014 IEEE SMC Hiroshima Chapter Young Researchers' Workshop Proceedings*, pp. 117-120, 2014.

[21] [Park13] D.-C. Park, "Structure optimization of bilinear recurrent neural networks and its application to ethernet network traffic prediction," *Information Sciences, 237, pp. 18-28, 2013.*

[22] [Prokhorov98] D.V. Prokhorov, E.W. Saad and D.C. Wunsch, "Comparative study of stock trend prediction using time delay, recurrent and probabilistic neural networks," *IEEE Transactions on Neural Networks*, 9, pp. 1456-1470, 1998.

[23] [QuocEtal11] V. Le Quoc, J. Ngiam, A. Coates, A. Lahiri, B. Prochnow and A.Y. Ng, "On optimization methods for deep learning", *Proceedings of the 28 th International Conference on Machine Learning*, Bellevue, WA, USA, 2011.

[24] [SrivastraSalakhutdinov12] N. Srivastava and R. Salakhutdinov, "Multimodal learning with deep Boltzmann machines," *Representation Learning Workshop*, 2012.

[25] R. Salakhutdinov and G. Hinton, "Deep Boltzmann machines," Proceedings of the 12th International Conference on Artificial Intelligence and Statistics, Clearwater Beach, Florida, USA, pp. 448-455, 2009.

[26] [ScullerEtal11] B. Sculler, S. Steidl, A. Batliner, F. Schiel and J. Krajewski, *The interspeech 2011 Speaker State Challenge,* Florence, Italy, 2011.

[27] N. Srivastave, G. Hinton, A. Krizhevsky, I. Sutskever and R. Salakhutdinov, "Drop out: A simple way to prevent neural networks from overfitting", *Journal of Machine Learning Research*, 15, pp. 1929-1958, 2014.

[28] [VriesPrincipe90] B. de Vries and J.C. Principe, "A theory for neural networks with time delays," *Proceedings of the 1990 Conference on Advances in Neural Information Processing Systems,* 3, pp. 162-168, 1990.

[29] [DoSSweb] "DoSS@d", http://mo161.soci.ous.ac.jp/@d/index.html, (accessed 2017-12-13).

[30] [UCweb] "UC Irvine Machine Learning Repositor", http://archive.ics.uci.edu/ml/, (accessed 2017-12-13).

SOCIETY FOR SCIENCE AND EDUCATION
UNITED KINGDOM

TMLAI TRANSACTIONS ON MACHINE LEARNING AND ARTIFICIAL INTELLIGENCE

Volume 6 No. 1
ISSN 2054-7390

Comparison Performance Evaluation of Modified Genetic Algorithm and Modified Counter Propagation Network for Online Character recognition

Adigun Oyeranmi. J[1], Fenwa Olusayo D[2], Babatunde. Ronke. S[3]
[1]Department of Computer Technology, School of Technology, Yaba College of Technology, Yaba, Lagos Nigeria
[2]Department of Computer Science and Engineering, Faculty of Engineering and Technology, Ladoke Akintola University of Technology, Ogbomoso, Nigeria.
[3]Department of Computer Science, College of Information and Communication Technology, Kwara State University, Malete, Nigeria.
odfenwa@lautech.edu.ng, ranmi.adigun@yabatech.edu.ng, ronke.babatunde@kwasu.edu.ng

ABSTRACT

This paper carries out performance evaluation of a Modified Genetic Algorithm (MGA) and Modified Counter Propagation Network (MCPN) Network for online character recognition. Two techniques were used to train the feature vectors using supervised and unsupervised methods. First, the modified genetic algorithm used feature selection to filter irrelevant features vectors and improve character recognition because of its stochastic nature. Second, MCPN has its capability to extract statistical properties of the input data. MGA and MCPN techniques were used as classifiers for online character recognition then evaluated using 6200-character images for training and testing with best selected similarity threshold value. The experimental results of evaluation showed that, at 5 x 7 pixels, MGA had 97.89% recognition accuracy with training time of 61.20ms while MCPN gave 97.44% recognition accuracy in a time of 62.46ms achieved. At 2480, MGA had 96.67% with a training time of 4.53ms, whereas MCPN had 96.33% accuracy with a time of 4.98ms achieved. Furthermore, at 1240 database sizes, MGA has 96.44 % recognition accuracy with 0.62ms training time whereas MCPN gave 96.11% accuracy with 0.75ms training time. The two techniques were evaluated using different performance metrics. The results suggested the superior nature of the MGA technique in terms of epoch, recognition accuracy, convergence time, training time and sensitivity.

Keywords: Character recognition, normalization, Modified genetic Algorithm(MGA), Modified counter propagation network (MCPN), Generation gap

1 Introduction

Characters recognition have been explored for many years and require many applications capabilities such as bank processing, person authentication, manufacturing of plastic cards, food industry etc. So far, online character recognition remains an open problem, in spite of a dramatic boost of research in this field and the latest improvement in recognition methodologies. Character recognition is an ongoing research which has motivated researchers from various aspect of human endeavors such as image processing, computer

DOI: 10.14738/tmlai.61.3976
Publication Date: 06th January, 2017
URL: http://dx.doi.org/10.14738/tmlai.61.3976

vision and machine learning [1]. Online character recognition system is the transformation process, that will allow extraction of input characters, from character image database to digitalize and translates the handwritten text into a machine editable form [2]. Due to global security threat, person authentication, and retrieving text, there is need of adopting techniques that could enhance the recognition performance of the system. Features extraction techniques and classifiers had been researched upon to have contributed to the performance of character recognition. At the same time, evaluation of the system with some selected metrics such as recognition accuracy, sensitivity, computation time etc. have been considered. Recognition of handwriting in online mode is usually accomplished using temporal spatial information obtained from the operativeness of a stylus on the surface of an electrostatic or electromagnetic tablet. Timing knowledge is accessible from Coordinate information of strokes [3]. The most popular method used in online character recognition is backpropagation network. Genetic algorithm was used to optimize backpropagation architecture and used to find an optimal solution in complex problems [4] that mimic the principle of natural genetics and natural selection that competes for survival to make up the next generation of population [5][6]. However, backpropagation neural network weakness are slow convergence and long time for recognition of characters. Character recognition consists mainly of four stages: data acquisition, pre-processing, feature extraction and classifications [7]. A modified counter-propagation (MCPN) algorithm is one the neural network classifiers with a supervised and unsupervised training which are closely related to the Nearest-Neighbor classifier. The network essentially functions as a nearest match lookup table. MCPN network has an interesting capability to extract the statistical properties of the input data, and can usually be trained very rapidly [8]. This study used statistical and structural features to extract features at MGA from the image characters and Integration of Geometrical and Statistical features had been used as feature extraction at MCPN. The classification of the developed system was carried out using both techniques. Hence, the focus of this paper is to evaluate performance of MGA and MCPN classifiers to recognize character images and establish the efficiency of the two techniques.

2 Related Work

High recognition accuracy and less training time are the bane of online character recognition. This could be done based on the feature extraction techniques and classifiers algorithm applied. [9] compared two different algorithms with combination of genetic algorithm with backpropagation network(GABPN) and correlation method with genetic algorithm(GA) to achieve both accuracy and training swiftness for recognizing alphabets. The performance evaluation of the two algorithms showed that correlation with GA technique gave the best recognition accuracy of 95-97%. [10] implemented two different techniques (structural and statistical) and MGA was used to extract optimized feature subset of the character for classification task. Two classifiers (C1 and C2) were formulated from MGA-MOBP. The overall results indicated that C2 achieved better performance than C1. A modified CPN was proposed by Fenwa in 2012 for online character recognition, which was faster than the existing conventional CPN. In the modified CPN model, character parameters were not trained like backpropagation architecture which is an interactive method that suffer long training. The system was experimented on different handwritten characters. The performances of proposed techniques were evaluated in terms of recognition time and recognition rate. [11] implemented a research on integration of PSO with hybrid of Counterpropagation and modified Optical Backpropagation Neural Network (COMOB) model to enhancement the

performance of the classifier in terms of recognition accuracy and recognition time. Results of the proposed system show that there was a better performance of recognition accuracy. However, [12] presented a paper for recognition of English characters based on features derived from partitioning the characters into non-overlapping cells. This work applied a dynamic window sliding for feature vector generation and uses four passes of window that led to the creation of a 30-element feature vector. A neural network (multi-layered perceptron) was used for classifying the 26 alphabets of the English language. The average recognition rate achieved was about 97.33%. [13] developed a technique for recognition of an offline handwritten character using grip approach. Extracted features are trained by neural network as classifier of the character in classification stage. The overall experimental result in recognition rate was 96.9%. [14] designed handwritten digits recognition system with combination of genetic algorithm. The proposed neural network was trained with data set containing 25 sample images of each digit. The average accuracy result of the proposed system was 99% and this was then compared with currently existing techniques with various constraints.

3 Methodology

The paper focuses on implementing MGA and MCPN classifiers on image dataset, evaluates the recognition accuracy, training time performance of the algorithms and comparing their results. The images recognition process consist of four stages: image acquisition, image preprocessing, feature selection &extraction and the classification.

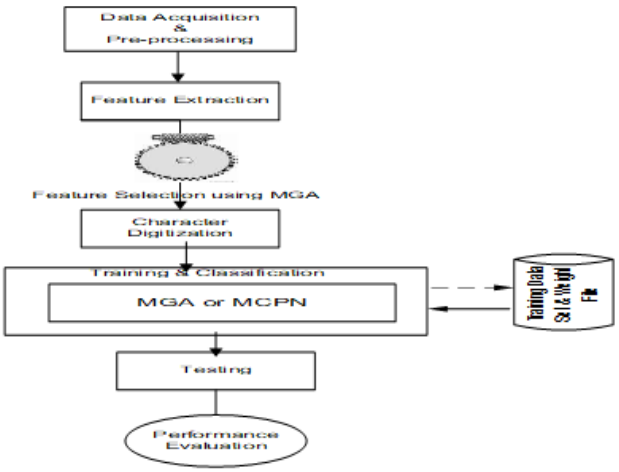


Figure 1: The Block Components of Comparison Evaluation of MGA and MCPM

3.1 Acquisition and Preprocessing

In this article, the first step is the character images were acquired using a pen digitizer from datasets of 6200 samples collected by (Adigun et al., 2016) was used for training. Three preprocessing techniques were employed: binarization, extreme coordinate measurement and grid resizing were used to convert into binary form, measure extreme coordinate of the space and matrix standard respectively. The normalization the images to remove the noise, gaps and character enhancement. Finally, classification of individual images based on input image was tested using MGA and MCPN classifiers. The extracted feature provided the characteristics of input type to classifier by considering the description of the relevant properties of image into feature space. The block diagram procedures to achieve this research work is shown in figure 1

(a) (b)

Figure 2a & b: Preprocessing System Result

3.2 Feature Selection and Extraction

This research work conducts a comparison evaluation of MGA and modified CPN techniques to recognize online characters. Summarily, MGA and MCPN were used as classifiers. The more efficient of these two techniques was checked and their recognition rate were evaluated. The character image in Fig. 2a is trained by undergoing preprocessing stage as shown above through conversion into grayscale and histogram equalization. This is to enhance the intensity of the images without losing any important information. The paper used hybrid (Struct-Statistical) Feature Extraction Algorithm developed by [4] to complement each other and reduce errors as shown figure 3. The MGA as feature selection was used to select optimized features subset to reduce redundant features which improves the recognition accuracy of the characters. The structural features used in this paper consists of stroke information, projection and invariant moments. This measures the pixel distribution around the centre of gravity of the character and allow capturing the global character shape information that improved the recognition accuracy. To calculate moment invariants as a

$$M_{pq} = \int \int \mathrm{x}^p y^q f(\mathrm{x},\mathrm{y})\mathrm{dxdy}$$

where f (x, y) is the intensity function representing the image, the integration is over the entire image and the F (x, y) is same function of x and y for example $x^p y^q$, or a sin(xp) and cos(yq). This is to determine the position, size and orientation of the character images. The statistical features adopted in this paper was hybrid Zoning algorithms of modified Image Centroid and Zone-based (ICZ) and modified Zone Centroid and Zone-based (ZCZ) distance metric feature extraction based on model developed by Fenwa *et al.,* (2012) was adopted see below:

Input: Pre-processed character image
Output: Features for Classification and Recognition
Begins
Step 1: Divide the input image into 25 equal zones
Step 2: Compute the input image centroid
Step 3: Compute the distance between the image centroid to each pixel present in the zone
Step 4: Repeat step 3 for the entire pixel present in the zone
Step 5: Compute average distance between these points
Step 6: Compute the zone centroid
Step 7: Compute the distance between the zone centroid to each pixel present in the zone.
Step 8: Repeat step 7 for the entire pixel present in the zone

Step 9: Compute average distance between these points
Step 10: Repeat the steps 3-9 sequentially for the entire zones
*Step11: Finally, 2*n (50) such features are obtained for classification and recognition. Ends*

As a result of global security threat and criminal activities, there is need of adopting techniques that could enhance the recognition performance of the system. Features extraction algorithms and classifiers had been researched upon to have contributed to the performance of the system.

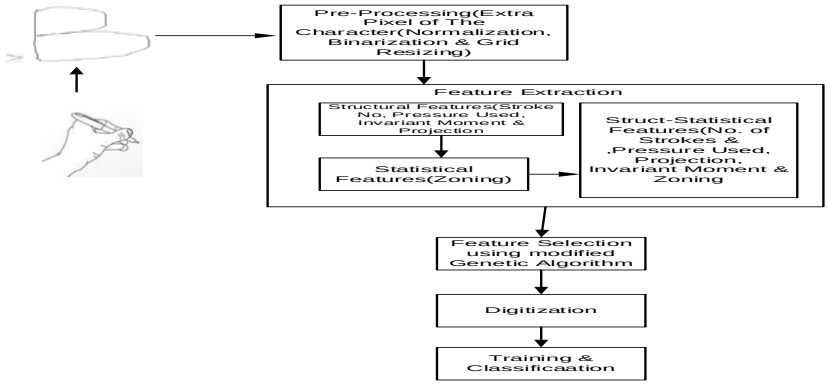


Figure 3: Developed Feature Extraction Model source: (Adigun et al., 2016)

3.3 Character Classification Method

The classification stage comprises of the training stage and the testing stage. The output of the features extracted from the training and testing images were saved and fed into the MGA and MCPN classifiers for comparison. The feature vectors generated are then compared with stored pattern and find out the best matching class for input. However, evaluation of the developed system with some selected metrics such as recognition accuracy, sensitivity, training time and computation time etc. have been considered. A threshold is determined by the continuous modification of the threshold until significant accuracy is observed. The threshold set implies correct match of a character which dependent on the minimum distance is less or equal to a threshold. The testing phase of the implementation was done in a straightforward manner. The program was coded into modular parts and the same routines of loading, analyzing, and computation of network parameters of input vectors in the training phase were reused in the testing phase as well. All the 6200 images of English characters acquired were used for training in this research work. The testing (900) images were introduced by 25 different persons one by one to see by if it will be recognized by the character recognition system.

Algorithm:

The basic steps in testing input images for characters can be summarized as follows:

i. Start
ii. Load character from the database
iii. Convert images into grayscale, then map into an input vector form and normalize image vector
iv. Apply Stat-struct technique for feature extraction and MGA for feature selection and store features
v. Apply MGA or MCPN classifier and compute output
vi. Match the introduced image with the ones in the database template

vii. Classify the image into (CR, FR and RF)

viii. Convert the binary output to the corresponding character and display to a message box

ix. Test the next character image and repeat until all characters were visited

The simulation tool used for this research is C# programming language, an objected oriented programming language, derived from C++. It supports window based application, 64bits operating system, 8.00GB RAM and run under Windows 7 Professional Operating system on Intel® Core(TM) i5-42000M CPU @2.50GHz processor and built for .Net platform. The performance of the results was evaluated using generation gap, database sizes and overall accuracy (correct recognition(CR), false recognition(FR) and recognition failure(RF) to determine the performance and accuracy of the system. The sensitivity of the character recognition is determined by

$$\text{Sensitivity} = \frac{CR}{CR + FR}$$

4 Results and Discussion

The performance of MGA and MCPN on trained and recognized character were measured to determine its efficiency in terms of training time, epoch, convergence time, sensitivity and recognition accuracy,540 images were used for testing. The results obtained by using MGA and MCPN classifiers with respect to metrics mentioned above were evaluated as follows: Training time was estimated for different database sizes of the same 5*7-pixel resolution. The result obtained was as shown in Table 1, in which the training time of MGA was smaller than MCPN due to its stochastic capability and ability to achieve feature selection reduction. The average training time of the developed system is very much less when compared with MCPN classifier. The MGA recorded 0.62 ms averagely with 1240 database size, 4.53 ms with 2480 database size, 61.20 ms with 6200 database sizes whereas MCPN at 1240 database size had 0.75ms averagely, at 2480 the training time of 4.98ms and 62.46 with 6200 database sizes.

Table 1 Evaluation variation of database size on training time (milliseconds) accuracies of MGA and MCPN

Character Samples		MCPN				MGA		
	Training time	CR	FR	RF	Training time	CR	FR	RF
1240	0.75	865	30	5	0.62	868	27	5
2480	4.98	867	27	4	4.53	870	26	4
6200	61.20	877	21	1	61.20	881	18	1

From Table 2, the recognition accuracy obtained using MGA with different generation gap threshold values of 0.1, 0.3, 0.5 and 0.7and the study revealed that MGA has better performance in convergence time and accuracy than MCPN as computed in section 3.1. The recognition accuracy at 6200 database sizes with MGA recognition accuracy of 97.89% at 0.1, 97.34% at 0.3, 96.81% at 0.5 and 96.18 % at 0.7 whereas, MCPN obtained 97.44 % at 0.1, 97.1% at 0.3, 96.37% at 0.5 and 95.93% at 0.7 generation gap. Table 2 deduced the performance of MGA against MCPN different database sizes of 1240, 2480, and 6200.The average convergence time reported at 6200 database sizes with MGA are 193.41ms, while MCPN produced 199.01 ms. However, MGA was noticed to classified faster because of its feature selection

reduction capability than MCPN. Table 2 shows that values generated in terms of convergence time by MGA took less time than that of MCPN at different database sizes level employed. Furthermore, Table 2 shows that MGA in terms of sensitivity at different sizes had an increase value compared with MCPN. The MGA generated high recognition accuracy at a less time with MCPN. The MCPN also has its capability to extract statistical properties of the input data. Finally, the results of evaluation showed that MGA distinctively outperformed MCPN in terms of recognition accuracy, faster convergence time and less training time.

Table 2 showing combined results with MGA and MCPN at best selected threshold value

.

Database Size	Algorithm	Epoch	Convergence Time (milliseconds)	Sensitivity (%)	Accuracy (%)
1240	MGA	403	49.79	96.98	96.44
	MCPN	413	49.98	96.65	96.11
2480	MGA	771	201.19	97.1	96.74
	MCPN	780	200.46	96.9	96.33
6200	MGA	1838	193.41	98.0	97.89
	MCPN	1898	199.01	97.66	97.44

It was shown in Table 2 that the higher the database sizes, the better the recognition accuracy due to fact the network training was able to attribute the test character to larger character sample in the vector space. Usually, the complex and large sized input sets require a large topology network with more number of iterations (Epochs). The epochs are directly proportional to the training time, this implies that the larger the image size, the more the training time. However, this would also imply more number of iterations were required to reach its optimal state.

5 Conclusion

The study has presented a comparative performance evaluation of modified genetic algorithm and modified counter propagation Neural Network as well as their application with online handwritten character recognition. MGA, an optimization technique was used to extract salient features from online handwritten character images at the initial stage before the application of MGA and MCPN classifiers. This was done to reduce the insignificant features for enhancing and efficient character recognition. This research work was implemented and evaluated in order to determine their effectiveness. The results suggest that MGA recorded better convergence time and recognition accuracy than MCPN. In view of the MGA would find the near global optimal solution in a large solution space quickly. It can also be used extensively in many application areas, such as image processing, pattern recognition, feature selection, criminal activities and machine learning.

REFERENCES

[1] Abed Majida Ali, Ismail Ahmad Nasser and Hazi Zubadi Matiz., *Pattern recognition Using Genetic Algorithm,* International Journal of Computer and Electrical Engineering, 2013. 2(3): p. 583-588.

[2] Adigun J.O., Fenwa O.D., Omidiora E.O., Olabiyisi S.O. Optimized features for genetic based neural network model for online character recognition. British Journal of Mathematics & Computer Science, 2016, 14(6):1-13.

[3] Herekar Rachana R. , S. R. Dhotre, *Handwritten Character Recognition Based on Zoning Using Euler Number for English Alphabets and Numerals*. In IOSR e-ISSN: 2278-0661, 2014, 16(4), p.75-88.

[4] Adigun J.O., Omidiora E.O., Olabiyisi S.O., Fenwa O.D., Oladipo O, Rufai M.M. *Development of a Genetic based neural network system for online character recognition*. International Journal of Applied Information Systems (IJAIS), 2015, 9(3), p.1-8.

[5] Noaman, Khaled M.G, Saif, Jamil Abdulhameed M., Alqubati, Ibrahim A.A, *Optical Character Recognition Based on Genetic Algorithms*, Journal of Emerging Trends in Computing and Information Sciences, 2015, 6(4), p.204-208.

[6] Yeremia Hendy, Niko Adrianus Yuwono, Pius Raymond and Widodo Budiharto, *Genetic Algorithm and Neural Network for Optical character recognition*, Journal of computer science, 2013, 9 (11),1435-1442.

[7] Fenwa, O.D., Omidiora, E.O. and Fakolujo, O.A. *Development of a Feature Extraction Technique for Online Character Recognition System*, Journal of Innovative System Design and Engineering, International Institute of Science, Technology and Education, New York, USA, 2012, 3(3), p.10-23.

[8] Kumar D, Rai CS, Kumar S. *An experimental comparison of unsupervised learning techniques for face recognition*. International Journal of Computer and Information Science and Engineering. 2007,1(3):158-166.

[9] Inamdar Farhan, and Bagal, S. B. (2016): *Comparative Study of Optical Character Recognition Techniques,* International Journal of Innovative Research in Computer and Communication Engineering, 2016,4(11), p.19831-19837

[10] Oyeranmi Adigun, Elijah Omidiora and Mohammed Rufai, *Modified Genetic Algorithm Parameters to Improve Online Character Recognition*, British Journal of Applied Science & Technology, 2016, 18(5): 1-8.

[11] Olusayo D. Fenwa, Funmilola A. Ajala and Alice O. Oke, *A PSO-Based Modified Counter Propagation Neural Network Model for Online Handwritten Character Recognition System*, International Journal of Emerging Technology and Advanced Engineering, 2014, 4(6), p.768 -776

[12] Biswas, Mithun and Parekh, Ranjan, *Character Recognition using Dynamic Windows,* International Journal of Computer Applications, 2012, 41(15):47-52.

[13] Chaudhari Prasad P, Sarode KR. *Offline handwritten character recognition by using grid approach.* International Journal of Application or Innovation in Engineering & Management, 2014, 3(4), p.71-73.

[14] Kaur Tarandeep and Chabbra Amit, Genetic Algorithm Optimized Neural Network for Handwritten Character Recognition, International Journal of Computer Applications, 2015, 119(24), p.22-26