HIERARCHICAL LOCATION CLASSIFICATION OF TWITTER USERS WITH A CONTENT BASED PROBABILITY MODEL

by

Mounika Nukala

Submitted in partial fulfilment of the requirements
for the degree of Master of Computer Science

at

Dalhousie University
Halifax, Nova Scotia
March 2015

To my beloved parents Mr. Upender Reddy Nukala and Mrs. Vasumathi Reddy, who have always inspired me to live an independent life with dignity and respect.

TABLE OF CONTENTS

LIST OF TABLES

LIST OF FIGURES

ABSTRACT

Extraction of geographical information from the content is gaining importance due to the huge growth of textual data on social media and a phenomenal increase in the location based personalized services. Knowledge of online user's content and location enables location based personalized services. Existing approaches to predict the location of Twitter users have not incorporated geographical information from geo-tagged tweets and are content driven only. A hybrid approach using a combination of hierarchical location classification and tweet geo-location is proposed to predict location based on tweet content and metadata. Our approach uses an ensemble of content based statistic classifiers trained on words, hashtags, places and heuristic classifiers for place names, geo-coordinates in tweets to predict locations at different granularities like time zone, state and city. Experimental results suggest that our hybrid approach achieves a city prediction accuracy of 70.7% for Twitter users and outperforms the existing hierarchical location classification methods.

LIST OF ABBREVIATIONS USED

MNB	MultiNomial Naïve Bayes
US	United States
ID	Identifier
API	Application Programming Interface
URL	Uniform Resource Locator
PMI	Point wise Mutual Information
BC	Betweenness Centrality
JI	Jaccard Index
TI	Tversky Index
IDE	Integrated Development Environment
NLTK	Natural Language Tool Kit
NLP	Natural Language Processing
EST	Eastern Standard Time
CST	Central Standard Time
PST	Pacific Standard Time
MST	Mountain Standard Time

ACKNOWLEDGEMENTS

I would like to express my sincere gratitude to my supervisor Dr. Srini Sampalli for his tremendous support, encouragement and patience. His dedication and enthusiasm towards research is truly inspirational. It would not have been possible for me to complete my research without his valuable suggestions and proper guidance.

Thanks to the entire team at MYTech Lab for their support and making my stay a memorable experience. I would also like to thank Raghav Sampangi and Mathavan Kumar for helping me through all the stages of research.

I would like to thank especially my parents, Upender Reddy Nukala and Vasumathi Reddy, my husband Vikas Reddy Merugu, my brother Varun Kumar and my parents-in-laws, Merugu Karunaker Reddy and Sarala Merugu for their endless encouragement and unconditional support.

CHAPTER 1 INTRODUCTION

1.1 MOTIVATION

Over the past few years, a phenomenal growth has been observed in the user bases of the micro blogging social media such as Twitter, which has more than 200 million user accounts in various geographical locations [1]. Micro blogging is described as a platform where users post the details of their daily activities, messages for public or specific friends and discuss about the events, political and social trends at the local or global level in less than 140 characters. The huge growth of the user base and popularity of Twitter social media has led to a significant growth in applications to adapt the services they offer based on the location of users [2]. Location of users enables targeted advertisements, e.g. mobile couponing based on the user's geographical scope and posting related news recommendations, targeted public health Web mining system that analyzes tweets for health monitoring and local emergency detection system that analyzes tweets about earthquakes and fires [3]. The online location of users is integral to these applications as it is important for them to target the appropriate users and provide personalized services.

Twitter allows users to provide a text description of their locations in their Twitter profiles. The reliability of utilizing the self-reported locations present in the user twitter profiles for location based applications is very important before these applications make use of the user's geographic information in order to provide specialized services. The user's behavior of reporting locations in the Twitter profiles is investigated by Hecht et al. [4] in their research work to understand the reliability of the available geographic information and is presented in Figure 1. From their study, they have demonstrated that only 66% users have entered valid geographic information in twitter profiles, 18% users have not entered any information and 16% users have entered irrelevant information.

Among the 66% users who have specified geographic information in twitter profiles, it is observed that only 64% and 20% users have mentioned locations at city and state levels respectively. The study of the user's behavior indicates that the self-reported locations of twitter users are ad hoc and unreliable [3, 4, 5,7]. Unreliability of the locations specified by twitter users enforces the need for the researchers to look out for other ways to extract location information. These issues have paved the way for researchers to study and analyze the content posted by users in order to extract geographic information.

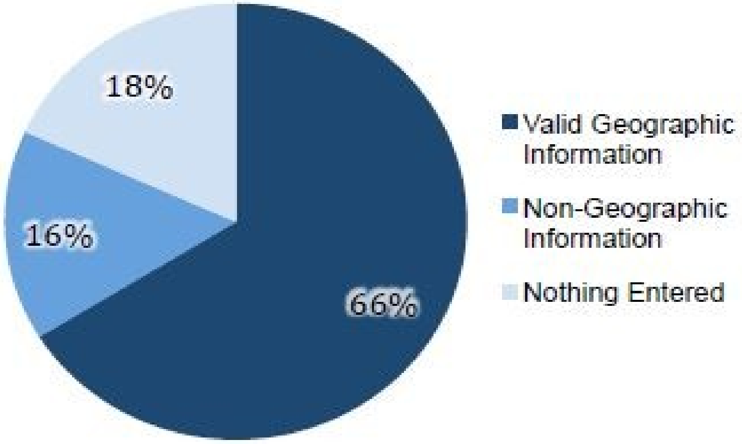


Figure 1 Distribution of manually entered location field data in twitter profiles [4]

1.2 OVERVIEW OF EXISTING SOLUTIONS

Content posted by the users tends to have geographical references or geospatial expressions that are specific to locations. Efforts are made by researchers to leverage geographical references in the online content to infer a location for users by identifying geographical entities and analyzing geospatial expressions in the content. This approach has limitations due to its inability to capture the informal geospatial expressions like Brissie for Brisbane and capture the non-geospatial terms associated with specific locations like ferry for Seattle or Sydney [6].

Other than the location inferring approaches for twitter users using geographical gazetteers, there are various approaches like content based probabilistic models [3,4,5,7] and location inference models via social network [10] proposed by many researchers. Hecht et al. [4] built a MNB probabilistic model based on user tweets to predict the location of Twitter users at the country and state levels. They have extracted selective terms for their proposed model by choosing top 10000 terms which have exhibited the maximum conditional probability of terms for states and countries. They have achieved 89% and 27% prediction accuracy at the country and state levels respectively. Predicting cities for twitter users is challenging than predicting at the state and country levels as the number of states or countries considered for inferring a location via models are less than the number of cities. Cheng et al. [5,7] demonstrated the importance of identifying local words for particular locations and accounting for less tweets in a city with less population. They have proposed a method to identify the focus and dispersion of the words using the geographical distribution of words from the geo-tagged tweets and then suggested smoothing models to overcome the issue of tweet sparseness for less populated cities. Similar to Hecht et al. [4] work, they have also built a content based probabilistic model on words selected by the suggested methods and estimated the cities for 51% of users within 1000 miles of their actual locations.

Han et al. [3] explored various feature selection methods to identify the best feature selection method for extracting location indicative words from the content available on twitter. They have also employed a naive Bayes probability model which is trained on the extracted location indicative words picked by their best feature selection method to improve location prediction accuracy. They have estimated cities for 45% of users within 100 miles of their actual locations. A hierarchical location classification model based on the content is explored by Mahmud et al. [8] to first predict time zone for a user based on the tweeting behavior of sampled twitter users in the US time zones and then utilize an ensemble of content based probabilistic classifiers to predict city or state for user by confining the search to the locations present in the predicted time zone only. They have achieved a prediction accuracy of 73% and 58% at time zone and city level respectively.

Krishnamurthy et al. [9] proposed Wikipedia knowledge enabled unsupervised approach to predict location for users. They have identified entities or local topics for a city listed as a hyperlink on the Wikipedia page for that city and came up with four scoring methods to measure the correlation shared between a city and its entities. They have calculated the score by aggregating the scores of the matched entities to find the city with a maximum score as the location for the user. They have achieved a city prediction accuracy of 54.48% with their best scoring method. Jurgens et al. [10] proposed a location inference model via social network to predict location for users. They use a spatial label propagation method to assign locations of nearest neighbors to users with no locations. They have considered various methods to identify nearest neighbors and evaluated their performance, and picked geographic median as their best method to find the nearest neighbor for a user. They have estimated cities for 50% of users within 10 km of actual locations.

A significant amount of work has been performed by researchers to predict locations of twitter users at country, time zone, state and city levels. The task of achieving higher prediction accuracy at the city level for users based on content still remains as a challenge.

1.3 RESEARCH PROBLEM AND OBJECTIVES

The objective of the proposed work is to estimate the home location of a Twitter user based purely on the content of tweets by adopting a hybrid approach of the available location classification methods [3,8] to enhance location prediction accuracy for twitter users. The proposed work is considered as a text classification task to address the location classification for twitter users.

The proposed work performs hierarchical location classification for Twitter users based on an ensemble of content based statistical and heuristic classifiers to estimate the location of Twitter users at different granularities such as time zone, state and city. The idea is to estimate a user's location based on a collection of tweets since the content of a single tweet posted by a user may not have sufficient information about user location. In the proposed location classification for twitter users, first a time zone is predicted for a user and then search is confined to only states present in the predicted time zone to predict state for a user followed by searching cities present in the predicted state to predict city for a user. The task of predicting location classification at different granularities is achieved by training the text classifiers on the MNB algorithm at time zone, state and city level from the tweets of users with known locations and using these classifiers at different hierarchy levels to estimate the city for a Twitter user.

1.4 THESIS OUTLINE

The rest of the thesis is organized as follows. Chapter 2 presents the background to social media, text analytics in social media and describes the related work for inferring twitter users' locations using content based probabilistic models. Chapter 3 presents the proposed work and the architecture of the location classification for twitter users. Chapter 4 presents the design and implementation details of the proposed architecture. Chapter 5 describes the scenarios considered in evaluating the performance of the proposed location classification for twitter users, results and evaluation. Chapter 6 concludes the thesis work and describes the possible future work in brief.

CHAPTER 2 BACKGROUND AND RELATED WORK

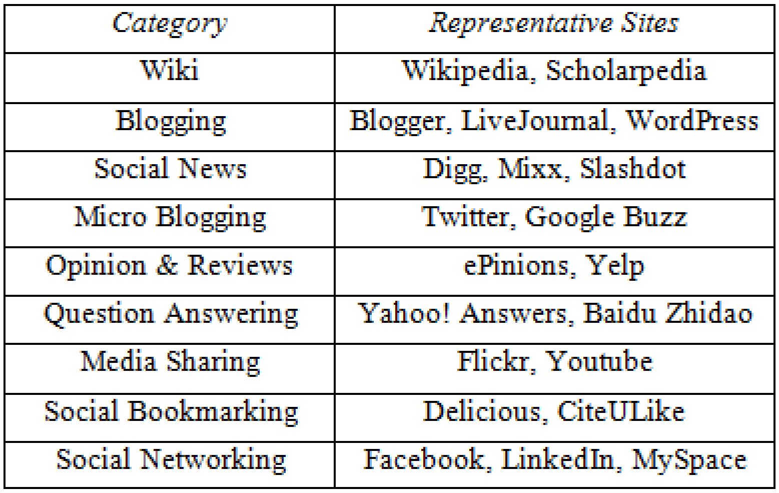
In this section, an overview of the social media, text analytics in social media, the challenges faced with the text analytics in the social media, introduction to Twitter and its API's are discussed in detail in order to present a background to the research problem. The related work on the research problem of analyzing the online content to predict the location of users is also described in detail to understand the models and approaches adopted by various researchers.

2.1 SOCIAL MEDIA

Social media provides a web-based and mobile platform for the individuals and the communities to interact, share and exchange information. The proliferation of social media applications such as micro blogs, blogs, social networks, video and photo sharing sites enables the social interaction on one side and on the other side the hugely generated data from the social media provides rich information about the individuals and their collective behavior [11]. The listed social media applications in the Table 1 differ in terms of the data they offer such as text, image and video. While the social media such as blogs and micro blogs facilitate the users to post frequently and create textual data, the media sharing services such as YouTube and Flickr facilitate the users to incorporate the videos and images in their posts.

The rich information from the social media can be exploited for the research and commercial purposes. Especially the textual content posted by the users is a potential source of data for the various applications like public polls, news recommendation and epidemic dispersion. Before exploiting the raw data available in the social media and utilizing for various purposes, it is important to perform the mining and analyzing of the user generated textual content and extract knowledge from the raw data. This is where the significance of the text analytics in social media comes into the picture.

Table 1 Types of Social Media [11]



Category	*Representative Sites*
Wiki	Wikipedia, Scholarpedia
Blogging	Blogger, LiveJournal, WordPress
Social News	Digg, Mixx, Slashdot
Micro Blogging	Twitter, Google Buzz
Opinion & Reviews	ePinions, Yelp
Question Answering	Yahoo! Answers, Baidu Zhidao
Media Sharing	Flickr, Youtube
Social Bookmarking	Delicious, CiteULike
Social Networking	Facebook, LinkedIn, MySpace

2.2 TEXT ANALYTICS

Text Analytics is the process of applying statistical and machine learning techniques to analyze unstructured text and extract relevant information to classify documents [12]. Text Analytics in social media gained more importance due to the increase in the textual data available on the social media. Thus, the combination of the social media and text analytics techniques can be used to find the opinion of users regarding any subject, political or social trends, news recommendations, local events and targeted advertising [13].

2.2.1 Architecture of text analytics framework

In this sub-section, the architecture of a general text analytics framework is presented and also the way a text document is processed by means of text analytics framework is described. A general text analytics framework as in Figure 2 comprises of mainly three phases. They are text preprocessing, text representation and knowledge discovery [11].

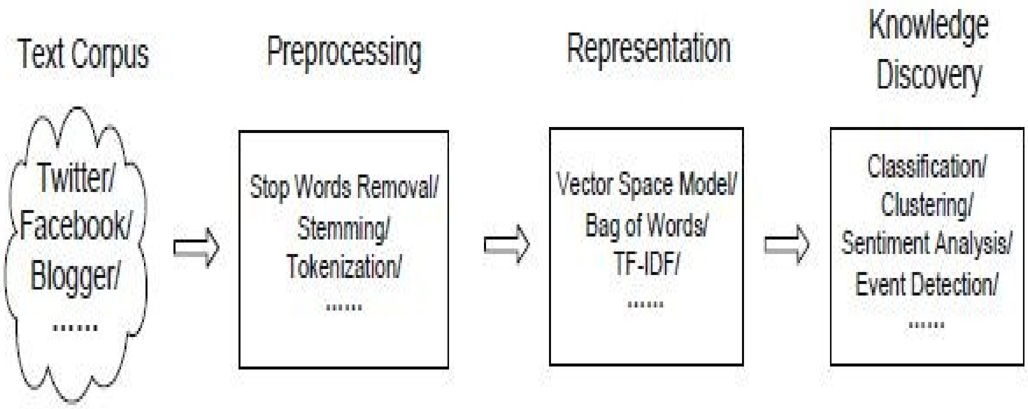


Figure 2 General Framework for Text Analytics [11]

Preprocessing

The main objective of the preprocessing phase is to extract the significant words from a document and discard the words in a document that does not distinguish between the available set of documents. There are several methods that are used in the process of text preprocessing such as tokenization, stop word removal, stemming [11,14].

Tokenization

Tokenization is the process of splitting a text document into a set of individual words called unigrams or a sequence of words called n-grams by removing the white spaces between the words. Figure 3 is an example of unigram tokens.

Stemming

Stemming is the process of finding the root or stem of a word. English words such as ‘look’, ’looks’, ’looking’, ’looked’ can be stemmed to the word ‘look’. The objective of this process is to find the exact matching stems and thereby reduce the number of words, save memory and time. Stemming of words present in the text can be performed by using stemming algorithms such as Open NLP. Sample stemming of a word is presented in the Figure 4.

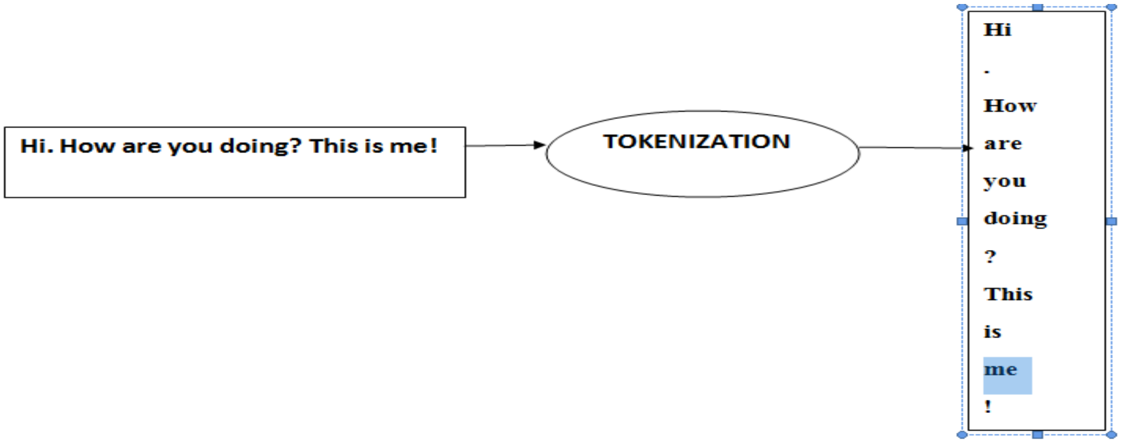


Figure 3 Sample Tokenized Tweet

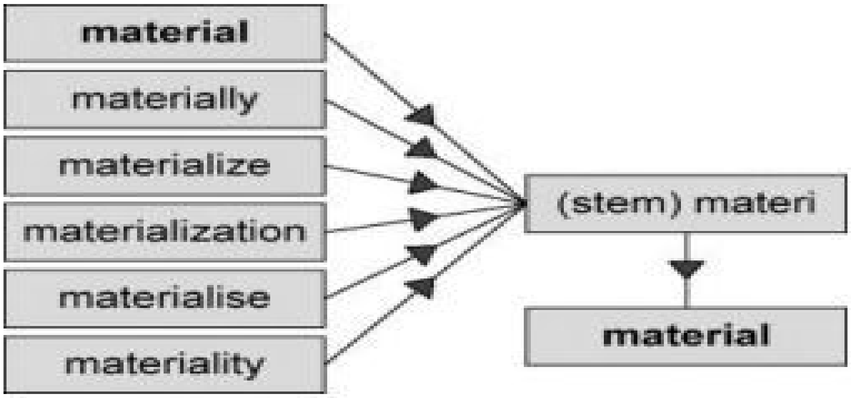


Figure 4 Sample Stemming of a word [29]

Stop Word Removal

Stop Words are the most frequently used words in any language. These words do not carry any information and hence they are not helpful in the text analysis. Stop words can be of types such as verbs, pronouns, conjunctions and prepositions [8, 29]. Thus the stop words are removed from the considered text document during the phase of text preprocessing.

Representation

Text Representation is the process of using a model to effectively represent a text document. Bag of Words is one of the basic models used in representing a document. A document is broken into tokens called terms or features and these groups of features after performing stop words removal and stemming form a feature vector to represent this document [19]. In this Bag of Words model, the exact occurrence of words in a document is ignored.

Knowledge Discovery

Various data mining techniques can be applied to the represent the test document for discovering the knowledge. In this section, text classification technique is described.

Text Classification

Text Classification is defined as a supervised approach to classify an unlabeled text document into one of the labels or classes by using a pre-defined set of labeled text documents [15]. It is a two step process. In the first step, a classifier is built based on the pre-defined set of text documents which are already categorized into a particular class. This is also called a learning step where the classification algorithms are employed to build a text classifier based on a training set of text documents with associated classes. Since the class of each of the text documents present in the training set is provided and fed to the classifier, text classification is called a supervised approach [19]. In the second step, the trained classifier is used to classify the unlabeled set of documents.

The size of the training and testing data sets in the text classification is important. If the classifier is built with a small set of documents, the trained model may not have the sufficient knowledge to classify a test document. Similarly, if the training data set is too huge compared to the testing data set, it leads to a problem called over fitting. Hence, it is important to take the size of both the training and testing data sets into consideration for training the classifier. Some of the popular classifier algorithms are Naive Bayes classifier, Support Vector Models and Decision Trees [22].

2.2.2 Challenges for Text Analytics in social media

Application of traditional text analytic methods to the textual content in social media comes across few challenges due to the distinct features of the social media. The challenges are time sensitivity, short length, unstructured phrases and abundant information [12].

- Time sensitivity

  The main facility provided by the social networking applications is the real time communication between the people. Thus the users could be talking about a range of topics starting from their daily activities to the events, movies, sports, music, newly launched products in the market. These real time updates provide information about the list of current topics that are trending over twitter. Also a change has been observed in the way of the content used in the communication styles of the social networks over the years. Thus, the real time information and the variety of texts used in social media in comparison to the regular text with the evolution of time poses a challenge to text analytics in analyzing the social media content.
- Short length

  Twitter has imposed a limit of 140 characters for each tweet posted by a user. In comparison to the regular text which is a combination of many words, tweets consists of only few words. This may not provide the sufficient information required for the analysis methods.
- Unstructured phrases

  The quality of the content in a social media is much different from the regular text. This may also not provide the sufficient context information required for analysis methods.

- Abundant information

  The enormous amount of information is available on the social media. It consists of both content and non-content information. In addition to the text shared by the users, they also post the hyperlinks, images, hashtags and videos. This combination of both content and non-content information may not be observed in the regular text.

2.3 INTRODUCTION TO TWITTER

Twitter is a social networking and microblogging application which enables the users to post about a variety of topics and connect with other users on Twitter [14]. Micro-blogging is defined as a platform to brief about ones updates in less than 200 characters. The posts shared by the users on the twitter have a limit of 140 characters. Hence, twitter is called a micro-blog [16].

2.3.1 Twitter Concepts

User

Twitter users become the owner of their accounts once they sign up to twitter and they can use their accounts to publish posts called tweets, follow or friend other users on twitter. The tweets shared by the users are public by default unless the visibility option is set to be private. Users are uniquely identified by either user ID or screen name. For a user A to see the posts shared by the user B, user A has to either follow or friend the user B. If the user B prefers not to follow the user A, user B cannot see the posts shared by user A but user A can see the posts of user B as user A is a follower of user B. Only when both the users A and B agree to be friends by choosing the friend option, each of them can see the posts shared by the other. Hence, the relation between any two users is asymmetric. Twitter imposed a limit on the number of friends for a user, whereas there is no limit on the number of followers for a user. In addition to the posts, the information such as a profile picture, location, description, web page and language can be optionally shared by users in their Twitter profiles [16].

Twitter terms

The following are the various terms and their descriptions that are used in a twitter.

Tweet	A tweet is a twitter message of length 140 characters or less that is posted through the Twitter service.
Follower	Users who follow a specific user A to receive user A tweets
Following	The following list of a user A indicates the users whose tweets user A opts to receive.
@reply	In response to a tweet shared by user B, user can post a public message which is directed at user b with @userB
Private account	A user account can be set to private so that the tweets cannot be public. An account whose tweets are not public. Only those twitter users who are approved as a follower by private user can see the private user tweets
Retweets (RTs)	Users have the option to share an interesting tweet they came across by re-tweeting it
Hashtag	A word which a flagged with a hash character is called a hashtag. Hashtags are created by Twitter to help in keyword search
Trending Topics	Twitter displays the most popular topics which are trending over the user accounts
Direct messages (DMs)	Direct message is a short message of length 140 characters or less which can be sent privately by users to one of their followers.

2.3.2 Twitter API's

Twitter provides an open API or Application Programming Interface for the external developers to design a technology that relies on Twitter's data. Twitter API's are categorized into types based on their design and access method for accessing data on Twitter [16,17]. They are REST API and Streaming API. REST APIs makes use of the pull strategy for data retrieval from Twitter. A user has to explicitly request the data in order to collect it. This API provides a facility for the developers to access Twitter data such as user timelines and user information. Streaming APIs makes use of the push strategy for data retrieval from Twitter. This API provides a continuous stream of information with no further input required from a user.

Twitter enables access to the APIs only through authenticated requests. Twitter provides Open Authentication and each request to the API has to be signed with valid Twitter user credentials. Twitter has imposed a limit on the number of requests that can be issued to access Twitter APIs within a time window called the rate limit. Access to Twitter API's can be issued on an individual or an application level. The rate limits are enforced at both the levels and they are different for each of these levels. Usually a time window or rate limit is for every 15 minutes.

2.3.3 Why mine Twitter?

Twitter has a huge user base of 200 million user accounts [1]. A significant growth in the user bases has been observed over the past few years and this has led to a huge interest in analyzing and extracting the information such as geographic location of the Twitter users. Geographic locations of the users have become an important factor in the applications such as event detection, targeted advertising, public polls, news recommendation [17] and epidemic dispersion for providing the users with personalized services. Many applications such as search engines and targeted advertising try to adapt the services they offer based on the location of the users [2]. Hence, mining the twitter content posted by users enables location based personalized services.

2.4 RELATED WORK

Interest in the geographic scope of the web resources by researchers has increased in the last few years. Relevant work in this area of research can be categorized into the following groups based on the strategies used in geo-locating online users. They are content analysis using probabilistic models, content analysis using an unsupervised approach and location inference with the analysis of social relations.

2.4.1 Content analysis with probabilistic models

The existing approaches to estimate the location of Twitter users based on the tweet content are of intuition that the location of a user influences their tweets. They most probably tweet about the restaurants, local sport teams, shops, local events and tweets can contain local slang words [9]. Most of these approaches are data driven and they require huge trained data set comprising tweets for building models to find the location of a user.

Hecht et al. [4] investigated the user behavior of the location field in the Twitter user profiles. They have explained how the location specified by the users in their profiles is not reliable in most of the cases and can affect the location based services which rely completely on the location entered in the Twitter profiles. To demonstrate the quality and the geographic scale of the geographic information entered in the twitter profiles, location details of a total of 10000 Twitter users who tweet in English is collected. Upon using a coding scheme to validate the available Twitter profile locations against search engines and online mapping sites, it is observed that 66% of users have entered valid geographic information in Twitter profiles, 18% of users have not entered any information while 16% of users have entered irrelevant information. Of the 66% of users with valid geographic information, users who are outside the United States and have reported multiple locations are filtered out and the filtering process left them with 3149 users. Further the geographic scale of the geographic information entered by 3149 users is examined. It is observed from Figure 5 that 64% of 3149 users have mentioned the location in their profiles at city level and 20% of the users at the state level.

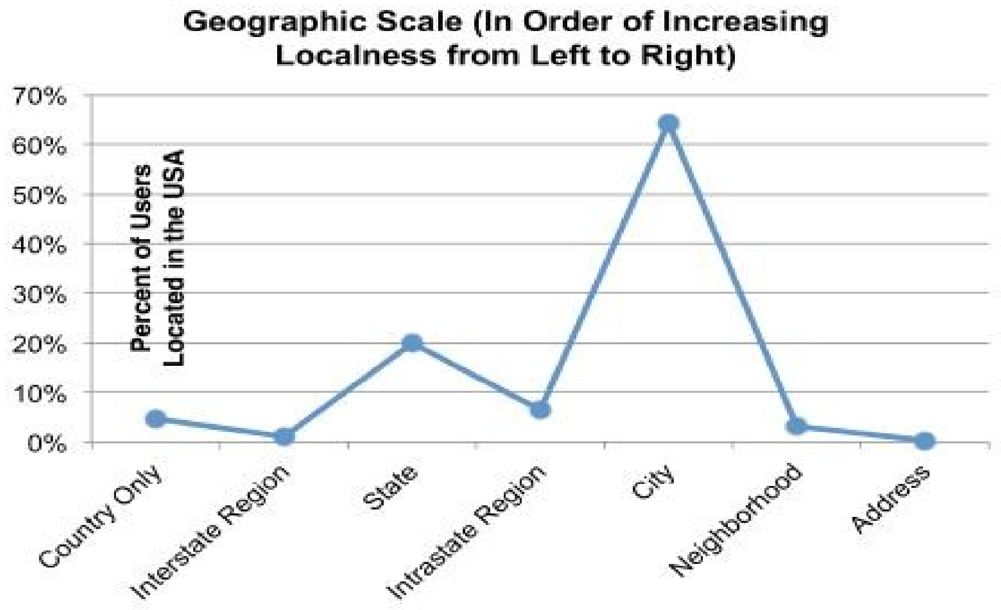


Figure 5 The scale of the geographic information entered by the United States Twitters users [4]

Hecht et al. [4] proposed a content based probabilistic model to predict the location of a Twitter user at state and country level. Term frequency based MNB model is used to train the classifier based on the tweets. They have fixed on using 10000 terms for building the classifier. Two methods are decided to be used for extracting the terms for the classifier. One is count method, which selects the 10000 terms based on the highest term frequency while the other is calgari method, which selects the first 10000 terms based on the maximum conditional probability of term for each of the states or the countries considered in the data set. To evaluate the accuracy of both the methods in predicting the location of user at state and country level, a data set is formed by collecting the English tweets of 99296 users who have valid state and country information. The location information of these users is validated with the help of Wikipedia based Geo-coder. It is observed that the calgari method has given 89% accuracy for predicting countries and 27% accuracy for predicting states while an accuracy of 72% for countries and 24% for states is observed for count method. The better performance of the calgari method is accounted for the ability to extract the features which are local to a country or state.

Inferring location of users at the city level is challenging compared to inferring at state and country level. This is because the number of cities typically used in a dataset for classification is more compared to the number of states or countries used [5,8].

Cheng et al. [5] proposed content based probabilistic model for predicting the location of Twitter user at the city level by considering only user tweets. The main features of their work are (i) to use only the content posted by the users with no other external input from users or external Web based knowledge (ii) to identify the words in tweets that have a strong local focus and dispersion by using a model based on the observed geographic distribution of words in tweets (iii) to apply smoothing models in order to account for the data sparseness. The training data set is formed by collecting around 4 million tweets from 130689 users whose profile locations are validated against a gazetteer of cities within the United States. Stop words are removed from the tweets and Jaccard co-efficient is used for stemming the words present in the data set. The probability distribution of each word for a city is calculated. Maximum likelihood of a user in a city is calculated to determine the location for 5119 users based on their tweets. It is observed that only 10.12% of users are located within 100 miles of their actual locations.

Upon examining their baseline model, they have identified two issues. One issue is that the most of the words used for classification are consistently distributed across all cities and another issue is that their model could not account for tweet sparsity in the cities with less population. To overcome the first issue in their model, they have proposed a spatial focus and dispersion method with the goal to extract the local words for a city based on the values of two parameters for a word such as focus and dispersion. Focus of a word is defined to be the geo-center at which a word is frequently used while dispersion is defined to be the speed at which the frequency of the word falls on moving away from the observed geo-center of that word. Using the spatial focus and dispersion method, they have identified the words that exhibited the location indicativeness as presented in Figure 6. To overcome the tweet sparsity issue, they have considered three smoothing models such as Laplace smoothing, state-level smoothing and neighborhood smoothing. To evaluate their work, they have built probabilistic models for cities based on the words selected by the spatial focus and dispersion method. A test data set comprising 5000 users who have at least 1000 tweets and have reported location as latitude/longitude coordinate is considered. They have estimated the cities of 51% of users within 1000 miles of their actual location using lattice based neighborhood smoothing model.

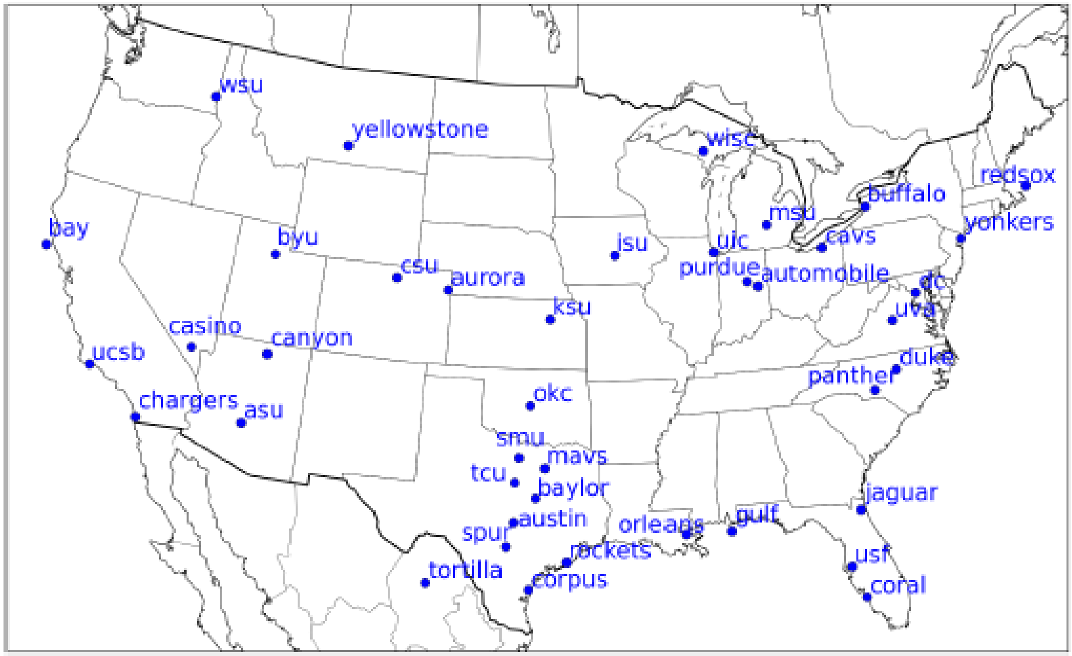


Figure 6 Geographical Centers of Local Words discovered in sampled Twitter data set[5]

Han et al. [3] has also approached the task of estimating geo-location as a text classification problem and demonstrated the importance of the location indicative words in accurately estimating the location of the user based only on the tweets. Multinomial Naive Bayes is chosen as the model for building the text based classifier. Three feature selection methods such as Inverse City Frequency (ICF), Information Gain Ratio (IGR) and Maximum Entropy (ME) are employed to observe the comparative performance of these methods in selecting the location indicate words.

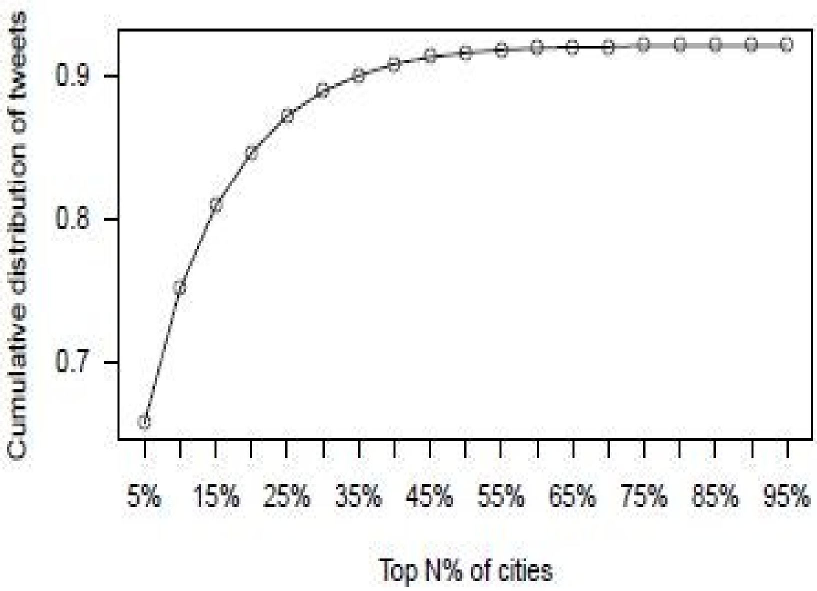


Figure 7 Cumulative coverage of tweets for an increasing number of cities based on 26 million geo-tagged tweets [3]

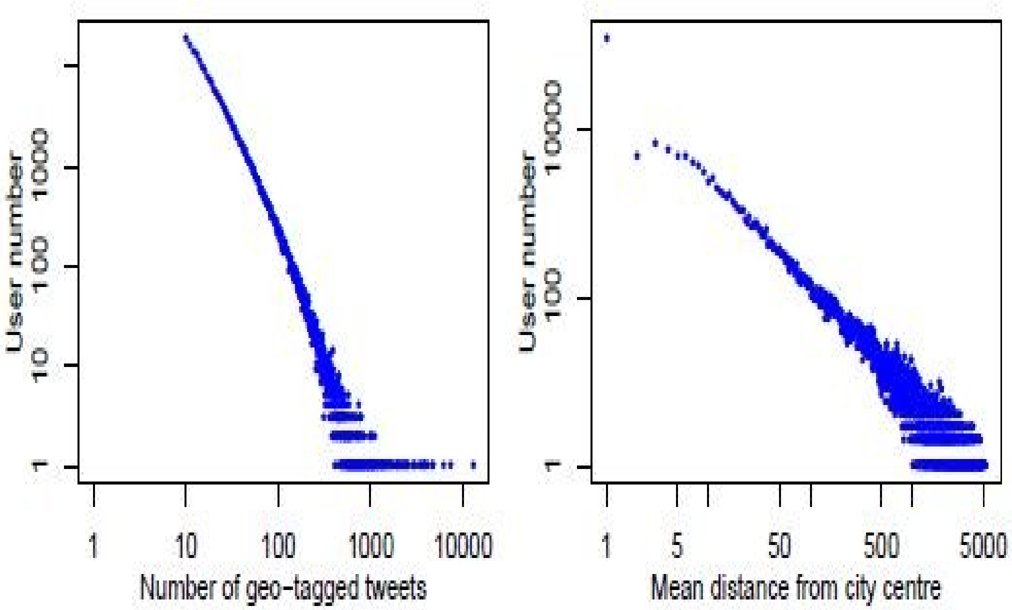


Figure 8 The number of users with different number of tweets, and different mean distances from the city center [3]

With the help of the publicly available geo-names dataset, they have determined the city ‘c’ with the highest population in a region as the top most cities have more geo-tagged tweets that can be seen in Figure 7. They have collapsed the rest of the cities in the same region within a range of 50km under the city c. They have followed the same procedure for the next largest city in the same region and repeated the similar process for all the regions in the world. Finally, they have identified 3709 cities all over the world. They have formed a data set by collecting geo-tagged English tweets from the users who are closely located in the identified cities. Only those tweets that are close to the city are considered by dividing the earth into 0.5’ and 0.5’ grids and the tweets that don’t fall into any of the neighboring 8 grid cells are removed. A sample of 26 million tweets is collected in total and a user is assigned a city in which the majority of their tweets is occurring. They have observed that the top 40% of the cities contained 90% of the sampled tweets. The built data set was analyzed to find the ratio of the number of users over their number of tweets and also to observe the number of users with various levels of geographical spread in their tweets, which is calculated as the average distance between a user’s tweets and the center of the city to which that user is assigned. According to their analysis in Figure 8, most of the users have comparatively less number of geo-tagged tweets and the most of users stay nearby city.About 96K features are manually extracted from the training data set and for each of the feature selection methods, the top N% of the manually selected features is considered for training the MNB classifier. For the top 92%, 88% and 20% features, an optimal accuracy is achieved for the IGR, ICF and ME methods respectively. A testing data set of 10K users is used and the accuracy of estimating the location correctly for users within 100 miles of their actual location is observed at 45%, 35.9% and 32.6% respectively.

Mahmud et al. [8] proposed a model built on a combination of statistical and heuristic classifiers to find locations based on tweets. They have approached the problem of finding location as a hierarchical classification task and estimated the time zone or geographic region for a user first followed by estimating location. User tweets are broken into three categories as words, hash tags and places. Places in the tweets are identified with the aid of a US geographical gazetteer. Each of these categories is used to train the statistical classifiers by Multinomial Naive Bayes algorithm. Statistical classifier is built for a time zone in order to account for the different tweet volumes in the different time zones. Local place classifier and visit history classifier are the heuristics in their proposed approach. A heuristic is that users are most likely to mention their home city or province name in tweets compared to other cities. Local place heuristic is to find the most frequent city/province name mentioned in the user tweets and predict this city/province as the location for the user. Visit heuristic is to find all the places visited by a user by resolving the URLs generated by Foursquare location check in service with Foursquare API and then predict the most frequently visited city as the location. They have formed the data set by collecting the tweets from the users corresponding to the top 100 most populous cities of the United States by using Google geo-coding API to obtain the geo-coordinates of a city. A total of 1.5 million tweets is collected from 9551 users. As part of the data cleaning process, all the stop words are removed from tweets and parts of speech of the words in the tweets are obtained using an Open NLP. Parts of speech, such as adjectives, verbs, prepositions are considered to be of no value as they don't correspond to any specific city and are used all over the world and hence they are removed from the tweets. Further the conditional probabilities of terms for each city are calculated and used as a measure to determine the terms that have to be retained and passed as an input to the statistical classifier. They have specified an empirically chosen value for the maximum and average conditional probabilities of terms for a city by experimenting for different values. Each of the three statistical classifiers is trained on different terms for each city. To evaluate the accuracy of their approach, they have evaluated the accuracy of the hierarchical classification at time zone and city levels. An accuracy of 73% was observed at the time zone level and 58% at city level.

Wing et al. [30] proposed a hierarchical discriminative classification method which employs logistic regression classifiers based on text to improve the performance of geolocation prediction. The primary reason behind adopting logistic regression classifiers is due to the consideration for the interdependence between the features unlike the commonly employed classifiers. The other reason is that the logistic regression classifiers extracts the features naturally by assigning more weights to the features that discriminates among the target classes well. They have evaluated the performance by considering with four methods. They are naïve bayes (NB) baseline method, naïve bayes which uses the features selected by the information measure method called information gain ratio (IGR), the FlatLR method which employs logistic regression classifier over the nodes or features present in the last hierarchy level and the HierLR method which considers the product of logistic regression classifiers at all nodes in the hierarchy. Accuracy of predicting locations in a range of 161 kilometers (Acc@161) is calculated for all the methods and the Acc@161 for the NB, IGR, FlatLR, HierLR methods are reported as 36.6%, 45.9%, 47.5% and 49% respectively. The hierarchy classification method with logistic regression classifiers outperformed the other considered methods and promised optimal performance.

2.4.2 Content analysis with unsupervised approach via external knowledge base

Krishnamurthy et al. [9] proposed a novel knowledge based approach to predict the location of Twitter users. Their approach is to use Wikipedia as a knowledge base in aiding the prediction of the user's location on Twitter without having to build large training data set of tweets. Wikipedia is chosen as the knowledge base as it is observed to have dedicated pages for geographical locations at various levels of granularity such as town, city, county, state, time zone and country and also it is dynamically updated by users. The key components of their proposed approach are Knowledge Base Generator, User Profile Generator and Location Predictor and is displayed in Figure 9. The knowledge base is generated by extracting the entities for each of the cities from the considered list of locations where an entity for a city in Wikipedia is defined to be all the local topics that are listed as a hyperlink on the Wikipedia page for that city. The entities extracted from the Knowledge base generator are scored by four scoring measures to determine the correlation shared between an entity and a city. The four measures are PMI, BC, JI, and JI. PMI is a measure used to find the correlation between a city and its local entities by considering their co-occurrences in the entire dump of Wikipedia. BC is a measure used to find the significant entities among the graph of entities for a city. A graph is formed with the nodes as the entities found in a city and an edge between two nodes is present in case of occurrence of these two entity names in the respective pages. JI is used to measure the overlap between the entities of a city and a local entity. Similar to JI, TI also measures similarity of the local entity to the city, but it looks for an overlap only in the category to which the local entity belongs. All the local entities for each of the listed cities are scored by the four scoring measures. For all the users for whom the location is to be predicted, the User Profile Generator is used in order to extract Wikipedia entities that matches with the tweets of each of the users. The outputs of the User Profile Generator and Knowledge base generator are utilized in the Location Predictor to calculate the score for a user for each city by aggregating the matched local entities of the city found in the user's tweets and then selecting the city with the maximum score as the location for the user.

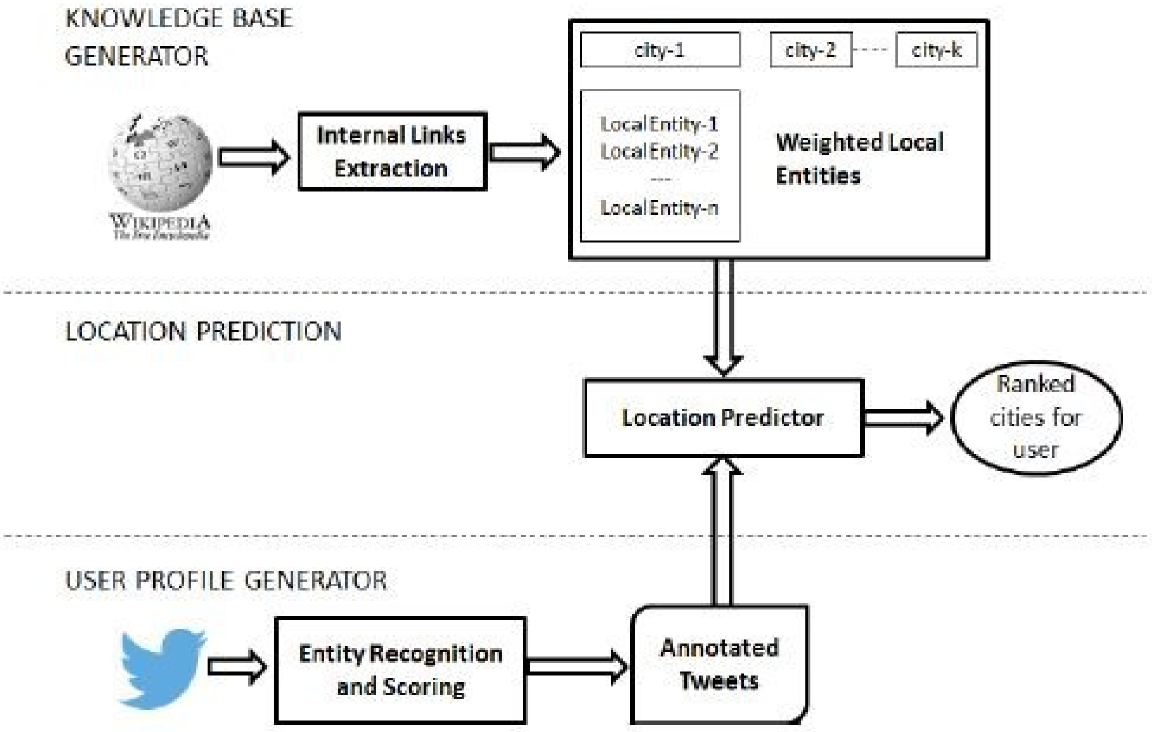


Figure 9 Location Prediction Framework using Wikipedia [9]

To evaluate the proposed approach with respect to the four scoring measures and also with Cheng et al. [5] approach, Cheng et al. [5] data set is used. The knowledge base is created by considering all the cities with a population of at least 5000 as per the census information of 2012. It is observed that the accuracy of the location prediction using local entities for PMI, BC, JI and TI is 32.46%, 47.91%, 53.21% and 54.48% respectively. Compared to an accuracy of 51% achieved by Cheng et al. [5] on the same data set, the proposed approach with a TI score performed better with an accuracy of 54.48%.

2.4.3 Location Inference via online social network

Jurgens et al. [10] demonstrated that a user's location can be inferred accurately by leveraging the online social network of a user. The intuition is that offline relations formed by the users are translated to online relations on social platforms and hence a user's social network is most likely to have friends who live geographically close by. They have proposed an algorithm called spatial label propagation which is a semi-supervised iterative algorithm to determine the locations of the users with no location specified on an online social platform by considering the known locations of their directly connected neighbors as input. A select function is employed to identify a neighbors' location and five methods such as geographic median, Oja et al. [18] Simplex Median, Triangle Heuristic, Random Neighbor and Traditional Label Propagation are used to aid in the process of identifying the nearest neighbor. Cumulative distribution functions (CDF) of the distance to the closest neighbor of a user is calculated for each of these methods and CDF is used as a parameter to demonstrate that a less distance between a user's location and their nearest neighbor is indicative of the fact that the method adopted to select the nearest neighbor is accurate. Upon evaluating each of these methods in estimating the nearest neighbor for a user and calculating the cumulative distribute function, it is observed that the geographic median has offered significant performance compared to the other methods. It has located half of the test users accurately within a range of 10 kilometers. Further, they have discussed that the location information of users on one social platform can be leveraged to infer the locations of the users on other social platform.

2.4.4 Summary

The researchers have approached the problem of location estimation of Twitter users in different ways. While few researchers have considered classification based approaches or hierarchical discriminative classification methods to determine the location via text, some researchers have employed unsupervised approaches via external knowledge or online social network of users to estimate location. Naïve Bayes, Naïve Bayes Multinomial and Logistic Regression are the popular classifiers employed for building the learned classifiers. The importance of the feature extraction methods for selecting the salient features which have the ability to discriminate among the locations is demonstrated in the existing classification or supervised approaches. Not only the recent updates or the real time information posted by the users is exploited, but also a history of the user tweets is considered for location estimation. The reason being a single tweet or a few recent tweets may not reveal enough geographic information. Thus, a collection of user tweets is explored by the approaches in order to have sufficient information for accurately finding the locations where the user posts are active.

Many researchers have proposed hierarchical classification methods to predict locations based on the tweet content only. But the tweet metadata which contains the information like location and time zone are not considered at all hierarchy levels in the existing hierarchical classification methods. The existing research work has employed the available tweet creation time of users belonging to different time zones in order to estimate the locations at higher hierarchy level. A combination of content and tweet metadata based classifiers at higher and lower hierarchy levels in the location classification methods is not looked upon. The raised issues are explored as the research objectives in the proposed methodology, which is presented in the chapter 3 to estimate the location of the Twitter users via available content and tweet metadata.

CHAPTER 3 PROPOSED WORK

To improve the prediction accuracy of the home city of Twitter users based on tweet content, a hybrid approach which comprises the hierarchical location classification and the geographical information from the geo-tagged tweets, is proposed to estimate the location of Twitter users at different granularities such as time zone, state and city. The ability to predict the location of users in a hierarchical fashion is achieved by employing a content based probability model for training the classifiers at time zone, state and city level on the tweets of users with known locations and further using these trained classifiers to estimate accurate city of users. The goal of the proposed hierarchical location classification method is to first predict a time zone of a user and then confine the search to the states only present in the predicted time zone to predict the state of a user followed by searching only the cities present in the predicted state to predict the city of a user. The content of a single tweet posted by a user may not reveal much geographical information about a user. Hence, the proposed work considers a collection of user tweets for predicting the location of users.

3.1 ARCHITECTURE OF HIERARCHICAL LOCATION CLASSIFICATION MODEL

The architecture of the proposed hierarchical location classification of Twitter users is presented in this section and in Figure 10. It consists of four phases. They are data extraction, data preprocessing, feature extraction and hierarchical location classification. The processes and algorithms involved within the phases are discussed in detail for the overall proposed architecture.

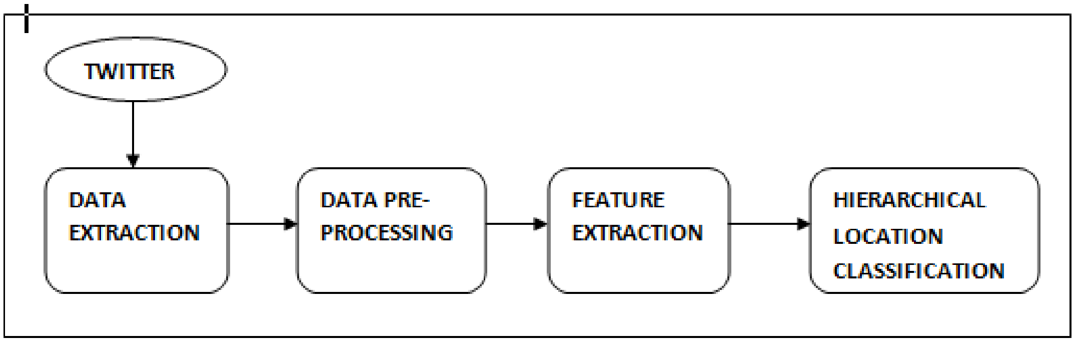


Figure 10 Architecture of Hierarchical Location Classification

3.1.1 Data Extraction

Firstly, the list of cities that are to be considered for location classification is selected. The geo-coordinates of the selected cities are obtained by using Google's geo-coding API.The extracted geo-coordinates are fed to the Twitter's Search API in order to find the users who have reported one among these cities as their locations in their Twitter profiles. Further, Twitter's Rest API is used for collecting the tweets of the found users to form a training data set and test data set.

3.1.2 Data Pre-processing

Tokenization, stop words removal and parts of speech tagger are the methods used in the preprocessing of tweets present in the training and test data sets.

Tokenization

All the user tweets present in the training and test data sets are broken down into tokens as presented in Figure 11. To identify the place names in the user tweets, tokens are further classified into unigrams, bigrams and trigrams. White spaces, URLs and all special characters excluding '#' symbol are removed. Places in the user tweets are identified by matching the unigram, bigram and trigram tokens with the list of selected cities and their provinces. Tokens are placed into different categories such as words, hashtags and place names.

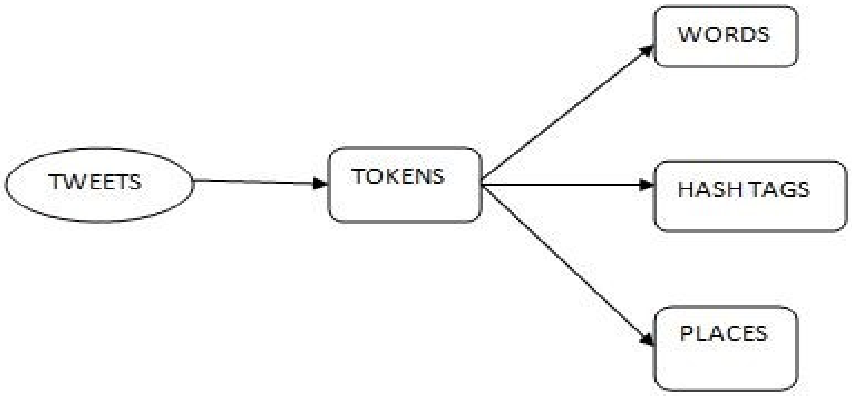


Figure 11 Tokenization of tweets into different categories

Stop Words Removal

Stop words are commonly used all over the world and they do not distinguish a particular location. Hence, stop words are removed from the user tweets by using a standard list of 319 stop words.

Parts of speech tagger

Parts of speech of the terms present in the word category are identified using Open NLP tagger [21]. Words that are identified to be adjectives, pronouns, verbs, prepositions are removed as these words are commonly used all over the world and hence they don't distinguish between the locations.

3.1.3 Feature Extraction

The objective of the feature extraction process is to select the terms from the categories such as words, hashtags and places, which are local to a location. The intuition is that the selected terms help in the process of location classification. The selected terms are called features and they are selected by using heuristics.

Firstly, the frequency of each of the terms present in the three categories is computed and also the number of people who have used these terms in the respective categories is computed. The terms which are used by at least K% of people are retained for further feature extraction where the value of K is chosen empirically. Secondly, the average and the maximum condition probability of locations for the terms is calculated. The difference between the average and maximum condition probability of locations for the terms is checked to see if the values are above a threshold. And also a threshold value is set for the maximum conditional probability of the locations for the terms. The threshold values are also chosen empirically. The goal is to extract the features or local terms which are frequently used by a considerable number of people in each of the locations. At the end of feature extraction, features or local terms of the three categories present in the user tweets such as words, hashtags and places are extracted.

3.1.4 Hierarchical Location Classification

To predict the location of a Twitter user at the lower level granularity like city, classifier has to discriminate against all the cities used in the training data set. This is a large classification problem. The prediction accuracy of a classifier depends on the number of classes to be discriminated against in order to predict the class. More the number of classes less is the prediction accuracy of the classifier. Hence, the need arises for the large location classification problem to be divided into smaller location classification problems by organizing the classifiers in a hierarchy.

The objective is to employ a three level hierarchy for location classification of Twitter users in order to improve the prediction accuracy. The first, second and third levels of hierarchy in the location classification method are time zone, state and city respectively. For each of the hierarchy levels, location classifier is a combination of content based statistical classifiers and content based hierarchy classifiers, and the trained models are built for the statistical classifiers at different hierarchy levels with the MNB classifier algorithm. The high level classifier in the hierarchical location classification method is trained with the classes as time zones. For the time zones considered as classes in the high level classifier, the intermediate level classifier is trained with the classes as states belonging to these time zones. Further, for each of the states present in a time zone, the low level classifier is trained with the classes as cities belonging to these states.

Classifier Algorithm

The Naive Bayes Multinomial (MNB) algorithm is employed for building the statistical classifiers. The way MNB is used to classify a location of a Twitter user by calculating the probability distribution of location for the selected terms in the user tweets is demonstrated by describing the algorithm in general as below [20].

Let C be the set of the classes or locations, N be the size of the vocabulary or unique terms present in a class and t_i be a tweet. Equation 1 [20] presents the equation to compute the MNB conditional probability of a user belonging to a city given a tweet.

$$\boxed{\Pr(c|t_i) = \frac{\Pr(c)\Pr(t_i|c)}{\Pr(t_i)}, \qquad c \in C} \tag{1}$$

Pr(c) is estimated by dividing the number of user tweets belonging to a class c by the total number of user tweets in the training data set. Equation 2 [20] presents the equation to compute the MNB conditional probability of obtaining a user's tweet t_i in a class of city c.

$$\boxed{\Pr(t_i|c) = \left(\sum_n f_{ni}\right)! \prod_n \frac{\Pr(w_n|c)^{f_{ni}}}{f_{ni}!},} \tag{2}$$

F_{ni} is the count of word ‘n’ in the user tweet t_i and Pr (w_n/c) is the conditional probability of a word ‘n’ given a class c and is calculated according to the equation 3 [20].

$$\hat{\widetilde{\mathrm{Pr}}}(w_n|c) = \frac{1 + F_{nc}}{N + \sum_{x=1}^{N} F_{xc}}, \tag{3}$$

F_{xc} is the count of word ‘x’ in all the training user tweets corresponding to the class c and the Laplace smoothing method is applied to avoid zero frequency problem for any word count by adding one. Pr (t_i) is the probability of a user tweet t_i and is calculated according to the equation 4 [20].

$$\mathbf{Pr}(t_i) = \sum_{k=1}^{|C|} \mathbf{Pr}(k)\mathbf{Pr}(t_i|k). \tag{4}$$

Pr (t_i/c) for a city is independent of summation and product of fn i.e. the count of word ‘n’ in user tweet t_i. Hence, the Pr(Ti/c) is calculated as in Figure 16 where alpha is a constant.The simplified conditional probability of a user belonging to a city given a tweet is calculated as per the formulae in the equation 5 [20]. The city for which the maximum conditional probability of a user in a city is observed is estimated to be the location of the user.

$$\Pr(t_i|c) = \alpha \prod_{n} \Pr(w_n|c)^{f_{ni}}, \tag{5}$$

3.2 HIERARCHICAL LOCATION CLASSIFICATION

The hierarchical location classification method is an ensemble of content based statistical classifiers and heuristic classifiers.

3.2.1 Content based Statistical Classifiers

At the end of the preprocessing phase, terms present in the user tweets of the training data set are categorized into three groups such as words, hashtags and places. Further, features or local terms extracted for these categories during the process of feature extraction are now passed as the input to the corresponding statistical classifiers for building the trained classifiers with the classes as locations i.e. time zone, state and city, as in Figure 12.

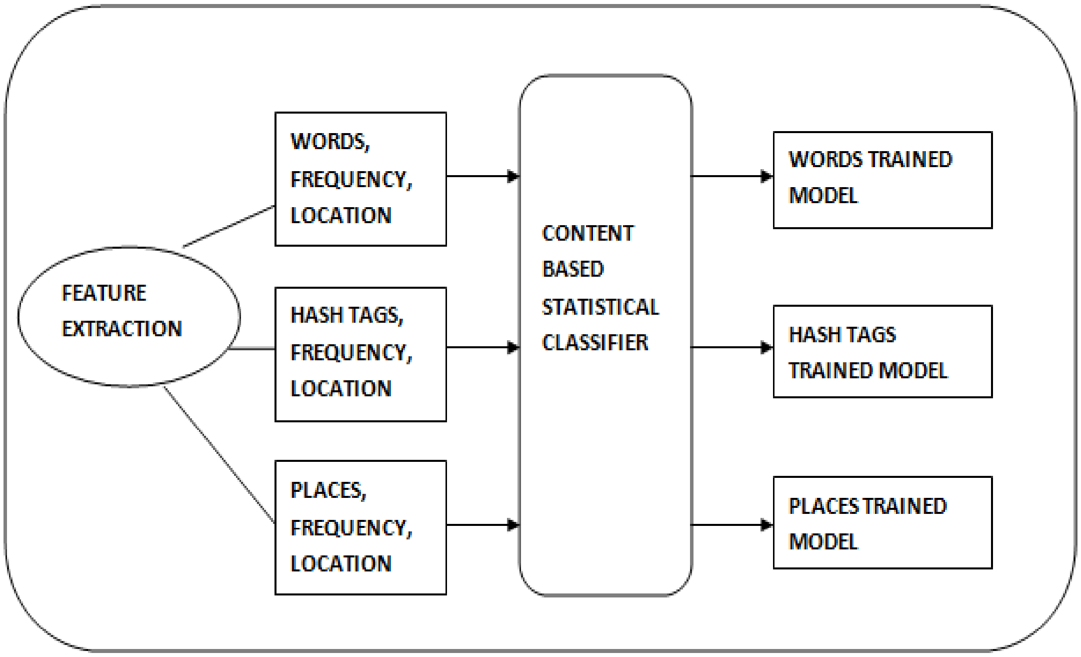

FEATURE EXTRACTION
WORDS, FREQUENCY, LOCATION
HASH TAGS, FREQUENCY, LOCATION
PLACES, FREQUENCY, LOCATION
CONTENT BASED STATISTICAL CLASSIFIER
WORDS TRAINED MODEL
HASH TAGS TRAINED MODEL
PLACES TRAINED MODEL

Figure 12 Content based Statistical Classifiers

Words Statistical Classifier

Words statistical classifier is trained on the words which are selected by feature extraction to build the trained model using MNB machine learning algorithm. Let the words be w, C be the set of classes (cities) in a state and N be the vocabulary or the unique words present among all the classes (cities) in C. The conditional probability of a word given a class is computed according to MNB's rule according to the equation 6.

P (w/c) = (1 + frequency of word w in city c)/ (N + frequency of all words in city c) (6)

Similarly, the conditional probability of a word given a class is calculated for the classes as states and time zones.

Hashtags Statistical Classifier

Hashtags statistical classifier is trained on the extracted hashtags to build the trained model using MNB machine learning algorithm. Let the hashtags be h, C be the set of classes (cities) in a state and N be the vocabulary or the unique hashtags present among all the classes (cities) in C. The conditional probability of a hashtag given a class is computed according to MNB's rule according to the equation 7.

P (h/c)=(1 + frequency of hashtag h in city c)/ (N + frequency of all hashtags in city c) (7)

Similarly, the conditional probability of a hashtag given a class is calculated for the classes as states and time zones.

Places Statistical Classifier

Places statistical classifier is trained on the extracted place names to build the trained model using MNB machine learning algorithm. Let the place names be p, C be the set of classes (cities) in a state and N be the vocabulary or the unique place names present among all the classes (cities) in C. The conditional probability of a place name given a class is computed according to MNB's rule according to the equation 8.

P (p/c) = (1 + frequency of place 'p' in city c) / (N + frequency of all places in city c) (8)

Similarly, the conditional probability of a place given a class is calculated for the classes as states and time zones.

3.2.2 Content Based Heuristic Classifiers

Two heuristic classifiers are built to predict the location of Twitter users. They are places heuristic classifier and geo-coordinates heuristic classifier.

Places Heuristic Classifier

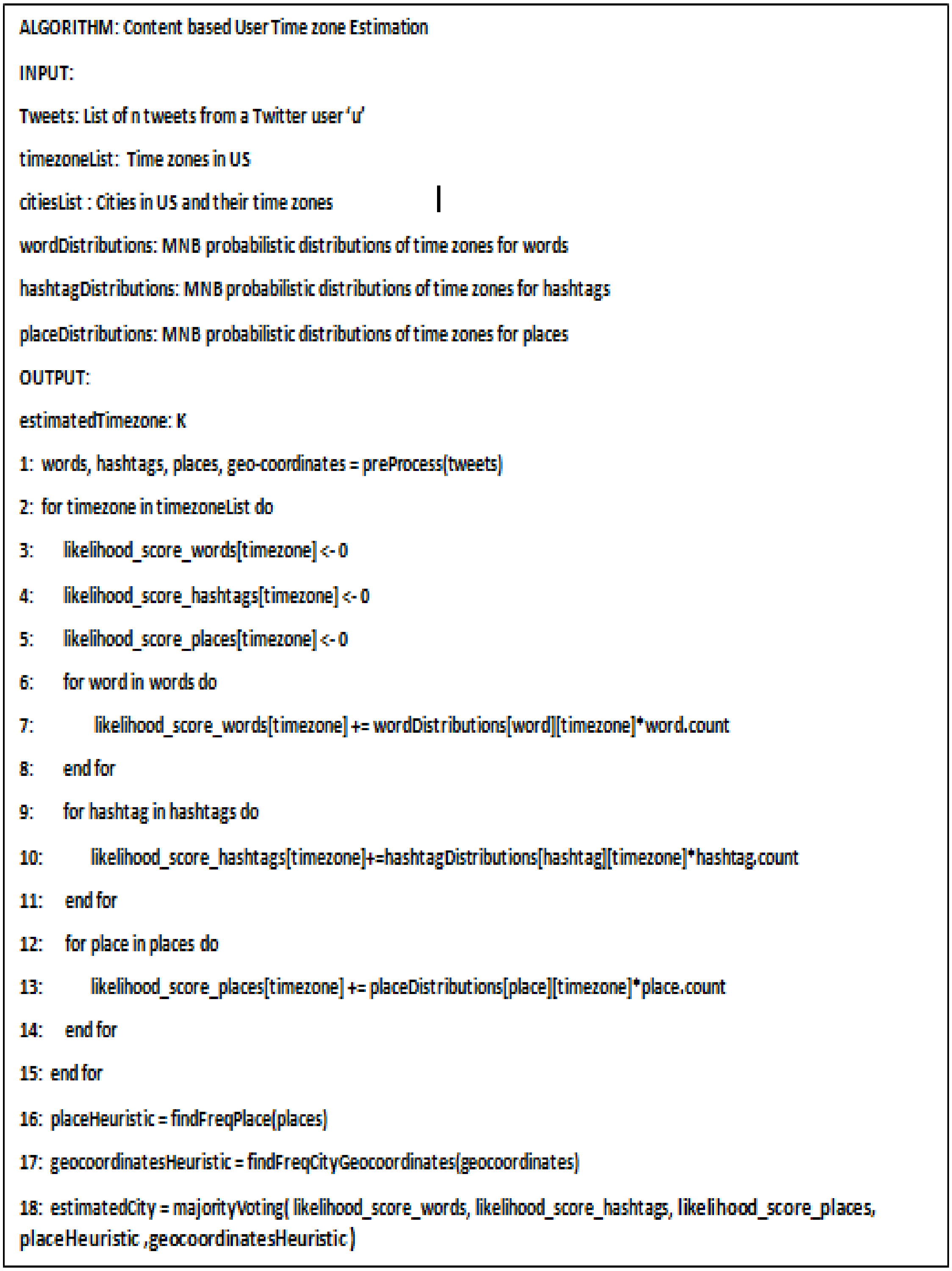
The heuristic is that users would specify their home city or state more compared to the other cities and states. For each of the cities and states used as classes in the training data set, their frequency of occurrence in user tweets is computed. The city or state with highest frequency in the user tweets is classified to be the home location for that user.

Geo-coordinates Heuristic Classifier

The heuristic is that users would tweet mostly from their home city or state compared to the other cities or states. For each of the cities used as the classes in the training data set, the geo-coordinates of the city center are found using the Google's geo-coding API. Further, the geo-coordinates from the geo-tagged tweets of a user are extracted and mapped to the closest city in a range of 50 miles using the Google's geo-coding API. Only those cities that are identified to be one among the classes used in the training data set are retained and the frequencies of geo-tagged tweets in the identified cities are computed. The city in which the geo-tagged tweets of a user have occurred most often is classified to be the home location for that user.

3.2.3 Location Prediction

To predict a home location of a Twitter user, first a size of 'n' user tweets are extracted from the user's timeline in Twitter where n is a number chosen empirically. These user tweets, time zones in the US and MNB conditional probability distributions of time zones for words, hashtags and places are passed as input to the content based user time zone estimation algorithm.



```
ALGORITHM: Content based User Time zone Estimation
INPUT:
Tweets: List of n tweets from a Twitter user 'u'
timezoneList:  Time zones in US
citiesList : Cities in US and their time zones
wordDistributions: MNB probabilistic distributions of time zones for words
hashtagDistributions: MNB probabilistic distributions of time zones for hashtags
placeDistributions: MNB probabilistic distributions of time zones for places
OUTPUT:
estimatedTimezone: K
words, hashtags, places, geo-coordinates = preProcess(tweets)
for timezone in timezoneList do
    likelihood_score_words[timezone] <- 0
    likelihood_score_hashtags[timezone] <- 0
    likelihood_score_places[timezone] <- 0
    for word in words do
        likelihood_score_words[timezone] += wordDistributions[word][timezone]*word.count
    end for
    for hashtag in hashtags do
        likelihood_score_hashtags[timezone]+=hashtagDistributions[hashtag][timezone]*hashtag.count
    end for
    for place in places do
        likelihood_score_places[timezone] += placeDistributions[place][timezone]*place.count
    end for
end for
placeHeuristic = findFreqPlace(places)
geocoordinatesHeuristic = findFreqCityGeocoordinates(geocoordinates)
estimatedCity = majorityVoting( likelihood_score_words, likelihood_score_hashtags, likelihood_score_places,
placeHeuristic ,geocoordinatesHeuristic )
```

Figure 13 Content based User Time zone Estimation Algorithm

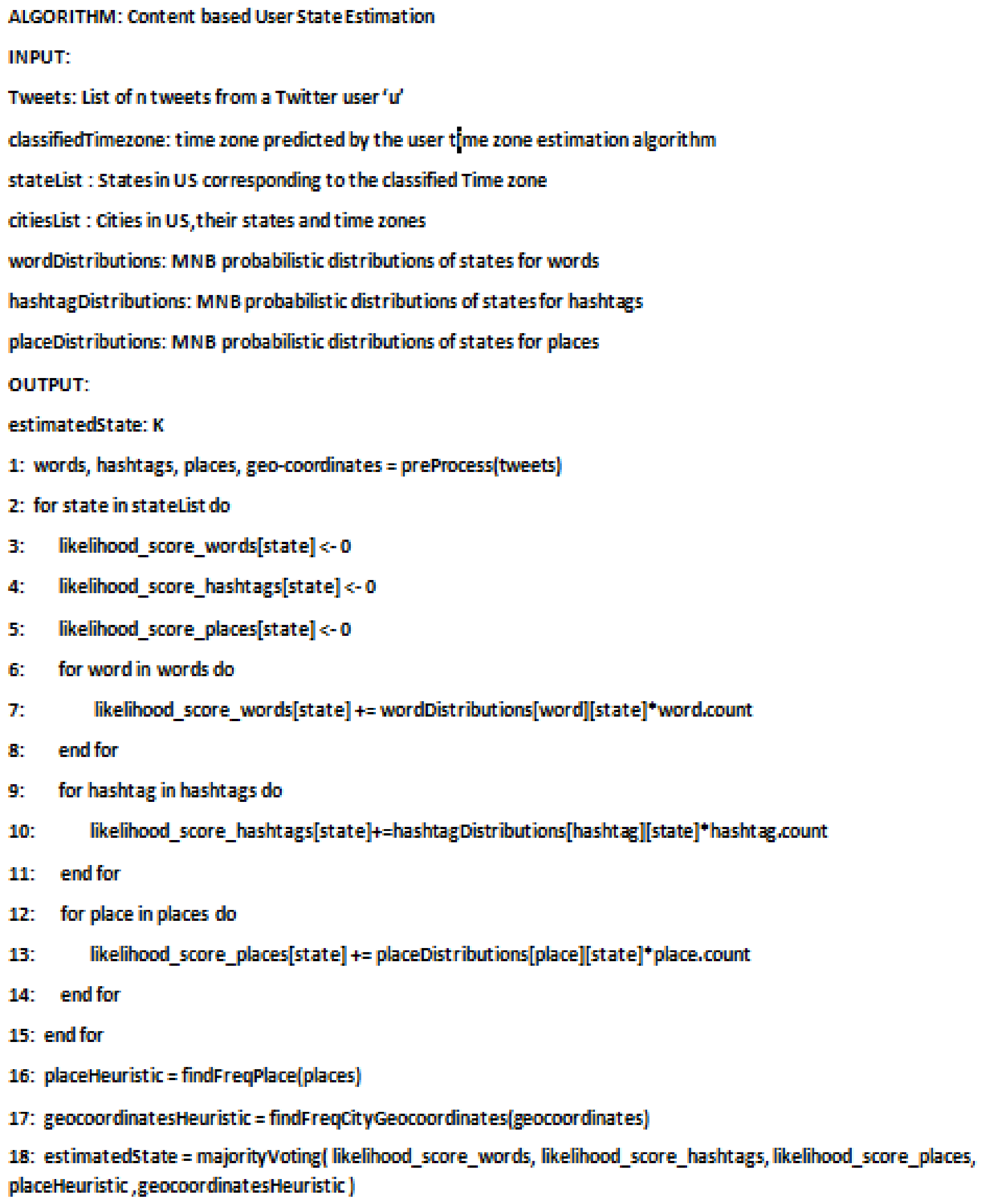
Time zone Estimation

The purpose of the time zone estimation algorithm is to estimate the accurate time zone for a Twitter user based on user tweets and MNB time zone classifier trained models for words, hashtags and places. The content based user time zone estimation algorithm is in Figure 13.

Firstly, user tweets are pre-processed to extract the words, hashtags and place names. For all the geo-tagged tweets present among user tweets, geo-coordinates are extracted. Secondly, the MNB conditional probability of a user in a time zone is calculated using the extracted words and MNB time zone classifier trained model for words. Similarly, the MNB conditional probability of a user in a time zone is calculated for the other categories like hashtags and places. Thirdly, the city or state name which is identified to be in the city list and mentioned more often by the user in tweets is selected as place heuristic. The corresponding time zone of the place heuristic is found. Fourthly, the extracted geo-coordinates are mapped to city centers using the Google's geo-coding API and only those cities which are present in the city list are retained along with their frequency. The city in which the geo-tagged tweets are found most often is selected as the geo-coordinates heuristic and the corresponding time zone is found. The time zones predicted by the content based statistical and heuristic classifiers are passed to the majority voting method. Majority voting is a method to find the time zone, which has received more votes from the classifiers. The time zone found by the majority voting method is estimated to be the accurate time zone of the Twitter user.

State Estimation

The purpose of the state estimation algorithm is to estimate the accurate state for a Twitter user based on user tweets, the estimated time zone selected by the time zone estimation algorithm and MNB state classifier trained models for words, hashtags and places. The content based user state estimation algorithm is in Figure 14.



```
ALGORITHM: Content based User State Estimation
INPUT:
Tweets: List of n tweets from a Twitter user 'u'
classifiedTimezone: time zone predicted by the user time zone estimation algorithm
stateList : States in US corresponding to the classified Time zone
citiesList : Cities in US,their states and time zones
wordDistributions: MNB probabilistic distributions of states for words
hashtagDistributions: MNB probabilistic distributions of states for hashtags
placeDistributions: MNB probabilistic distributions of states for places
OUTPUT:
estimatedState: K
words, hashtags, places, geo-coordinates = preProcess(tweets)
for state in stateList do
    likelihood_score_words[state] <- 0
    likelihood_score_hashtags[state] <- 0
    likelihood_score_places[state] <- 0
    for word in words do
        likelihood_score_words[state] += wordDistributions[word][state]*word.count
    end for
    for hashtag in hashtags do
        likelihood_score_hashtags[state]+=hashtagDistributions[hashtag][state]*hashtag.count
    end for
    for place in places do
        likelihood_score_places[state] += placeDistributions[place][state]*place.count
    end for
end for
placeHeuristic = findFreqPlace(places)
geocoordinatesHeuristic = findFreqCityGeocoordinates(geocoordinates)
estimatedState = majorityVoting( likelihood_score_words, likelihood_score_hashtags, likelihood_score_places,
placeHeuristic ,geocoordinatesHeuristic )
```

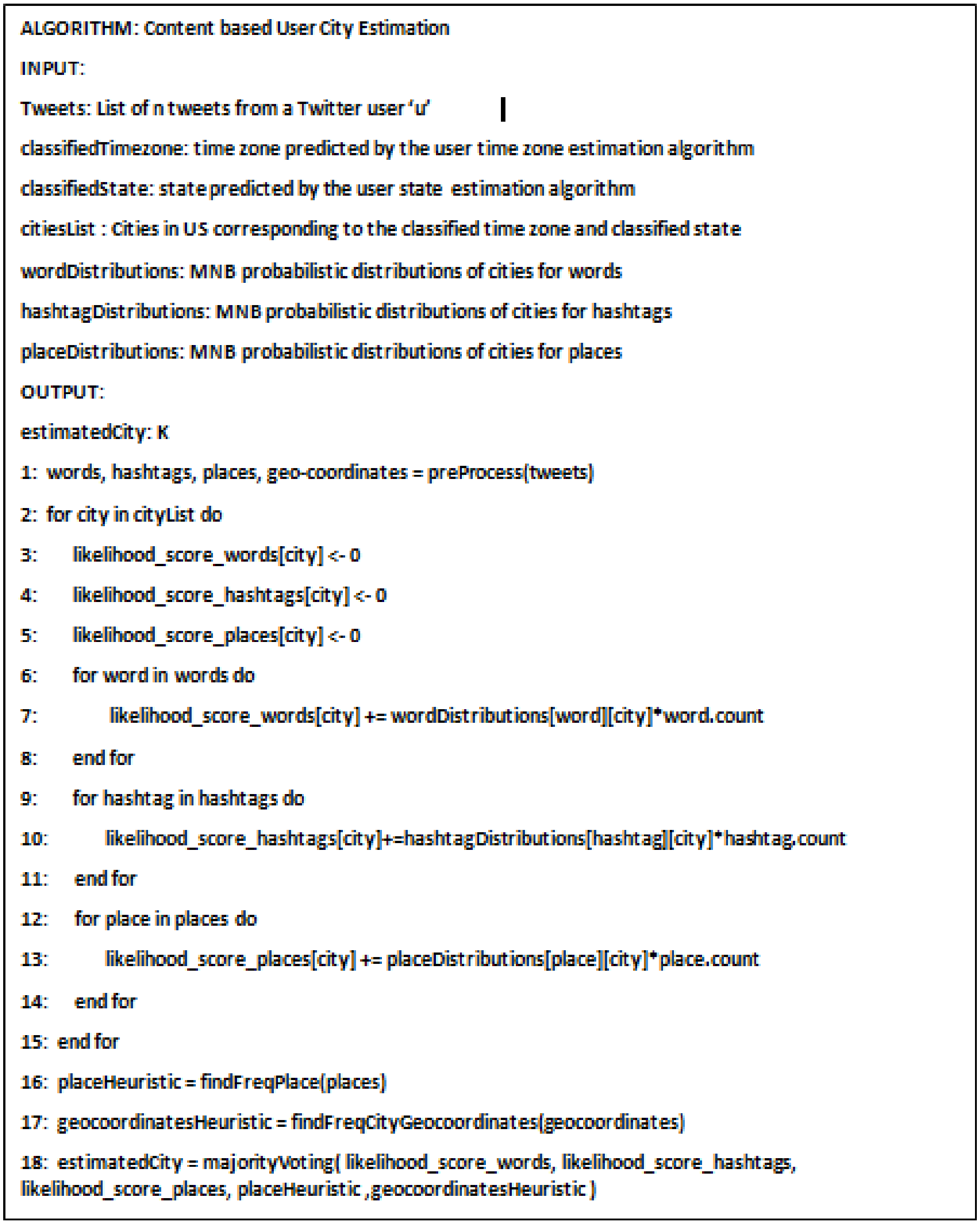
Figure 14 Content based User State Estimation Algorithm

The extracted words, hashtags, place names and geo-coordinates of user tweets are considered as input. The MNB conditional probability of a user in a state present in the estimated time zone is calculated using extracted words and MNB state classifier trained model for words. Similarly, the MNB conditional probability of users in a state, present in the estimated time zone is calculated for the other categories like hashtags and places. Further the city or state name which is identified to be in the city list corresponding to the estimated time zone and mentioned most often by the user in tweets is selected as place heuristic. The corresponding state of the place heuristic is found. Further the extracted geo-coordinates are mapped to city centers using the Google's geo-coding API and only those cities which are present in the city list corresponding to the estimated time zone are retained along with their frequency. The city in which the geo-tagged tweets are found most often is selected as the geo-coordinates heuristic and the corresponding state is found. The states predicted by the content based statistical and heuristic classifiers are passed to the majority voting method. Majority voting is a method to find the state which has received more votes from the classifiers. The state found by the majority voting method is estimated to be the accurate state of the Twitter user.

City Estimation

The purpose of the city estimation algorithm is to estimate the accurate city for a Twitter user based on user tweets, the estimated state selected by the state estimation algorithm and MNB city classifier trained models for words, hashtags and places. The content based user city estimation algorithm is in Figure 15.

The extracted words, hashtags, place names and geo-coordinates of user tweets are considered as input. The MNB conditional probability of a user in a city present in the estimated state is calculated. Similarly, the MNB conditional probability of user in a city present in the estimated state is calculated for the other categories like hashtags and places. Further the city or state name which is identified to be in the city list corresponding to the estimated state and mentioned more often by the user in tweets is selected as place heuristic. Further the extracted geo-coordinates are mapped to city centers using the geo-coding API and only those cities which are present in the city list corresponding to the estimated state are retained along with their frequency. The city in which the geo-tagged tweets are found most often is selected as the geo-coordinates heuristic.



```
ALGORITHM: Content based User City Estimation
INPUT:
Tweets: List of n tweets from a Twitter user 'u'
classifiedTimezone: time zone predicted by the user time zone estimation algorithm
classifiedState: state predicted by the user state estimation algorithm
citiesList : Cities in US corresponding to the classified time zone and classified state
wordDistributions: MNB probabilistic distributions of cities for words
hashtagDistributions: MNB probabilistic distributions of cities for hashtags
placeDistributions: MNB probabilistic distributions of cities for places
OUTPUT:
estimatedCity: K
words, hashtags, places, geo-coordinates = preProcess(tweets)
for city in cityList do
    likelihood_score_words[city] <- 0
    likelihood_score_hashtags[city] <- 0
    likelihood_score_places[city] <- 0
    for word in words do
        likelihood_score_words[city] += wordDistributions[word][city]*word.count
    end for
    for hashtag in hashtags do
        likelihood_score_hashtags[city]+=hashtagDistributions[hashtag][city]*hashtag.count
    end for
    for place in places do
        likelihood_score_places[city] += placeDistributions[place][city]*place.count
    end for
end for
placeHeuristic = findFreqPlace(places)
geocoordinatesHeuristic = findFreqCityGeocoordinates(geocoordinates)
estimatedCity = majorityVoting( likelihood_score_words, likelihood_score_hashtags,
likelihood_score_places, placeHeuristic ,geocoordinatesHeuristic )
```

Figure 15 Content based User City Estimation Algorithm

The cities predicted by the content based statistical and heuristic classifiers are passed to the majority voting method. Majority voting is a method to find the city which has received more votes from the classifiers. The city found by the majority voting method is estimated to be the accurate city of the Twitter user.

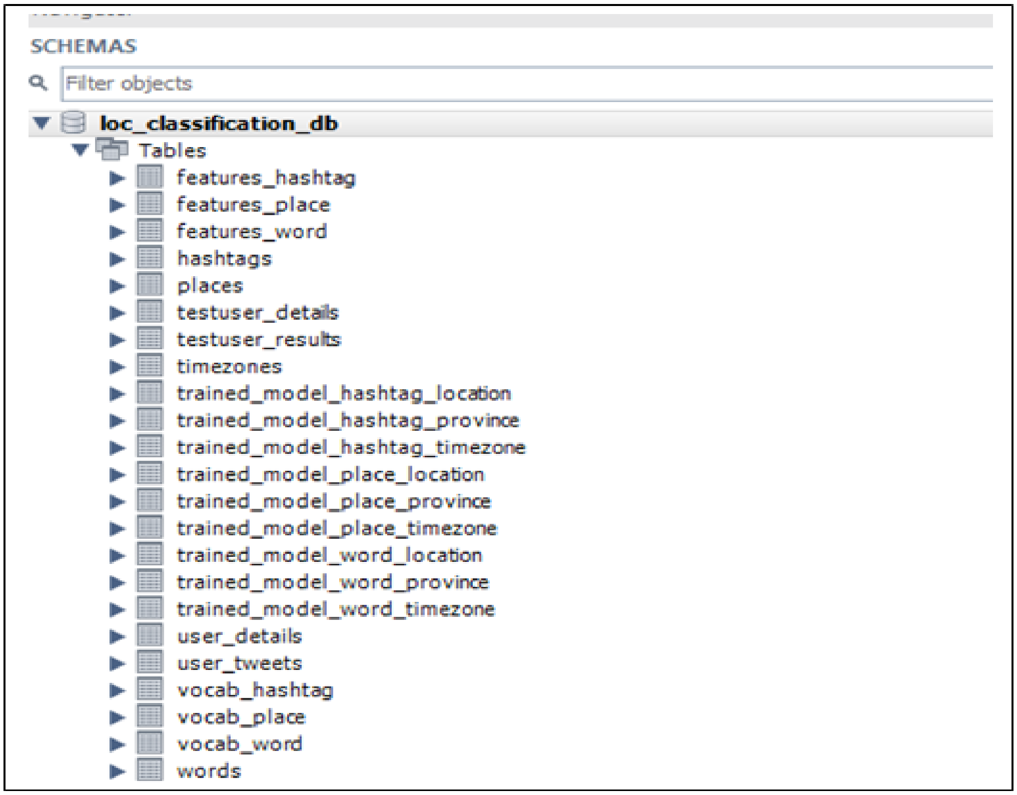
CHAPTER 4 IMPLEMENTATION

Our solution for the hierarchical location classification of Twitter users has been implemented on Pycharm IDE using Python as the programming language and MySQL for the database. The development environment used for the implementation of the proposed hierarchical location classification is presented in the Table 2. The implementation details of the phases of the proposed hierarchical location classification are discussed in this chapter.

Table 2 The development environment used for the implementation of the proposed Hierarchical Location Classification model

Python 3.4.1 [23]	Programming language
Jetbrains Pycharm Community Edition 4.0 [24]	IDE for creating and building the python programs
MySQL Workbench 6.2 [25]	Database
NLTK 3.0 [21]	NLTK provides the text processing libraries for tokenization and tagging.
Twython 3.2.0 [26]	Python wrapper for Twitter API. This library is used to access Twitter API's for sampling the user tweets to create datasets.
Google's Geocoding API v3 [27]	Google's geocoding API is used to obtain the geo-coordinates of the cities considered for hierarchical location classification.

The MySQL database schema and the Pycharm project structure used for the implementation of the proposed solution are present in the Figure 16 and Figure 17 respectively. A database called loc_classification_db is created with 23 tables. The tables user_details and user_tweets are created for storing the training data set information about the Twitter users, their locations and tweets. The tables testuser_details and testuser_results are used to store the test data set tweets and results. The table for time zones has the list of the cities selected as classes for hierarchical location classification along with their associated states and time zones. To store the outputs of the data preprocessing of user tweets, i.e. words, hashtags and places, tables words, hashtags and places are created. Tables features_word, features_hashtag and features_place are created to store the outputs of the feature extraction for words, hashtags and places while the tables vocab_word, vocab_hashtag and vocab_place are created to store the unique terms present in each of these categories. The nine trained model tables are created to store the MNB conditional probability of locations, i.e. time zones, states and cities, for words, hashtags and places.



loc_classification_db
Tables
features_hashtag
features_place
features_word
hashtags
places
testuser_details
testuser_results
timezones
trained_model_hashtag_location
trained_model_hashtag_province
trained_model_hashtag_timezone
trained_model_place_location
trained_model_place_province
trained_model_place_timezone
trained_model_word_location
trained_model_word_province
trained_model_word_timezone
user_details
user_tweets
vocab_hashtag
vocab_place
vocab_word
words

Figure 16 MySQL Database schema of Hierarchical Location Classification

4.1 DATA EXTRACTION

To implement the proposed hierarchical location classification of Twitter users, firstly a data set comprising user tweets is required to build the training classifier models. The top 100 US cities by population are selected as the cities or classes for the location classification [28]. By using Google's geo-coding API [27], the latitude and longitude corresponding to these cities are obtained. A Twitter developer's account is created to have the required permissions to access the user tweets of Twitter users. Once the necessary permissions are obtained, a Twitter handle is created in the python project so that all the programs can access the Twitter handle.

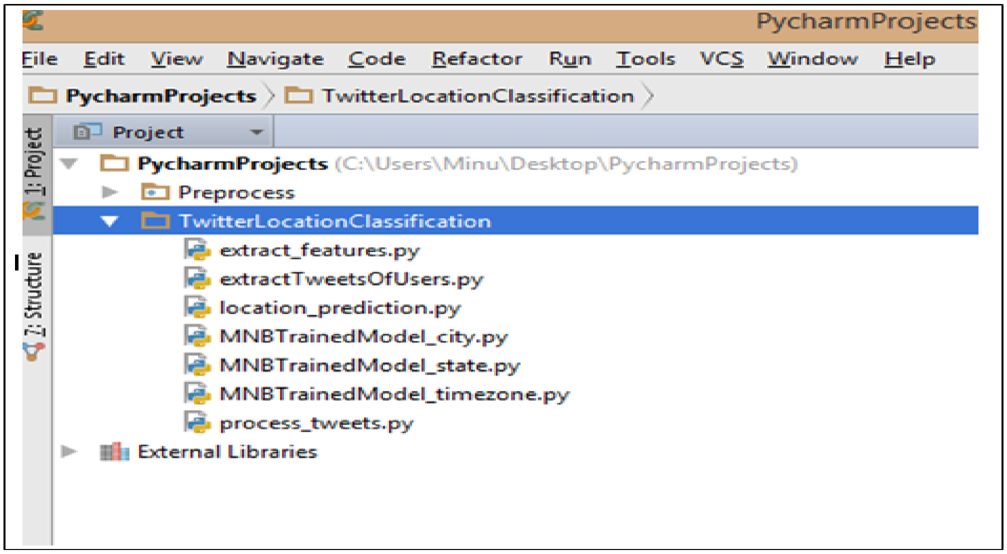
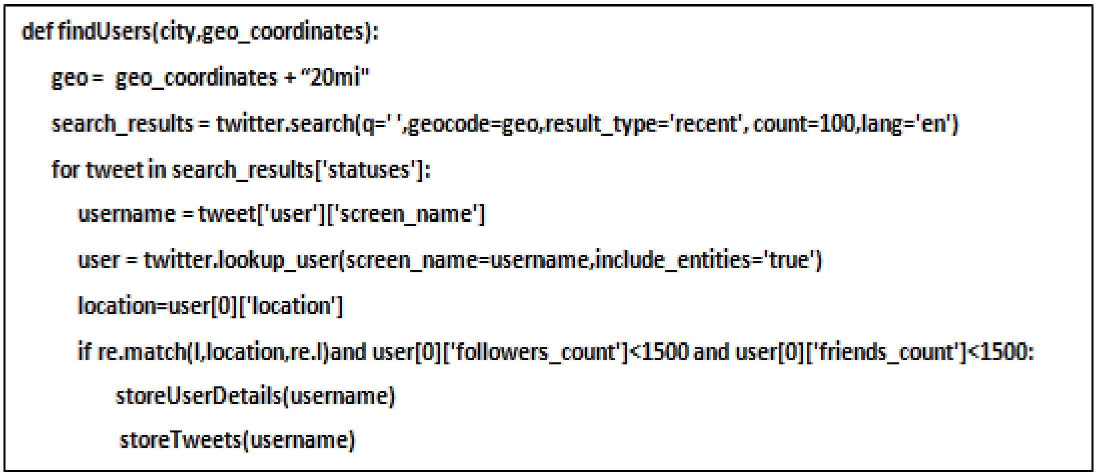

PycharmProjects
File Edit View Navigate Code Refactor Run Tools VCS Window Help
PycharmProjects › TwitterLocationClassification ›
Project
1: Project
2: Structure
PycharmProjects (C:\Users\Minu\Desktop\PycharmProjects)
Preprocess
TwitterLocationClassification
extract_features.py
extractTweetsOfUsers.py
location_prediction.py
MNBTrainedModel_city.py
MNBTrainedModel_state.py
MNBTrainedModel_timezone.py
process_tweets.py
External Libraries

Figure 17 Pycharm Python Project for Hierarchical Location Classification

To find the Twitter users who are located in the selected cities and to collect their tweets from their corresponding timelines on Twitter, python program extractTweetsFromUsers present in the Figure 17 is executed. There is one method called findUsers method in this program for finding the users of a particular city.



```
def findUsers(city,geo_coordinates):
   geo =  geo_coordinates + “20mi"
   search_results = twitter.search(q=' ',geocode=geo,result_type='recent', count=100,lang='en')
   for tweet in search_results['statuses']:
      username = tweet['user']['screen_name']
      user = twitter.lookup_user(screen_name=username,include_entities='true')
      location=user[0]['location']
      if re.match(l,location,re.I)and user[0]['followers_count']<1500 and user[0]['friends_count']<1500:
          storeUserDetails(username)
           storeTweets(username)
```

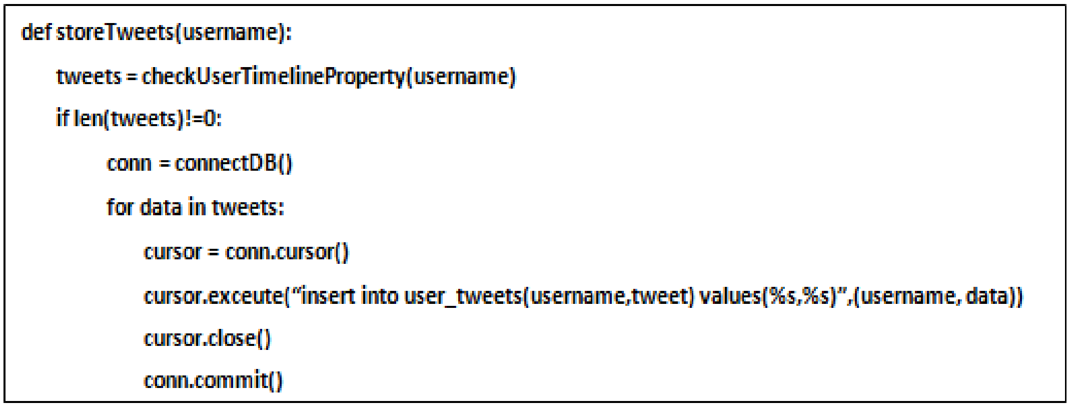
Figure 18 Python method for finding Twitter users

Method findUsers

Users are recorded for each of the selected cities by using the search API [26] of Twitter. To use the search API of Twitter for finding the users, three parameters are to be specified such as the geo-tag filter, number of users and the language. As described in Figure 18, geo-tag filter is set to the geo-coordinates of one of the selected cities as well as range of 20 miles in order to search for the users in a range of 20 miles away from the city center. Number of users is set to 100 as the aim is to store the tweets of 100 users per city. And the language is set to English since the objective is to find the location of a Twitter user based on the English tweets only. A search call is performed using the obtained Twitter handle and the results are stored. The screen name is a unique identifier used by the Twitter to identify a Twitter user. For each of the screen names present in the stored search results, a lookup_user method call is performed using the Twitter handle to find the information corresponding to a user. The results of the loopup_user method call are stored to retrieve the values of the location, the number of friends and followers pertaining to each user in the stored search results for each city. Only those users whose number of friends and followers are less than 1500 are retained to filter the accounts which are related to news, celebrity or events.

Method storeTweets

Once the users are found, storeTweets method is called at the end of the findUsers method to store 100 recent tweets from a user’s timeline. To remove the users who have private profiles, checkUserTimelineProperty method is called from the storeTweets method as described in Figure 19 to find whether a user has a public or private profile. In case a user’s profile is public, checkUserTimelineProperty method returns the 100 recent tweets from a user’s timeline. Else, it returns nothing.

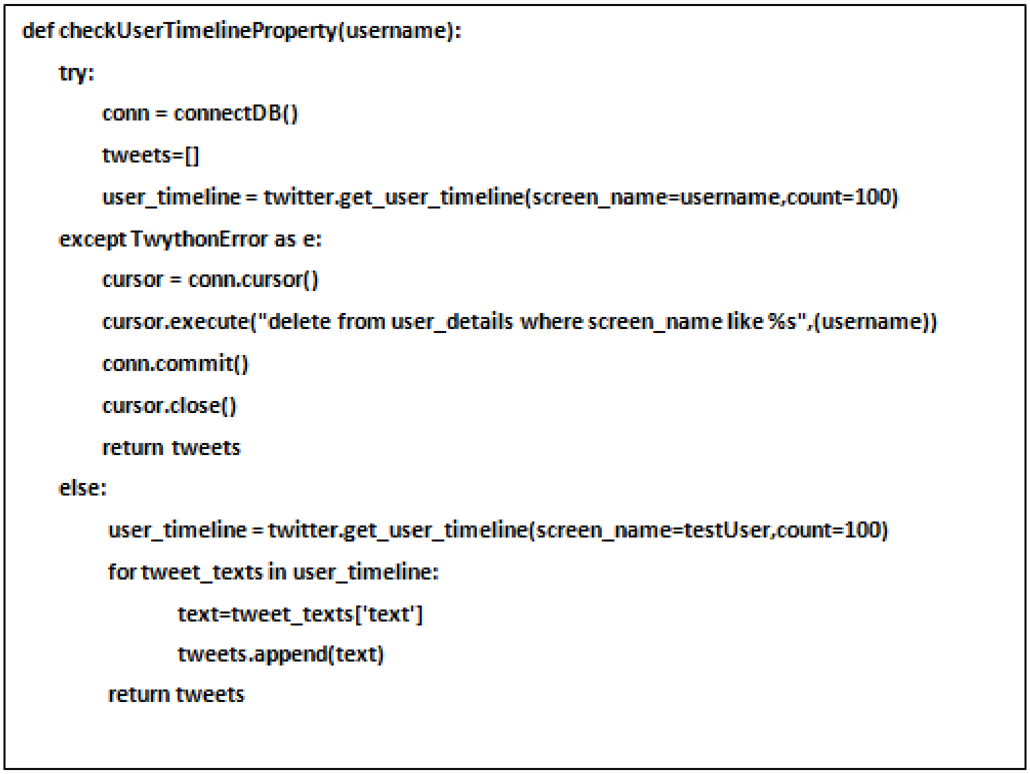


```
def storeTweets(username):
    tweets = checkUserTimelineProperty(username)
    if len(tweets)!=0:
        conn = connectDB()
        for data in tweets:
            cursor = conn.cursor()
            cursor.exceute(“insert into user_tweets(username,tweet) values(%s,%s)”,(username, data))
            cursor.close()
            conn.commit()
```

Figure 19 Python method for storing user tweets

Method checkUserTimelineProperty

In the checkUserTimelineProperty method, Twitter REST API is invoked via the Twitter handle to collect recent 100 tweets of a user from a user’s timeline. If the user’s profile is private, Twitter REST API [26] call throws an exception indicating that the user’s profile is private and hence the account cannot be accessed to retrieve tweets from the timeline. This exception is caught in the python program by including try except else blocks for handling the exception thrown by Twitter. The Twitter REST API call is performed to retrieve tweets in the try block. If the Twitter API call throws an exception, control of the program goes to except block as shown in Figure 20 where the user details are deleted from the database. The code in the else block executes if the try block in the program does not throw any exception. If the user’s profile is public, the control of the program returns to else block upon executing the try block. In the else block, user tweets are fetched from a user's timeline and stored. The stored user tweets are returned to the storeTweets method and the user tweets are stored into the database.

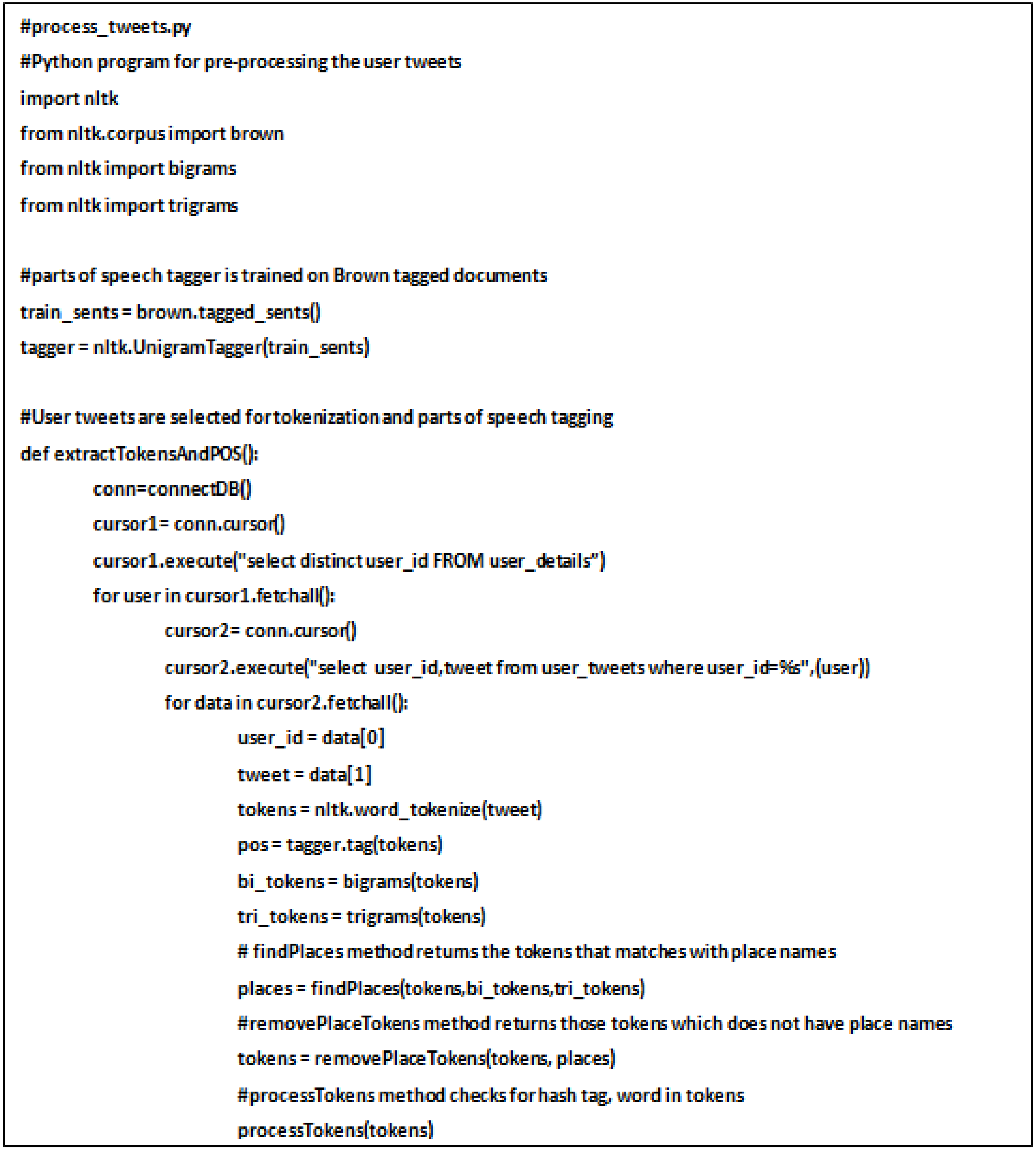


```
def checkUserTimelineProperty(username):
    try:
        conn = connectDB()
        tweets=[]
        user_timeline = twitter.get_user_timeline(screen_name=username,count=100)
    except TwythonError as e:
        cursor = conn.cursor()
        cursor.execute("delete from user_details where screen_name like %s",(username))
        conn.commit()
        cursor.close()
        return tweets
    else:
        user_timeline = twitter.get_user_timeline(screen_name=testUser,count=100)
        for tweet_texts in user_timeline:
            text=tweet_texts['text']
            tweets.append(text)
        return tweets
```

Figure 20 Python method for checking the privacy property of Twitter users

4.2 DATA PREPROCESSING

Natural Language Tool Kit called NLTK [21] is a platform available for building python programs to work with text data. It provides text processing libraries for tokenization, stemming, tagging, parsing and classification. NLTK libraries are downloaded and integrated with the Pycharms IDE [24]. To achieve the data preprocessing of user tweets, NLTK library is imported to perform the tokenization and parts of speech tagging.



```
#process_tweets.py
#Python program for pre-processing the user tweets
import nltk
from nltk.corpus import brown
from nltk import bigrams
from nltk import trigrams

#parts of speech tagger is trained on Brown tagged documents
train_sents = brown.tagged_sents()
tagger = nltk.UnigramTagger(train_sents)

#User tweets are selected for tokenization and parts of speech tagging
def extractTokensAndPOS():
        conn=connectDB()
        cursor1= conn.cursor()
        cursor1.execute("select distinct user_id FROM user_details")
        for user in cursor1.fetchall():
                cursor2= conn.cursor()
                cursor2.execute("select  user_id,tweet from user_tweets where user_id=%s",(user))
                for data in cursor2.fetchall():
                        user_id = data[0]
                        tweet = data[1]
                        tokens = nltk.word_tokenize(tweet)
                        pos = tagger.tag(tokens)
                        bi_tokens = bigrams(tokens)
                        tri_tokens = trigrams(tokens)
                        # findPlaces method returns the tokens that matches with place names
                        places = findPlaces(tokens,bi_tokens,tri_tokens)
                        #removePlaceTokens method returns those tokens which does not have place names
                        tokens = removePlaceTokens(tokens, places)
                        #processTokens method checks for hash tag, word in tokens
                        processTokens(tokens)
```

Figure 21 Python program for pre-processing of tweets

Program process_tweets displayed in Figure 21 is executed for categorizing the terms in user tweets into words, hashtags and places. Using the NLTK library, user tweets are broken into unigrams, bigrams and trigrams. For the unigram tokens, parts of speech are found using the POS tagger of the NLTK library [21] where the NLTK POS tagger is trained on the brown corpus which is built on the trained tagged documents. Further unigrams, bigrams and trigrams are passed to findPlaces method in order to find the tokens which matches with the list of cities and their associated provinces that are selected for location classification. This method returns all the places found among the tokens passed to it. The found places are removed from the tokens in the removePlaceTokens method. Now the selected tokens after removing place names are sent to the processTokens method for classifying a token as either word or hashtag. The words, hashtags, place names are extracted from tweets for each user present in each city of our data set and are stored in the database. The term frequency and user frequency of the extracted words, places and hashtags are computed with respect to a city and are also stored.

4.3 FEATURE EXTRACTION

Program extract_features is executed to extract the local terms or features for the categories such as words, hashtags and places. In case of words category, only those words which are tagged as nouns, proper nouns and none are considered. For each of these categories, the terms which are used by at least 5% of people are retained for further feature extraction. Then the average and the maximum condition probability of locations for each of the terms is calculated. The difference between the average and maximum condition probability of locations for terms is computed and only those terms in locations for which the calculated difference is above 0.1 and the maximum conditional probability of locations for terms is above 0.5 are retained. Terms which meet the specified criteria are extracted and stored.

4.4 HIERARCHICAL LOCATION CLASSIFICATION

Program MNBTrainedModel_timezone is executed to find the MNB probability of time zones for the extracted features and the calculated MNB probability value is stored for the features in relation to time zones. Similarly, the programs MNBTrainedModel_state and MNBTrainedModel_city are executed to find the MNB based probability of states for the extracted features and MNB probability of cities for extracted features respectively. And the calculated MNB probability values are stored into the database.

Program location_prediction is designed to estimate the city for the test user’s data set. This program estimates a home location for Twitter user by considering ‘n’ user tweets where n is chosen empirically, time zones in the US and MNB probability distributions of time zones, states and cities for words, hashtags and places. Firstly, content based user time zone estimation algorithm is called upon to estimate the time zone. Then the estimated time zone for a user is used by the content based user state estimation algorithm to confine the search only to the states present in the estimated timezone and estimates the state for the user. Lastly, the content based user city estimation algorithm takes the estimated state and time zone into consideration to confine the search only to the cities present in the estimated state and time zone and estimates the city for the user. The estimated time zones, states and cities for test users are stored in the testuser_results table.

CHAPTER 5 EVALUATION AND RESULTS

Experiments are conducted to evaluate the performance of the proposed hierarchical location classification method of Twitter users. The evaluation of various aspects of the proposed method is performed by calculating the accuracy of location classification. Accuracy is defined to be the number of test users for whom the location estimations by the proposed method are correct by the total number of test users. Equation 9 presents the equation to calculate the accuracy of location classification of Twitter users. If the total test users are ‘n’ and the number of test users who are located correctly to their actual locations is ‘n0’, then

$$\boxed{\textbf{Accuracy of location classification} = n_0 \div n} \qquad (9)$$

The following are the scenarios that are put into implementation for demonstrating the performance of the proposed method

- To investigate the overall performance of the hierarchical location classification and the performance of the individual classifiers present in the location classification at three hierarchy levels i.e. time zone, state and city.

- To investigate the accuracy of location classification of Twitter users over different numbers of cities considered as classes for location classification.

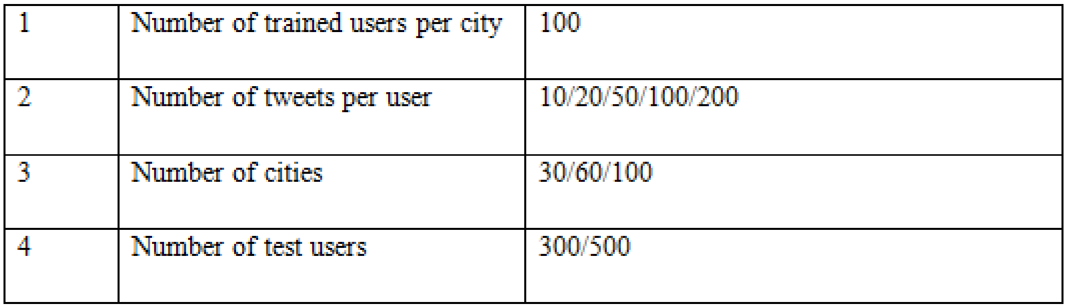
- To evaluate and analyze the accuracy of location classification for an increase in the size of trained model built for location classification.

- To investigate the variation in the performance of location prediction over the increase in the number of tweets for a test user.

- To compare the performance of the proposed method with an existing work.

The experimental setup for evaluating the proposed method is presented in the Table 3.

Table 3 Experimental setup details



1	Number of trained users per city	100
2	Number of tweets per user	10/20/50/100/200
3	Number of cities	30/60/100
4	Number of test users	300/500

5.1 OVERALL PERFORMANCE OF THE HIERARCHICAL LOCATION CLASSIFICATION

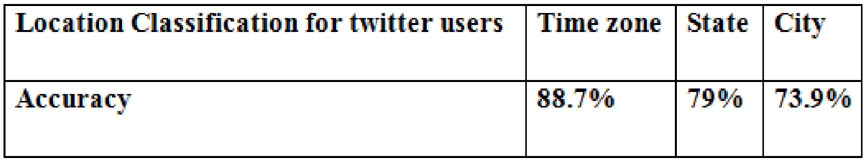
The objective of this scenario is to evaluate the overall performance of the hierarchical location classification at three hierarchy levels, i.e. time zone, state and city, and the also to evaluate the performance of the individual classifiers present in each of the three hierarchical levels of the hierarchical location classification of Twitter users.

The training data set for this scenario is prepared by first acquiring the list of the users belonging to the selected 20 cities in each of the three time zones in the US such as EST, CST and PST. In total, the selected cities are 60 and states are 32. A maximum of 100 tweets if available or less than 100 tweets are collected from each of the 100 users belonging to the selected cities which adds up to a total of 6000 users and 583472 tweets. This data set is processed and broken into three categories such as words, hashtags and places using the preprocessing python program. Further, the features for each of these categories are extracted using the feature extraction python program. The extracted features are sent to the respective MNB location based classifier algorithm to build the trained classifiers for time zone, state and city hierarchy levels.

By taking the cities of the data set into consideration, a test data set is formed by collecting a maximum of 200 tweets if available or less than 200 tweets from each of the selected 500 users who are not considered for the training data set and for whom the home locations are already available in their Twitter profiles. The location prediction python program is executed to estimate the location for the users present in the test dataset and the estimated locations such as time zone, state and city are stored into the testuser_results table for analysis.

Accuracy of location classification of test users is calculated and presented in the Table 4. 88.7% of 500 users are located correctly to their actual time zones, 79% users are located correctly to their actual states and 73.9% of users are located correctly to their actual cities. As the location classification of Twitter users goes from time zone to city level, a drop in the accuracy percentage levels is observed.

Table 4 Accuracy of location classification at different location hierarchy levels



Location Classification for twitter users	**Time zone**	**State**	**City**
Accuracy	**88.7%**	**79%**	**73.9%**

The performance of location classification for users at lower hierarchical locations depends on the performance at higher hierarchical locations. This is because the proposed location classification approach first estimates a time zone for a user and then performs the content based user state estimation by searching only among the states present in the estimated time zone to zero in on an accurate state for the user. Similarly, accurate city is estimated for the user by performing the content based user city estimation with the cities present in the estimated state only. The possibility of correctly estimating states for users is more when their time zones are correctly estimated where as there is no possibility of correctly estimating states for users when their time zones are incorrectly estimated.

One more difference between the content based user time zone estimation, state estimation and city estimation classifiers is the number of comparisons that are to be made with that of locations in respective classifiers in order to estimate user's location. As the location classification proceeds from time zone to city level, there is an increase in the number of comparisons with that of locations. In a content based user time zone estimation classifier, the algorithm performs the comparison of user tweets to that of 3 time zones in order to estimate time zone for the user. While content based user city estimation algorithm requires 20 comparisons as there are 20 cities in a time zone. Less number of comparisons or classes in a location classification, more the accuracy of the classification. Since the number of comparisons or classes in a time zone level classification is less compared to that of state and city levels, the accuracy of location classification of Twitter users at time zone level is more. Hence, the accuracy of location classification of lower hierarchical location in any case is lesser than or equal to the accuracy at higher hierarchical location.

These are the reasons for observing a drop in the accuracy percentage levels of location classification as the classification goes from time zone to city level. Thus, a higher accuracy of time zone classification for Twitter users helps in achieving a high accuracy for both state and city classification.

The accuracy of the individual classifiers, such as words statistical classifier, hashtag statistical classifier, place statistical classifier, place heuristic classifier and geo-coordinates heuristic classifier, at each of three location hierarchy levels in location classification approach is calculated to observe the contribution of these individual classifiers in estimating Twitter user's location. It is observed from Figure 22 that the geo-coordinates heuristic classifier has given the best accuracy in estimating Twitter user's location compared to the other individual classifiers at each of the three location hierarchy levels. Geo-coordinates heuristic classifier reported an accuracy of 86.3%, 76.5% and 72.9% at time zone, state and city levels in location classification. After geo-coordinates heuristic classifier, places heuristic and places statistic classifiers achieved a better accuracy of above 40% at all three location hierarchy levels in location classification. Words statistic and hashtags statistic classifiers reported a less accuracy in estimating user locations. While words statistic classifier achieved an accuracy of above 25% and below 36%, hashtags statistic accuracy went down from 35.7% to 11.4% as the location classification moved from time zone to city level. Thus, the contribution of the geo-coordinates heuristic, places heuristic and places statistic classifiers in estimating the user location is significant.

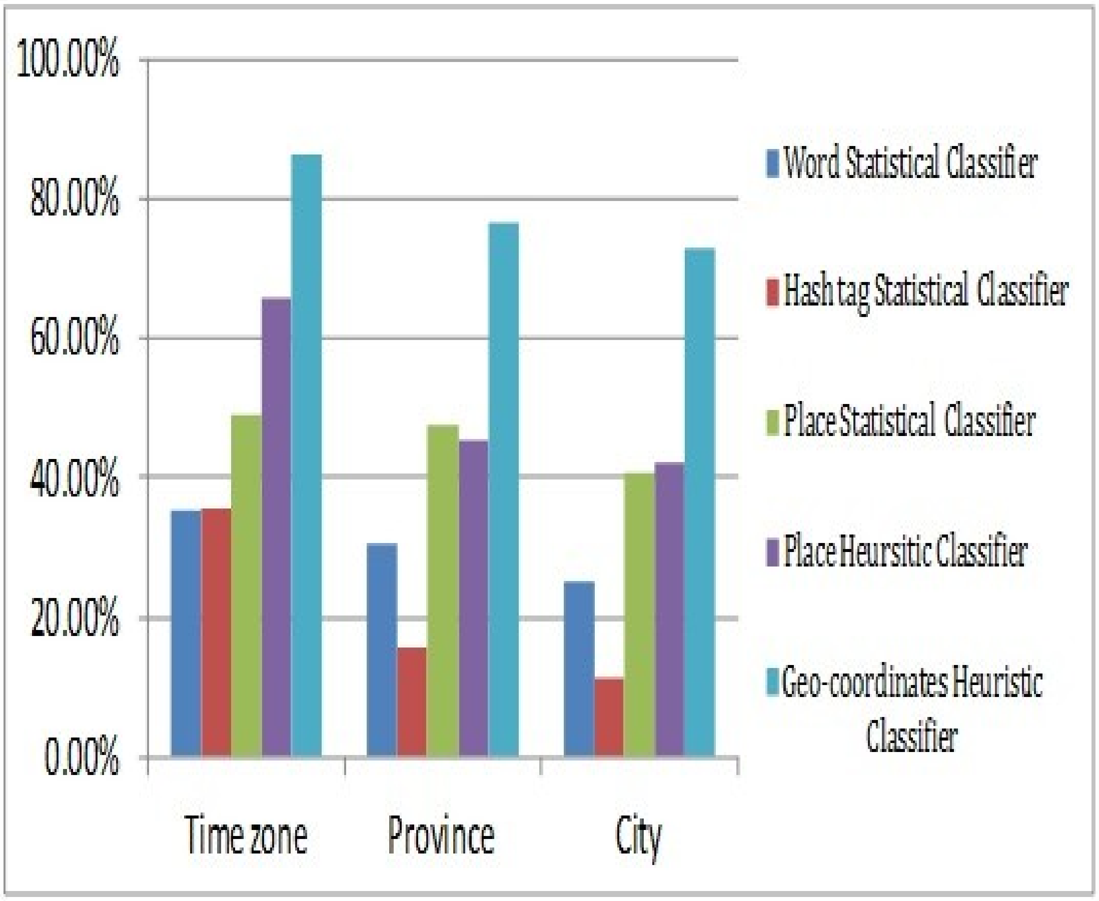


Figure 22 Accuracy of individual classifiers at different location levels in hierarchical location classification

5.2 ACCURACY OF PROPOSED METHOD OVER AN INCREASE IN NUMBER OF CITIES

The objective of this scenario is to observe the impact of the increase in the trained classes or cities on the accuracy of location classification for Twitter users.

Three training data sets are prepared for this scenario. For the first data set, 3 time zones in the US i.e. EST, CST and PST are selected and then 10 cities are selected from each of these time zones. For the second data set, 3 time zones in the US i.e. EST, CST and PST are selected and then 20 cities are selected from each of these time zones. Thirdly, 4 time zones in the US i.e. EST, CST, PST and MST are selected and then 25 cities are selected from each of these time zones. Once the time zones and cities are selected for the data sets, each of these data sets is formed by collecting a maximum of 100 tweets if available or less than 100 tweets for each of the 100 users belonging to the respective selected cities. These data sets are separately passed as inputs to the preprocessing python program and feature extraction python program. For each dataset, the extracted features are sent to the respective MNB location based classifier algorithm to build the trained classifiers for time zone, state and city hierarchy levels.

By taking the cities of three datasets into consideration, three test data sets are separately formed by collecting a maximum of 200 tweets if available or less than 200 tweets from selected 300, 500 and 1000 users who belongs to the cities considered for the respective training data sets and who are not considered for the respective training data sets. For each of the training data sets and its respective testing dataset, location prediction python program is executed to estimate the location for the users present in the test data sets and the estimated locations such as time zone, state and city are stored into the testuser_results table for analysis.

For the three test data sets, the accuracy of the content based user time zone estimation classifiers, content based user state estimation classifiers and the content based user city estimation classifiers are calculated to analyze the variation in performance of location classification at different location hierarchy levels with respect to the number of cities used as training classes for location classification and is presented in Figure 23.

It is observed that with an increase in the number of cities used as training classes for location classification for Twitter users, there is a slight decrease in the accuracy percentage levels of location classification at time zone, state and city levels. For location classification with trained classes at 30, 60 and 100 cities, the city prediction accuracy is observed at 81.6%, 73.90% and 70.7% respectively. With the proposed location classification approach of Twitter users, a slight fall in accuracy is observed over an increase in trained number of classes or cities used for location classification.

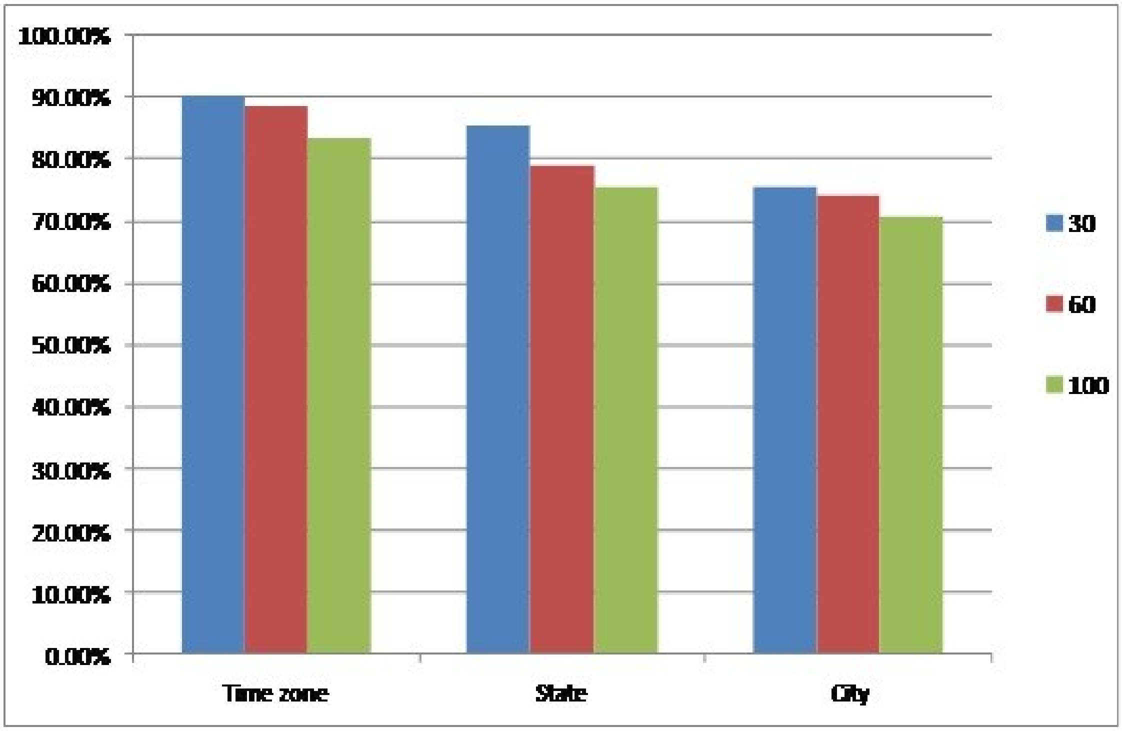


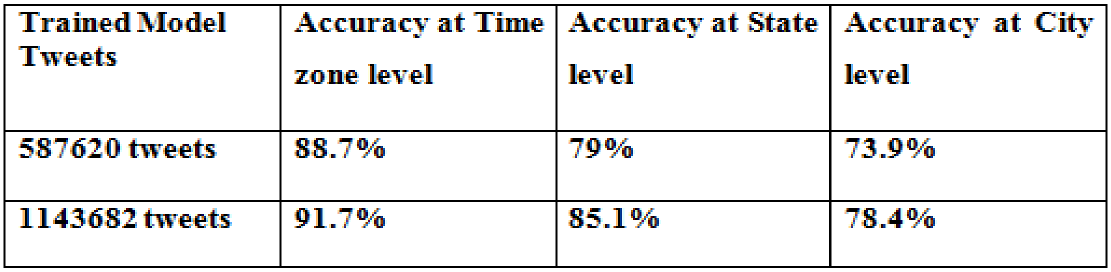
Figure 23 Accuracy of hierarchical location classification over increase in training cities

5.3 ACCURACY OF PROPOSED METHOD OVER AN INCREASE IN SIZE OF TRAINING MODEL

The objective of this scenario is to analyze the impact of the size of trained model, i.e. user tweets on the accuracy of location classification for a fixed number of trained classed or cities used for classification.

Two data sets are prepared to evaluate this scenario. Firstly, three time zones in the US such as EST, CST and PST are selected and then 20 cities belonging to each of these time zones are selected. Then a list of 100 users belonging to each of the selected cities is obtained. A total of 3 time zones, 60 cities and 100 users per city are considered for both the data sets. The first data set is formed by collecting a maximum of 100 tweets if available or less than 100 tweets from each of the selected users. A total of 587620 tweets is collected for the first data set. Similarly, the second data set is formed by collecting a maximum of 200 tweets if available or less than 200 tweets from each of the selected users. A total of 1143682 tweets is collected for the second dataset. These datasets are separately passed as inputs to the preprocessing python program and feature extraction python program. For both the data sets, the extracted features are sent to the respective MNB location based classifiers algorithm to build the trained classifiers for time zone, state and city hierarchy levels.

Table 5 Accuracy of hierarchical location classification for different sizes of training tweets



Trained Model Tweets	Accuracy at Time zone level	Accuracy at State level	Accuracy at City level
587620 tweets	88.7%	79%	73.9%
1143682 tweets	91.7%	85.1%	78.4%

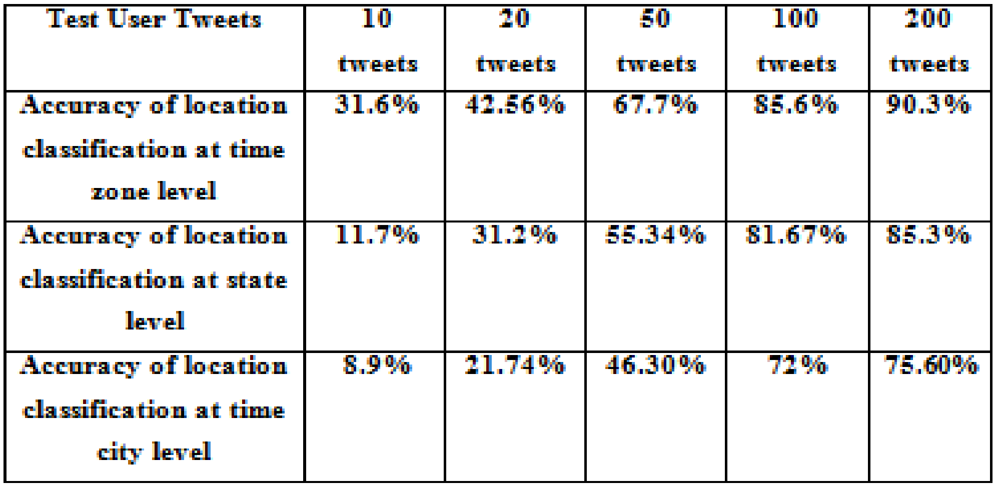
Test data set is prepared by collecting a maximum of 200 tweets if available or less than 200 tweets from each of the selected 300 users who belong to the cities considered for the training data sets and who are not considered for the training data sets. Using both the training data sets, location prediction python program is executed once to estimate the location for the users present in the test data set and the estimated locations such as time zone, state and city are stored into the testuser_results table and is presented in Table 5 for analysis.

With an increase in the number of user tweets used for building the training classifiers for a fixed trained classes or cities used in location classification, an increase in the accuracy of the location classification for Twitter users in time zone, state and city levels are observed. For the same 60 cities used as training classes in location classification, the accuracy of estimating city of Twitter users by the proposed approach went up from 73.9% to 78.4% on increasing the size of the tweets in the trained model by almost two times. A considerable increase of 5% accuracy is achieved with the twice the number of tweets used for training the classifiers for location classification for Twitter users. Trained classifiers are built on the features or the local terms which are extracted from the user tweets. By increasing the number of tweets used for training the classifiers for location classification, more number of features are extracted and in turn the training classifiers are built on more features. Since the training classifiers are built on a significant set of extracted features, the ability of the location prediction to estimate Twitter user locations has significantly improved. Therefore, it is observed by using a significant number of tweets for building the training classifiers for location classification, a better accuracy of location estimation for Twitter users with the proposed location classification approach can be achieved.

5.4 ACCURACY OF PROPOSED METHOD OVER AN INCREASE IN NUMBER OF TEST USER TWEETS

The objective of this scenario is to analyze the impact of the number of test user tweets used on the accuracy of location prediction.

Table 6 Comparative evaluation of accuracy of location classification at different levels over an increase in test user tweets



Test User Tweets	10 tweets	20 tweets	50 tweets	100 tweets	200 tweets
Accuracy of location classification at time zone level	31.6%	42.56%	67.7%	85.6%	90.3%
Accuracy of location classification at state level	11.7%	31.2%	55.34%	81.67%	85.3%
Accuracy of location classification at time city level	8.9%	21.74%	46.30%	72%	75.60%

Firstly, three time zones in the US such as EST, CST and PST are selected and then 10 cities belonging to each of these time zones are selected. Then a list of 100 users belonging to each of the selected cities is obtained. A total of 3 time zones, 30 cities and 100 users per city are considered and training data set is formed by collecting a maximum of 100 tweets if available or less than 100 tweets from each of the selected users. Further, the training data set is passed to the preprocessing python program, feature extraction python program and the extracted features are sent to the respective MNB location based classifier algorithm to build the trained classifiers for time zone, state and city hierarchy levels.

Secondly, a set of 300 users who belong to the cities used for location classification and who are not considered for the training datasets is selected. Then five test data sets are prepared by collecting 10, 20, 50, 100 and 200 tweets respectively for each of the 300 users. For all the test data sets, location prediction python program is executed once to estimate the location for the users present in the test dataset and the estimated locations such as time zone, state and city are presented in the Table 6 for analysis. From the Figure 24, it can be observed that the accuracy of location classification at time zone, state and city levels has improved. An increase in the city level prediction accuracy from

8.9% to 75.60% is achieved by our method upon increasing the number of tweets of test users from 10 tweets to 200 tweets.

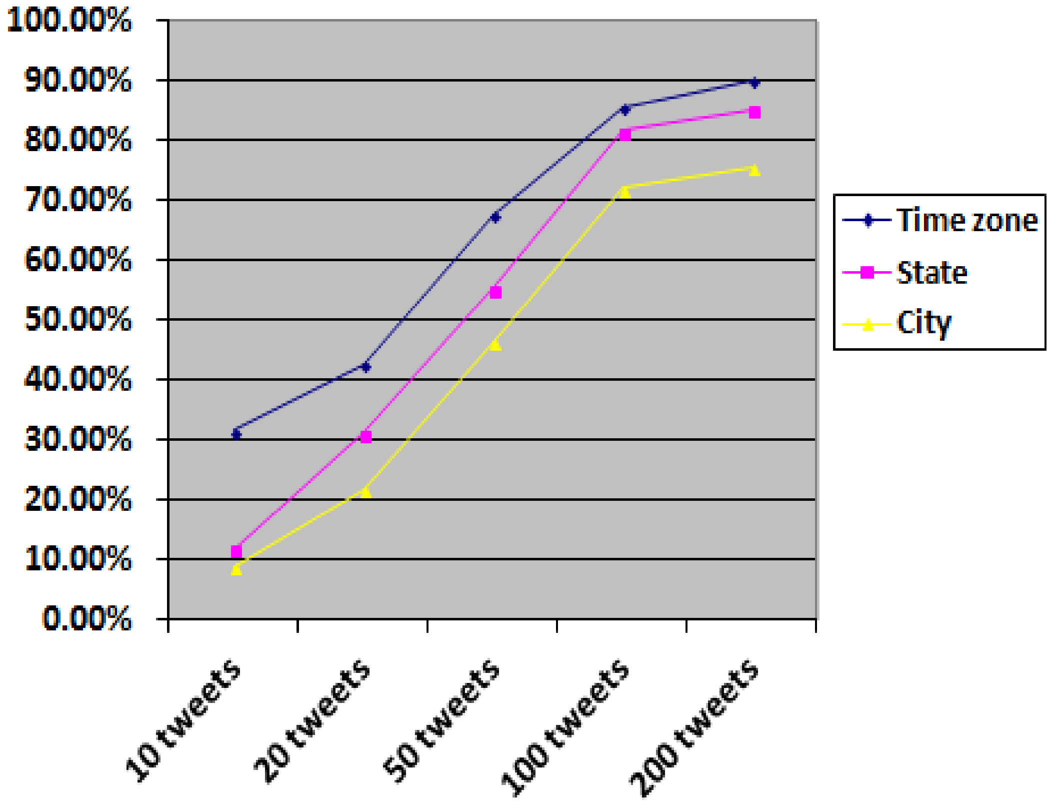
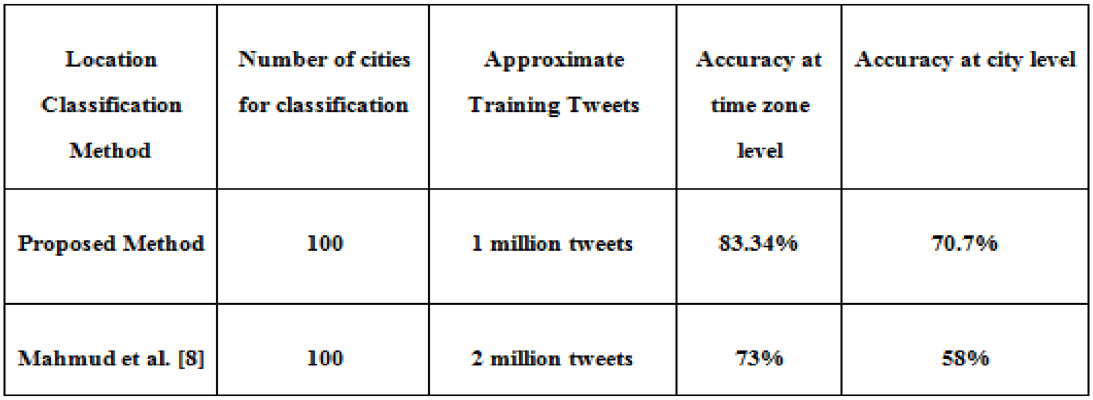


Figure 24 Accuracy of hierarchical location classification for different number of test user tweets

5.5 COMPARITIVE EVALUATION OF PROPOSED METHOD WITH EXISTING WORK

Evaluation of our work with Mahmud et al. [8] content based hierarchical location classification method is performed an the results are presented in the Table 7. The results demonstrate that for the same number of the US cities considered for location classification, our method achieved a better accuracy of 83.34% and 70.7% at time zone and city level respectively, compared to 73% and 58% accuracy achieved by Mahmud et al. [8] method at time zone and city level respectively. The size of the trained model used in our method is almost half of that of the Mahmud's method and our method is able to achieve a better accuracy with less number of tweets used as a training model for hierarchical location classification of Twitter users.

Table 7 Comparison of the proposed and Mahmud et al. [8] Methods of hierarchical location classification of Twitter users



Location Classification Method	Number of cities for classification	Approximate Training Tweets	Accuracy at time zone level	Accuracy at city level
Proposed Method	100	1 million tweets	83.34%	70.7%
Mahmud et al. [8]	100	2 million tweets	73%	58%

5.6 SUMMARY

The performance of the proposed hierarchical location classification model of Twitter users is investigated by evaluating and analyzing the prediction accuracy at the city level for the considered scenarios. For a fixed number of cities i.e. 60 cities, considered for location classification, we have observed that our method has achieved a prediction accuracy of 88.7%, 79% and 73.9% at the time zone, state and city levels respectively. The contribution of the individual classifiers to the location classification in each of the location levels is evaluated and it is observed that the geo-coordinates heuristic classifier, places heuristic and places statistic classifiers achieved a better accuracy of above 40% at all the location levels in location classification compared to the word statistic and hashtag statistic classifiers which have reported a less accuracy in estimating user locations. Thus, the contribution of the geo-coordinates heuristic, places heuristic and places statistic classifiers to our hierarchical location classification with a content based probability model is significant in estimating the Twitter user locations.

The accuracy of our method is investigated for a different number of cities, states and time zones used for location classification. With our method, a city prediction accuracy of 75.6%, 73.90% and 67.7% is achieved by 30, 60 and 100 cities respectively considered for location classification. Upon increasing the number of cities for location classification, a drop is observed in the city level accuracy. Only a slight fall in accuracy is reported by our method over an increase in trained number of classes or cities used for location classification. Hence, our method demonstrates to perform better even in a location classification with the most number of cities.

For an increase in the number of tweets used for training the classifiers in our method, the prediction accuracy is evaluated. Our method has achieved an accuracy of 73.9% and 78.4% upon increasing the size of the tweets in the trained model by almost two times. An increase in the number of features extracted for training the classifiers is reported with an increase in the tweets considered for location classification. The advantage of considering a huge trained model for classification improved the prediction accuracy of our method. The disadvantage is that the process of building classifiers for huge trained model is time consuming. Thus, it is important to consider an appropriate training model size for location classification that ensures a high prediction accuracy.

Further, the performance of our method is evaluated by considering a various number of test user tweets. Our method has reported an increase in the city level prediction accuracy from 8.9% to 75.60% on considering an increase in the number of test user tweets from 10 tweets to 200 tweets. Thus, the number of test user tweets used for location classification plays an important role in the prediction accuracy of our method. The drawback is that for Twitter users with less number of tweets, our method performs with a less accuracy.

Our method is compared with the existing hierarchical location classification method to observe the performance achieved by these methods. On comparison with the Mahmud et al. [8], our method has performed better with an accuracy of 70.7% compared to 58% achieved by Mahmud et al. [8]. The existing method proposed two levels of hierarchy for location classification where the tweet creation times of the users belonging to different time zones is considered to estimate time zones for users. The tweet metadata is not employed by the city level classifier to estimate cities for users while our proposed method considers an ensemble of content and tweet metadata based classifiers at three hierarchy levels in the classification method. Tweet metadata gives away the location information of users, which is exploited by our method to estimate location at different hierarchy levels of classification. These factors have boosted the performance of the proposed method and led to a jump in the accuracy of city prediction. The time taken by the proposed method in order to estimate the location of Twitter users is computed for a user and also 500 users. The location prediction program took 45 seconds to predict the location with our proposed method while three hours for estimating locations for 500 users.

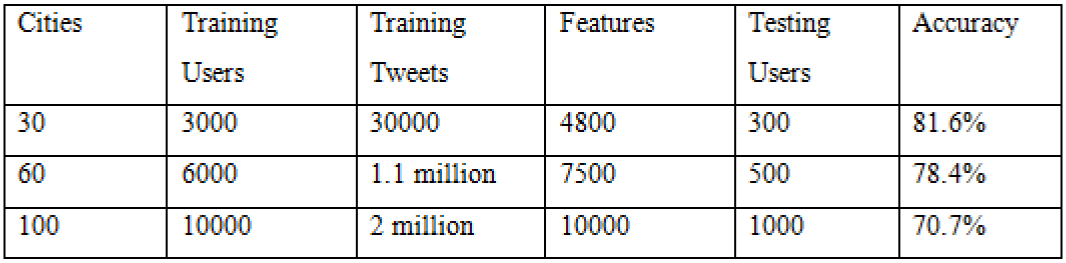
CHAPTER 6 CONCLUSION

To overcome the unreliability of locations reported by Twitter users and to enable the location based personalized services, a hybrid approach of hierarchical location classification and tweet geo-location is proposed to predict home location based only on the tweet content and metadata posted by users.

Our approach is a hierarchical location classification method which uses an ensemble of content based statistic classifiers trained on words, hashtags, places and heuristic classifiers for place names, geo-coordinates in tweets to predict locations at different granularities like time zone, state and city. MNB algorithm is used for training the content based statistic classifiers due to the huge size of the classes considered for location classification. Three classification algorithms such as content based user time zone estimation algorithm, content based user state estimation algorithm and content based user city estimation algorithm are used as a part of the overall location classification method to predict city for users. With the proposed location classification method, first a time zone is estimated for a user by using user's content and content based user time zone estimation algorithm. Then the estimated time zone is used to limit the search of the content based state estimation algorithm to states present only in this time zone followed by restricting the search of content based user city estimation algorithm to cities present only in the estimated state to estimate city for the user.

The location classification method is evaluated on a sampled Twitter data set consisting of user tweets for 100 cities of the US and a city prediction accuracy of 70.7% is observed. Evaluation of our work with Mahmud's content based hierarchical location classification method [8] demonstrates that our method achieved a better accuracy of 83.34% and 70.7% at time zone and city level respectively, compared to 73% and 58% accuracy achieved by Mahmud's method at time zone and city level respectively. The accuracy results for the proposed method are presented in the Table 8.

Table 8 Accuracy of city prediction for different test cases



Cities	Training Users	Training Tweets	Features	Testing Users	Accuracy
30	3000	30000	4800	300	81.6%
60	6000	1.1 million	7500	500	78.4%
100	10000	2 million	10000	1000	70.7%

Several areas of future research are identified to better the performance of our proposed hierarchical location classification model of Twitter users.

- Firstly, future work will consider detecting users who are on travel via tweets and then eliminate them to train the classifiers for our method and evaluate the performance of our method without travelling users' information.

- Secondly, future work will consider applying our location classification method to countries other than US and other languages in order to find its performance with respect to a different size of cities, states, time zones in other countries and also to evaluate the performance of the proposed method over languages other English.

- Thirdly, future work will leverage the social network information of Twitter users for inferring location. Our proposed hierarchical location classification method cannot find an accurate location for users in case of no tweets or less tweets. It may not promise good results when the features extracted f the trained model do not match with that of terms present in the user tweets. To overcome this issue, the social network information can be leveraged to find the nearest neighbor of users in order to predict the location of users [10].

- Fourthly, our future work will consider to identify the entities or topics for a city on the Wikipedia page for that city similar to Krishnamurthy et al. [9] work. The identified cities can be extracted to improve the feature extraction stage of our method and then evaluate the overall performance of our method by using Wikipedia as a knowledge base to improve the feature extraction for building the training classifiers for location classification of Twitter users.

- Fifthly, our future work will employ the US geographical gazetteer to identify the geographical entities in tweets to evaluate its impact on the performance of our method.

- Lastly, our future work will consider various machine learning algorithms for training the classifiers for location classification and will identify the best algorithm that can be incorporated into our hierarchical location classification model of Twitter users.

BIBLIOGRAPHY

[1] Twitter Usage. https://about.twitter.com/company. Accessed: 2015-03-05.

[2] Chandra, S., Khan, L., & Muhaya, F. B. Estimating twitter user location using social interactions--a content based approach. *In Privacy, Security, Risk and Trust (PASSAT) and 2011 IEEE Third Inernational Conference on Social Computing (SocialCom), 2011 IEEE Third International Conference on pp. 838-843, IEEE.*

[3] Bo, H. A. N., & BALDWIN, P. C. T. Geolocation prediction in social media data by finding location indicative words. *In Proceedings of COLING 2012: Technical Papers, pp. 1045-1062.*

[4] Hecht, B., Hong, L., Suh, B., & Chi, E. H. Tweets from Justin Bieber's heart: the dynamics of the location field in user profiles. *In Proceedings of the SIGCHI Conference on Human Factors in Computing Systems, pp. 237-246, ACM.*

[5] Cheng, Z., Caverlee, J., & Lee, K. A content-driven framework for geolocating microblog users. *ACM Transactions on Intelligent Systems and Technology (TIST), 4(1), 2.*

[6] Leidner, J. L., & Lieberman, M. D. Detecting geographical references in the form of place names and associated spatial natural language. *SIGSPATIAL Special, 3(2), pp. 5-11.*

[7] Cheng, Z., Caverlee, J., & Lee, K. You are where you tweet: a content-based approach to geo-locating twitter users. *In Proceedings of the 19th ACM international conference on Information and knowledge management, pp. 759-768, ACM.*

[8] Mahmud, J., Nichols, J., & Drews, C. Home location identification of twitter users. *ACM Transactions on Intelligent Systems and Technology (TIST), 5(3), 47.*

[9] Krishnamurthy, R., Kapanipathi, P., & Sheth, A. P. Location Prediction of Twitter Users using Wikipedia.

[10] Jurgens, D. That's What Friends Are For: Inferring Location in Online Social Media Platforms Based on Social Relationships. *In ICWSM.*

[11] Hu, X., & Liu, H. Text analytics in social media. *In Mining text data, pp. 385-414, Springer US.*

[12] Text Analytics : The Hurwitz Victory Index Report. http://provalisresearch.com/ Documents/HurwitzProvalis.pdf. Accessed: 2015-03-05.

[13] Agarwal, A., Xie, B., Vovsha, I., Rambow, O., & Passonneau, R. Sentiment analysis of twitter data. *In Proceedings of the Workshop on Languages in Social Media, pp. 30-38, Association for Computational Linguistics.*

[14] Ramasubramanian, C., & Ramya, R. Effective Pre-Processing Activities in Text Mining using Improved Porter's Stemming Algorithm. *International Journal of Advanced Research in Computer and Communication Engineering, 2(12).*

[15] Han, J., Kamber, M., & Pei, J. Data mining, southeast asia edition: Concepts and techniques. Morgan kaufmann.

[16] Peri, C., & Ho, B. Sams Teach Yourself the Twitter API in 24 Hours. *Sams Publishing.*

[17] Kumar, S., Morstatter, F., & Liu, H. Twitter data analytics. *Springer.*

[18] Oja, H. Descriptive statistics for multivariate distributions. *Statistics & Probability Letters, 1(6), pp. 327-332.*

[19] Harish, B. S., Guru, D. S., & Manjunath, S. Representation and classification of text documents: A brief review. *IJCA, Special Issue on RTIPPR (2), pp. 110-119.*

[20] Kibriya, A. M., Frank, E., Pfahringer, B., & Holmes, G. Multinomial naive bayes for text categorization revisited. *In AI 2004: Advances in Artificial Intelligence, pp. 488-499, Springer Berlin Heidelberg.*

[21] Natural Language Toolkit http://www.nltk.org/. Accessed: 2015-03-05.

[22] Ikonomakis, M., Kotsiantis, S., & Tampakas, V. Text classification using machine learning techniques. *WSEAS Transactions on Computers, 4(8), pp. 966-974.*

[23] Python. https://www.python.org/. Accessed: 2015-03-05.

[24] PyCharm: The Most Intelligent Python IDE. https://www.jetbrains.com/pycharm/. Accessed: 2015-03-05.

[25] MySQL Workbench. http://www.mysql.com/products/workbench/. Accessed: 2015-03-05.

[26] Twython Usage. https://twython.readthedocs.org/en/latest/. Accessed: 2015-03-05.

[27] Google Maps API Web Services.https://developers.google.com/maps/documentation /geocoding/. Accessed: 2015-03-05.

[28] List of United States cities by population. http://en.wikipedia.org/wiki/List_of_ United_States_cities_by_population. Accessed: 2015-03-05.

[29] Ramasubramanian, C., & Ramya, R. Effective Pre-Processing Activities in Text Mining using Improved Porter's Stemming Algorithm. *International Journal of Advanced Research in Computer and Communication Engineering, 2(12).*

[30] Wing, B., & Baldridge, J. Hierarchical Discriminative Classification for Text-Based Geolocation. *Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing (EMNLP), pp. 336–348.*