Cluster Analysis of Twitter Data: A Review of Algorithms

Noufa Alnajran, Keeley Crockett, David McLean and Annabel Latham
School of Computing, Math and Digital Technology, Manchester Metropolitan University,
John Dalton Building, All Saints, Manchester, M1 5GD, U.K.

Keywords: Clustering, Social Network Analysis, Twitter, Data Mining, Machine Learning.

Abstract: Twitter, a microblogging online social network (OSN), has quickly gained prominence as it provides people with the opportunity to communicate and share posts and topics. Tremendous value lies in automated analysing and reasoning about such data in order to derive meaningful insights, which carries potential opportunities for businesses, users, and consumers. However, the sheer volume, noise, and dynamism of Twitter, imposes challenges that hinder the efficacy of observing clusters with high intra-cluster (i.e. minimum variance) and low inter-cluster similarities. This review focuses on research that has used various clustering algorithms to analyse Twitter data streams and identify hidden patterns in tweets where text is highly unstructured. This paper performs a comparative analysis on approaches of unsupervised learning in order to determine whether empirical findings support the enhancement of decision support and pattern recognition applications. A review of the literature identified 13 studies that implemented different clustering methods. A comparison including clustering methods, algorithms, number of clusters, dataset(s) size, distance measure, clustering features, evaluation methods, and results was conducted. The conclusion reports that the use of unsupervised learning in mining social media data has several weaknesses. Success criteria and future directions for research and practice to the research community are discussed.

1 INTRODUCTION

The rapid evolution of web 2.0 technologies such as OSN applications, has led to the continuous generation of an enormous volume of digital heterogeneous data being published at an unprecedented rate. These technologies have significantly changed the way people communicate and share information among each other in various domains. Millions of people have shifted from the traditional media channels such as newspapers, to online social media. In this context, Twitter has gained massive popularity as it provides an informal platform where people can easily publish and broadcast messages on different areas across the world. It had a prominent role in spreading awareness of natural disasters such as Hurricane Sandy and socio-political events such as the Arab Spring (Kumar et al., 2014). This has made Twitter an important source of information for synthesizing evidence in argumentation, and a goldmine of potential cross-domain opportunities for both businesses and decision makers. However, the exponential amount of user generated content on this site is too vast for manual analysis. More than 500 million short-text messages, referred to as "tweets", are published every day (Krestel et al., 2015). This requires an automated and scalable mining process to discover patterns in the unstructured data.

Cluster analysis is the unsupervised process of grouping data instances into relatively similar categories, without prior understanding of the groups structure or class labels (Han et al., 2011). It is a prominent component of exploratory data analysis. A subfield of clustering includes text mining, where large volumes of text are analysed to find patterns between documents (Godfrey et al., 2014). The growth of these unstructured data collections, advances in technology and computer power, and enhanced software capabilities, has made text mining an independent academic field. Moreover, the emergence of OSNs has yielded new frontiers for academic research, where researchers in the broad area of Natural Language Processing consider text analysis one of the most important research areas. Recent studies in various disciplines have shown increasing interest in micro-blogging services, particularly Twitter (Sheela, 2016). The applications of text mining tools for studying features of content

Alnajran N., Crockett K., McLean D. and Latham A.
Cluster Analysis of Twitter Data: A Review of Algorithms.
DOI: 10.5220/0006202802390249
In *Proceedings of the 9th International Conference on Agents and Artificial Intelligence (ICAART 2017)*, pages 239-249
ISBN: 978-989-758-220-2

and semantics in tweets propagating through the network has been widely studied (Kumar et al., 2014). Several studies have aimed at analysing social data from Twitter through performing data mining techniques such as classification (Castillo et al., 2011). However, these techniques could be considered to have limited capabilities due to the unpredictable nature of the dataset. Cluster analysis of tweets has been reported to be particularly suitable for this kind of data for two reasons (Go et al., 2009): (1) the amount of data for training is too vast for manual labelling. (2) The nature of the data implies the existence of unforeseen groups that may carry important nuggets of information which can only be revealed by unsupervised learning.

Among the research conducted around clustering tweets' short-text and other text mining applications on Twitter, researchers aim to find relevant information such as inferring users' interests and identifying emergent topics. However, several natural challenges of the data prevent standard clustering algorithms being applied with their full potentials:

- Sparseness –unlike traditional clustering of documents which are rich in context, tweets are restricted to 140 characters.
- Non-standardization –people invented many ways to expand the semantics that are carried out by the tweet. This implies the usage of slangs, misspelled, and connected words. Users also use self-defined hashtags to identify topics or events.
- Volume –the rapid generation of tweets results in high volumes of data.

Therefore, due to the textual length restriction of the text, the content in tweets is limited, however it still may contain rich meanings. Therefore, tweets require intelligent techniques, such as incorporating semantic technologies that can analyse datasets with such complex characteristics and convey meanings and correlations.

The main purpose of this paper is to:

- Review various clustering algorithms that are implemented on different features of Twitter datasets.
- Review various domains of applications and success criteria that are used for measuring and evaluating the accuracy of the algorithm.
- Compare relevant approaches in terms of clustering methods, algorithms, number of clusters, dataset(s) size, distance measure, clustering features, evaluation methods, and results.
- Recommend future directions for research and practice to the research community.

To the best of our knowledge, there does not exist research that reviews the prominent clustering algorithms available to use on challenging, large, and unstructured data such as Twitter. Thus, this shall provide a thorough literature review and a valuable source of information on the state of the art for relevant research in this field.

The rest of this paper is organized as follows: section 2 describes the methods that are used in this review. Section 3 includes the techniques of mining Twitter datasets that use four clustering methods: (1) partition-based, (2) hierarchical-based, (3) hybrid-based, and (4) density-based. Section 4 contains the discussion and section 5 has the conclusion and future work. A table providing a summary of the studies featured in this review is located at the end of the paper.

2 METHODS

2.1 Literature Search Procedures

In this review, multiple research databases were investigated, such as Google Scholar and DeepDyve, to conduct online searches. This process includes searching for the following terms: 'mining Twitter short-text', 'clustering tweets', 'unsupervised learning on Twitter', and 'categorization of tweets'.

2.2 Inclusion Criteria

The inclusion criteria for this paper includes research that involve:

- An implementation of one of the following clustering methods: partition, hierarchical, hybrid, and density, on Twitter short-text messages. The reason for choosing these methods is that these generally cover the major clustering algorithms and have not been reviewed previously in the context of Twitter data.
- An approach to find hidden patterns and similar groups of information in tweets using models of unsupervised learning.

A total of 13 articles from 2011 to present have met the inclusion criteria as Twitter text mining applications using unsupervised learning.

3 CLUSTER-BASED TWITTER MINING

Many clustering methods exist in the literature, and it is difficult to provide a crisp categorization of these methods as they may overlap and share features. Nevertheless, the major clustering methods are included in this review (Han et al., 2011).

Clustering has been widely studied in the context of Twitter mining. It has been applied to analyse social behaviours in a variety of domains to achieve different tasks, such as tailoring advertisements for groups with similar interests (Friedemann, 2015), event detection (De Boom et al., 2015), and trending issues extraction (Purwitasari et al., 2015). This review focuses on the major clustering methods: partition, hierarchical, hybrid, and density, which have been used in the context of Twitter data.

3.1 Partition-based Clustering

Partitioning algorithms attempt to organize the data objects into *k* partitions ($k \leq n$), each representing a cluster, where *n* is the number of objects in a dataset. Based on a distance function, clusters are formed such that objects within the cluster are similar (intra-similarity), whereas dissimilar objects lie in different clusters (inter-similarity). Partitioning algorithms can be further divided into hard and fuzzy (soft) clustering. In this section, six articles are summarized in which partitioning-based clustering algorithms has been applied in the exploratory analysis of Twitter.

3.1.1 Hard Clustering

Methods of hard partitioning of data assign discrete value label 0, 1, in order to describe the belonging relationship of objects to clusters. These conventional clustering methods provide crisp membership assignments of the data to clusters. *K*-means and *k*-medoids are the most popular hard clustering algorithms (Preeti Arora, 2016).

K-means is a centroid-based iterative technique which takes the number of representative instances, around which the clusters are built. Data instances are assigned to these clusters based on a dissimilarity function (i.e. distance measure). In each iteration, the mean of the assigned points to the cluster is calculated and used to replace the centroid of the last iteration until some criteria of convergence is met.

K-means has been adapted in numerous ways to suit different datasets including numerical, binary, and categorical features. In the context of Twitter mining applications, *k*-means approach for clustering customers of a company using social media data from Twitter was proposed (Friedemann, 2015). The technique constructs features from a massive Twitter dataset and clusters them using a similarity measure to produce groupings of users. The study performed *k*-means clustering and produced satisfactory experimental results. It is considered to be relatively computational efficient. In (Soni and Mathai, 2015), a 'cluster-then-predict' model was proposed to improve the accuracy of predicting Twitter sentiment through a composition of both supervised and unsupervised learning. After building the dataset, *k*-Means was performed such that tweets with similar words are clustered together. This unsupervised phase was performed after a feature extraction process. After the clustering phase, classification was done on the same data. The data was divided into training and testing sets, with 70% and 30% of the data respectively. Finally, the Random Forest learning algorithm was used for building the learning model, which was applied to each of the training datasets individually (Breiman, 2001). This algorithm has been chosen as it provides satisfactory trade-off between accuracy, interpretability, and execution time. Empirical evaluation shows that combining both supervised and unsupervised learning (k-Means then Random Forest) performed better than various stand-alone learning algorithms.

K-medoids is an object-based representative technique that deals with discrete data. It is an improvement to *k*-means in relation to its sensitivity to outliers. Instead of referring to the mean value of cluster objects, *k*-medoids picks the nearest point to the center of data points as the representative of the corresponding cluster. Thus, minimizing the sum of distances between each object, *o*, and its corresponding center point. That is, the sum of the error for all objects in each cluster is calculated as (Han et al., 2011),

$$E = \sum_{j=1}^{k} \sum_{p \in Cj} |p - o_j| \tag{1}$$

Where *k* is the number of clusters, *p* is an object in the cluster C_j, while o_j is the representative objects of C_j. The lower the value of *E*, the higher clustering quality.

A recent study focused on the usage of *k*-medoids algorithm for tweets clustering due to its simplicity and low computational time (Purwitasari et al., 2015). In this study, the author applied this algorithm to extract issues related to news that is posted on Twitter such as "flight passengers asking for refund" in Indonesia. Their proposed methodology for Twitter trending issues extraction consists of clustering tweets with *k*-medoids, in which they divided the tweets dataset into groups and used a representative tweet as the cluster center. Issue terms are then selected from the clusters result and assigned higher weight values.

The terms that weigh over a certain threshold are extracted as trending issues. Weight score is calculated as the frequency of word occurrences in the dataset. Average Silhouette Width (Rousseeuw, 1987), a method for validating clusters' consistency, was used to measure and evaluate the clustering performance (Ramaswamy, [no date]). In the work, the experiments demonstrated good results of using *k*-medoids for this purpose, however, re-tweets (i.e. duplicates) had influenced the clustering results. Another study used *k*-means and *k*-medoids respectively to cluster a single Twitter dataset and compare the results of each algorithm (Zhao, 2011). Initially, *k*-means was applied, which took the values in the matrix as numeric, and set the number of clusters, *k*, to eight. After that, the term-document matrix was transformed to a document-term one and the clustering was performed. Then, the frequent words in each cluster and the cluster centers were computed in order to find what they are about. The first experiment showed that the clusters were of different topics. The second experiment was conducted using *k*-medoids, which used representative objects instead of means to represent the cluster center.

K-medoids has the advantage of robustness over *k*-means as it is less influenced by noise and outliers. However, this comes at the cost of efficiency. This is due to the high processing time that is required by *k*-medoids compared to *k*-means. Both methods require the number of clusters, *k*, to be fixed. In terms of clustering sparse data such as tweets, *k*-medoids may not be the best choice as these do not have many words in common and the similarities between them are small and noisy (Aggarwal and Zhai, 2012). Thus, a representative sentence does not often contain the required concepts in order to effectively build a cluster around it.

3.1.2 Fuzzy Clustering

This partition-based method is particularly suitable in the case of no clear groupings in the data set. Unlike hard clustering, fuzzy algorithms assign a continuous value [0, 1] to provide reasonable clustering. Multiple fuzzy clustering algorithms exist in the literature, however fuzzy *c*-means (FCM) is the most prominent.

FCM provides a criteria on grouping data points into different clusters to varying degrees that are specified by a membership grade. It incorporates a membership function that represents the fuzziness of its behaviour. The data are bound to each cluster by means of this function.

In the context of Twitter analysis, a recent study presented a simple approach using fuzzy clustering for pre-processing and analysis of hashtags (Zadeh et al., 2015). The resulting fuzzy clusters are used to gain insights related to patterns of hashtags popularity and temporal trends. To analyse hashtags' dynamics, the authors identified groups of hashtags that have similar temporal patterns and looked at their linguistic characteristics. They recognised the most and least representative hashtags of these groups. The adopted methodology is fuzzy clustering based and multiple conclusions were drawn on the resulting clusters with regards to variations of hashtags throughout a period of time. Their clustering was based on the fact that categorization of hashtags is not crisp, rather, most data points belong to several clusters according to certain degrees of membership. Another study compared the performance of supervised learning against unsupervised learning in discriminating the gender of a Twitter user (Vicente et al., 2015). Given only the unstructured information available for each tweet in the user's profile, the aim is to predict the gender of the user. The unsupervised learning involved the usage of soft in conjunction with hard clustering algorithms. *K*-means and FCM were applied on a 242K Twitter users' dataset. The unsupervised approach based on FCM proved to be highly suitable for detecting the user's gender, achieving a performance of about 96%. It also has the privilege of not requiring a labelled training set and the possibility of scaling up to large datasets with improved accuracy.

Experiments have shown that fuzzy-based clustering is more complex than clustering with crisp boundaries. This is because the former requires more computation time for the involved kernel (Bora et al., 2014). Fuzzy methods provide relatively high clustering accuracy and more realistic probability of belonging. Therefore, they can be considered an effective method that excludes the need of a labelled dataset. This is particularly useful for sheer volumes of tweets, where human annotations can be highly expensive. However, these methods generally have low scalability and results can be sensitive to the initial parameter values. In terms of optimization, fuzzy clustering methods can be easily drawn into local optimal.

3.2 Hierarchical-based Clustering

In hierarchical clustering algorithms, data objects are grouped into a tree like (i.e. hierarchy) of clusters. These algorithms can be further classified depending on whether their composition is formed in a top-down (divisive) or bottom-up (agglomerative) manner. This section reviews three studies that performed hierarchical-based clustering algorithms in applications of Twitter mining.

Hierarchical clustering was used for topic detection in Twitter streams, based on aggressive tweets/terms filtering (Ifrim et al., 2014). The clustering process was performed in two phases, first the tweets and second the resulting headlines from the first clustering step. Their methodology is composed of initially computing tweets pair-wise distances using the cosine metric. Then computing a hierarchical clustering so that tweets belonging to the same topic shall cluster together, and thus each cluster is considered as a detected topic. Afterwards, they controlled the tightness of clusters by cutting the resulting dendrogram at 0.5 distance threshold. In this way they will not have to provide the number of required clusters a-priori as in *k*-Means. The threshold was set to 0.5 in order to avoid having loose or tight clusters, rather, a value of 0.5 worked well for their method. Each resulting cluster is then assigned a score and ranked according to that score. The top-20 clusters are then assigned headlines, which are the first tweet in each of them (with respect to publication time). The final step involved re-clustering the headlines to avoid topic fragmentation, also using hierarchical clustering, the resulting headlines are then ranked by the one with the highest score inside a cluster. The headlines with the earliest publication time are selected and their tweet text is presented as a final topic headline. Another research implemented a hierarchical approach for the purpose of helping users parse tweets results better by grouping them into clusters (Ramaswamy, [no date]). The aim was for fewer clusters that are tightly packed, rather than too many large clusters. The work involved using a dataset of tweets to see how the choice of the distance function affects the behaviour of hierarchical clustering algorithms. Ramaswamy conducted a survey of two clustering algorithms that are both hierarchical in nature but different in their core implementation of the distance function has been conducted. A total of 925 tweets comprising of various topics with common keyword have been used in the experiments. In the first algorithm, the author considered each of the given objects to be in different clusters. Then determining if the object *o* is close enough to cluster *c*, and if so, add *o* to *c*. This process continues until the maximum size of the desired clusters is reached or no more new clusters can be formed. In this first algorithm, the notion of the distance between an object and a cluster has been defined using concepts from association rule problems –support and confidence. The second algorithm maintained the average distance of an object from each element in the cluster as the similarity measure. If the average is small enough, the object is added to the cluster. Both clustering algorithms were implemented using C# and involve reading the tweets, tokenizing them, clustering them and returning the clustered output. Although the overall behaviour was found to be similar for both algorithms, the second one seemed to fare better for each of the confidence and support level value. An integrated hierarchical approach of *agglomerative* and *divisive* clustering was proposed to dynamically create broad categories of similar tweets based on the appearance of nouns (Kuar, 2015). The bottom-up approach merges similar clusters together to reduce their redundancy. The technique adopted a recursive and incremental process of dividing and combining clusters in order to produce more meaningful sorted clusters. It has shown an increase in clustering effectiveness and quality compared to standard hierarchical algorithms.

In this context, empirical evaluations provided that hierarchical methods performed slower than hard partition-based clustering, particularly *k*-means (Manpreet Kaur, 2013). Therefore, for massive social media datasets, hard partitioning methods are considered to be relatively computationally efficient as well as producing acceptable experimental results.

3.3 Hybrid-based Clustering

Because hierarchical clustering algorithms tend to compare all pairs of data, their robustness is relatively high. However, this makes them not very efficient due to their tendency to require at least $O(n^2)$ computation time. On the other hand, partitioning algorithms may not be the optimal choice despite being more efficient than hierarchical algorithms. This is because the former may not be very effective as they tend to rely on small number of initial cluster representatives. This trade-off has led researchers to propose several clustering algorithms that combined the features of hierarchical and partitioning methods in order to improve their performance and efficiency. These hybrid algorithms include any aggregations between clustering algorithms. In general, they initially partition the input dataset into sub clusters and then construct a new hierarchical cluster based on these sub clusters.

There is not much research conducted using a hybrid clustering approach in the area of Twitter mining. Nevertheless, one approach implemented clustering of keywords that are presented in the tweets using agglomerative hierarchical clustering and crisp *c*-means (Miyamoto et al., 2012). The clustering features was based on a series of tweets as one long sequence of keywords. The approach involved building two datasets, each composed of 50 tweets in different timeframes. Several observations of agglomerative clusters obtained by cutting the dendrogram and *c*-means clusters, with and without pair-wise constrains were analysed. Better clustering results are provided using pair-wise constrains, however, the size of datasets is relatively small for a generalization.

3.4 Density-based Clustering

This method groups data located in the region with high density of the data space to belong to the same cluster. Therefore, it is capable of discovering clusters with arbitrary shape. DBSCAN (Density-Based Spatial Clustering of Applications with Noise) is the prominent density-based algorithm. It grows regions with sufficiently high density into clusters (Ester et al., 1996). In this section, three articles are summarized in which density-based algorithms has been applied in the exploratory analysis of Twitter.

A density-based clustering has been adopted in the context of Twitter textual data analysis to discover cohesively the information posted by users about an event as well as the user's perception about it (Baralis et al., 2013). The provided framework adopts a multiple-level clustering strategy, which focuses on disjoint dataset portions iteratively and identifies clusters locally. DBSCAN has been exploited for the cluster analysis as it allows discovering arbitrarily shaped clusters, and increases cluster homogeneity by filtering out noise and outliers. Additionally, it does not require prior specification of the number of expected clusters in the data. In this approach, DBSCAN has been applied iteratively on disjoint dataset portions and all the original dataset is clustered at the first level. Then, tweets labelled as outliers in the previous level are re-clustered at each subsequent level. To discover representative clusters for their Twitter dataset, they attempt to avoid clusters containing few tweets. They also attempt to limit the number of tweets labelled as outliers and thus un-clustered, in order to consider all different posted information. Through addressing these issues, DBSCAN parameters were properly set at each level. A recent study employed DBSCAN as part of its novel method for creating an event detection ground truth through utilizing tweets hashtags (De Boom et al., 2015). The authors clustered co-occurring hashtags using DBSCAN. The method required setting two thresholds: the minimum number of hashtags per cluster and a minimum similarity measure between two hashtags, above which the two hashtags belong to the same neighbourhood. A collection of clusters of sufficiently co-occurring hashtags on the same day were obtained by running DBSCAN for every day in the dataset. A recent study has introduced the application of DBSCAN for representing meaningful segments of tweets in batch mode (Anumol Babu, 2016). The segmentation was done based on calculations of the stickiness score. This score considers the probability of a segment being a phrase within the batch of tweets (i.e. local context) and the probability of it being a phrase in English (i.e. global context) (Weng et al., 2015). Sentimental variations in tweets were then analysed based on these segments. Each word in the text is assigned a sentiment score according to a predetermined sentiment lexicon. The sentiment of a tweet is then denoted as the summation of the most positive score and the most negative score among individual words in the tweet. In this approach, the core of the clustering consisted of integrating DBSCAN with Jaccard Coefficient similarity function. Empirical evaluations indicated an enhancement of the existing system as a result of using DBSCAN for clustering,

It can be observed from the literature surrounding Density-based algorithms in Twitter mining, that they are highly efficient and can be particularly suitable for clustering unstructured data, such as tweets, as it allows the identification of clusters with arbitrary shape. Moreover, it is less prone to outliers and noise, and does not require initial identification of the required number of clusters. However, clustering high data volumes requires big memory size.

4 DISCUSSION

Several approaches of unsupervised learning applications for mining unstructured social media data have been reviewed and presented in table 1. The table provides a comparison on the features that are used in the studies including: research approach, clustering method, algorithm, number of clusters, dataset size, distance measure, clustering features, evaluation methods, and results. The review comprises 13 studies spanning from 2011 to the present. These studies have different approaches, in which the clustering of Twitter data was performed in various settings and domains to achieve different business values or satisfy certain requirements. These approaches range from pure clustering perspectives, such as determining the impact of a distance function choice on clustering behaviour, to a more general pattern recognition application, such as targeting advertisements and events detection. The majority of the studies performed clustering in order to detect news, topics, events, and facts and to predict sentiments. Different clustering methods and algorithms were implemented in these studies, each of different dataset and number of clusters. From the 13 reviewed datasets, it can be observed that the average dataset size is 162,550 for tweets textual data, ranging from 50 to 1,084,200 and average of 126,329 for Twitter user accounts, ranging from 10,000 to 242,658 distinct user accounts. The majority of the dataset sizes observed in the surveys are relatively small, which means that the high volume challenge of Twitter data has not been taken into consideration. Therefore, in order for these algorithms to be effective, they should be able to scale well to the massive amounts of Twitter data. In this matter, the scalability (in terms of clustering performance) of most of the algorithms implemented in the surveys is questionable as these algorithms have not been tested on considerably large datasets.

As partitioning algorithms require the number of clusters, *c*, to be pre-set, *c* has been included in the review to provide an indication on the number of clusters that might be appropriate for similar tasks. From the provided comparisons, the average number of clusters maintained can be derived, which is 7, with 2 as the minimum clusters and 10 as the maximum. The table additionally compared the different distance measures used. It can be observed that Euclidean distance is the prominent for partitioning algorithms, whereas hierarchical algorithms commonly implemented the cosine similarity measure. In terms of clustering features, different sets were used depending on the implemented approach. The features observed from the review include some or all of the following:

- Hashtags –31% of the reviewed surveys included hashtags in the features set and considered their impact, 23% treated hashtags as normal words in the text, and 31% removed hashtags before analysis (excluding the 15% studies that are clustering upon user accounts).
- Account metadata –username, date, status, latitude, longitude, followers, and account followings.
- Tweet metadata –tweet id, published date, and language.
- Maintaining a BOW of the unique words contained in each textual data of a tweet and their frequencies as the feature vector. Some included hashtags in the BOW while others ignored them.

None of the surveys studied the impact of retweets nor “@mentions”. Rather, some datasets did not remove the retweeted tweets which affected the resulting clustering credibility. Because tweets commonly get large number of retweets, keeping them in the dataset will produce large clusters containing redundant tweets rather than tweets with similar features. This will consequently reinforce false patterns and increase run time.

Evaluation methods vary from robust measures, such as ASW to manual observations, such as manually comparing an algorithm’s detected topics with Google news headlines. ASW has been utilised by most of the studies to measure the clustering performance. Some of the evaluation methods are derived from other data mining techniques such as association rules and classification. These methods include clustering based on confidence and support levels, and calculating precision, recall and the F measure from a confusion matrix.

5 CONCLUSION

The review contributes to the literature in several significant ways. First, it provides a comparative analysis on applications that utilized and tuned text mining methods, particularly clustering, to the characteristics of Twitter unstructured data. Second, the review concentrated on algorithms of the general clustering methods: (1) partition-based, (2) hierarchical-based, (3) hybrid-based, and (4) density-based, in Twitter mining. Third, unlike existing reviews which provides high level and abstract specification of surveys, this review was comprehensive in that it provided comparative information and discussion across the dataset size, approach, clustering methods, algorithm, number of clusters, distance measure, clustering feature, evaluation methods, and results.

Thirteen articles were reviewed in this paper, and the results indicated that there is a sufficient improvement in the exploratory analysis of social media data. However, many of the existing methodologies have limited capabilities in their performance and thus limited potential abilities in recognising patterns in the data:

- Most of the dataset sizes are relatively small which is not indicative of the patterns in social behaviours and therefore generalised conclusions cannot be drawn. Because of the sparsity of Twitter textual data, it is difficult to discover representative information in small datasets.

Therefore, future studies should aim to increase the size of the dataset.

- Some of the algorithms implemented may have provided effective results in terms of efficiency and accuracy. However, this may be attributed to the small size of dataset as the scalability has not been evaluated.
- Some of the reviewed datasets included redundant tweets (i.e. retweets) which yields inaccurate clustering. Therefore future studies should perform a comprehensive pre-processing phase in which retweets and other noise, such as URLs, are removed from the dataset prior to clustering.
- Most of the studies implemented keyword-based techniques, such as term frequencies and BOW which ignores the respective order of appearance of the words and does not account for correlations between text segments. Therefore, future research should incorporate and measure the underlying semantic similarities in the dataset.

In conclusion, after conducting this review it can be clearly noticed that clustering is an important element of exploratory text analysis in which unstructured data can be useful for pattern recognition as well as identification of user potentials and interests. However, future research must demonstrate the effectiveness of such approaches through acquiring larger datasets in order for the algorithms to be useful in discovering knowledge and applicable in several contexts and domains. A meta-analysis review is recommended as a future work, which will provide a quantitative estimate for the impact and usefulness of clustering methods in providing insights from social media data.

Table 1: Summary of the studies featured in this review.

Author & Year	Approach	Method		Algorithm & Number of Clusters (C)	Dataset Size	Distance Measure	Clustering Features	Evaluation Methods	Results
(Friedemann, 2015)	Targeting advertise-ments	Partitioning-Based Clustering	Hard Partitioning	*k*-Means C: 5	10,000 Twitter user account	Euclidean distance	posted status, number of followers and account followings, latitude, longitude, whether a popular Twitter account (*influencer*) is followed	Computing a metric of clustering quality *q*. The lower the value of *q*, the better clustering performance	Achieved clustering is midway between ideal and randomized data. Experiments emphasized the credibility of Twitter data for market analysis
(Soni and Mathai, 2015)	Sentiment prediction			*k*-Means C: 2	1200 “Apple” tweets	Squared Euclidean distance	Bag-of-Words (BOW) from twitter corpus (frequency of word occurrences)	Confusion matrix and ROC (Receiver Operator Characteristic) graph	Model integration of supervised and unsupervised *k*-Means learning improved twitter sentiment prediction
(Purwitasari et al., 2015)	News summary			*k*-Medoids C: 10	200 tweets (geo-location: Indonesia)	Cosine similarity	Term frequencies and weight in tweet text. Hashtags omitted	The larger ASW value, the more homogeneous the cluster result	Inclusion of retweets affected cluster result quality
(Zhao, 2011)	R Data Mining			*k*-Means C: 8	1st 200 tweets from @rdatamining account	Euclidean distance	Term frequencies in tweet text (document-term matrix). Hashtags omitted	Checked the top 3 terms in every cluster	Clusters of different topics
				k-Medoids C: 9		Manhattan distance		ASW	Clusters overlap and not well separated

Table 1: Summary of the studies featured in this review. (cont.)

Author & Year	Approach	Method	Algorithm & Number of Clusters (C)	Dataset Size	Distance Measure	Clustering Features	Evaluation Methods	Results
(Vicente et al., 2015)	Gender detection	Fuzzy Partitioning	*k*-Means C: 2	242,658 unique Twitter users	Euclidean distance	Screen name and user name	Two experiments: 1st used labelled data for building clusters and evaluating performance. 2nd used unlabelled data for clustering and labelled for evaluation	*C*-Means provided better clustering performance than *k*-Means. More usage of unlabelled data significantly enhanced *c*-means but got *k*-Means worse
			c-Means C: 2					
(Zadeh et al., 2015)	Events and facts detection		FANNY (Kaufman and Rousseeuw, 2009) C: 6	40 distinct hashtag	Manhattan Distance	Temporal aspects of hashtags	Defined a *misfit* measure to identify elements' degree of "not fitting" into a cluster. Clustering performance measured using ASW	Insights into patterns associated with each cluster for hashtags changing popularities over time
(Ifrim et al., 2014)	Topic detection	Hierarchical-Based Clustering	Agglomerative (dendrogram cut at 0.5)	1st dataset: 1,084,200 tweets. 2nd: 943,175 JSON format English tweets	Cosine similarity	Date, tweet id, text, user mentions, hashtags, URLs, media URLs, and retweet or not	(1) a subset of ground truth topics, (2) google for the automatically detected topic headline, in the manual assessment of how many detected topics are actually published news in traditional media	Application of agglomerative clustering can detect topics with 80% accuracy. However, not efficient for real-time data analysis.
(Ramaswamy, [no date])	Impact of distance function choice on clustering behaviour		Two Ward (Jr., 1963) algorithms C: 5	925 tweets	Ratio of tweets appearing in different clusters Avg. distance between tokens and clusters	Tokenization of tweets' texts	Several experiments conducted to determine appropriate values of confidence and support levels which determine further clustering	Generally similar behaviour of the 2 algorithms. In terms of fewer, tightly packed clusters, 2nd algorithm fared better for confidence and support values
(Kaur, 2015)	Noun-based tweet categorization		Agglomerative	15062 "Stem Cell" tweets	Inter-cosine similarity	Frequency of occurrences for nouns in tweets. Hashtags omitted	Experimental comparisons of clustering quality against: k-means, Ward, and DBSCAN clustering.	Combinatorial approach provided higher accuracy compared to existing methodologies, however, at the cost of performance. Clustering runtime: 1hour
			Divisive		Intra-cosine similarity			
(Miyamoto et al., 2012)	Keyword clustering	Hybrid-based Clustering	Hard *c*-Means (partitioning) C: 2	1st dataset: 50 tweets (35 terms occur > 8 times) 2nd: 50 tweets (38 terms occur > 5 times)	Squared Euclidean distance	Sequence of word occurrences in a set of tweets	Several observations of: clusters with and without pair-wise constraints clusters obtained by cutting the dendrogram with and without pair-wise constraints	Application of pair-wise constraints improved clustering quality. However, dataset size is arguably small
			Agglomerative (hierarchical) C: 2					

Table 1: Summary of the studies featured in this review. (cont.)

Author & Year	Approach	Method	Algorithm & Number of Clusters (C)	Dataset Size	Distance Measure	Clustering Features	Evaluation Methods	Results
(Baralis et al., 2013)	Cohesive information discovery	Density-Based Clustering	DBSCAN	"Paralympics" dataset: 1969 tweets "Concert" dataset: 2960 tweets	Cosine similarity	BOW of tweets including hashtags	ASW	Effective in discovering knowledge. Performance relatively low for not very large dataset. Clustering runtime: 2min 9sec May not scale well to massive datasets
(De Boom et al., 2015)	Event detection		DBSCAN	63,067 tweets (geolocation: Belgium)	Sum of avg. occurrences of both hashtags per day/2	Hashtags co-occurrence matrix	Precision, recall, and F measures	Improvement in event detection and clustering through high-level semantic information
(Anumol Babu, 2016)	Sentiment Analysis		DBSCAN	100 synthetic tweets	Jaccard similarity	Tweet text and publication time. Hashtags omitted	Evaluating tweets segmentation and its accuracy through an experiment	Enhancement of the present system as DBSCAN was integrated

REFERENCES

Aggarwal, C. C. & Zhai, C. 2012. *Mining Text Data*, Springer Science & Business Media.

Anumol Babu, R. V. P. 2016. Efficient Density Based Clustering of Tweets and Sentimental Analysis Based on Segmentation. *International Journal of Computer Techniques,* 3**,** 53-57.

Baralis, E., Cerquitelli, T., Chiusano, S., Grimaudo, L. & Xiao, X. Analysis of Twitter Data Using A Multiple-Level Clustering Strategy. *International Conference on Model and Data Engineering, 2013*. Springer, 13-24.

Bora, D. J., Gupta, D. & Kumar, A. 2014. A Comparative Study Between Fuzzy Clustering Algorithm and Hard Clustering Algorithm. *Arxiv Preprint Arxiv:1404.6059*.

Breiman, L. 2001. Random Forests. *Machine Learning,* 45**,** 5-32.

Castillo, C., Mendoza, M. & Poblete, B. Information Credibility On Twitter. *Proceedings Of The 20th International Conference On World Wide Web*, 2011. ACM, 675-684.

De Boom, C., Van Canneyt, S. & Dhoedt, B. Semantics-Driven Event Clustering In Twitter Feeds. Making Sense Of Microposts, 2015. Ceur, 2-9.

Ester, M., Kriegel, H.-P., Sander, J. & Xu, X. A Density-Based Algorithm for Discovering Clusters in Large Spatial Databases with Noise. *KDD,* 1996. 226-231.

Friedemann, V. 2015. Clustering A Customer Base Using Twitter Data.

Go, A., Bhayani, R. & Huang, L. 2009. Twitter Sentiment Classification Using Distant Supervision. *Cs224n Project Report, Stanford,* 1**,** 12.

Godfrey, D., Johns, C., Meyer, C., Race, S. & Sadek, C. 2014. A Case Study in Text Mining: Interpreting Twitter Data from World Cup Tweets. *Arxiv Preprint Arxiv:1408.5427*.

Han, J., Pei, J. & Kamber, M. 2011. *Data Mining: Concepts And Techniques*, Elsevier.

Ifrim, G., Shi, B. & Brigadir, I. Event Detection in Twitter Using Aggressive Filtering and Hierarchical Tweet Clustering. *Second Workshop on Social News on the Web (Snow)*, Seoul, Korea, 8 April 2014, 2014. ACM.

Jr., J. H. W. 1963. Hierarchical Grouping to Optimize an Objective Function. *Journal of the American Statistical Association,* 58**,** 236-244.

Kaufman, L. & Rousseeuw, P. J. 2009. *Finding Groups in Data: An Introduction to Cluster Analysis*, John Wiley & Sons.

Kaur, N., 2015. *A Combinatorial Tweet Clustering Methodology Utilizing Inter and Intra Cosine Similarity* (Doctoral Dissertation, Faculty Of Graduate Studies And Research, University Of Regina).

Krestel, R., Werkmeister, T., Wiradarma, T. P. & Kasneci, G. Tweet-Recommender: Finding Relevant Tweets for News Articles. *Proceedings of the 24th International Conference on World Wide Web*, 2015. ACM, 53-54.

Kumar, S., Morstatter, F. & Liu, H. 2014. *Twitter Data Analytics*, Springer.

Manpreet Kaur, U. K. 2013. Comparison Between K-Mean And Hierarchical Algorithm Using Query Redirection. *International Journal of Advanced Research in Computer Science and Software Engineering* 3**,** 54-59.

Miyamoto, S., Suzuki, S. & Takumi, S. Clustering In Tweets Using A Fuzzy Neighborhood Model. Fuzzy Systems (Fuzz-Ieee), 2012 *IEEE International Conference On*, 2012. IEEE, 1-6.

Preeti Arora, D. D., Shipra Varshney. Analysis of k-Means and k-Medoids Algorithm for Big Data. 2016 India. *Procedia Computer Science*, 507-512.

Purwitasari, D., Fatichah, C., Arieshanti, I. & Hayatin, N. k-Medoids Algorithm on Indonesian Twitter Feeds for Clustering Trending Issue as Important Terms in News Summarization. *Information & Communication Technology And Systems (ICTS), 2015 International Conference On*, 2015. IEEE, 95-98.

Ramaswamy, S. Comparing The Efficiency of Two Clustering Techniques.

Rousseeuw, P. J. 1987. Silhouettes: A Graphical Aid to the Interpretation and Validation of Cluster Analysis. *Journal of Computational and Applied Mathematics,* 20**,** 53-65.

Sheela, L. 2016. A Review of Sentiment Analysis in Twitter Data Using Hadoop. *International Journal of Database Theory And Application,* 9**,** 77-86.

Soni, R. & Mathai, K. J. 2015. Improved Twitter Sentiment Prediction Through Cluster-Then-Predict Model. *Arxiv Preprint Arxiv:1509.02437*.

Vicente, M., Batista, F. & Carvalho, J. P. Twitter Gender Classification Using User Unstructured Information. *Fuzzy Systems (Fuzz-IEEE), 2015 IEEE International Conference On*, 2015. IEEE, 1-7.

Weng, J., Li, C., Sun, A. And He, Q., 2015. Tweet Segmentation and Its Application to Named Entity Recognition.

Zadeh, L. A., Abbasov, A. M. & Shahbazova, S. N. Analysis Of Twitter Hashtags: Fuzzy Clustering Approach. Fuzzy Information Processing Society (Nafips) Held Jointly With 2015 *5th World Conference On Soft Computing (WCONSC)*, 2015 Annual Conference of the North American, 2015. IEEE, 1-6.

Zhao, Y. 2011. R and Data Mining: Examples and Case Studies.