Journal of Theoretical and Applied Information Technology
31st March 2021. Vol.99. No 6

ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

REAL-TIME BIG DATA CLUSTERING USING SPARK: UBER CASE STUDY

[1]M. EL-SAID EL-BARBEER, [2]AMIRA REZK, [3]A. ABU-ELFETOUH SALEH
[1,2,3]Information System Department, Faculty of Computers and Information, Mansoura University, Egypt
E-mail: [1]mohammed.elbarbeer@mans.edu.eg, [2]amira_rezk@mans.edu.eg, [3]elfetouh@mans.edu.eg

ABSTRACT

We live in a world flooded with data. Some even see it as the fuel that drives all companies to reach their goals. Business Intelligence is introduced to enable a company to get the power of its data to be able for the competition in the rougher market. Because business needs to make decisions in a fast and reliable manner, analysis the big data in real time become interested issue. Although the significant efforts that done in this area, big data analysis in real time is still need additional effort to enhance the performance and reduce the required time. This paper introduces a framework to analyze big data in real-time using the K-means clustering technique. Although the K-mean is widely used in clustering, its processing requirement can be a problem in big data and real-time systems. In this research, the K-mean algorithm is adapted to be suitable for the case of big data and real-time systems. The proposed framework introduces two models the first one uses historical data to create a model which deployed to real-time data and the second one analyzes the data in real-time without historical data. Experimental results show that the accuracy of the proposed framework with its two models is approximately 0.5, 0.34 respectively using the Silhouette Coefficient measurement.

Keywords: *Business Intelligence (BI), Big Data, Real-time Analytics, Clustering, Apache Spark*

1. INTRODUCTION

The term Business Intelligence (BI) dates to at least the 1860s in order to satisfy the executives' demands for analyzing the enterprise data efficiently to better realize the situation of their business [1, 2]. BI transforms raw data into useful information and, through human analysis, into rich knowledge. The acquired knowledge may be about what are customers' needs, customers' reviews / opinions / feedbacks about something like a new product or about news, etc., the competitors and the market, the industry conditions, and technological, economic trends [3]. That knowledge would then be able to be utilized to make vital business choices that improve efficiency, increment income, and quicken development. Other potential advantages of business intelligence include; enhancing the decision-making process either for time or quality; upgrading inside business forms; expanding operational effectiveness; driving new incomes; increasing upper hand over business rivals; helping organizations in the distinguishing proof of market patterns and spotting business issues that should be tended to [1].

BI data can incorporate historical data, as well as new data accumulated from different sources. A lot of companies have structured and unstructured data [4], internal and external data which forms a huge amount. These unprecedented and complicated data have given birth to the concept of "Big Data". Big data is high-volume, high-variety, and high-velocity data assets. This type of data requires effective and innovative forms of processing for enhancing insights and decision-making process [5]. Although Big Data has a lot of characteristics [6], this research will focus on its velocity. Velocity refers to the rate at which data is generated and the pace at which it should be analyzed and acted upon. The multiplication of computerized gadgets such as smartphones, tablets, sensors as so on and the evolution of 5G technologies recently are a breeding ground for big data and are driving a growing need for batch or real-time analytics and evidence-based planning [7, 8].

Big data analytics (BDA) is the process of examining vast and varied datasets to discover hidden patterns, unknown correlations, market trends, customer preferences, customer segmentation, and other useful information that can help organizations make more-informed business decisions. Companies and enterprises that proceed with big data analytics can gain several benefits, such as the discovery of new revenue opportunities, effective marketing campaigns, improved customer service delivery, more proficient activities, and other competitive advantages [9].

A lot of enterprises from different domains are facing some challenges of handling fast-growing amount of data and data diversity, therefore the conventional BI is not adequate in dealing with these challenges [10]. Real-time analytics and big data come to the top of Business Intelligence Trends in 2021 [11]. Real-Time Big Data analysis is basically processing the stream of data in motion and analyzing that data to conclude, or decision and that decision is used in different applications [12]. Real-time is often confused with instantaneous. As far as data is generated, a real-time processing engine can be intended to either push or pull data. The real-time processing engine is not always able to accommodate streaming data. Otherwise, it can be designed to pull data by requiring the arrival of any new data. The time between such requests depends on business requirements and can vary from milliseconds to hours (usually one second) [13].

This paper introduces a framework that helps organizations to analyses their big data in real-time/ near real-time. This type of analysis is based on a machine learning technique called clustering that separates objects into significant clusters without the prior knowledge of the data objects. There are a variety of methods for achieving clustering and one of them is called the partitioning method. The K-Means algorithm is considered a partitioning method that is suitable for numeric data only. The proposed framework makes use of the K-Means algorithm.

The rest of this paper will be organized as follow; section 2 related works, section 3 proposed framework, section 4 results, section 5 conclusion, and finally section 6 future work.

2. RELATED WORK

Real-time big data analysis and business intelligence are inevitable development trends and critical research points for many researchers. For example, XIANG LI, et. al [14] proposed a streaming clustering computing algorithm for big data that manipulates real-time data clustering throughout two phases. In the first phase, coarse clustering is used for the preprocessing of the streaming data. In the second phase, the macro cluster set is sampled and then applied K-means parallel clustering. In the end, the clustering algorithm and the edge computing algorithm are fully integrated to accommodate clustering analysis within the edge computing framework. The proposed algorithm does well in improving the performance of real-time big data clustering, but it does not take into account the issues of advanced computing and cloud computing.

Darshan M. Tank [15] explored by visualization how Real-Time Business Intelligence (RTBI) is beneficial for supporting the operational and tactical layers of decision-making within an organization and how it would provide the decision-maker with fresh and reliant data to base the decisions on.

Jiwat Ram, et. al [16] explored implications of Big Data analytics of data collected from Social Media for enhanced business intelligence within the context of Chinese businesses. But the research had some limitations like the limited number of people interviewed for data collection and the location of interviewees could not cover all the regions of China.

Deniz Kılınç [17] proposed a framework based on Apache Spark for detecting Twitter fake accounts and sentiment analysis for Twitter streaming with sentiment classification performances (80.93%). The Naïve Bayes classification algorithm is used for training and testing the model not different classification algorithms and choose the best performance.

Divya Sehgal, et. al [18] used the Hadoop framework for applying sentiment analysis of real-time Twitter data including tweets, emoticons, and hashtags with a total accuracy of 72.22%.

Based on the previous works, the real-time big data analysis still needs a lot of efforts to increase its accuracy and enhance its overall performance.

3. PROPOSED FRAMEWORK

The main goal of this research is proposing a general framework to analyzing big data in real-time using clustering that can be deployed in efficient manner in business intelligent system. To achieve this aim, a framework with two models is introduced. The first proposed model applies the K-means clustering algorithm [19, 20] to group/cluster big data in real-time based on a deployed model using historical data. The second model applies Streaming K-means to group/cluster big data in real-time based on a real-time model using real-time data.

3.1 The First Model

This model is suitable if the system has historical data that can be analyzed and used to predict new data arrived in real-time. It consists of two phases: The model Creation phase and the Real-time analysis phase as shown in Figure 1.

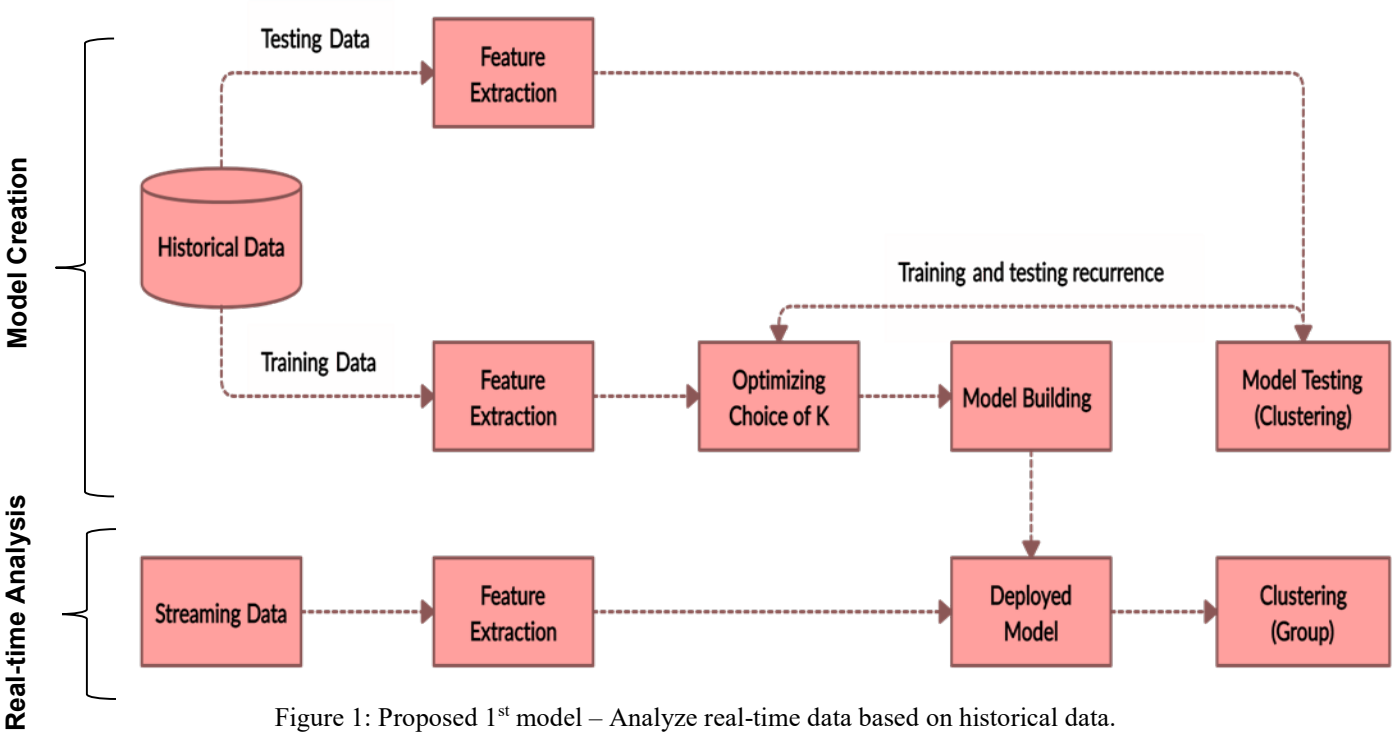

Figure 1: Proposed 1st model – Analyze real-time data based on historical data.

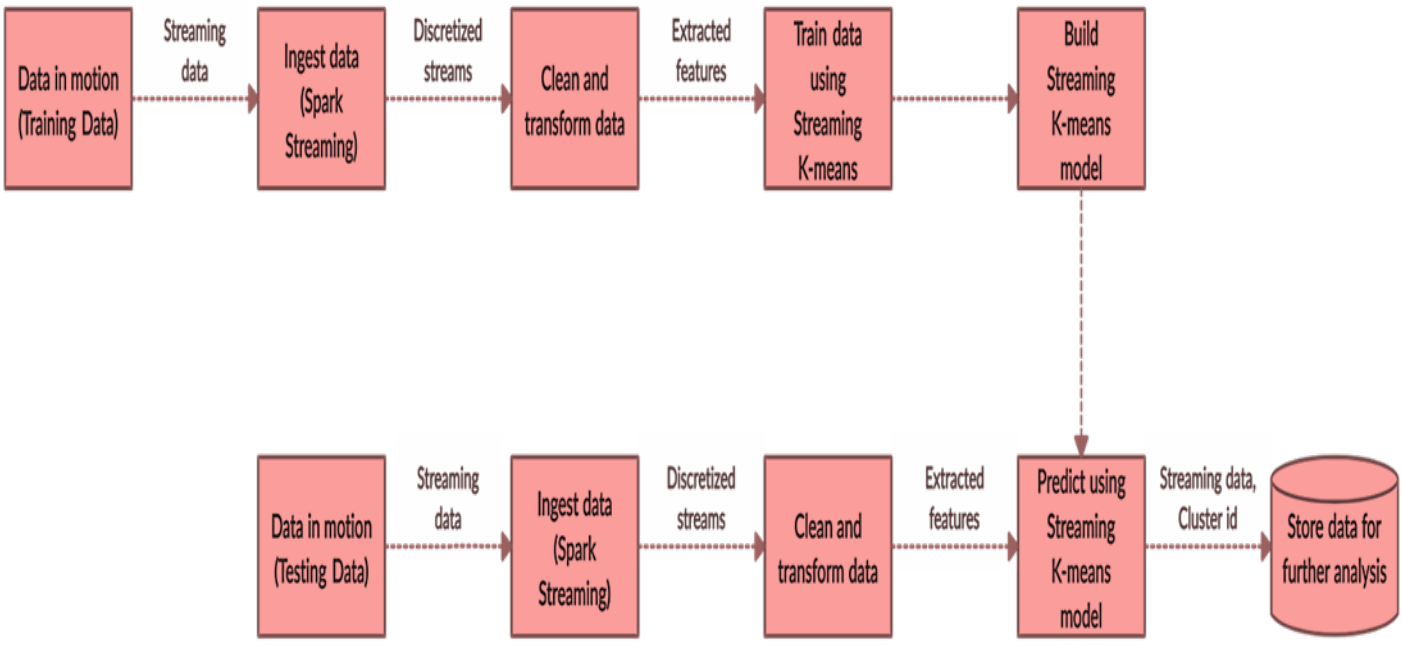

Figure 2: Proposed 2nd model - The architecture for clustering streaming data based on streaming k-means.

The creation phase aims to construct a learned model. It starts with partitioning the historical data into two portions of training and testing, then the most important features are extracted. One of the issues of the k-means algorithm is the predefined number of keys. In the proposed model, some methods are used to help in optimizing the choice of the number of keys. The result of this step is a deployed k-means clustering model.

In the real-time analysis phase, the streaming data is clustered based on the deployed model after extracting the features, and the result of this phase is the clustering label that belongs to.

3.2 The Second Model

This model is suitable for a system that has a high-speed stream of data with a large size that is not stored and needed to be analyzed in the real-time as shown in Figure 2 In this model, the training data is a streaming data that is used to build the clustering model in real-time. The testing data is also a streaming data that is passed to the clustering model. The clusters are updated in real-time with the new data.

The proposed framework with the two models makes use of Spark framework to analysis big data in batch and real-time manner [21].

4. IMPLEMENTATION

To evaluate the proposed model, it should be implemented. In this section, the implementation factors will be discussed in detail and testing the model with real-time data.

4.1 Dataset

The use case dataset is Uber organization dataset [22]. It has the following schema:

- Date/Time: The date/time of the Uber pickup.
- Lat: The coordinates of the Uber pickup (latitude).
- Lon: The coordinates of the Uber pickup (longitude).
- Base: NYC Taxi & Limousine Commission (TLC) base company code affiliated with the Uber pickup.

The dataset consists of 829275 pickups (records) which sizes 34.6 MB. An example of pickups' data: 2014-08-01 00:00:00,40.729, -73.9422, B02598

4.2 Working Environment

The proposed is performed using these specifications, computer with CPU Intel Core i5, RAM 4 GB, hard disk 500 GB, and Windows operating system. Apache Spark with python (PySpark 2.4.4) [23] is used that is running in single-machine mode with Spyder as an Integrated Development Environment (IDE).

4.3 Implementation of The First Model

The implementation starts with applying k-means algorithm to the historical dataset to build a clustering model that helps us to predict pickups that occurred in each cluster according to the proposed shown in Figure 3 which corresponding to first step in Figure 1.

Firstly, the required packages for Spark Machine learning and SQL are imported. It should specify the schema for the dataset with a Spark Structype. The schema includes the attributes' names and related data types. Next, the data should be loaded from CSV file into Spark DataFrame (that is distributed collection of data organized into named fields that affords operations to filter, group, or compute aggregates, and can be used with Spark SQL) according to the predefined schema.

For the features to be used by a machine learning algorithm, the attributes (features) are transformed and put into Feature Vectors, which are vectors of numbers representing the value for each feature.

VectorAssembler is used to transform and return a new DataFrame with all the feature columns in a vector column. When the k-means clustering algorithm runs, it uses a randomly generated seed to determine the starting centroids of the clusters. The elbow method is used to determine the optimal number of clusters in k-means clustering [24,25].

The elbow method also plots the value of the cost function produced by different values of k. In this use case, different choices of k are set from 2 to 20. Then, for each different value of K, K-means algorithm is applied, and the cost function is computed. After exploring the results of the elbow function and the dataset, K=7 is suitable and acceptable according to the cost as mentioned in the results section.

So, the proposed model applies the K-means algorithm with number of clusters equals seven. The next step is to start with fitting the

clustering model with the selected features and the number of selected keys.

To evaluate the clustering model, it should be tested with different data than the trained one. The input dataset is randomly split to 70% for training and 30% for testing. The testing data is transformed according to the clustering model to get the clusters for these data to further analyze the clustering. Finally, a machine learning model using Apache Spark's K-means algorithm is created in order to cluster Uber data based on location.

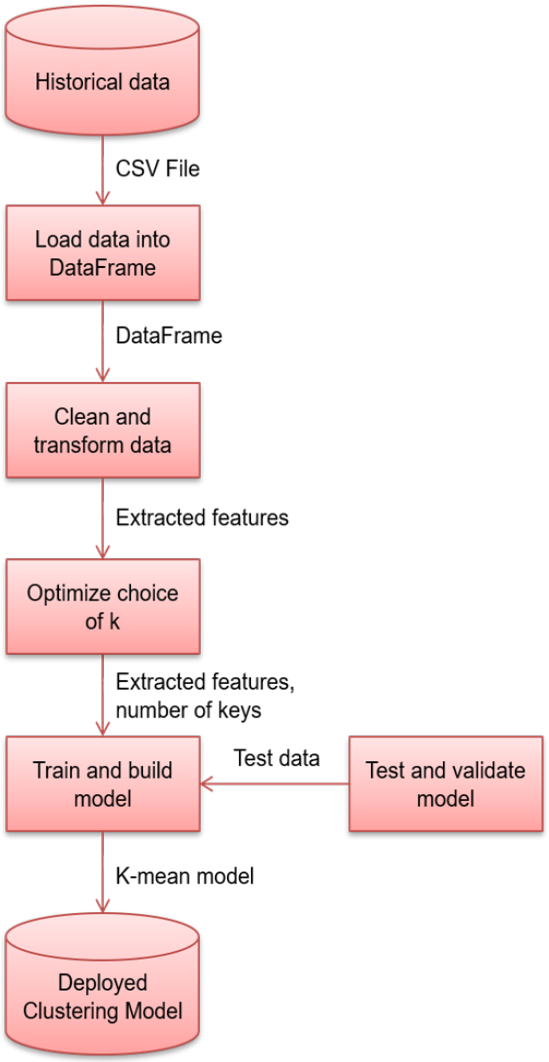

Figure 3: The Architecture For Applying Clustering Analysis On Historical Data.

The next phase in the implementation is to use the deployed K-means model with data in motion (streaming data) to do real-time analysis of where and when Uber vehicles/items are clustered.

According to the architecture described in Figure 4 (corresponding to the second step in Figure 1): Uber trip data (data in motion) is passed to spark through socketTextStream that creates an entry from source TCP hostname: port. Data is received using a TCP socket, and the received byte is interpreted as UTF8 \n delimited lines [26]. This, in turn, creates a Discretized Stream (DStream) [27] that represents the stream of incoming data that ingested every one second. The DStream foreachRDD function is used to apply processing to each Resilient Distributed Datasets (RDD) [28] in this DStream. Similarly, each RDD is converted to DataFrame that allows the usage of DataFrames and SQL operations and functions on streaming data.

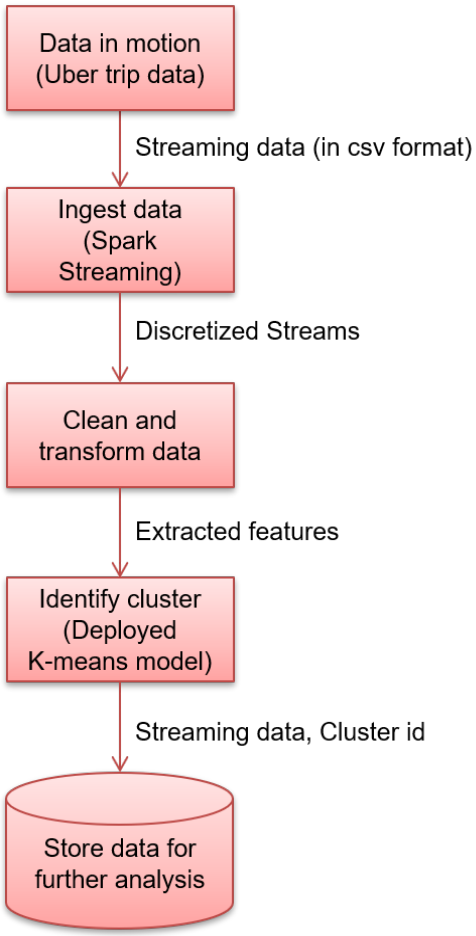

Figure 4: The Architecture For Clustering Streaming Data Based On Deployed K-Means Model.

The data cleaning and transformation is applied for each DataFrame Using VeactorAssembler. The values of latitude and longitude are extracted in the preparation for clustering. The deployed k-means model (which formed from the previous step) is loaded by KMeansModel class (clustering model derived from

the k-means method, i.e., Model fitted by KMeans). The Extracted features are passed to the model in order to know the predictions which constitute the cluster of these values. The final step in the proposed architecture is to store the data and the corresponding predictions for further analysis. The Apache Hive is used for storing these data, data queries and analyses as well.

4.4 Implementation of The Second Model

Real-time data clustering based on k-means is continued with a new concept supported by Spark which called streaming k-means. Streaming data may be needed to cluster dynamically and updating these clusters with the arrival of new data. For each batch of data, all objects are allocated to the nearest cluster, then the new centroids are calculated, and finally, each cluster is updated based on the following equations:

$$c_{t+1} = \frac{c_t n_t \alpha + x_t m_t}{n_t \alpha + m_t} \quad (1)$$

$$n_{t+1} = n_t + m_t \quad (2)$$

Where c_t is the previous centroid of the cluster, n_t is the number of objects allocated to the cluster till now, x_t is the new cluster centroid from the current batch, and m_t is the number of objects added to the cluster in the current batch. Decay factor α can be used to ignore the formed. When $\alpha = 1$, all data will be used from the launch; with $\alpha = 0$ only the recent data will be used [29].

Real-time Uber trip data (as a training data) is ingested by spark through socketTextStream as shown in Figure 2. This data may be a file that contains records for every trip or passed as separated records. After receiving the streamed data, data cleaning and transformation are applied to the data and feature extraction as well.

Streaming k-means is initialized with k equals to seven and the decay factor equals 1.0 (which means the forgetfulness of the previous centroids). The training data is passed to the streaming k-means model in order to be trained. The streaming k-means model can be updated with every new data. Similarly, Real-time Uber trip data (as a testing data) is ingested by spark through socketTextStream across different port number. The data is cleaned and transformed. Features extraction is also performed to select latitude and longitude values. Then, the data is passed to the streaming k-means model for clustering and then storing data for further analytics.

5. RESULTS

In the beginning, the K-means algorithm is well suitable for the mentioned case study because of the numeric nature of the dataset. Before implementing the k-means algorithm, the number of clusters should be determined. The elbow method is implemented on the historical dataset, and the cost function is computed as showed in Figure 5. The idea behind the elbow method is to find “elbow” or “knee” which likely corresponds to the optimal value of k. Therefore, K=7 is a good candidate with an acceptable cost for this case study.

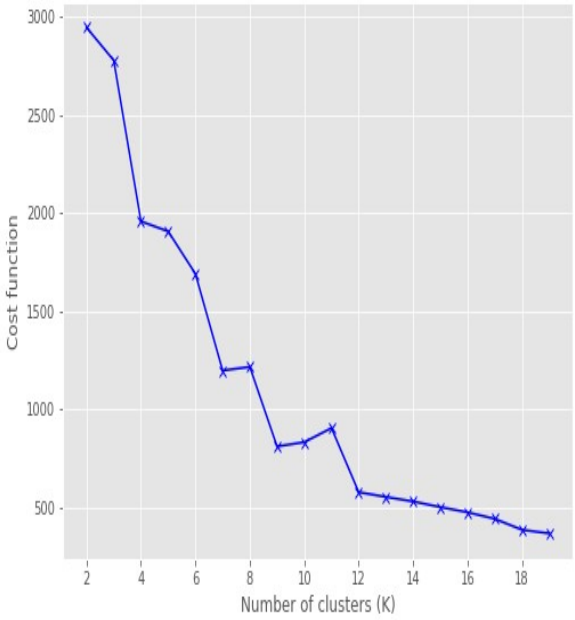

Figure 5: Elbow method with different keys.

The elbow cannot always be unambiguously determined, especially for complex data. In many cases, the error curve will not have a clear suggestion for one value, but for multiple values. In order to also ensure the choice of K, the knee point detection algorithm is used that return the knee point of the function [30]. The point is labeled with a dashed red line as shown in Figure 6, which assures that K = 7 is a good choice.

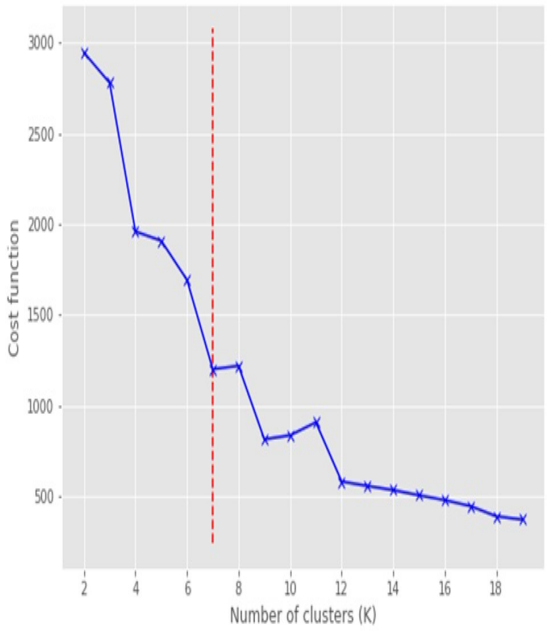

Figure 6: Elbow Method With Knee Point Detection.

After implementing the k-means, the center of every cluster is calculated and Google Map (using gmplot package [31]) is used to explore these clusters' centers as a type of visualization. As shown in Figure 7, the center of every cluster is plotted with a blue marker.

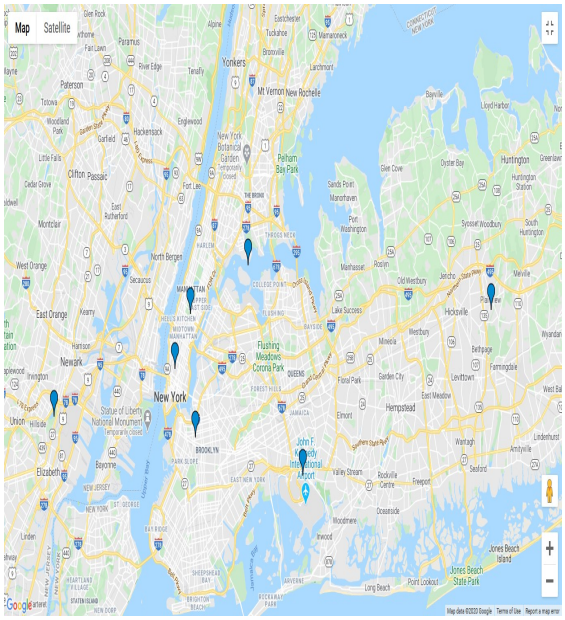

Figure 7: Plotting The Clusters' Centers On Google Map.

For evaluating the clustering, the ClusteringEvaluator class is used. This evaluator calculates the Silhouette Coefficient measurement using the Euclidean distance.

The Silhouette is a measure for the validation of the stability within clusters. It varies between (1) and (-1), where a value close to 1 indicates that the objects (pickups) in a cluster are close to the other objects within the same cluster and far from the objects of the other clusters whilst, if the value close to -1, means the objects may be assigned to wrong cluster or there some issues in the data preprocessing step [32, 33].

The proposed is applied 5 times on the historical dataset and the Silhouette with squared Euclidean distance is calculated at each time as shown in Figure 8 with average value equals to 0.524307499983973.

According to the calculated Silhouette values, the objects (pickups) are close to each other in the same cluster.

For the evaluation of clustering for real-time data, the proposed discussed a framework with two models. The first one is clustering streaming data based on the deployed k-means model and the second one is clustering streaming data based on streaming k-means.

When evaluating the two models, we have taken into consideration two factors the batch size (in seconds) and the number of objects alongside the Silhouette value.

For both models, the Silhouette metric is measured in different batch sizes (10s, 20s, 30s, 40s, 50s, 60s) as shown in Figure 9 and Figure 10 We cared about the same number of objects to be passed for the two models. The choice also for the number of objects was random.

As shown in Figure 11, which describes the change in Silhouette value with the change in the number of objects and batch size for the real-time clustering based on the deployed k-means model. We can notice that the Silhouette value settles a little above 0.5 which is considered a good indicator for the stability in the clustering.

Figure 12 is not much different from Figure 11 except the dramatic change in the Silhouette values for the real-time clustering based on the streaming k-means model.

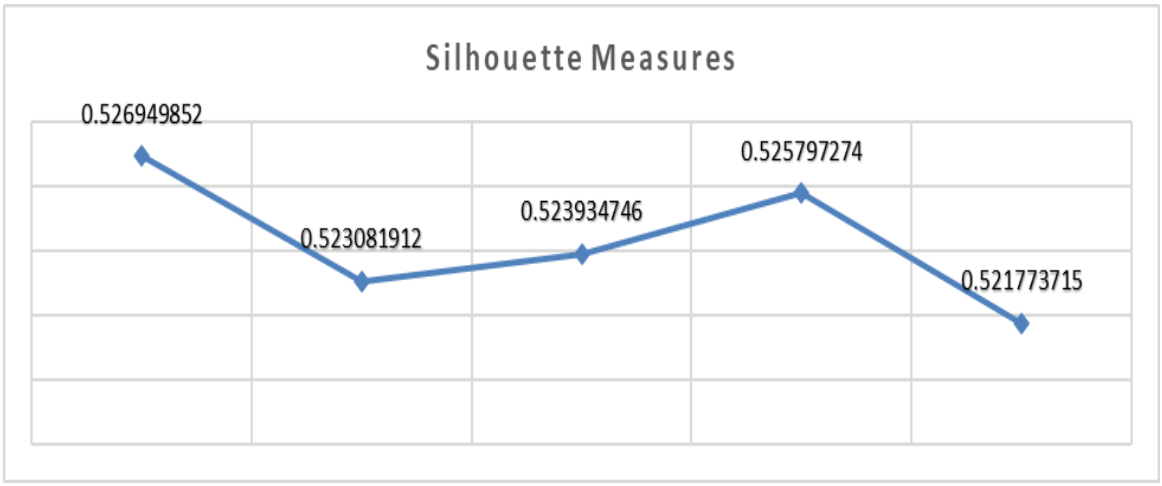

Figure 8: Different Silhouette measures for historical dataset.

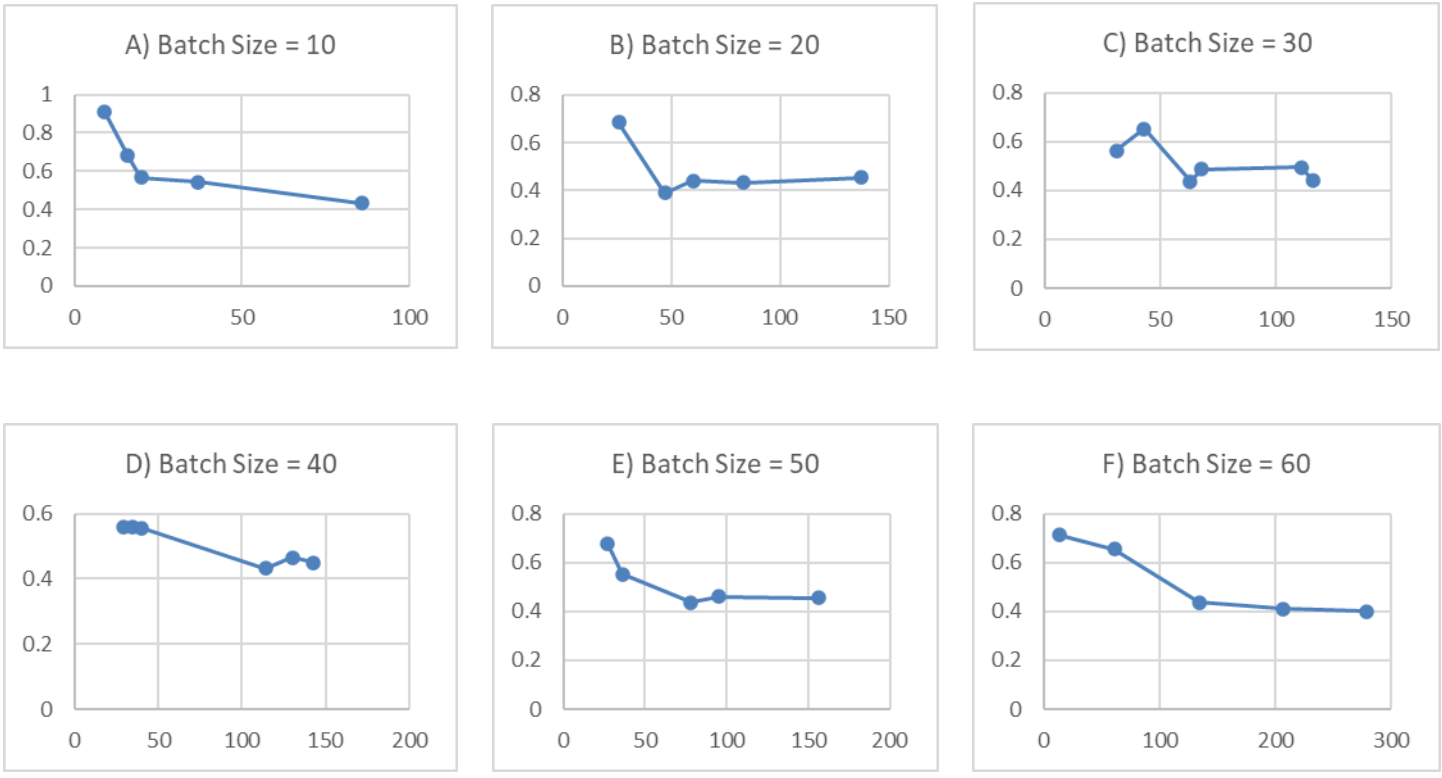

Figure 9: Different Silhouette Measures For The Deployed Clustering Model (X-Axis: No. Objects, Y-Axis: Silhouette).

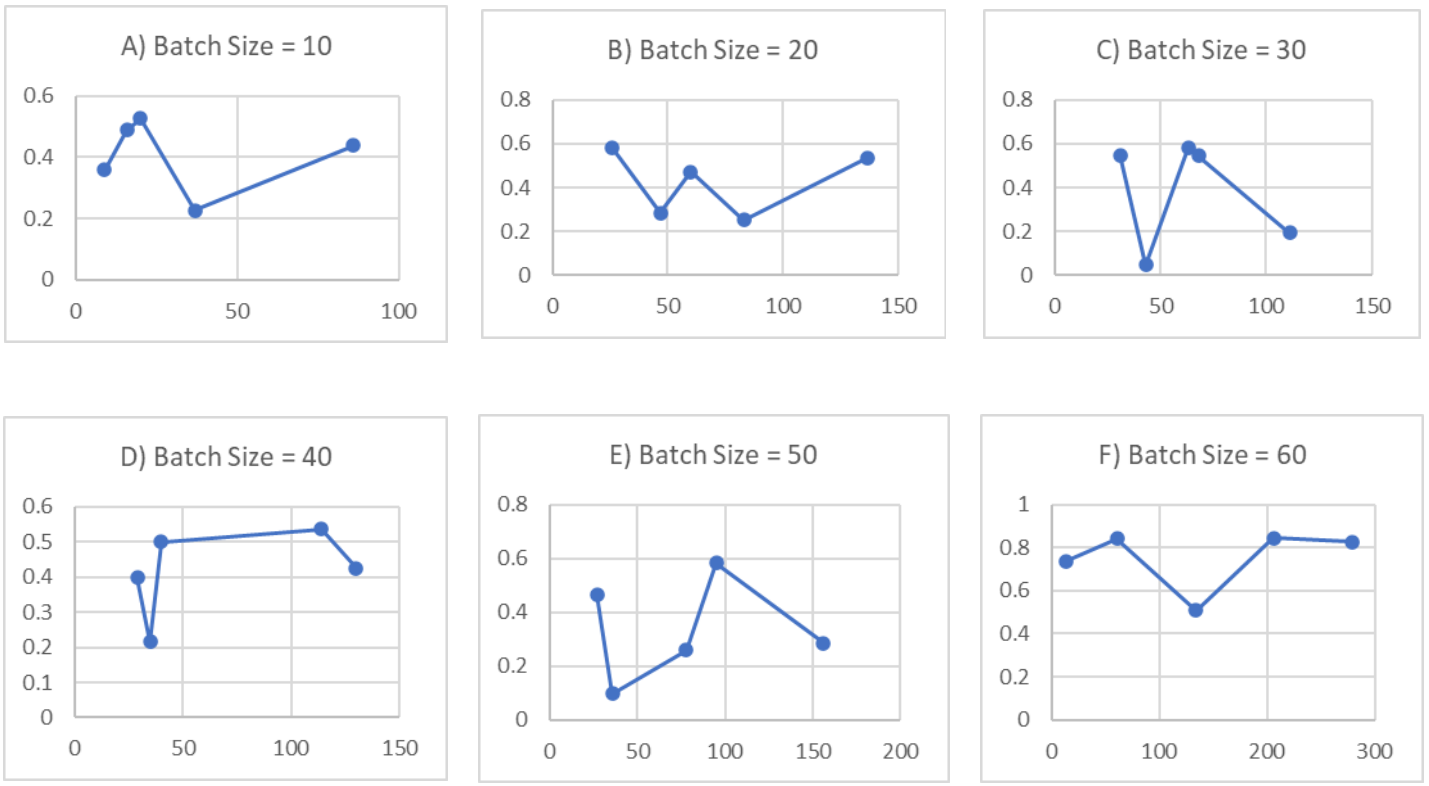

Figure 10: Different Silhouette Measures For Clustering Streaming Data Based On Streaming K-Means (X-Axis: No. Objects, Y-Axis: Silhouette).

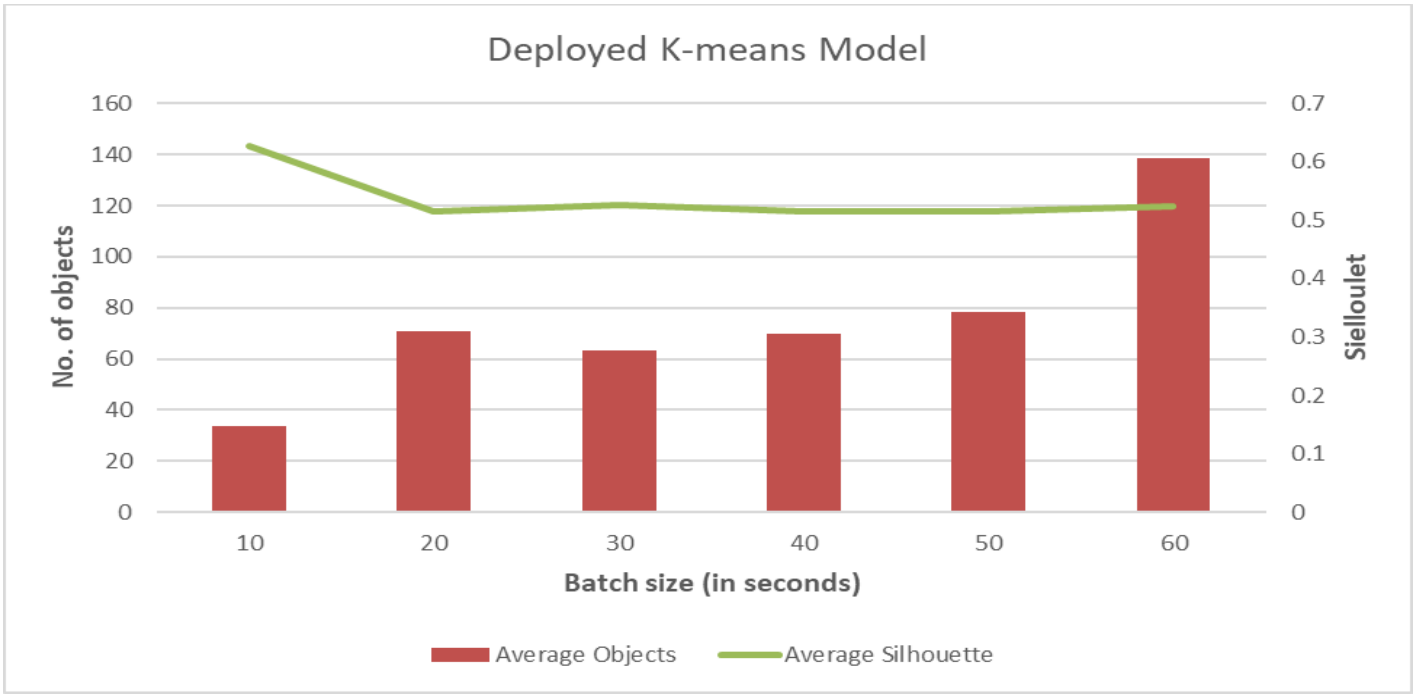

Figure 11: Summary Statistics Of The Real-Time Clustering Based On The Deployed K-Means Model.

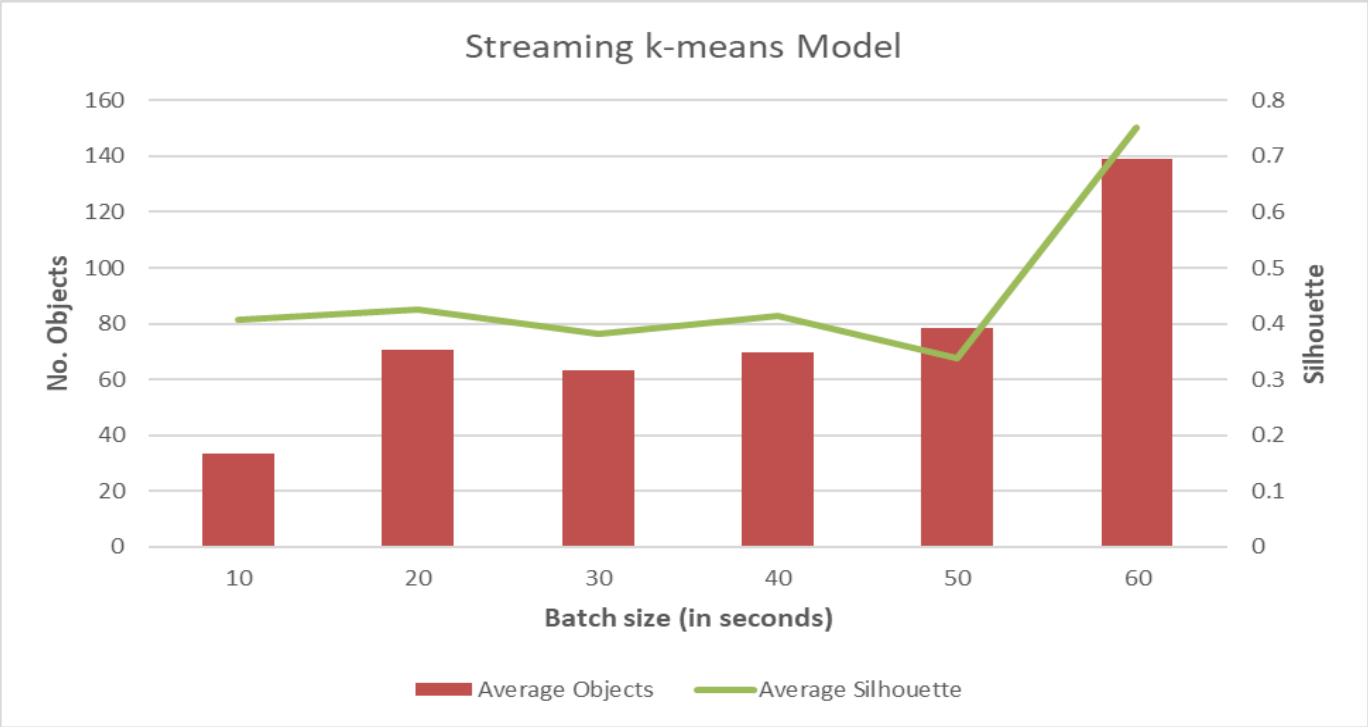

Figure 12: Summary Statistics Of The Real-Time Clustering Based On The Streaming K-Means Model.

Although all Silhouette values exceed 0.338, it might be a good indicator for some applications and not good for other applications for ensuring stability in the clustering.

In the light of the literature review, Gunawardena, et al [34] conducted a machine learning model based on Spark similar to the first model in the proposed framework in order to identify the prevalent Uber locations pickups and use this model to analyze real-time streaming Uber data in Kubernetes environment [35] in a distributed manner. They used Apache Kafka for data ingestion, unlike the proposed framework that uses socket streaming. The proposed framework with its two models is not implemented in a distributed environment and but in local machine and not in the Google Cloud Platform (GCP) like their model. They also used elbow plot to select the number of clusters but there is some confusion about the choice. Their model is evaluated by CPU utilization with the time of real-time data. Santhi et al [36] overcome the estimation of the initial centroids of the K-Means by implementing in the pre-processing step the Bat and the Firefly techniques using Spark. The performance is evaluated based on the execution time. In the proposed, the number of clusters is determined by the elbow, and knee methods and the performance is evaluated by the Silhouette Coefficient that measures the validation of consistency within clusters. Backhoff, et al [28] used StreamingKMeans in a different dataset with a decay factor is equal to 0 in the experiments unlike the second model in the proposed framework that is equal to 1. The K-Means algorithm is used with Silhouette Coefficient measure based on Euclidean distance (like the proposed framework) in the model of analyzing customer behavior by Anitha P et al [33].

For the business domain and the use case study (Uber trips/pickups), some analysis can be applied to help businesses and decision-makers for answering for some questions like which cluster had the highest number of pickups (that may consider the most crowded areas that had a lot of pickups), how many pickups occurred in each cluster, which hours of the day and which cluster had the highest number of pickups, etc. Executives can decide based on the results such as targeting ads, increase the number of cars in a specific base, the crowded areas, and avoid them and more and more.

ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

6. CONCLUSION

A machine learning clustering algorithm is applied to big data specifically Uber trips pickups as a case study.

Firstly, k-means is applied to historical data in order to build k-means model that will be used later for real-time clustering. The k-means is applied after choosing the suitable number of clusters by using elbow method.

Secondly, the proposed research made use of the deployed k-means model with the concept of streaming to apply clustering on real-time data. pyspark.streaming provides support for StreamingContext which in turn supports socketTextStream.

Thirdly, to enhance the clustering process, StreamingKMeans is used to dynamically grade the clusters, and update them as new data arrives.

The predictions can help businesses and the executives in the decision-making process as a concept of implications of real-time big data analytics on business intelligence that facilitates self-service analytics.

In closing, we cared to build general clustering models that not suitable only for the used dataset but any numeric dataset that needs to apply clustering on.

7. FUTURE WORK

The proposed framework describes the clustering for streaming data using two different models that are applied with spark in single-machine mode. The work will continue multiple-machines mode. The future work will also include how to apply real-time classification with multiple class labels using a variety of machine learning classification algorithms like Decision Tree, Naive Bayes classifier and Support Vector Machine. It will continue also with Kafka as a way for data ingestion instead of socket streaming. As the business evolves every day and the businesses and decision-makers need to make decisions in the proper time and as soon as the data is available, real-time dashboards with dynamic and interactive charts will be taken into consideration as a part of the future work. At last, Different types of data will be taken into considerations as part of the future.

REFERENCES

[1] Rouse M., business intelligence (BI), [Internet]. Available from: https://searchbusinessanalytics.techtarget.com/definition/business-intelligence-BI (last accessed on 3/1/2021)

[2] Techopedia, Business Intelligence (BI), [Internet]. Available from: https://www.techopedia.com/definition/345/business-intelligence-bi (last accessed on 3/1/2021)

[3] Golfarelli M, Rizzi S, Cella I. Beyond data warehousing: what's next in business intelligence?. InProceedings of the 7th ACM international workshop on Data warehousing and OLAP 2004 Nov 12 (pp. 1-6).

[4] Rouse M., big data, [Internet]. Available from: https://searchdatamanagement.techtarget.com/definition/big-data (last accessed on 3/1/2021)

[5] Gartner IT Glossary (n.d.)., Big Data, [Internet]. Available from: https://www.gartner.com/en/information-technology/glossary/big-data (last accessed on 3/1/2021)

[6] Oussous A, Benjelloun FZ, Lahcen AA, Belfkih S. Big Data technologies: A survey. Journal of King Saud University-Computer and Information Sciences. 2018 Oct 1;30(4):431-48.

[7] Gandomi A, Haider M. Beyond the hype: Big data concepts, methods, and analytics. International journal of information management. 2015 Apr 1;35(2):137-44.

[8] Saeed MM, Al Aghbari Z, Alsharidah M. Big data clustering techniques based on Spark: a literature review. PeerJ Computer Science. 2020 Nov 30;6:e321.

[9] Rouse M., big data analytics, [Internet]. Available from: https://searchbusinessanalytics.techtarget.com/definition/big-data-analytics (last accessed on 3/1/2021)

[10] Wani MA, Jabin S. Big data: issues, challenges, and techniques in business intelligence. InBig data analytics 2018 (pp. 613-628). Springer, Singapore.

[11] BI-SURVEY, Top Business Intelligence Trends, [Internet]. Available from: https://bi-survey.com/top-business-intelligence-trends (last accessed on 3/1/2021)

[12] Sharma N, Agarwal M. Real-Time Big Data Analysis Architecture and Application. InData Science and Big Data Analytics 2019 (pp. 313-320). Springer, Singapore.

[13] Bekker A., A Comprehensive Guide to Real-Time Big Data Analytics, [Internet]. Available

from: https://www.scnsoft.com/blog/real-time-big-data-analytics-comprehensive-guide (last accessed on 3/1/2021)
[14] Li X, Zhang Z. Research and Analysis for Real-Time Streaming Big Data Based on Controllable Clustering and Edge Computing Algorithm. IEEE Access. 2019 Nov 26;7:171621-32.
[15] Tank DM. Enable better and timelier decision-making using real-time business intelligence system. International Journal of Information Engineering and Electronic Business. 2015;7(1):43.
[16] Ram J, Zhang C, Koronios A. The implications of big data analytics on business intelligence: A qualitative study in China. Procedia Computer Science. 2016 Jan 1;87:221-6.
[17] Kılınç D. A spark-based big data analysis framework for real-time sentiment prediction on streaming data. Software: Practice and Experience. 2019 Sep;49(9):1352-64.
[18] Sehgal D, Agarwal AK. Real-time sentiment analysis of big data applications using Twitter data with Hadoop framework. InSoft computing: theories and applications 2018 (pp. 765-772). Springer, Singapore.
[19] MacQueen J. Some methods for classification and analysis of multivariate observations. InProceedings of the fifth Berkeley symposium on mathematical statistics and probability 1967 Jun 21 (Vol. 1, No. 14, pp. 281-297).
[20] Harifi S, Byagowi E, Khalilian M. Comparative study of apache spark MLlib clustering algorithms. InInternational Conference on Data Mining and Big Data 2017 Jul 27 (pp. 61-73). Springer, Cham.
[21] Salloum S, Dautov R, Chen X, Peng PX, Huang JZ. Big data analytics on Apache Spark. International Journal of Data Science and Analytics. 2016 Nov 1;1(3-4):145-64.
[22] Uber Dataset, [Internet]. Available from: https://drive.google.com/file/d/14675RC0xILCVoyUM1WCwcF545JuO2d6o/view?usp=sharing (last accessed on 3/1/2021)
[23] Documentation for spark and pyspark, [Internet]. Available from: https://spark.apache.org/docs/latest/api/python/index.html (last accessed on 3/1/2021)
[24] Kodinariya TM, Makwana PR. Review on determining number of Cluster in K-Means Clustering. International Journal. 2013 Nov;1(6):90-5.
[25] Bholowalia P, Kumar A. EBK-means: A clustering technique based on elbow method and k-means in WSN. International Journal of Computer Applications. 2014 Jan 1;105(9).
[26] Spark documentation for socketTextStream, [Internet]. Available from: https://spark.apache.org/docs/2.1.0/api/python/pyspark.streaming.html (last accessed on 3/1/2021)
[27] Spark documentation for DStreams, [Internet]. Available from: https://spark.apache.org/docs/latest/api/java/org/apache/spark/streaming/dstream/DStream.html (last accessed on 3/1/2021)
[28] Backhoff O, Ntoutsi E. Scalable online-offline stream clustering in apache spark. In2016 IEEE 16th International Conference on Data Mining Workshops (ICDMW) 2016 Dec 12 (pp. 37-44). IEEE.
[29] Spark documentation and examples for streaming k-means, [Internet]. Available from: https://spark.apache.org/docs/2.2.0/mllib-clustering.html#streaming-k-means (last accessed on 3/1/2021)
[30] Satopaa V, Albrecht J, Irwin D, Raghavan B. Finding a" kneedle" in a haystack: Detecting knee points in system behavior. In2011 31st international conference on distributed computing systems workshops 2011 Jun 20 (pp. 166-171). IEEE.
[31] Google Map Tutorial, [Internet]. Available from: https://pypi.org/project/gmplot/ (last accessed on 3/1/2021)
[32] Bui SM, Gorro K, Aquino GA, Sabellano MJ. An analysis of DRR suggestions using K-means clustering. InProceedings of the 2017 International Conference on Information Technology 2017 Dec 27 (pp. 76-80).
[33] Anitha P, Patil MM. RFM model for customer purchase behavior using K-Means algorithm. Journal of King Saud University-Computer and Information Sciences. 2019 Dec 25.
[34] Gunawardena TM, Jayasena KP. Real-Time Uber Data Analysis of Popular Uber Locations in Kubernetes Environment. In2020 5th International Conference on Information Technology Research (ICITR) 2020 Dec 2 (pp. 1-6). IEEE.
[35] What is Kubernetes, [Internet]. Available from: https://kubernetes.io/docs/concepts/overview/what-is-kubernetes/ (last accessed on 10/2/2021)
[36] Santhi V, Jose R. Performance analysis of parallel k-means with optimization algorithms for clustering on spark. InInternational Conference on Distributed Computing and Internet Technology 2018 Jan 11 (pp. 158-162). Springer, Cham.