Sentimental Analysis using Big Data on Social Dilemma

**Chapter 1: Introduction**

Television is viewed as a lot more than just a conventional broadcast medium by this age. The way people view television has significantly changed with the introduction of digital technology as well as internet streaming services. Today, a television can refer to more than just a physical equipment in a living area. It has developed into a versatile and open kind of media that can be viewed on a range of screens, including computers, smartphones, and tablets. Younger generations, who favor simplicity and variety in their viewing habits, have welcomed this move.

Streaming services like Prime, Netflix, and others are becoming more and more well-liked because they provide a huge selection of video on demand that can be tailored to indivudual tastes. This makes it possible for people to watch their preferred television programmed and films at any time they desire. Thanks to the practicality of streaming platforms, binge-watching full seasons, or uncovering new shows has become a frequent practice.

Additionally, social media and television have grown deeply entwined, with viewers actively participating in real-time conversations and comments. A feeling of community and sharing viewing experiences have been established by hashtags, live tweeting, and internet communities devoted to shows.

Twitter is a networking platform that has done an excellent job of thoroughly ingraining itself into everyday lives of those who use it. Twitter typically complements current media channels, such broadcasting, or online mainstream media, in these applications rather than necessarily replacing them, giving its users additional possibilities to actively participate in the larger media landscape. This is especially true when Twitter is used in conjunction with television, either as a basic backchannel to live broadcast or for more advanced applications [1].

This immediate viewer feedback, which was originally designed for other consumers but is now readily accessible to broadcasters as well as researchers, in turn offers possibly a very rich flow of information to understand how people feel about television programmes. Analysis can be done on the detailed, minute-by-minute qualitative and quantitative information that currently surrounds television in the shape of tweets. It offers the foundation for an advanced, real-time measurement and comprehension of audience activity. As a result, Twitter no longer just serves as a backchannel for the television programme but also becomes an integral component of it. Sentiment analysis is a method for examining the feelings, viewpoints, and attitudes contained in a text. In order to apply sentiment analysis to TV shows, it is necessary to examine user-generated content (UGC) connected to particular TV series, including reviews, comments from viewers, social media posts, and more.

TV programmer sentiment analysis can offer insightful information about how viewers experience and respond to various elements of the plot, language, and production value. The ability to categories attitudes as favorable, negative, or neutral makes it feasible to assess a TV show's general reception and pinpoint strengths or places for improvement.

This paper focus upon understanding sentiments of the popular TV show named Social Dilemma. Popular documentary "The Social Dilemma" examines how social media networks and their algorithms affect society. The movie, that was released in 2020, included interviews with former workers of well-known internet firms, such as Google, Facebook, and Twitter. These individuals’ express worries about the influence of user behaviour, the dissemination of inaccurate data, and the erosion of privacy. It describes how social media sites exploit user information as well as algorithms to increase engagement and revenue, frequently encouraging addictive behaviors and amplifying divisive content. The impact of these practices on democracy, societal well-being, and mental health is also covered in the movie [2].

This paper utilizes Linear Regression, Random Forest, Decision Tree along with Subject Vector Regression across a window size of 10, 20 and 30 with preprocessing of Apache Spark, MySQL and Cassandra to predict the sentiments based on the documentary of ‘Social Dilemma’.

**Chapter 2: Literature Review**

Assessing audience thoughts and feelings towards TV shows requires careful sentiment analysis. With an emphasis on the approaches, techniques, datasets, and assessment criteria used in these studies, this literature review seeks to give an overview of the current research in sentiment evaluation for TV episodes.

Early studies on TV show sentiment analysis frequently used a bag-of-words methodology. [3] classified sentiment based on the frequency of positive and negative phrases in TV show reviews using a straightforward frequency-based technique.

Support vector machines (SVM) were investigated for the purpose of classifying TV programme evaluations' sentiments in [4], with encouraging outcomes. Deep learning techniques have drawn a lot of attention in TV programme and other sentiment analysis studies. A Convolutional Neural Network (CNN) model was suggested by [5] to analyse sentiment in TV show conversation while taking context and dependencies into account.

A Twitter dataset created exclusively for TV show sentiment analysis was introduced by [6]. Researchers were able to examine sentiment during particular episodes or seasons of popular TV shows because to the dataset's inclusion of tweets relevant to those shows. Lexicon-based methods make use of sentiment lexicons, which are pre-defined word lists with corresponding sentiment scores. The use of emotion lexicons with TV show analysis of sentiment has been studied by researchers. As an example, [7] assigned ratings for sentiment to TV show language and examined emotional arcs throughout episodes using sentiment lexicons. In addition, [8] increased performance by classifying sentiment in TV programme reviews using Word2Vec embeddings.

In TV programme sentiment analysis, RNNs, especially Long Short-Term Memory (LSTM) networks, have proven their potential in collecting sequential information and dependencies. An LSTM-based model for categorising sentiment in TV show captions was proposed by [9]. Since SVM has been frequently used for categorising sentiment in TV show data, [10] also used SVM to categorise sentiment in TV show tweets while taking into account both textual and non-textual elements.

**Chapter 3: Data and Methodology**

**3.1 Data**

The "The Social Dilemma Tweets" dataset, which is accessible on Kaggle, is a thorough compilation of tweets specifically on the documentary "The Social Dilemma." The documentary looks at how social media sites affect several facets of daily life and their effects on society. The dataset includes tweets that were published between September 4 and the 11th of September, representing the online discussion and dialogues surrounding the movie at that time. Each of the entries in the collection represents a single tweet and contains important details including the tweet's distinctive identifier, the posting timestamp, the author's user ID, the tweet's content, and other pertinent metadata. The attitudes, viewpoints, and discussions surrounding "The Social Dilemma" on Twitter have been examined using this dataset [11].

**3.2 Methodology**

**A picture containing text, screenshot, design

Description automatically generated**

Figure : Framework

In the methodology above, we can see that a holistic framework of utilising the Big data tools along with two separate databases has been done. Hadoop MapReduce's scalability, tolerance for failures, and distributed processing characteristics make it useful for sentiment evaluation of TV broadcasts. MapReduce enables for parallel processing over numerous machines, which speeds up execution when there are enormous amounts of data to analyse. The fault tolerance feature guarantees data integrity and continuous processing. Performance is further improved by Hadoop's data localization, which reduces data transmission and network overhead. Hadoop is also a cost-effective solution because it can be installed on low-cost hardware. Overall, Hadoop MapReduce offers a scalable and effective framework for TV show data sentiment analysis, making it a great option for processing massive datasets.

* + 1. **Hadoop**

In order to handle the Big Data programmes in the Hadoop cluster, Hadoop is defined as a set of Software Utilities that run across an ensemble of computers employing Software Frameworks using a distributed storage environment. The following are the three parts of Hadoop, Yarn, HDFS and MapReduce but we shall discuss the last two in this paper [12].

1. MapReduce

Hadoop Hadoop's processing engine is called MapReduce. The processing for the MapReduce method is carried out at the slave nodes, while the finished product is transferred to the master node.

The full set of data is processed using a code that contains data. In contrast to the data itself, this coded data is typically extremely little. A computationally intensive procedure can be carried out on computers with as little as a few kilobytes of code.

The following MapReduce steps for Big Data are performed on the data.

1. Input split

The input splits used in a MapReduce job for big data are fixed-size portions of the input. An input split is a portion of an input that just one map uses.

ii. Mapping

This is the initial stage of the map-reduce program's execution. Data from each split is provided to a mapping function during this phase, which generates output values. In our example, the mapping phase's task is to count how many times each word appears in the input divides (additional information about the input split is provided below) and to create a list in the format of <word, frequency>

iii. Shuffling

The result from the mapping phase is used in this phase. Its duty is to compile the pertinent results from the Mapping step. The same terms and their corresponding frequencies are combined in our case.

iv. Reducing

The results from the shuffling phase are collected in this step. A single value for output is produced after this phase combines the data from the shuffling phase. This stage, in a nutshell, summarises the entire dataset [13].

On an overall scale, Hadoop breaks the project up into tasks. Two categories of tasks exist:

* **Map tasks** -that include splitting and mapping
* **Reduce tasks** -that include shuffling and [reducing](https://www.guru99.com/introduction-to-mapreduce.html) [14].

1. HDFS

In HDFS, data is distributedly stored. Name node and data node are the two parts of HDFS. There can be several data nodes, but there's only one name node. The Hadoop Distributed File System (HDFS), which offers distributed storage, data protection, and high failure tolerance, is also available.

The Hadoop cluster is made up of two HDFS nodes, one of which is the master node, or name node, and the other is the slave node, or data node. The operation of the data nodes is handled by the name node. It keeps the metadata as well.

Data is read, written to, processed, and replicated by the data nodes. Additionally, they transmit to the name node impulses known as heartbeats. The data node's state is indicated by these heartbeats. The name node distributes input data among the data nodes and replicates it among the data notes if you have a specific store of input data. By default, data replication is done three times. This is done so that if a commodity machine malfunctions, a new machine with the same data can take its [place](https://www.simplilearn.com/tutorials/hadoop-tutorial/what-is-hadoop#:~:text=It%20is%20the%20most%20commonly,resource%20management%20unit%20of%20Hadoop.) [15].

* + 1. **Databases**

Databases are organised, maintained collections of structured data that allow for effective information retrieval, modification, and analysis. They act as the core repositories for the management and organisation of enormous amounts of data, offering tools for data organisation, retrieval, and upkeep [16].

There are many kinds of databases, such as:

1. SQL databases: Databases with specified relationships between its tables are known as relational databases. They define, query, and work with the data using the SQL language. This essay employs MySQL.
2. NoSQL databases: Non-relational databases that offer versatility in storing and retrieving information include NoSQL (Not simply SQL) databases. They offer great scalability and performance while handling unstructured and semi-structured data. Key-value stores, storage of documents (like MongoDB), columnar databases (like Cassandra), and other types of NoSQL databases are examples.
   * + 1. MySQL

The relational database management system (RDBMS) MySQL is free and open source which is reachable to every people , and it offers a platform for maintaning, organising, and retrieving structured data. Due to its dependability, effectiveness, and simplicity of use, it is widely employed in a variety of applications and industries [17].

The client, the server, and the storage engine make up the three core parts of MySQL's architecture. For tasks like obtaining data, insertion, updating, and deletion, the client apps communicate with the MySQL server via SQL queries.

The server could have takes these queries, interprets them, and communicates with the real storage engine in order to read or change the data stored in the database. There are a lot of factors that contribute to MySQL's popularity.Thats why it is open source, users can freely edit and adapt the source code to meet their what they need. This is the first advantage. Furthermore, MySQL has high-performance characteristics that make it appropriate for applications that need quick and effective data processing. MySQL is well known for its dependability, stability, and extensive community support. With that kind of security updates and bug fixes, it is continuously can being developed and updated. With its support for a variety of programing languages and operating systems, it is flexible and easy for developpers to use. Becuse of its dependability, scalability, and excelent data management capabilities, MySQL has been used in this study to store data following the MapReduce() cluster's "reduce()" function. The relational database system that is well-established and often used, MySQL, can effectively store and retrieve structured data. The characteristics it offers, such query optimisation, transaction support, and indexing, make it suited for managing big amounts of Twitter data. Additionally, MySQL's adaptability to a variety of frameworks and programming languages makes it simpler to integrate with the MapReduce cluster. Overall, MySQL guarantees data integrity and makes it possible to easily store and analyse the condensed Twitter data to gain additional insights [18]. Due to MySQL's interoperability with a variety of programming languages and frameworks, it is possible to integrate it with MapReduce cluster as well as other data processing systems, which is an additional intriguing element of using this.

* + - 1. Cassandra

Cassandra is a highly scalable and distributed NoSQL database management system known for its ability to handle massive amounts of data across multiple servers. It was developed by Facebook and is now open-source. Cassandra's architecture is designed to provide high availability and fault tolerance by distributing data across a cluster of nodes. Each node in the cluster is capable of handling read and write requests independently, enabling linear scalability as more nodes are added. Cassandra follows a column-family data model, allowing for flexible schema design and dynamic column modifications without downtime.

Cassandra excels in handling large reading and writing throughput, which qualifies it for applications requiring real-time access to large amounts of data. It offers adjustable consistency levels so users may regulate data accessibility and uniformity according to the requirements of their applications. Furthermore, even in the event of node failures, Cassandra's decentralized architecture and data replication provide high availability. Applications dealing with huge and changing data sets frequently choose it because of its adaptable schema model and horizontal scalability. Cassandra is an excellent choice for applications that need high availability, tolerance for faults, scalability, and effective data operations because to its design and feature set.

Cassandra's distributed and scalable architecture makes it useful for storing data after PySpark consumes it. Because of its high write throughput capacity, it makes it possible to store processed data in huge numbers effectively. Data availability and durability are ensured by Cassandra's robust fault tolerance and regeneration features. By introducing more nodes, it may scale horizontally to accommodate an expanding data volume. Additionally, Cassandra's adaptable data model permits dynamic schema modifications, facilitating the effective storing of various data kinds that PySpark processes. A dependable and scalable option for storing data that PySpark processes is provided by this combo [19].

* + 1. **Apache Pyspark**

An interface for Python programming is provided by Apache PySpark, a free and open-source distributed computing platform built on top of Apache Spark. On huge datasets, it makes it possible to do scalable and powerful processing of information, data analysis, and machine learning operations [20]

PySpark's architecture is organized in a master-worker fashion. It consists of a cluster of worker nodes that carry out computations and a driver programme that manages task execution. To obtain resources and assign tasks to the workers, the driver programme interacts with the cluster management. A Spark executor process is active on each worker node, controlling how tasks are carried out and how data is stored there. Resilient Distributed Datasets (RDDs) are used by PySpark as the primary data model for distributed computation. Scalable and parallel data processing are made possible by PySpark thanks to the cooperation of the driver programme, cluster manager, worker nodes, and RDDs [21,22]

There are several reasons why TV show sentiment analysis information from a MySQL database can be read using Apache PySpark. First, PySpark offers an easy-to-use interface for navigating and analyzing massive datasets, which is essential for managing sentiment data analysis that can be rather enormous. Second, PySpark's distributed computing abilities allow for simultaneous data processing across a cluster, facilitating quicker sentiment analysis of data from the MySQL database. Additionally, seamless connectivity and data retrieval are made possible by PySpark's interaction with numerous data sources, including MySQL. The extraction, modification, and evaluation of sentiment data are further facilitated by PySpark's comprehensive set of tools for data manipulation and machine learning packages.All things considered, PySpark provides an adaptable, distributed, and packed with features environment for conducting sentiment analysis on TV show data saved in a MySQL database.

* + 1. **Modelling**

The following machine-learning models are used for sentiment analysis.

3.2.4.1 *Logistic Regression (LR):* For binary or multi-class classification, LR is a linear model. It simulates the association among the input characteristics and the likelihood of falling into a specific class. To create predictions, LR employs a logistic function and calculates coefficients for each feature. It works well with linearly separated data, is computably efficient, and is comprehensible.

*3.2.4.2 Decision Trees (DT):* To categorise data, DT creates a hierarchical structure of decision nodes based on attributes. Each node in the leaf represents a class label, and each internal node provides a judgement based on a feature. DTs can capture non-linear correlations, handle category and numerical information, and are adaptable. However, they may lack generalisation and be prone to overfitting.

*3.2.4.3 Random Forest (RF):* RF is a decision tree-based ensemble learning technique. Using random portions of training data and features, it creates a collection of decision trees. By minimising overfitting through ensemble averaging, RF outperforms DT. It offers feature importance measures, manages high-dimensional data, and is reliable.

*3.2.4.4 Support Vector Machines (SVM):* SVM creates a hyperplane in the feature space that maximally separates several classes. By utilizing kernel functions, it is capable of handling non-linear relationships and high-dimensional data. SVM seeks to increase the space between classes, improving generalization. SVM can, however, be computationally costly and hyperparameter selection sensitive.

* + 1. **Python Dashboard**

A Python dashboard for "Social Dilemma" TV program analysis of sentiment is a graphical user interface that shows and examines sentiment-related data taken from the show's data. It offers a simple and interactive way to investigate sentiment scores, sentiment trends, and other insights linked to sentiment. The dashboard functions by initially gathering information about TV shows, like tweets or reviews. After that, sentiment analysis techniques are used to categorize each entry's sentiment as either positive, negative, or neutral. To produce sentiment statistics and trends, the sentiment analysis data are processed and compiled. Visualizations of the sentiment analysis data, including line graphs or bar plots, are made using Python modules like Dash. Users can filter data, discover sentiment trends, engage with the dashboard, and learn more about the analysis of sentiment of the TV show material. The ability to create robust and interactive dashboards solely in Python is made possible by Python Dash's capability to easily link Python's data processing abilities with dynamic web-based visualizations [23].

**Chapter 4: Results and Discussion**

4.1 Results

Text

Description automatically generated

Figure : Hadoop Installation

Text

Description automatically generated

Figure : Creation of Hadoop Home

Graphical user interface, text, application

Description automatically generated

Figure : Hadoop Overview

The figures above showcase the installation of Hadoop.

A screenshot of a computer screen

Description automatically generated with medium confidence

Figure : PySpark Preprocessing

Figure 5 showcases the preprocessing done by PySpark.

A screenshot of a computer

Description automatically generated

Figure : Date Sentiment Aggregation

A screenshot of a computer program

Description automatically generated with low confidence

Figure : Save to Cassandra

A screenshot of a computer

Description automatically generated

Figure : Cassandra Output

A screenshot of a computer program

Description automatically generated with medium confidence

Figure : Code for Dashboard

A picture containing line, plot, screenshot

Description automatically generated

Figure : Forecast Result for window size of 10 days for Linear Regression

A picture containing line, plot, text, diagram

Description automatically generated

Figure : Forecast Result for window size of 10 days for Decision Tree

A picture containing line, screenshot, plot, diagram

Description automatically generated

Figure : Forecast Result for window size of 20 days for Decision Tree

A picture containing text, font, line, handwriting

Description automatically generated

Figure 13: Window size 10,20 and 30 with models LR, DT and RF for 7, 30 and 90 Days

Figures 10, 11, 12, and 13 all show the dashboard represenations of the time-series data.

4.2 Discussion

The paper utilises MySQL and Cassandra as the 2 pillars of processing. The MySQL serves as a raw data warehouse to store a semi structured or semi processed data however with its fast response time the class based PySpark lazy running enhances the data and drives it to Cassandra in a preprocessed format for aggregation and data storage. Post which the dashboard utilized the Cassandra data via CSV for ARIMA modelling and forecasting. Further the preprocessing pipeline uses NLTK library to label and stem the sentences for labelling along with labialization. Post which the sentiment aggregate is calculated across dates allowing the generic potential shift of sentiment per day. This is generalized and predicted by random forest, linear regression and decision tree across window size of 10, 20 and 30. The database test comparison is in relevance to the insertion deletion and update time with reference to the CPU and memory utilization as discussed in the compare code. It showcases that the Cassandra data base has a faster insertion and MySQL 1has a faster response time

**Chapter 5: Conclusion**

Finally, the sentiment analysis of the television programme "Social Dilemma" using a Hadoop MapReduce cluster, MySQL data storage, Apache PySpark for processing and storing data in Cassandra, and including linear regression, decision tree, random forest, and SVM models for the time series analysis provides a thorough framework for comprehending and forecasting sentiment trends related to the programme.

Faster sentiment analysis is made possible by the deployment of a Hadoop MapReduce cluster, which enables the efficient processing of massive volumes of data in a dispersed and parallel fashion. MySQL data storage offers a dependable and organised database solution for quick data access and storage. The data processing features of Apache PySpark enable effective data reading and storing in Cassandra and enable smooth connection with Hadoop. With this setup, sentiment data management and analysis are scalable and flexible. The underlying patterns and trends in sentiment over time can be captured by using time series analysis with linear regression, decision tree, random forest, and SVM models. These models provide several methods for deriving predictions from historical sentiment data and temporal dynamics analysis.

The overall goal of this integrated strategy is to get understanding of the sentiment patterns of the TV show "Social Dilemma" by combining the strength of Hadoop, MySQL, Apache PySpark, and other predictive models. It gives a thorough framework for analysing sentiment, making it easier to comprehend how the audience reacted to the performance and forecast sentiment trends in the future.

**Chapter 6: References**

1. Barba González, C., García Nieto, J.M., Navas Delgado, I. and Aldana Montes, J.F., 2016. A fine grain sentiment analysis with semantics in tweets. International Journal of Interactive Multimedia and Artificial Intelligence, 3 (6), 22-28.
2. Ariska, E. and Yanti, Y., 2022. An Analysis of Directive Speech Acts and Cultural Values Reflected in the Netflix Documentary" The Social Dilemma" (Doctoral dissertation, Universitas Bung Hatta).
3. Pang, B. and Lee, L., 2008. Opinion mining and sentiment analysis. Foundations and Trends® in information retrieval, 2(1–2), pp.1-135.
4. Hu, M. and Liu, B., 2004, August. Mining and summarizing customer reviews. In Proceedings of the tenth ACM SIGKDD international conference on Knowledge discovery and data mining (pp. 168-177).
5. Tang, D., Wei, F., Yang, N., Zhou, M., Liu, T. and Qin, B., 2014, June. Learning sentiment-specific word embedding for twitter sentiment classification. In ACL (1) (pp. 1555-1565).
6. Balahur, A., Steinberger, R., Kabadjov, M., Zavarella, V., Van Der Goot, E., Halkia, M., Pouliquen, B. and Belyaeva, J., 2013. Sentiment analysis in the news. arXiv preprint arXiv:1309.6202.
7. Socher, R., Perelygin, A., Wu, J., Chuang, J., Maning, C.D., Ng, A.Y. and Potts, C., 2013, October. Recursive deep models for semantic compositionality over a sentiment treebank. In Proceedings of the 2013 conference on empirical methods in natural language processing (pp. 1611-1692).
8. Park, H.J., Song, M.C. and Shin, K.S., 2018. Sentiment analysis of korean reviews using cnn: Focusing on morpheme embedding. Journal of intelligence and information systems, 24(2), pp.59-83.
9. Murthy, G.S.N., Allu, S.R., Andhavarapu, B., Bagadi, M. and Belusonti, M., 2020. Text based sentiment analysis using LSTM. Int. J. Eng. Res. Tech. Res, 9(05).
10. Sabariah, M.K. and Effendy, V., 2015, May. Sentiment analysis on Twitter using the combination of lexicon-based and support vector machine for assessing the performance of a television program. In 2015 3rd International Conference on Information and Communication Technology (ICoICT) (pp. 386-390). IEEE.
11. Kash (2021) The social dilemma tweets - text classification, Kaggle. Available at: https://www.kaggle.com/datasets/kaushiksuresh147/the-social-dilemma-tweets (Accessed: 06 May 2023).
12. Madani, Y., Erritali, M. and Bengourram, J., 2019. Sentiment analysis using semantic similarity and Hadoop MapReduce. Knowledge and Information Systems, 59, pp.413-436.
13. Garg, K. and Kaur, D., 2019. Sentiment analysis on Twitter data using Apache Hadoop and performance evaluation on Hadoop MapReduce and Apache Spark. In Proceedings on the International Conference on Artificial Intelligence (ICAI) (pp. 216-261). The Steering Committee of The World Congress in Computer Science, Computer Engineering and Applied Computing (WorldComp).
14. Parveen, H. and Pandey, S., 2016, July. Sentiment analysis on Twitter Data-set using Naive Bayes algorithm. In 2016 2nd international conference on applied and theoretical computing and communication technology (iCATccT) (pp. 410-418). IEEE.
15. Sehgal, D. and Agarwal, A.K., 2016, November. Sentiment analysis of big data applications using Twitter Data with the help of HADOOP framework. In 2016 international conference system modeling & advancement in research trends (SMART) (pp. 251-255). IEEE.
16. Cattell, R., 2011. Scalable SQL and NoSQL data stores. Acm Sigmod Record, 39(4), pp.12-27.
17. (No date) MySQL. Available at: https://www.mysql.com/ (Accessed: 08 May 2023).
18. Vora, M.N., 2011, December. Hadoop-HBase for large-scale data. In Proceedings of 2011 International Conference on Computer Science and Network Technology (Vol. 1, pp. 601-605). IEEE.
19. Kanavos, A., Vonitsanos, G., Mohasseb, A. and Mylonas, P., 2020, October. An entropy-based evaluation for sentiment analysis of stock market prices using Twitter data. In 2020 15th International Workshop on Semantic and Social Media Adaptation and Personalization (SMA (pp. 1-7). IEEE.
20. Shetty, S.D., 2021, March. Sentiment analysis, tweet analysis and visualization on big data using Apache Spark and Hadoop. In IOP Conference Series: Materials Science and Engineering (Vol. 1099, No. 1, p. 012002). IOP Publishing.
21. Di Martino, B., Colucci Cante, L., Graziano, M. and Enrich Sard, R., 2021. Tweets analysis with big data technology and machine learning to evaluate smart and sustainable urban mobility actions in Barcelona. In Complex, Intelligent and Software Intensive Systems: Proceedings of the 14th International Conference on Complex, Intelligent and Software Intensive Systems (CISIS-2020) (pp. 510-519). Springer International Publishing.
22. Stančin, I. and Jović, A., 2019, May. An overview and comparison of free Python libraries for data mining and big data analysis. In 2019 42nd International convention on information and communication technology, electronics and microelectronics (MIPRO) (pp. 977-982). IEEE.
23. Dash documentation & user guide (no date) Plotly. Available at: https://dash.plotly.com/ (Accessed: 06 May 2023).