# Sentiment Analysis Based on Smart Human Mobility: a comparative study of ML models

Luís Rosa[1], Hugo Faria[1], Reza Tabrizi[1], Simão Gonçalves[1], Fábio Silva[2], and Cesar Analide[1]

[1] [1]University of Minho, ALGORITMI Center, Dep. of Informatics, Braga, Portugal
[2] CIICESI, ESTG, Politécnico do Porto, Felgueiras, Portugal
pg44415@alunos.uminho.pt, pg33877@alunos.uminho.pt,
pg42850@alunos.uminho.pt,
id8123@alunos.uminho.pt,fas@estg.ipp.pt,analide@di.uminho.pt

**Abstract.** The great social development of the last few decades has led more and more to free time becoming an essential aspect of daily life. As such, there is the need to maximize free time trying to enjoy it as much as possible and spending it in places with positive atmospheres that result in positive sentiments. In that vein, using Machine Learning models, this project aims to create a time series prediction model capable of predicting which sentiment a given place cause on the people attending it over the next few hours. The predictions take into account the weather, whether or not an event is happening in that place, and the history of sentiment in that place over the course of the previous year. The extensive results on dataset illustrate that Long Short-Term Memory model achieves the state-of-the-art results over all models. For example, in multivariate model, the accuracy performance is 80.51% when it is applied on the LinkNYC Kiosk dataset.

**Keywords:** Machine Learning · Smart Cities · Sentiment Analysis · Human Mobility

## 1 Introduction

Currently, the number of Smarty Cities (SCs) is increasing and these cities have as main objective to improve the lives of citizens using Internet of Things (IoT) devices [5]. Examples of which are the implementation of existing lighting control systems in cities or the management of the flow of electricity or traffic [4]. A very common problem in cities is to control the density people in urban areas, that is, human mobility in cities that are not always the same throughout the day and that can affect people's daily lives due to existence of a high population density [6, 10].

This density of people in the same geographic area, as well as the data they can generate from the interaction with digital services available in cities, allows us to make predictions at the most different levels. One of these levels is the emotion associated with a certain location at a given future time. If proven

effective, it may allow for further investigation, with the goal of helping people better choose how to spend their time.

The polarity of a sentiment is a subjective topic, but there are still aspects known to affect it [12]. One the most important ones being the weather. The relevancy of weather in people's sentiment is something that can used to predict how the environment at a given place feels. One other important aspect to be considered for the sentiment a place is the possibility of there being an event occurring at said place. Additionally, all the information present in a date can help predict sentiments In other words, the day of the week, the month or even the hour of the day are all aspects that can influence sentiments.

The rest of the paper is organized as follows. Section 2 contextualizes key concepts as SCs, Artificial Intelligence (AI) and surveys about Sentiments analyzes. In Section 3, we detail the case study experiment used, and it explains all the processes carried out during the exploration and pre-treatment of data, address the various models developed to predict the human sentiments on New York City center. In the end of this stage (Section 4), we discuss the results of the case study. Finally, we summarize the work done and note on future work that be implemented to improve our work.

## 2 State of The Art

In this first stage of the article, we contextualize the two key concepts, the concept of Smart Cities (SCs) and Internet of Things (IoT). We intend that with the realization of the contextualization of these two concepts it is possible to have better understand the purpose of our study and the various reasons that led us to carry it out. Then, some surveys related with area of human sentiment are presented.

### 2.1 Smart Cities and Internet of Things

One of the most important concepts to understand is the concept of smart cities. SCs use IoT devices such as smart sensors, cameras or traffic lights. All this to improve the quality of life of citizens, stimulate the local economy and raise development indicators. Thus, the main objective of these cities is to develop innovative responses to improve infrastructure, public services and much more [3].

Through Table 1, we can see some use cases of SCs and we can easily observe cases that are constantly present in our daily lives, such as smart parking or smart street lighting. That said, building a true Smart City (SC) can be incredibly complex, not only because of the numerous tasks and functions that a city can have, but also because of the huge financial hurdle. Furthermore, to be a true SC, cities need to have an integrated approach, where various projects are interconnected, and, above all, data and IoT platforms are brought together to get all the benefits that SCs make possible.

**Table 1.** Examples of Use Cases and Applications of Smart Cities

| Areas | Examples |
|---|---|
| Public Services | Citizen Services, Tourist Services, ... |
| Transportation | Smart Roads, Smart Parking, ... |
| Sustainability | Environment Monitoring, Smart Energy, ... |
| Public Safety | Smart Lighting, Emergency Response, ... |
| Infrastructure | Smart Buildings, Structure Health, ... |

### 2.2 Developed Studies

This exact field of Artificial Intelligence (AI) models predicting human sentiment associated with time and place is rather scarce in terms of past research work. However, there are a few works in area of human sentiment.

E. Asani et al. [2] proposed a context-aware recommender system to extracts the food preferences of individuals from their comments and suggests restaurants in accordance with these preferences. For this purpose, the semantic approach is used to cluster the name of foods extracted from users' comments and analyze their sentiments about them. Finally, nearby open restaurants are recommended based on their similarity to user preferences.

In its turn, B. AlBadani l et al. [1] presents a new method of sentiment analysis using deep learning architectures by combining the Universal Language Model Fine-Tuning (ULMFiT) with Support Vector Machine (SVM) to increase the detection efficiency and accuracy. The method introduces a new deep learning approach for Twitter sentiment analysis to detect the attitudes of people toward certain products based on their comments.

Gihan Weeraprameshwara and Vihanga Jayawickramas [13] establishes benchmarks with the goal of identifying the best model for Sinhala sentiment analysis. They test on Facebook posts a set configuration, other deep learning models catered for sentiment analysis. In this study they report that the 3 layer Bidirectional LSTM model achieves an F1 score of 84.58% for Sinhala sentiment analysis, surpassing the current state-of-the-art model; Capsule B, which only manages to get an F1 score of 82.04%. Further, since all the deep learning models show F1 scores above 75% they conclude that it is safe to claim that Facebook reactions are suitable to predict the sentiment of a text.

M. Mao et al. developed an group event recommendation engine [7], based on the social relations a group of users has, that recommends them inexperienced events. At it's core, its an recommendation algorithm with the purpose of making recommendations to persistent groups, according to their criteria, past events. Since events take place at predefined location and time it is important to identify the popularity much before the occurrence of the event.

That said, it was possible to observe with the aforementioned studies that the concepts of Smart Cities (SCs) are increasingly present in society, AI has applicability in several areas, and we saw how they contribute to forecasting people's sentiments.

## 3 Experimental Study Case

As we have seen before, our case study is about predicting sentiments through Machine Learning (ML) techniques. Thus, in this section we start by approaching the dataset that we chose to use for our forecast and the main reasons that led us to choose it. After that, we see the main conclusions that we obtained from the exploratory analysis to better understand the data we are using. Thus, we proceed with the main steps taken in the data preprocessing to leave our dataset "clean" to later apply our ML models. The project was implemented in Python via Google Colab framework.

### 3.1 Exploratory Data Analysis

To build our dataset we use a public service that provided relevant information for the development of this study. Firstly, we decided to choose a service provided by NYC OpenData API, more specifically the LinkNYC Kiosk Status. The Department of Information Technology and Telecommunications (DoITT) manages the technical operations with Socrata Open Data API, ensuring that technological capabilities are always evolving to better meet user needs. This API provides data about the LinkNYC kiosks in which various data such as location, and the status of the Link's WiFi, tablet, and phone of the person connected to the network are described. This dataset consists of 2161 rows and 29 columns.

311 Open API via NYC Open Data is another resource we used on our work [9]. Although, this dataset contains several attributes, we highlight the most useful for the context of this work: Created Date, Descriptor, Latitude and Longitude. Then, a simple Python code apply Valence Aware Dictionary and Sentiment Reasoner (VADER) Analysis. VADER is a lexicon and rule-based sentiment analysis tool that is specifically attuned to sentiments expressed in social media [11]. It also uses a combination of a sentiment lexicon is a list of lexical features (e.g., words) which are generally labeled according to their semantic orientation as either positive, neutral or negative. This math formula gives Descriptor attribute their weight in the calculation of the sentiment label for every entry in the data set.

The previous distinguishable sentiments were then given a label number in a way that created somewhat of a scale of excitement in order to facilitate handling the data as well as making it better for the model. The scale chosen was the following: -1 - Negative; 0 - Neutral; 1 - Positive. After calculating sentiment weight for Descriptor attribute, the column with these weights was joined on LinkNYC Kiosk Status dataset, giving the final sentiment value each corresponding entry (based on date index).

After we apply Correlation Matrix (CM) technique, attributes such as Sentiment, Generated On, Borough, Latitude and Longitude show an import correlation coefficients. The first attribute is an attribute that contains the date and time when a user entered the kiosk. The second attribute is where the kiosks are located. The last two, as the name implies, give us the exact location in terms of

latitude and longitude. We can view more detailed information about all dataset attributes in [8].

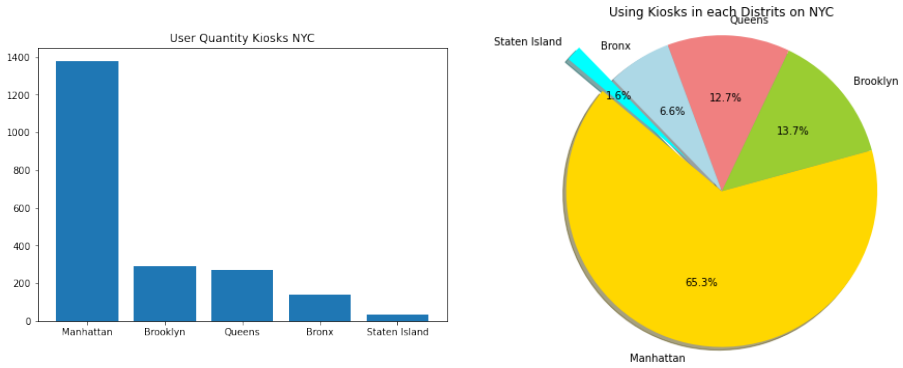


**Fig. 1.** Exploratory Analysis - User Quantity Kiosks NYC and Using Kiosks in each Districts on NYC.

When analyzing the Generated On attribute in detail, the most important interpretation we obtained was that this attribute had several old dates, and some of these dates even corresponded to dates from the last century. About the Borough attribute, it is important to mention that one of the conclusions we reached was that it had only 5 values, namely Manhattan, Brooklyn, Queens, Bronx and Staten Island. Thus, we can see through Fig. 1 some interesting data that we obtained in which it can be concluded that most users are from Manhattan (Manhattan has 65.3% of kiosk users).

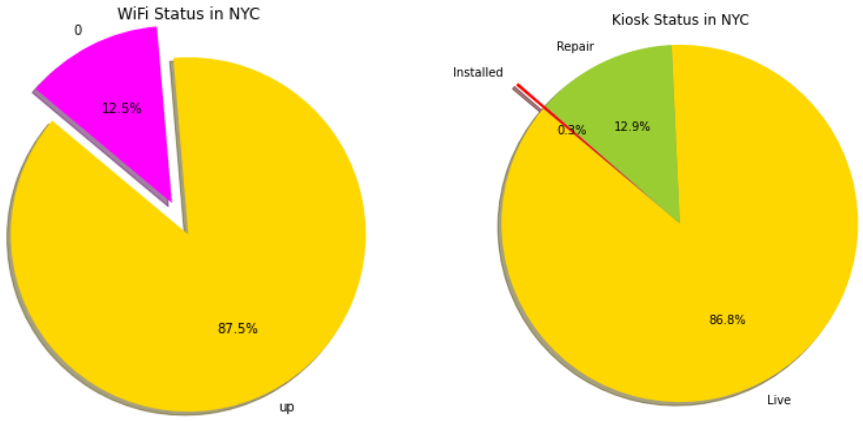


**Fig. 2.** Exploratory Analysis - Wifi Status in NYC and Kiosk Status in NYC.

To conclude our exploratory analysis, we also consider it interesting to observe the performance of the kiosks and WiFi. Having said that, we can observe in Fig. 2 the various states of both and we can easily conclude that both present good performances.

### 3.2 Data Preprocessing

As already mentioned, we started by carrying out an exploratory analysis where it was possible to conclude that the most important attributes for our study are the Generated On, Borough, Latitude and Longitude attributes. For this reason, one of the data preprocessing steps that was carried out was the elimination of the remaining columns.

Another thing that was possible to conclude when looking at the data in more detail, was that there were several null and "useless" data. One example of these "useless" data is the data present in the Generated On attribute with several dates that are too old. So, we proceeded to eliminate all lines with data below the beginning of the year 2017 and later we also proceed to the ordering of these dates. Regarding the data type of each attribute of the dataset, we also proceed to perform the conversion of the column data Generated On to the type of data that we consider most appropriate for our purpose.

To finish our data processing, in the Borough attribute that contained five distinct values, we used the fit_transform method of the scikit-learn library to improve the performance of our models. This method is used so that we can scale the training data and also learn the scaling parameters of these data and thus allow the built models to learn the mean and variance of the characteristics of the training set. These learned parameters are then used later to scale our test data. At the end of the Data Preprocessing, our dataset has 2123 rows and 4 columns and in the next step we address the various models that we implemented to achieve our goal.

### 3.3 Developed Models

The models of our project are based on two architectures: Long Short-Term Memory (LSTM) and Convolutional LSTM (ConvLSTM). The main difference between ConvLSTM and LSTM is the number of input dimensions, because the LSTM input data is one-dimensional and ConvLSTM is designed for 3D data as input data. However, other considerations about implementation of these models are explained on this section.

**LSTM** Relative to the LSTM, were implemented three models. The first model was the classic LSTM with a single layer. The second model was an extension of the classic LSTM model, which is the LSTM Bidirectional. In the LSTM Bidirectional, instead of training just one model, are introduced two models. The first model learns the sequence from the input and the second model learns the inverse of that sequence. For this, it is necessary to have a mechanism that

can combine both models, and this step is called the merge step, which can be done with addition, multiplication, average or concatenation functions. The third and last model implemented was another extension of the classic LSTM model, which is the Stacked LSTM. The classic LSTM consists of a single hidden LSTM layer followed by a feed-forward output layer, whereas Stacked LSTM has multiple hidden LSTM layers, where each layer contains multiple memory cells.

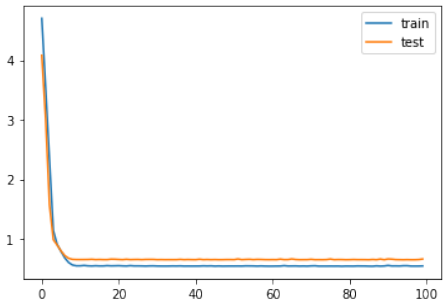


**Fig. 3.** Results from LSTM.

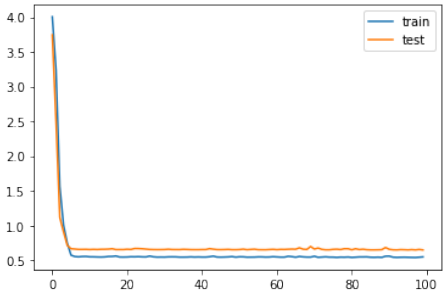


**Fig. 4.** Results from Stacked LSTM.

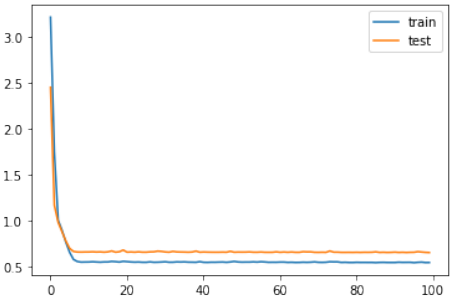


**Fig. 5.** Results from LSTM Bi-direction.

In these three graphs are exposed the behaviours of the models by loss function Mean Squared Error (MSE) and to optimize the models we use Adam optimizer. As we can see in the three models of LSTM, first we have a quickly reduce of the loss and then they stabilize. In a fast observation we can see that Stacked LSTM is slightly better than LSTM Bidirectional and LSTM single layer. Still, the computational cost of training of the models is higher comparing with LSTM single layer.

**ConvLSTM** Relative to the ConvLSTM, were implemented two models. The first model was the CNN-LSTM, which is an integration of a Convolutional Neuronal Network (CNN) with an LSTM. In the first phase of this model, the CNN part of the model processes the data and int the second phase the one-dimensional result of the first phase is fed into an Long Short-Term Memory (LSTM) model. The second model was ConvLSTM2D, which is similar to LSTM but the input transformations and the recurring transformations are convolutional.

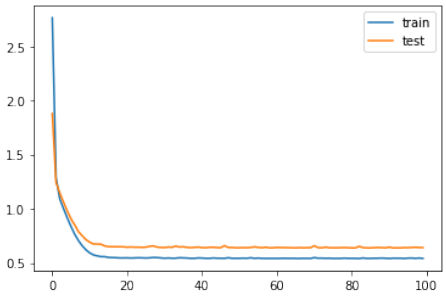


**Fig. 6.** Results from CNN-LSTM.

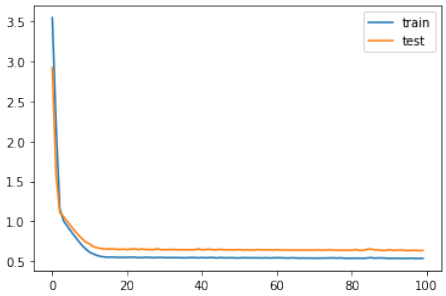


**Fig. 7.** Results from ConvLSTM2D.

In these two graphs we compare the behaviors of the CNN-LSTM model versus ConvLSTM2D. Such as the three others models that we saw of LSTM, first we have a quickly reduce of the loss and then they stabilize. We also used the loss function Mean Squared Error (MSE) and the Adam optimizer as well. The training result of these models is better than the training result of the three models of LSTM, but the computational cost to training the models is higher.

## 4 Results and Discussion

In this analysis we can see that we don't have big differences between the models, but of course we can see a little difference in the Convolutional LSTM (ConvLSTM) model that have a better performance.

If we have a huge and flexibility feature set to don't look at the computational cost of the training we will choose the ConvLSTM2D with a 0.632 value of loss. On the other hand, if we don't have this feature set and we need the best model with the lower computational cost we will choose Long Short-Term Memory (LSTM) single layer with a 0.650 value of loss, because in the set of the models with the lower computational cost, this is the model with the lower computational cost and just have a little bit higher value of loss. However, if we want a balance between the computational cost and the value of loss we will choose the CNN-LSTM with a 0.639 value of loss, because have a lower computational cost than ConvLSTM2D and still have one of the bests values of loss.

In the end, we made a comparison of the Multivariate models. We compare Multivariate LSTM model with the Multivariate CNN-LSTM model and how can we see in the Fig.8 the loss function in LSTM model is much better than CNN-LSTM model. In addition, the computational cost of LSTM is extremely less than Convolutional Neural Network (CNN) models.

| Models | Val_Loss |
|---|---|
| LSTM | 0.650 |
| CONVLSTM2D | 0.632 |
| STACKED LSTM | 0.648 |
| LSTM BIDIRECTIONAL | 0.649 |
| CNN-LSTM | 0.639 |

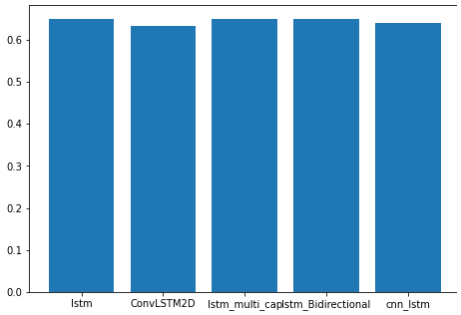


**Fig. 8.** Comparison of the results obtained.

## 5 Conclusion

This article approaches the modelling and prediction of sentiment associated with time an place. Starting with finding and processing data relevant to the problem in question, this step proved incredibly important, since finding the appropriate data for this particularly subjective topic was as difficult if not more than the making of the prediction model. The second phase of the work consisted of the model implementation that through training and some optimization ended the practical work with the conclusion of the final Long Short-Term Memory (LSTM) prediction model.

However, a critical point about data size should be mentioned. Unfortunately, our dataset is extremely small and did not have enough information to train our model. For example, we did not have the internet connection time of each cell phone and time to terminate the network connection in each kiosk. Due to these inconveniences we try to solve this problem with the cross validation method with shuffel = true, and we try to compare the models in their behavior of error loss and loss function.

In future work, we intend to predict the future sentiment values into a classification problem. In this classification problem a possible type of model to use would be decision trees. The short tests made in the data processing phase showed that a simple, unoptimized decision tree was able to reach around 75% accuracy, achieving better results than the ones obtained through the LSTM model.

## Acknowledgments

This work has been supported by FCT - Fundacao para a Ciencia e Tecnologia within the R&D Units Project Scope: UIDB/00319/2020. It has also been supported by national funds through FCT – Fundação para a Ciência e Tecnologia through project UIDB/04728/2020.

## References

1. AlBadani, B., Shi, R., Dong, J.: A Novel Machine Learning Approach for Sentiment Analysis on Twitter Incorporating the Universal Language Model Fine-Tuning and SVM. Applied System Innovation **5**(1), 13 (jan 2022). https://doi.org/10.3390/asi5010013, https://www.mdpi.com/2571-5577/5/1/13/htm https://www.mdpi.com/2571-5577/5/1/13
2. Asani, E., Vahdat-Nejad, H., Sadri, J.: Restaurant recommender system based on sentiment analysis. Machine Learning with Applications **6**, 100114 (dec 2021). https://doi.org/10.1016/j.mlwa.2021.100114
3. Balboni, C., Bryan, G., Morten, M., Siddiqi, B.: Transportation, Gentrification, and Urban Mobility: The Inequality Effects of Place-Based Policies. Preliminary Draft p. 3 (2020)
4. Garver, J.B.: National geographic society. American Cartographer **14**(3), 237–238 (1987). https://doi.org/10.1559/152304087783875921
5. Hultin, J.: Smart cities: acceleration, technology, cases and evolutions in the smart city, https://www.i-scoop.eu/internet-of-things-iot/smart-cities-smart-city/
6. Joshi, S., Saxena, S., Godbole, T., Shreya: Developing Smart Cities: An Integrated Framework. In: Procedia Computer Science. vol. 93, pp. 902–909. Elsevier (jan 2016). https://doi.org/10.1016/j.procs.2016.07.258
7. Liao, G., Huang, X., Mao, M., Wan, C., Liu, X., Liu, D.: Group Event Recommendation in Event-Based Social Networks Considering Unexperienced Events. IEEE Access **7**, 96650–96671 (2019). https://doi.org/10.1109/ACCESS.2019.2929247
8. NYC Open Data: LinkNYC Kiosk Status (2019), https://data.cityofnewyork.us/City-Government/LinkNYC-Kiosk-Status/n6c5-95xh
9. NYC Open Data: 311 Service Requests from 2010 to Present (2021), https://data.cityofnewyork.us/Social-Services/311-Service-Requests-from-2010-to-Present/erm2-nwe9
10. Rosa, L., Silva, F., Analide, C.: WalkingStreet: Understanding Human Mobility Phenomena Through a Mobile Application. In: International Conference on Intelligent Data Engineering and Automated Learning, pp. 599–610. Springer, Cham (nov 2021). https://doi.org/10.1007/978-3-030-91608-4_58,
11. Surender Dawra, Sumit Gumber: Sentiment Analysis using VADER (2021), https://www.geeksforgeeks.org/python-sentiment-analysis-using-vader/
12. Taj, S., Shaikh, B.B., Fatemah Meghji, A.: Sentiment analysis of news articles: A lexicon based approach. In: 2019 2nd International Conference on Computing, Mathematics and Engineering Technologies, iCoMET 2019 (2019). https://doi.org/10.1109/ICOMET.2019.8673428
13. Weeraprameshwara, G., Jayawickrama, V., de Silva, N., Wijeratne, Y.: Sentiment Analysis with Deep Learning Models: A Comparative Study on a Decade of Sinhala Language Facebook Data. arXiv preprint arXiv:2201.03941 (jan 2022). https://doi.org/10.48550/arxiv.2201.03941, https://arxiv.org/abs/2201.03941v2