Data Science Methods and Machine Learning Algorithm Implementations for Customized Pratical Usage

Erol Mrzic, Tarik Zaimovic

University of Sarajevo, School of Economics and Business, Department of Information Systems Development, Trg oslobođenja - Alija Izetbegović 1, Sarajevo, Bosnia and Herzegovina

Abstract **– Due to the unprecedented rise of data content over the last decade, an opportunity for data-based personalization and analysis has become a norm in the modern world. By implementing Machine Learning algorithms and Data Science methods no industry remained unchanged. This paper applied these methods and algorithms in personal, practical examples in order to see their benefits in our day-to-day lives. For the purpose of this case study, we analyzed three cases: a personal movie recommender, messages analysis and real estate trends and predictions on the local market. For this research we used global and personal data, and applied a suitable machine learning model. The purpose of this paper is to establish how one individual, and in what measure, with the use of these new technological tools, can ease his decision making process and manage a more tailored lifestyle.**

Keywords **– Data Science, Machine Learning, personal use, message analysis, movie recommender, price prediction.**

1. Introduction

In the recent decade we have witnessed the rise of social networks and growth of Internet-based media platforms resulting in vast amounts of data.

All that data has functioned as a fuel for data exploration and data-based predictive models, giving rise to better and more complex models as well as new and innovative use cases [1], [2]. Such cases of Machine Learning algorithms together with data science and analysis methods have initiated changes across many industrial aspects, resulting in developing novel business approaches in transportation [3], healthcare [4], education [5], production and political campaigns, referendums and governments [1], [6].

This techno-analytical golden age is evolving at an exponential pace given that the world we live in now is more interconnected and multi-faced than has ever been before and everything we use is a data generator. This presented a new way to transform entire systems, across (and within) countries, institutions and society. Scaling down from society to individual, this paper is trying to establish whether we could use these methods and algorithms, these ground-breaking tools that large institutions and companies use for much more important purposes, to improve seemingly unimportant personal tasks, and make our lives more productive and effective.

There has been extensive research and projects using Machine Learning across all fields, including the topics that we are discussing in our paper such as: movie recommendations for users that have connection and similar ratings [7], social media message analysis as in alert messages during crises [8] and sentiment analysis [9] and apartment price analysis and prediction for certain areas [10], [11]. Previous research has taken a rather wide approach, while we are trying to use these methods for personal use.

In this paper, we will give an overview of these complex methods and algorithms for non-scientific use, for improving the quality of our private decisions and management of our time and resources. The goal is to develop a base and structure to create a stand-alone, standardized Machine Learning algorithms that could use individuals' own data and create a customized experience only for this individual, therefore avoiding the bias and marketing

DOI: 10.18421/TEM93-44
https://doi.org/10.18421/TEM93-44

Corresponding author: Erol Mrzic,
University of Sarajevo, Department of Information Systems Development, Sarajevo, Bosnia and Herzegovina
Email: erol.mrzic@student.efsa.unsa.ba

Received: 29 April 2020.
Revised: 14 July 2020.
Accepted: 21 July 2020.
Published: 28 August 2020.

of businesses and corporations. Advantages of this approach would be not having marketing influences on our choices and our results, no business strategies or favoritism as well as political or society bias.

Limitations would be the heavy reliability on personal data, which, in today's world, is very valuable and must be dealt with very carefully. These algorithms were made for the individual by the individual and therefore avoided the controversy of 3rd party data gathering. This approach would take time to create a valuable and functional data sets that would, in the end, give valuable insights and answers.

In our case study, we focus on a number of topics, including:

A personal recommender algorithm for movies based on private viewing habits and ratings of a single user. Given that now most of our interaction is text-based, we will make an analysis of personal texting habits giving us an insight of contact priorities and better management for important people in our lives. Finally, we take an example of an important and complex decision such as buying or selling an apartment and we try to make it easier by analysing apartment prices and trends in Sarajevo real estate market based on web collected data.

Our methodology involves collection of global and personal data, its pre-processing followed up with a suitable machine learning model.

2. Methodology

In this section, we will introduce our process of data gathering and extraction, some methods which are widely used, then describe the pre-processing for every use case. We will continue to explain our features and feature selection and offer our decision for Machine Learning algorithm. Our methodology contains the following steps: Data Extraction, Data Pre-processing, Data Integration and Transformation, Feature Selection and ML algorithm

2.1. Data Extraction

Gathering data from various sources is defined as data extraction. Extended pre-processing, transformation and integration of data is required in order to further analyse it.

2.1.1. Use case #1: Movie recommender system

In this use case we have used an online kaggle.com dataset containing over 45.000 movie metadata, and over 270.000 user reviews to be the base for our future movie selection. And for our recommendation base we have used a personal 12 month dataset which consisted of 250 movies watched in the period of (May, 2018 – May, 2019), on which we then applied our recommender algorithm.

2.1.2. Use case #2: Message analysis

Here we have extracted data from private social media accounts, including Viber, WhatsApp and Facebook messenger. Total amount of data exceeded 200.000 messages. Data was extracted using official app extraction methods (WhatsApp, and Facebook) and third-party app designed for data extraction of specific app (Viber). Data collected was in html format, and further processing was needed.

2.1.3. Use case #3: Apartment analysis

This use case had us scraping web content for data. Making a custom web scraper script combined with a third party app (Parse Hub) for complicated web page maneuvering we selected a national web page (Olx.ba) for real estate information and applied our system. The final result gave us over 80 web pages content with 40 apartment links per page, from which we gained useful data for over than 2500 apartments.

2.2. Data Pre-processing

Data pre-processing is a method of transforming raw data into an understandable format, which includes resolving various issues, given that the extracted data is never perfect. It is very often the case to have missing data, inconsistent values, duplicates and outliers caused by flaws in the data collection process, human error, or simply the nature of the raw data itself. Data pre-processing is the first step in organizing data for further processing.

2.2.1. Use case #1: Movie recommender system

In this case we included removing unnecessary features and adequate formatting on other columns such as date-time formatting and rounding up values of ratings and votes.

2.2.2. Use case #2: Message analysis

This case requested more complicated approach. Our data collected was in html format for every contact in our messages. We used a custom script with Beautiful Soup library (bs4) for scraping every html document. Once done, we had a data frame for contact with selected features (Date, Time, Sender, and Content) info. A pipeline was made and applied to every single contact html document, which gave us adequate datasets with more suitable formatting (.csv). Also, a special emoji library was imported for extracting emoji-s from the content of the messages.

2.2.3. Use case #3: Apartment analysis

Pre-processing in case #3 consisted of extracting some of the data from the URLs in order for it to be useful (latitude and longitude) and making feature columns out of all raw data we scraped, again using Beautiful Soup library (bs4). Some of the features required type and formatting adjustments, from numerical/string to categorical and float to integer type adjustments. All missing data was given a special category to analyze the magnitude of the missing data, and then subsequently was removed if most of the items data was missing. Data was then formed into dataset with appropriate column names and exported as a single file.

2.3. Data Integration and Transformation

Data Integration stands for combining data from several separate sources, which is done using various methods, libraries and technologies to provide a unified view of our data. Data Transformation involves methods to transform or consolidate data into forms suitable for further data mining and analysis.

2.3.1. Use case #1: Movie recommender system

We grouped our personal dataset by custom users, in this case it was with family, alone or with girlfriend, was it on workdays or weekends. We reshaped our datasets so that our movies were now index rows instead of columns, so we could apply similarity algorithms on the movies that users watched instead of the usual finding similar users approaches. Once the movie ID number was our index rows we had to modify our columns to best describe the movie in question. Python's open source library Pandas and its function *get_dummies* was used to sort keywords and genres in this situation.

2.3.2. Use case #2: Message analysis

Data integration in case #2 was done by grouping all contact messages documents into one single dataset that represented its social media app. All missing values were removed from the dataset and column names were modified so they were the same across all datasets. Finally, all datasets were concatenated and exported as a one single file (allMsgs.csv).

2.3.3. Use case #3: Apartment analysis

Our initial dataset contains 2196 apartment entries, each described with 9 features. We first grouped the apartments per location (municipalities) and analyzed prices within each group (Fig. 1.). As seen on Fig.1.a, there are several outliers which could interfere in future analysis and model application. These outliers were removed from the dataset, resulting in a more adequate price distribution (Fig.1.b). In the next step, we studied the correlation between the features and the target price and removed features with insufficient percentile, i.e. features that do not impact the target price and therefore would not contribute to the model accuracy.

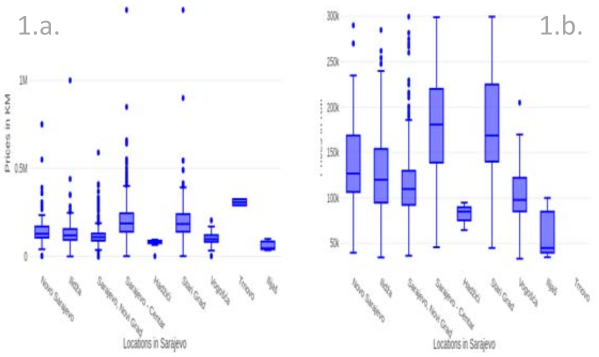


Figure 1. Price distribution per municipality in Sarajevo; 1.a: Price distribution including outliers; 1.b: Price distribution after removing outliers

2.4. Feature Selection and ML Algorithm

Feature selection, also known as attribute selection or subset selection, involves selecting specific data points and discarding redundant or irrelevant data, maximizing efficiency and making our Machine Learning model more precise in its predictions.

2.4.1. Use case #1: Movie recommender system

A few features were removed because there was insufficient data and would only hurt the algorithm, such were the budget and the revenue columns. Hopefully our taste was not based merely on those exact features.

We chose cosine similarity algorithm, which was used to predict similar items in previous papers [12], but this time it was modeled on movies and not on similar users [7].

Cosine similarity measures the similarity of two documents, ranging from 0 (no similarity) to 1 (identical documents), irrespective of their size. We have sorted our movie data in different datasets, based on the day and our company when the movie was watched.

2.4.2. Use case #2: Message analysis

This case again requested more complex approach. For Feature selection we already had a small number of columns to select from. Our data consisted of Date and time of the message, Sender as in the name of the one who sent the message, and Content as in the content of the message. We added a Label column so we could distinguish groups in those contacts such as family, friends or girlfriend, and Messages length column for the number of characters in the content of the message. Also, during specific analysis, such as emoji analysis, data was divided into sent and received categories, which was accomplished by grouping the messages by the Label column.

Data was then subjected to further processing before the NMF topic modeling method was applied which was selected among other methods [13]. Non-negative Matrix Factorization (NMF) is a method that is widely used when analyzing high-dimensional data and has proven to be successful in exploiting the similarity between users' interactions and preferences to give recommendations. [14].

Our NMF model required use of documents, as in the messages we will choose to analyze for creating topics based on their token content. We established that the messages are too short to find clear patterns in the usage of words, and also the texting habit of sending messages in several lines. We solved this issue by combining the messages into groups of maximum 5 messages based on the time they were sent and the sender. After we made our message groups we dealt with removing punctuation, duplicate letters, conversion to lower case and finally tokenisation (slicing documents into words by using *nltk* package and its word tokenize function) and removing stop words. On that dataset, TfidfVectorizer was fitted and NMF method was applied which got us our 30 items with 5 word each that was representing the topics in our messages. A handmade classification was then made to label each subsequent topic.

2.4.3. Use case #3: Apartment analysis

Our cleaned dataset included 1980 apartments and 9 feature columns. After using Python's Pandas *get dummies* method on some columns, we ended up with 994 feature columns. We decided on Random Forrest regressor in this use case because of its great performances so far [10], [11].

Random Forrest is an addition to the decision tree algorithm and comprises of a random collection of multiple decision trees across our data. A decision tree is a tree-like model of decision structure in which each internal node represents a condition of an attribute from which the tree will split into branches representing the outcome of the condition, and each leaf node that can't split anymore represents a decision or a class label.

We removed the prices from the dataset and applied the model on the selected features. 80% of the data has been used for training and 20% for testing, and the number of estimators in Random Forrest was 35.

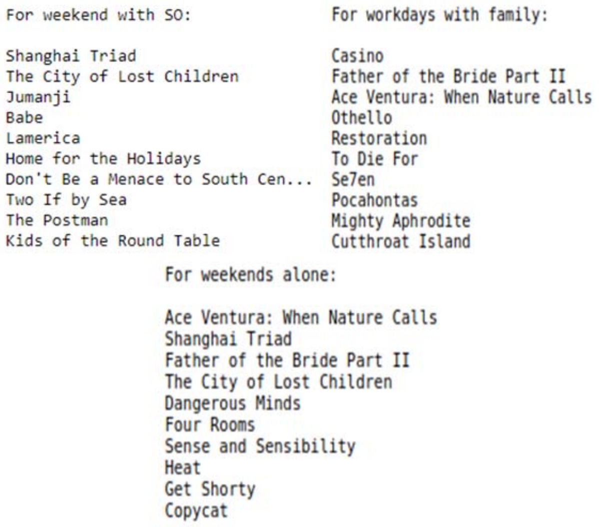
We also used k-fold cross-validation for our results so that we could use all of our data for testing and training. In k-fold cross validation, the data is divided into k number of subsets. Our method now is repeated that k number of times but each time with a different k subset performing as the test subset while the other k-1 subsets are merged together as the training test. The error estimation will be averaged over all k trials to get the total effectiveness of our model.

We used k-means unsupervised learning method on our error data to establish most likely clusters of data with substantial error margin. What this algorithm does is that it groups data points that are similar and then allocates those data points to the nearest cluster, k representing the fixed number of clusters in the dataset. Further use can be found here [15].

3. Results

3.1. Use case #1: Movie recommender system

From our dataset, movies with a personal rating higher than 3.0, based on our criteria (highest rating 5.0, lowest rating 1.0), were then pushed into our cosine similarity algorithm. A custom list of similar movies based on our company and specific days with common keywords, similar average popular rating, and genre was then returned from our database of 45.000 movies.



```
For weekend with SO:                    For workdays with family:

Shanghai Triad                          Casino
The City of Lost Children               Father of the Bride Part II
Jumanji                                 Ace Ventura: When Nature Calls
Babe                                    Othello
Lamerica                                Restoration
Home for the Holidays                   To Die For
Don't Be a Menace to South Cen...       Se7en
Two If by Sea                           Pocahontas
The Postman                             Mighty Aphrodite
Kids of the Round Table                 Cutthroat Island
                    For weekends alone:

                    Ace Ventura: When Nature Calls
                    Shanghai Triad
                    Father of the Bride Part II
                    The City of Lost Children
                    Dangerous Minds
                    Four Rooms
                    Sense and Sensibility
                    Heat
                    Get Shorty
                    Copycat
```

Figure 2. Results from our recommendation system based on day and company.

Results are now subjective (result sample shown in Fig. 2.), based on: do we really like the movies recommended, but the algorithm did find similar movies based on small datasets to work with. Our personal data was 250 movies all together, given that we divided our data into categories based on day and company when the movie was watched, our datasets were very small in comparison to the movie data we were finding similarities to (45.000 movies).

3.2. Use case #2: Message analysis

Our analysis showed our activity over 1-year period and we added the scale of sent and received messages. As shown in Fig. 3.

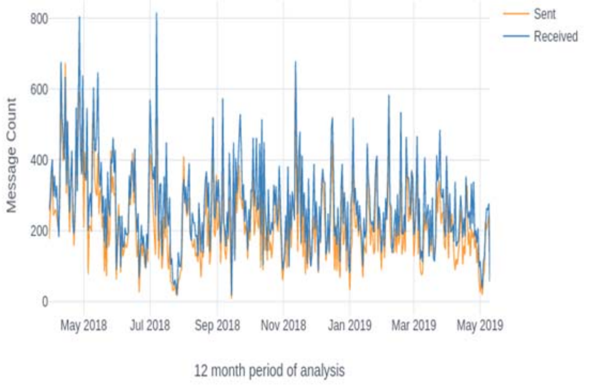


Figure 3. Graph showing full data distribution over the period of 12 months

We see that the number of received messages is higher than the number of sent messages, but the overall count of messages is quite high. The conclusion is that the subject relies on texting quite much but is less active than his contacts. Detailed view of our hourly message rate is offered in Fig. 4. and Fig. 5.

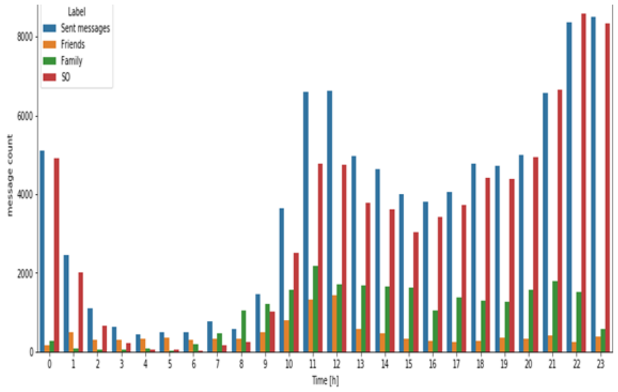


Figure 4. Showing hourly rate of messages with labelled data corresponding to our grouped contacts

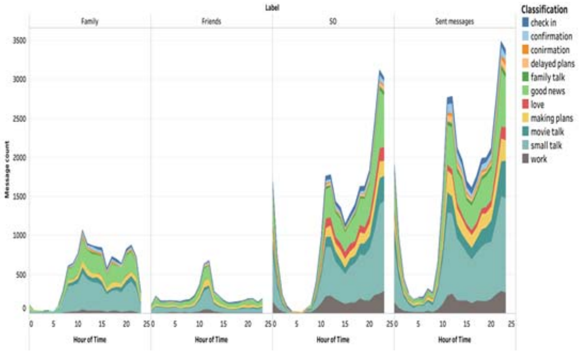


Figure 5. The volume of conversation topics per hour

From our hourly rate of messages analysis, we can conclude that the group labelled *Friends* is the least active group, but spikes during noon, as do all other messages. Conversation with the *Significant Other* is present during the entire day and increasing towards late hours. *Family* labelled conversations are usually after work hours until bed time. We can also see that our *Sent messages* peak occurs in the 10-12 both PM and AM time of day.

Overall view of our conversation topics classification in general, as well as for each group, can be seen in Fig. 6.

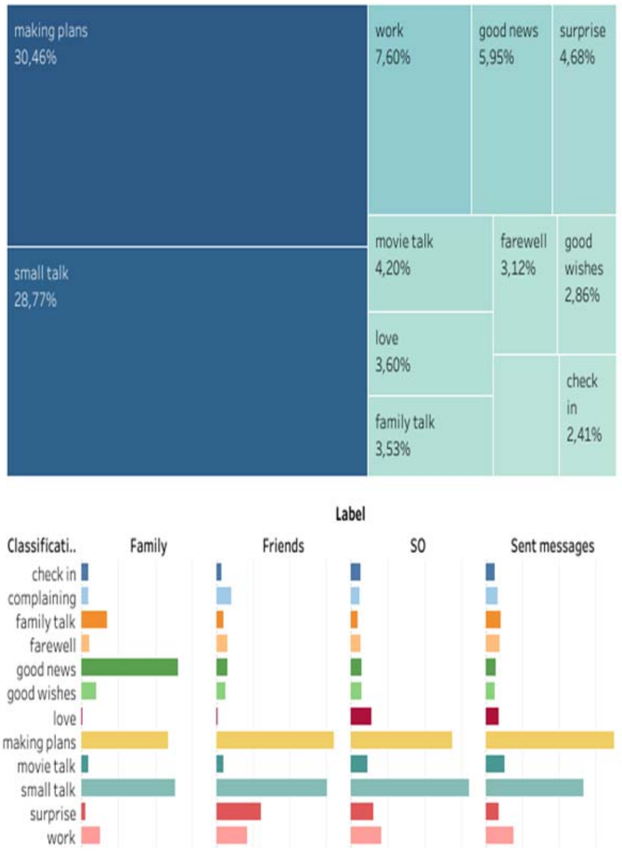


Figure 6. Distribution of topics across all messages in labelled groups

In our message topics analysis, we could establish that nearly 60% (59.23%) of our conversation data falls under the classification of ‘making plans’ and ‘small talk’. Those being among the most popular topics across all contact groups with small margin between them, except for the *Sent Messages* where ‘making plans’ topic and the *Family* messages where ‘good news’ topic have the highest percentage.

3.3. Use case #3: Apartment Analysis

Our model delivered a Cross-Validation result of 79% accuracy on price prediction (as seen in Fig. 7.), with an average percentage error in the range of 2-5%, but some high outliers (as seen in Fig. 8.).

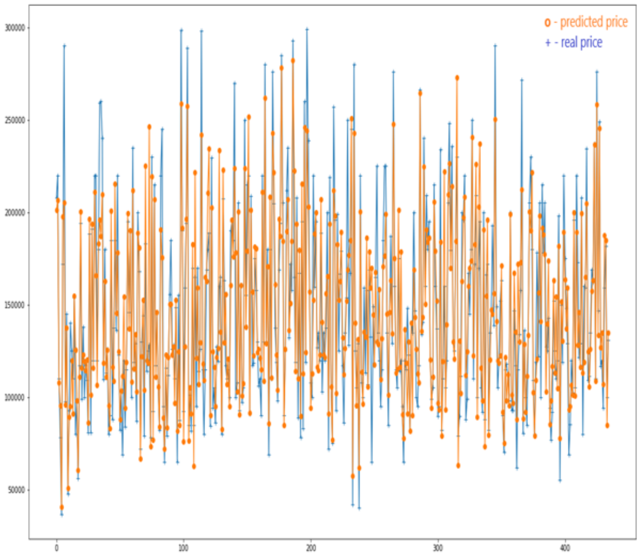


Figure 7. Random Forrest prediction model

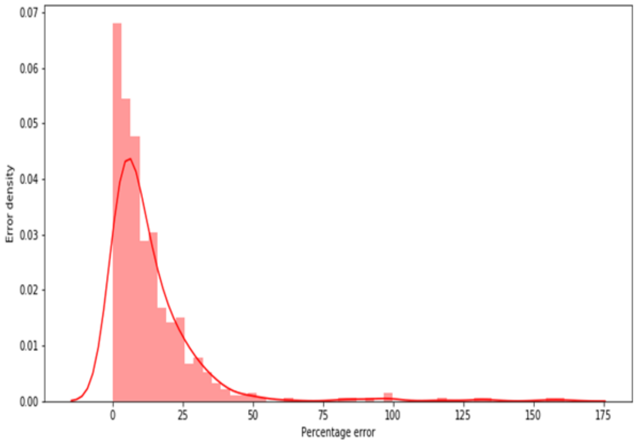


Figure 8. Error distribution for Apartment prediction

We went on to further analyse our error data by grouping apartments with over 30% error in price prediction and applying k-means method on it. As a result, we got clusters of apartments with high error percentage, and after looking into it we saw that the high error in prediction was mostly because of false description segment (e.g. 'Damaged construction', 'leaking') which we, avoiding text analysis in this occasion, did not include in our features.

4. Discussion

This paper consisted of 3 projects that served personal usage, and the aim was to conclude whether they could help manage our time as well as resource and help with our decision-making process.

In Use case #1 we made a movie recommender system that could help our viewing habits and ease the movie selection. We achieved the recommender system efficiency that we wanted, movies are selected on the basis of our watch history and weekday activity. Also, having labelled our users for these activities, we can expect over time to develop a full profile and tailored recommendations for each of them. Advantages of this particular approach was that it was made exclusively on the habits of one person's rating and viewing pattern while taking into account the company with which the movie experience was shared. If this was done by a corporation it would be considered a controversial subject, again asking the question where is the line of the users' privacy the users. For a good reason, there are limits that these businesses and corporations are limited with their data gathering but if these algorithms are available for individual and individual only, made by them for them, we have a chance to use all the potential of these prediction methods.

In use case #2 we gained great insight through our message analysis in terms of usage and conversation topics distributed through social media apps. We can now tell our communication style and timeline, and our preferred topics over different groups. We could see our messaging habits during the day and decide to act on them. This could eventually help making automated messages to a specific group of people, and easier communication overall. Additionally, it can be used as a base for methods to come or in combination with already existing automation algorithms and methods, such as real time translation typing or voice typing.

Use case #3 gave us a predicting model for apartment prices which we can use to establish if the apartment had a fair value, as whether the price was fair or not. If we came across an apartment with the price much lower than our predicted value, it could indicate a possible bargain and a good deal. We could then focus on specific areas as we had a visual geographical look at our data. Again, a highly personal and customized approach here would make the whole experience of finding a new apartment for rent or purchase much less tedious and less time consuming. Providing both a good overview of the price patterns and the ability to view the bigger picture of the real estate market so we can make a better decision in the end.

5. Conclusion

Our analysis showed how by rather simple use of Machine Learning and Data Science in regard to mundane problems with different degrees of complexity can help us make decisions or gain insight in our personal life. It can ease our research and simplify our selection, without any major setbacks. Each model can be further developed and more features can be added that would increase accuracy in our prediction.

For use case #1 personal dataset will have to be bigger to further test our algorithm. And adding features like movie budget, box office revenue, and cast and crew will certainly improve the prediction results on this model. It will be able to find more patterns in our data and hence recommend more similar movies in this customized personal model. Playlist for future viewings can be made depending on a day of the week and company included. Highly customized movie video store that would change on your every next viewing and deliver more optimal results. Further advances can be made for the experience itself, given that our general online viewing experience is stretched across multiple platforms (Amazon Prime, Netflix, etc.). By linking our own available platforms together with our recommender algorithm, we would have our own highly customized personal browser throughout all of these platforms and a list of recommendations from all the available viewing options.

Use case #2 can be tracked for a longer period than 1 year and data can be labelled to show which social media has more priority. With more data we could further develop sentiment analysis and the message topic analysis can be further dissected into smaller and more focused groups, such as friends, close friends, university friends, work friends etc. Once we reach this point, we would have a scalable option to choose from seeing a more general overview of our messaging habits or a more detailed view throughout all groups, selected groups, time periods, general topics or highly specific topics.

Use case #3 can be improved by adding features like proximity to train stations, nightclubs, parks, favourite restaurants and university or work distance. Adding a sentiment analysis on the apartment description section would make a valuable feature as well, where we could try and filter out words that would describe the condition of the apartment in a positive or negative manner (needs restructuring, fully equipped, no furniture included etc.).

Ongoing process of digitalization will necessitate even further industrialization and monetization of ML applications across variety of services and areas.

References

[1]. Dey, P., Kothari, P. K., & Nath, S. (2019, January). The social network effect on surprise in elections. In *Proceedings of the ACM India Joint International Conference on Data Science and Management of Data* (pp. 1-9).

[2]. Chen, J., & Li, M. (2019). Chained predictions of flight delay using machine learning. In *AIAA Scitech 2019 Forum* (p. 1661).

[3]. Jacob, C., Abdulhai, B., Hadayeghi, A., & Malone, B. J. (2006, September). Highway work zone dynamic traffic control using machine learning. In *2006 IEEE Intelligent Transportation Systems Conference* (pp. 267-272). IEEE.

[4]. Pollettini, J. T., Panico, S. R., Daneluzzi, J. C., Tinós, R., Baranauskas, J. A., & Macedo, A. A. (2012). Using machine learning classifiers to assist healthcare-related decisions: classification of electronic patient records. *Journal of medical systems*, *36*(6), 3861-3874.

[5]. Sun, J. M., Pei, X. S., & Zhou, S. S. (2008, July). Facial emotion recognition in modern distant education system using SVM. In *2008 International Conference on Machine Learning and Cybernetics* (Vol. 6, pp. 3545-3548). IEEE.

[6]. Patil, A. P., Doshi, D., Dalsaniya, D., & Rashmi, B. S. (2017, September). Applying Machine Learning Techniques for Sentiment Analysis in the Case Study of Indian Politics. In *International Symposium on Signal Processing and Intelligent Recognition Systems* (pp. 351-358). Springer, Cham.

[7]. Perny, P., & Zucker, J. D. (2001). Preference-based search and machine learning for collaborative filtering: the "film-conseil" movie recommender system. *Information, Interaction, Intelligence*, *1*(1), 9-48.

[8]. Brynielsson, J., Johansson, F., Jonsson, C., & Westling, A. (2014). Emotion classification of social media posts for estimating people's reactions to communicated alert messages during crises. Security Informatics, 3(1), 7.

[9]. Andriotis, P., Takasu, A., & Tryfonas, T. (2014, January). Smartphone message sentiment analysis. In *IFIP International Conference on Digital Forensics* (pp. 253-265). Springer, Berlin, Heidelberg.

[10]. Čeh, M., Kilibarda, M., Lisec, A., & Bajat, B. (2018). Estimating the performance of random forest versus multiple regression for predicting prices of the apartments. *ISPRS international journal of geo-information*, *7*(5), 168.

[11]. Pow, N., Janulewicz, E., & Liu, L. (2014). Applied Machine Learning Project 4 Prediction of real estate property prices in Montréal. *Course project, COMP-598, Fall/2014, McGill University*.

[12]. Sarwar, B., Karypis, G., Konstan, J., & Riedl, J. (2001, April). Item-based collaborative filtering recommendation algorithms. In *Proceedings of the 10th international conference on World Wide Web* (pp. 285-295).

[13]. Lee, M., Wang, W., & Yu, H. (2006). Exploring supervised and unsupervised methods to detect topics in biomedical text. *BMC bioinformatics*, *7*(1), 140.

[14]. Lee, D. D., & Seung, H. S. (2001). Algorithms for non-negative matrix factorization. In *Advances in neural information processing systems* (pp. 556-562).

[15]. Hartigan, J. A., & Wong, M. A. (1979). Algorithm AS 136: A k-means clustering algorithm. *Journal of the royal statistical society. series c (applied statistics)*, *28*(1), 100-108.