Enhancing Movie Recommenders by means of KNN-based Algorithms

Kamil Rojek[1], Rafał Ochorok[1] and Maciej Wiencis[1]

[1] *Faculty of Applied Mathematics, Silesian University of Technology, Kaszubska 23, 44-100 Gliwice, Poland*

Abstract

The project concerns a system recommending films using the KNN algorithm. The program in order to find movies, is based on the history of viewed items. Movie Recommender initially chooses the best ones from the history of films, in order to finally give the proposals best suited to the user's preferences. The model works with data from *IMDB* [?] data set downloaded from datasets.imdbws.com.

Keywords

Movies, Prediction model, Classification algorithms, Personalized movie prediction, Python

1. Introduction

Currently, the most dynamically developing IT tools are methods of artificial intelligence [1, 2]. Algorithms supporting decision-making or supporting inference based on fuzzy sets [3, 4] find a number of applications, among others, in the detection of anomalies on roads [5] or in the control of intelligent home management systems [6, 7]. At this point, one cannot fail to mention a wide class of heuristic algorithms based on the observation of animal behavior [8, 9, 10], which are widely used. The energy reduction applications [11, 12, 13, 14, 15] are very important. The most common applications relate to the use of [16] neural networks in a wide variety of applications that affect almost every area of life [17, 18, 19, 20]. Very interesting applications concern the care of the elderly [21, 22, 23, 24]. Often, neural networks are used in various types of detection tasks for certain features [25, 26, 27, 28]. The use of neural networks also plays a very important role in machine learning [29, 30].

Due to digitization of our modern world prediction models are extremely crucial in these days. That's because they can optimize some of the user's processes, that would facilitate comfort of using a given app. Movies are extremely complex thanks to a lot of variables into which they can be divided. People struggle with choosing a movie to watch, because they not only might not be aware of their preferences, but also they may not have enough time to check and compare all data[31]. Whole problem could be solved by a program doing all the necessary calculations for you, basing on user's watch history and reviews.

2. Assumptions for algorithms

Each of the algorithms should be prepared to meet the following criteria:

1. Prepared according to the mathematical description of the algorithm;
2. Optimized for the performance on our data set;
3. Returns an array containing information about top k (number of predicted movies) movies, basing on top 3 movies from our watch-list.

3. Program description

The task of our project is to create a system recommending films using the KNN algorithm. The program in order to find movies, is based on the history of viewed items. Movie Recommender initially chooses the best ones from the history of films. Then each video goes through the algorithm so that the program finally gives the proposals best suited to the user's preferences. (Including watch history). In this particular cases, KNN uses three metrics: Taxi cab, Cosine distances and Euclidean.

4. KNN history

The origins of KNN can be traced to research conducted for the U.S armed forces. Evelyn Fix (1904-1965) was a mathematician and statistician who taught at Berkeley. Joseph Lawson Hodges Jr. (1922-2000) was a Berkeley statistician who worked with the 20 United States Air Forces (USAF) from 1944. Combining their brilliant minds, in 1951 they wrote a technical analysis report for the USAF. He introduced a discriminant analysis, nonparametric classification method. However, the newspaper was never officially published - most likely due to confidentiality in the aftermath of World War II.

SYSYEM 2022: 8th Scholar's Yearly Symposium of Technology, Engineering and Mathematics, Brunek, July 23, 2022

kamiroj@polsl.pl (K. Rojek); rafaoch606@polsl.pl (R. Ochorok); maciwie234@polsl.pl (M. Wiencis)

5. Euclidean metrics history

Euclidean distance is the distance in Euclidean space; both concepts are named after ancient Greek mathematician Euclid, whose Elements became a standard textbook in geometry for many centuries.Concepts of length and distance are widespread across cultures, can be dated to the earliest surviving "protoliterate" bureaucratic documents from Sumer in the fourth millennium BC (far before Euclid),and have been hypothesized to develop in children earlier than the related concepts of speed and time.But the notion of a distance, as a number defined from two points, does not actually appear in Euclid's Elements. Instead, Euclid approaches this concept implicitly, through the congruence of line segments, through the comparison of lengths of line segments, and through the concept of proportionality.

6. Taxi Cab metrics history

Taxicab Geometry is a non-Euclidean Geometry that measures distance on horizontal and vertical lines. According to Taxicab Geometry - History, the taxicab metric was first introduced by Hermann Minkowski (1864-1909) over 100 years ago; however, it did not get its name until 1952. Taxicab is unique in that it is only one axiom away from being a Euclidean metric. In Euclidean Geometry the minimum distance between two points is the shortest line segment between those two points. However, in Taxicab Geometry there can be multiple minimal distances or 'shortest paths' made up of line segments perpendicular or parallel to the x-axis. Taxicab Geometry - History suggests that modern research on taxicab did not occur until as recent as the 1980s. The measurement of distance using vertical and horizontal lines rather than diagonal lines has sparked questions about its applications and encouraged more research and exploration of this simple yet unique metric

7. Description of the program's operation

Initially, we started the project by preparing the data in such a way that we could then carry out calculations on them. For this purpose, we downloaded the IMDB database, which consisted of four files containing data on:

- Data on the movie itself.
- Ratings for individual videos.
- The cast of the movie.
- Personal data of people participating in the film.

In order to optimize the algorithm, we do not use the full names of the cast at this stage of operation.

The next step is to create a person's profile to keep a history of the videos watched along with the user's rating. Before starting the algorithm, the program selects three top movies according to the user's rating. Then, based on this data, it performs calculations to find the best matching items in our database.
The metric used in the KNN algorithm is the sum of the cosine, taxicab and euclidean distances,between the values of the film elements we compare, i.e. genres, writers, directors. We use previously created numerical values. Formulas used to determine distances between successive parameters looks like this:

- Cosine distance
$$1 - \frac{u \cdot v}{\|u\|_2 \|v\|_2}, \tag{1}$$
- Taxi cab distance
$$|u - v| \tag{2}$$
- Euclidean distance
$$\sqrt{\sum_{i=1}^{n} (u - v)^2} \tag{3}$$
where **u** i **v** are the arrays to be compared

The next step is to use the KNN algorithm, which will use the previously described metric, in order to find the k-nearest neighbors of a given movie. In the algorithm itself, we predict finding k neighbors. As the algorithm can receive a maximum of 3 user top videos, it will therefore return a top 3k of the proposed positions. For example, if we add movies to the history:

- Coffee & Kareem, rating: 8
- Das Cabinet des Dr. Caligari, rating: 9
- The Kid, rating: 8.5

Program would output:
Recommended Movies basing on: Coffee & Kareem

- The F word
- It's All Gone Pete Tong

Recommended Movies basing on: Das Cabinet des Dr. Caligari

- Psycho
- 6 donne per 1'assassino

Recommended Movies basing on: The Kid

- The Circus
- Modern Times

8. Algorithms

In this section we will present pseudocodes of the most important algorithms used by us.

```
Data: Input: Id of the first movie $movieId1$, Id of
      the second movie $movieId2$
Result: The lack of data

genresA = genre of $movieId1$
genresB = genre of $movieId2$
genreDistance = the cosine distance between two
  values.

directorA = genre of $movieId1$
directorB = genre of $movieId2$
directorDistance = the cosine distance between
  two values.

writerA = genre of $movieId1$
writerB = genre of $movieId2$
writerDistance = the cosine distance between two
  values.

return genreDistance + directorDistance +
  writerDistance
```

Algorithm 1: Cosine distance metric pseudocode

```
Data: Input: Id of the first movie $movieId1$, Id of
      the second movie $movieId2$
Result: The lack of data

genresA = genre of $movieId1$
genresB = genre of $movieId2$
genreDistance = the taxi cab distance between
  two.

directorA = genre of $movieId1$
directorB = genre of $movieId2$
directorDistance = the taxi cab distance between
  two values.

writerA = genre of $movieId1$
writerB = genre of $movieId2$
writerDistance = the taxi cab distance between
  two values.

return genreDistance + directorDistance +
  writerDistance
```

Algorithm 2: Taxicab metric pseudocode

```
Data: Input: Id of the first movie $movieId1$, Id of
      the second movie $movieId2$
Result: The lack of data

genresA = genre of $movieId1$
genresB = genre of $movieId2$
genreDistance = the euclidean distance between
  two values.

directorA = genre of $movieId1$
directorB = genre of $movieId2$
directorDistance = the euclidean distance between
  two values.

writerA = genre of $movieId1$
writerB = genre of $movieId2$
writerDistance = the euclidean distance between
  two values.

return genreDistance + directorDistance +
  writerDistance
```

Algorithm 3: Euclidean metric pseudocode

```
Data: Input: The name of the movie $name$,
      Amount $k$, User Name $user$
Result: Featured Videos

newMovie = movie name
distances=[]
neighbors = []

for movie in $movies$ do
    if movie not in history then
        Add distances to the distances array using
          the 'Similarities' metric between the
          given movie and the rest of the movies
          in the database.
    end
end
distances.sort()
for $x$ in $k$ do
    Add to $neighbors$ calculated distances.
end
for $neighbor$ in $neighbors$ do
    View featured video data $neighbor$
end
```

Algorithm 4: An algorithm that returns Recommended Videos based on user preferences.

```
Data: Input: movie's name,
k - number of films searched
Result: Prediction: k - movies' name

movie = movie information database row
neighbors = KNN algorithm using the taxi metric,
given k - amount of movies to be found

for $neighbor$ in $neighbors$ do
    neighbors = KNN algorithm using the Cosine
      distance metric
end

for $neighbor$ in $neighbors$ do
    neighbors = KNN algorithm using the
      Euclidean metric
end

$recommendedMovies$ = []

for $neighbor$ in $neighbors$ do
    $avgRating$ = average rating of the movie
      (additional information from knn)
      $recommendedMovies$ += [$neighbor$,
      $avgRating$]
end

return $recommendedMovies$
```

Algorithm 5: An algorithm containing various metrics to find the best matches for the user.

9. Data base

9.1. Used Database

The following database was used for demonstration purposes in a non-commercial, scientific manner - *IMDB* [32] data set downloaded from datasets.imdbws.com. Tables used:

- name.basics.tsv
- title.basics.tsv
- title.crew.tsv
- title.ratings.tsv

The database, after our simplifications and prior preparation, contains a collection of 9,827 films with information:

9.2. Description of the columns

The set consists of 6000 rows and 7 columns.
A detailed description is provided below:

1. **OriginalTitle** - Original title of a movie.
2. **Genres** - List of movie genres.
3. **AverageRating** - Average rating of a movie.
4. **Writers** - A list of writers of a given movie.
5. **Directors** - A list of directors of a given movie.
6. **Genres_bin** - Converted column 'Genres' to a numerical form.
7. **Writers_bin** - Converted column 'Writers' to a numerical form.
8. **Directors_bin** - Converted column 'Directors' to a numerical form.

Based on the data base above, we have created several rankings that show the popularity ratio of the data that was used in the KNN algorithm:

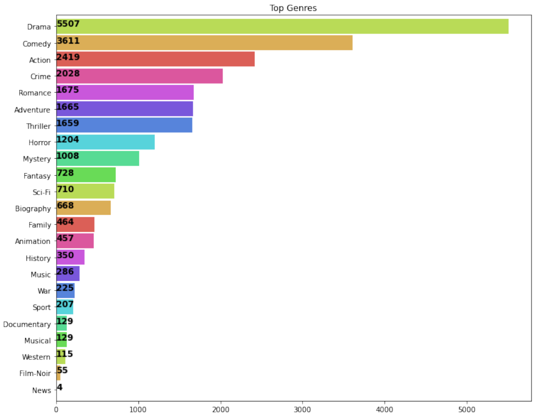

Figure 1: Top genres

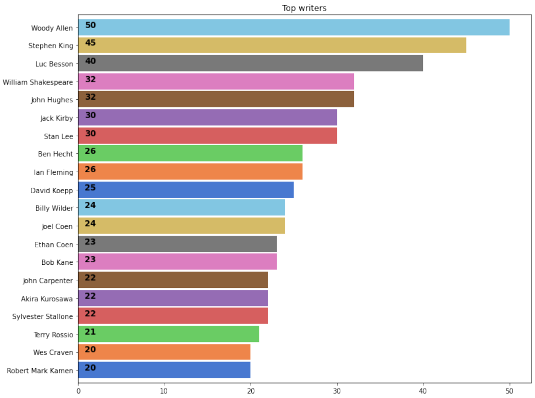

Figure 2: Top writers

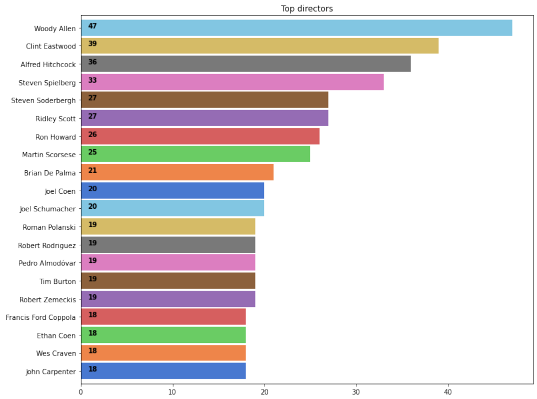

Figure 3: Top directors

10. Conclusion and future work

In order to improve the operation of the algorithm and to make the use of it more enjoyable, you can use a more friendly GUI in the future. To make the algorithm work better, it is also possible to use more data (more extensive user history) to further refine the metic used.

References

[1] M. A. Sanchez, O. Castillo, J. R. Castro, Generalized type-2 fuzzy systems for controlling a mobile robot and a performance comparison with interval type-2 and type-1 fuzzy systems, Expert Systems with Applications 42 (2015) 5904–5914.

[2] Q.-b. Zhang, P. Wang, Z.-h. Chen, An improved particle filter for mobile robot localization based on particle swarm optimization, Expert Systems with Applications 135 (2019) 181–193.

[3] Y. Li, W. Dong, Q. Yang, S. Jiang, X. Ni, J. Liu, Automatic impedance matching method with adaptive network based fuzzy inference system for wpt, IEEE Transactions on Industrial Informatics 16 (2019) 1076–1085.

[4] F. Qu, J. Liu, H. Zhu, D. Zang, Wind turbine condition monitoring based on assembled multidimensional membership functions using fuzzy inference system, IEEE Transactions on Industrial Informatics 16 (2019) 4028–4037.

[5] M. Woźniak, A. Zielonka, A. Sikora, Driving support by type-2 fuzzy logic control model, Expert Systems with Applications 207 (2022) 117798.

[6] M. Woźniak, A. Zielonka, A. Sikora, M. J. Piran, A. Alamri, 6g-enabled iot home environment control using fuzzy rules, IEEE Internet of Things Journal 8 (2020) 5442–5452.

[7] F. Bonanno, G. Capizzi, A. Gagliano, C. Napoli, Optimal management of various renewable energy sources by a new forecasting method, in: SPEEDAM 2012 - 21st International Symposium on Power Electronics, Electrical Drives, Automation and Motion, 2012, pp. 934–940. doi:10.1109/SPEEDAM.2012.6264603.

[8] Y. Zhang, S. Cheng, Y. Shi, D.-w. Gong, X. Zhao, Cost-sensitive feature selection using two-archive multi-objective artificial bee colony algorithm, Expert Systems with Applications 137 (2019) 46–58.

[9] M. Ren, Y. Song, W. Chu, An improved locally weighted pls based on particle swarm optimization for industrial soft sensor modeling, Sensors 19 (2019) 4099.

[10] D. Połap, M. Woźniak, C. Napoli, E. Tramontana, R. Damaševičius, Is the colony of ants able to recognize graphic objects?, Communications in Computer and Information Science 538 (2015) 376–387. doi:10.1007/978-3-319-24770-0_33.

[11] M. Woźniak, A. Sikora, A. Zielonka, K. Kaur, M. S. Hossain, M. Shorfuzzaman, Heuristic optimization of multipulse rectifier for reduced energy consumption, IEEE Transactions on Industrial Informatics 18 (2021) 5515–5526.

[12] F. Bonanno, G. Capizzi, C. Napoli, Some remarks on the application of rnn and prnn for the charge-discharge simulation of advanced lithium-ions battery energy storage, in: SPEEDAM 2012 - 21st International Symposium on Power Electronics, Electrical Drives, Automation and Motion, 2012, pp. 941–945. doi:10.1109/SPEEDAM.2012.6264500.

[13] G. M. Khanal, G. Cardarilli, A. Chakraborty, S. Acciarito, M. Y. Mulla, L. Di Nunzio, R. Fazzolari, M. Re, A zno-rgo composite thin film discrete memristor, in: 2016 IEEE International Conference on Semiconductor Electronics (ICSE), IEEE, 2016, pp. 129–132.

[14] F. Bonanno, G. Capizzi, S. Coco, C. Napoli, A. Laudani, G. Sciuto, Optimal thicknesses determination in a multilayer structure to improve the spp efficiency for photovoltaic devices by an hybrid fem - cascade neural network based approach, in: 2014 International Symposium on Power Electronics, Electrical Drives, Automation and Motion, SPEEDAM 2014, IEEE Computer Society, 2014, pp. 355–362. doi:10.1109/SPEEDAM.2014.6872103.

[15] S. Acciarito, A. Cristini, L. Di Nunzio, G. M. Khanal, G. Susi, An a vlsi driving circuit for memristor-based stdp, in: 2016 12th Conference on Ph. D. Research in Microelectronics and Electronics (PRIME), IEEE, 2016, pp. 1–4.

[16] V. S. Dhaka, S. V. Meena, G. Rani, D. Sinwar, M. F. Ijaz, M. Woźniak, A survey of deep convolutional neural networks applied for prediction of plant leaf diseases, Sensors 21 (2021) 4749.

[17] G. Capizzi, G. Lo Sciuto, C. Napoli, M. Woźniak, G. Susi, A spiking neural network-based long-term prediction system for biogas production, Neural Networks 129 (2020) 271 – 279.

[18] N. Brandizzi, S. Russo, R. Brociek, A. Wajda, First studies to apply the theory of mind theory to green and smart mobility by using gaussian area clustering, volume 3118, CEUR-WS, 2021, pp. 71–76.

[19] F. Bonanno, G. Capizzi, G. Lo Sciuto, C. Napoli, Wavelet recurrent neural network with semi-parametric input data preprocessing for micro-wind power forecasting in integrated generation systems, in: 5th International Conference on Clean Electrical Power: Renewable Energy Resources Impact, ICCEP 2015, 2015, p. 602 – 609.

[20] G. Borowik, M. Woźniak, A. Fornaia, R. Giunta, C. Napoli, G. Pappalardo, E. Tramontana, A software architecture assisting workflow executions on cloud resources, International Journal of Electronics and Telecommunications 61 (2015) 17–23. doi:10.1515/eletel-2015-0002.

[21] M. Wieczorek, J. Siłka, M. Woźniak, S. Garg, M. M. Hassan, Lightweight convolutional neural network model for human face detection in risk situations, IEEE Transactions on Industrial Informatics 18 (2021) 4820–4829.

[22] S. Illari, S. Russo, R. Avanzato, C. Napoli, A cloud-oriented architecture for the remote assessment and follow-up of hospitalized patients, in: CEUR Workshop Proceedings, volume 2694, CEUR-WS, 2020, pp. 29–35.

[23] N. Dat, V. Ponzi, S. Russo, F. Vincelli, Supporting impaired people with a following robotic assistant by means of end-to-end visual target navigation and reinforcement learning approaches, in: CEUR Workshop Proceedings, volume 3118, CEUR-WS, 2021, pp. 51–63.

[24] R. Brociek, G. Magistris, F. Cardia, F. Coppa, S. Russo, Contagion prevention of covid-19 by means of touch detection for retail stores, in: CEUR Workshop Proceedings, volume 3092, CEUR-WS, 2021, pp. 89–94.

[25] O. Dehzangi, M. Taherisadr, R. ChangalVala, Imu-based gait recognition using convolutional neural networks and multi-sensor fusion, Sensors 17 (2017) 2735.

[26] M. Wozniak, C. Napoli, E. Tramontana, G. Capizzi, G. Lo Sciuto, R. Nowicki, J. Starczewski, A multiscale image compressor with rbfnn and discrete wavelet decomposition, in: Proceedings of the International Joint Conference on Neural Networks, volume 2015-September, Institute of Electrical and Electronics Engineers Inc., 2015. doi:10.1109/IJCNN.2015.7280461.

[27] H. G. Hong, M. B. Lee, K. R. Park, Convolutional neural network-based finger-vein recognition using nir image sensors, Sensors 17 (2017) 1297.

[28] G. Capizzi, G. Lo Sciuto, M. Woźniak, R. Damaševičius, A clustering based system for automated oil spill detection by satellite remote sensing, Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics) 9693 (2016) 613 – 623.

[29] A. T. Özdemir, B. Barshan, Detecting falls with wearable sensors using machine learning techniques, Sensors 14 (2014) 10691–10708.

[30] K. G. Liakos, P. Busato, D. Moshou, S. Pearson, D. Bochtis, Machine learning in agriculture: A review, Sensors 18 (2018) 2674.

[31] G. Iannizzotto, L. Lo Bello, A. Nucita, G. M. Grasso, A vision and speech enabled, customizable, virtual assistant for smart environments, in: 2018 11th International Conference on Human System Interaction (HSI), IEEE, 2018, pp. 50–56.

[32] I. IMDb.com, Imdb datasets, 1990-2022. URL: https://www.imdb.com/interfaces/.