**Using Machine Learning Techniques to Predict Potential Customers and Their Purchasing Items**

Short Title: Machine Learning to Predict Customers and Purchases

Zihao Wang

Long Island University, zihao.wang3@my.liu.edu

Sevim Oztimurlenk

Long Island University, [sevim.oztimurlenk@my.liu.edu](mailto:sevim.oztimurlenk@my.liu.edu)

Ling Zhu

Long Island University, ling.zhu@liu.edu

The need has increased in B2C e-commerce for a more accurate recommendation system that targets customers. The system should be able to provide customers targeted purchase information based on their online behavior. A powerful and meaningful recommendation system can offer enormous value to customers and e-commerce providers. Customers can locate an item of their interest more easily and quickly. On the other hand, an e-commerce provider can achieve a higher customer conversion rate and thus increase sales. To this end this research developed a two-step machine-learning-based recommendation prototype with two data mining goals: to identify potential customers with purchase intention and to predict the products they may purchase based on an e-commerce log (i.e., a sequential record of products click-through events). In the first module, we used both the K-nearest neighbor (KNN) model and the multi-layer perceptron (MLP) neural network to classify customers with and without purchase intention. In the second module, we developed a first-order Markov chain model to predict products that those potential customers will purchase. We then tested the prototype with the dataset provided in the RecSys 2015 challenge. The results show that in the first module both the KNN and the MLP neural network have their own advantages and disadvantages in identifying those potential customers. In the second module the Markov model also performs well to predict the products purchased by a customer. The findings show that the prototype has a great potential to be developed into a machine-learning-based recommendation system for e-commerce usage. To our knowledge, this prototype is the first of its kind to combine a classification model with time series analysis.

CCS CONCEPTS • Recommendation system • Neural network • K-nearest neighbor • Markov chain

**Additional Keywords and Phrases:** e-commerce, purchase intention, customer preference, online behavior, targeted marketing.

1. Introduction

Online service personalization has become a growing research and commercial interest [5]. Both customers and businesses benefit from personalization systems [1]. For example, through a recommendation system, customers can locate an item of their interest quickly and easily. E-commerce service providers can achieve a higher customer conversion rate and thus increase sales. It is a big challenge, however, as how to precisely know customers’ needs.

This study aims to identify potential customers who have purchase intention and what products they may purchase. We propose two research questions in the study: 1) whether a customer has an intention to purchase items online in a given session; 2) If a customer does have the intention, what products are most likely to be purchased by the customer.

The paper is presented in the following sections: the first section presents the results of our literature review. We review the methods for mining potential customers with purchase intention and creating product recommendations. The second section describes the methodology, data collection, and analytical techniques we developed for the study. In the third section we discussed the results of our predictive model. The future applications are discussed in the conclusion section. The process of this study is illustrated in Figure 1.

**Select customers who have intention to purchase products**

**Select features for purchase intention prediction**

**Use neural network and KNN to analysis if customers want to purchase products**

**Generate a set of customers with purchase intention**

**Calculate accuracy for training and testing dataset**

**Compare the performance between 2 models**

**Predict items likely to be purchased by those customers**

**Implement Markov chain to predict customers’ preferable products**

**Generate a set of customers with predicted customers’ preferable products**

**Calculating accuracy of selected dataset**

Figure 1: The process of this research

1. Literature Review
   1. Mining potential customers with purchase intention

Romov and Sokolov proposed seven features that are related to purchase intention, see Table 1 [5].

Table : Features related to purchase intention [5]

|  |  |  |
| --- | --- | --- |
|  | Feature Description | Number/Type |
| 1 | Numerical time features of the start/end of the session (month, day, hour, minute, second, etc.) | 2 × 7 Number |
| 2 | Categorical time features of the start/end of the session (month, day, month-day, month-day-hour, hour, minute, weekday) | 2 × 7 Category |
| 3 | Length of the session in seconds | 1 Number |
| 4 | Number of clicks, unique items, categories, and item-category pairs in the session | 4 Number |
| 5 | Top 10 items and top 5 categories by the number of clicks in the session | 15 Category |
| 6 | IDs of the first/last item clicked at least k = 1, 2 . . ., 6 times in the session | 12 Category |
| 7 | Vector of click numbers and total durations for 100 items and 50 categories that were the most popular in the whole training set | 150 × 2 Number |

Pálovics et al. developed a linear model to find customers with purchase intention [4]. The model used features of numbers of clicks on the item, the item, the day of the week, the hour of the day, dwell time on the item, and the maximum dwell time in the session. It used “buy/click ratio,” a fraction of the number of buyer sessions in some given amount of sessions, to analyze the relationship between each feature and the buy/click ratio for all sessions.

Figure 2(a) shows the relationship between the week of days and the buy/click ratio. More customers are likely to make purchases on a Wednesday. Figure 2(b) shows the relationship between hours of days and the buy/click ratio. More people prefer shopping in the afternoon. Figure 2(c) describes the relationship between session length and the buy/click ratio. If the session length is between 1 and 55, with the increase of session length, it is more likely that a customer will make a purchase. Figure 2(d) shows the relationship between the dwell time and the buy/click ratio. The more dwell time, the more buy/click ratio. Figure 2(e) shows the relationship between the number of items in a session and the buy/click ratio. The buy/click ratio increases when the number of the products searched increases.

|  |  |
| --- | --- |
| (a) | (b) |
|  |  |
| (c) | (d) |
|  |  |
| (e) |  |

Figure : Relationship between each feature and the buy/click rate [4]

Chen et al. used the multi-Layer Perceptron (MLP) neural network with a single hidden layer to find customers with purchase intention [2]. The deep learning model used the tanh activation function in the hidden layer and the SoftMax activation function in the output layer. The structure of the MLP neural network is shown in Figure 3.

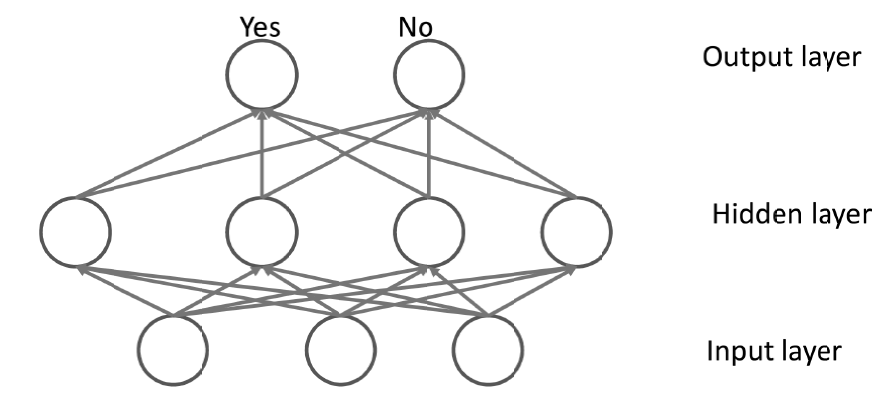


Figure 3: The single hidden layer MLP neural network [2]

Guo et al. proposed DeepFM, a deep learning algorithm that combines a factorization machine with a neural network, to predict the click-through rate in a recommender system [3]. The input is a multi-dimension vector and the output is a value of either 0 or 1, which represents click behaviors (1 means customer clicked the item, and 0 otherwise). The factorization model and deep learning model are combined and output a value by using the sigmoid function. The output function is defined as:

()

In this function, ,yFM is the output of the factorization machine and yDNN is the output of the neural network.

There are two hidden layers in the learning structure of DeepFM. Each layer has 32 neurons with a ReLU activation function. The structure is shown in Figure 4.



Figure 4: The structure of DeepFM [3]

* 1. Predicting products of the customers’ interests
     1. Association rule

Association rule learning is a rule-based machine learning method for product recommendations. Two measures are used to evaluate the association model: support and confidence. Definitions of these two measures are described below.

Let X, Y be an itemset, X⇒Y is an association rule, and T is a set of transactions.

Support is an indication of how frequently the itemset appears in the transactional dataset. The support of X with respect to T is defined as the proportion of transactions that contain X, as defined:

()

Confidence is an indication of how often the rule is true, as defined:

()

* + 1. Markov chain

Yang et al. developed a hybrid recommendation system that unifies similarity models with Markov chains for sequential recommendation [6]. The system used both sequential patterns and content similarity to analyze the relationship between items.

First, it used a high-order Markov chain to model sequential patterns over time series. Second, the most frequent items are predicted by the high-order Markov chain. Third, the content similarity is used to recommend other similar products from the item list. The process of this study is shown in Figure 5.

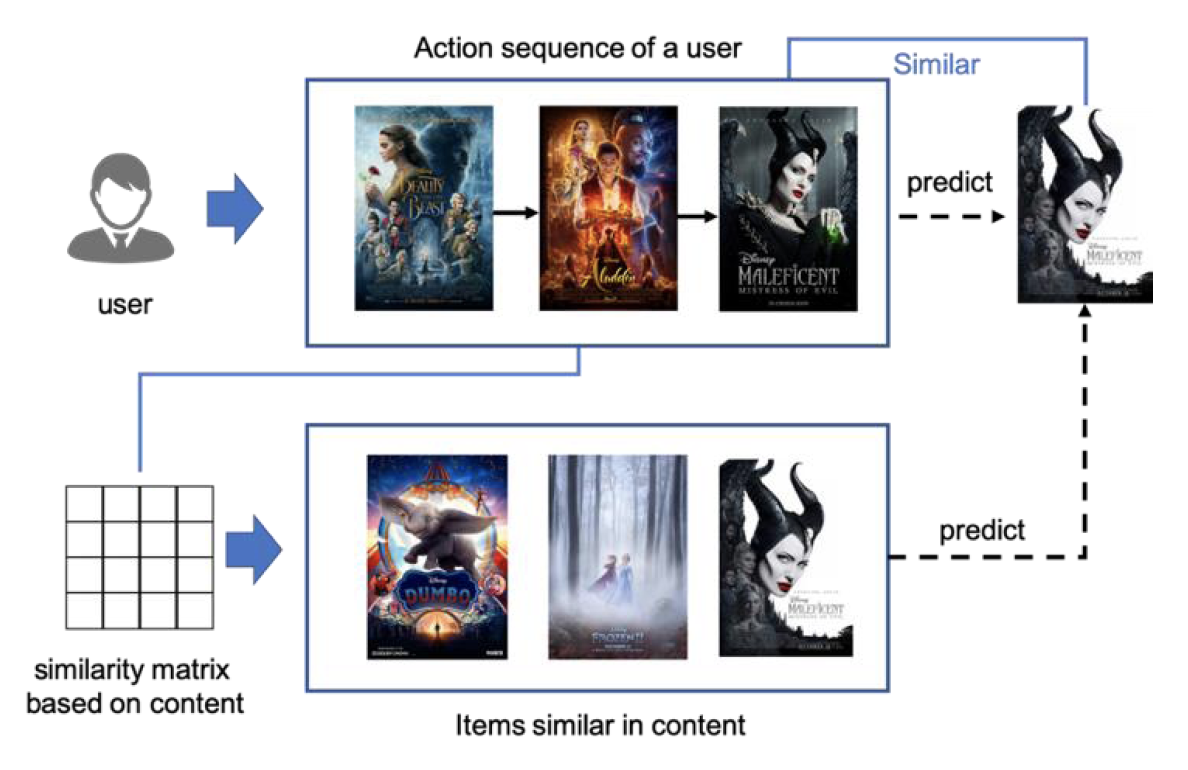


Figure 5: An example of recommendation system using Markov chain and similarity [6]

In Figure 5, a customer searched a sequence of movies (Beauty and Beast, Aladdin, and Maleficent 1). The Markov chain predicted that the movie “Maleficent 1” that this customer might watch next. Based on the category of “Maleficent 1”, similar movies were recommended from the movie list, for example, “Maleficent 2.”

The Markov chain represents sequential patterns by building the relationship between items and the transition probabilities from the transition matrix. The inner product between the vector of the probability distribution of the current state and the transition matrix represents the transition probability of the next step. Markov chain has better performance on sparse transaction data.

1. Methodology
   1. Data collection

This study used e-commerce search logs from RecSys 2015 challenge [1]. It contains two datasets: clicked events and bought events. The columns in the clicked events dataset are explained in Table 2.

Table 2: Description of clicked events dataset

|  |  |
| --- | --- |
| Item | Description |
| Session ID | The id of the session is represented as an integer number. One session ID represents an unique customer. In one session there are one or many clicks |
| Timestamp | The time when the click occurred. |
| Item ID | The unique identifier of the item that has been clicked, represented as an integer number |
| Category | The context of the click |

An example of clicked event data is shown in Figure 6.

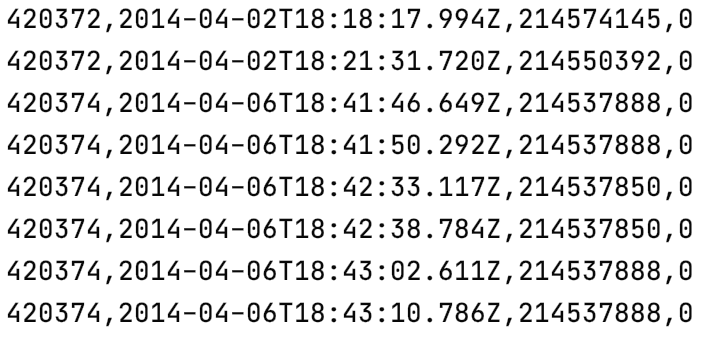


Figure 6. An example of click event data

The bought events dataset contains the products bought. The columns in the dataset are described in Table 3.

Table 3: The description of bought events data

|  |  |
| --- | --- |
| Item | Description |
| Session ID | The id of the session is represented as an integer number. One session ID represents an unique customer. In one session there are one or many buying events |
| Timestamp | The time when the buy occurred. |
| Item ID | The unique identifier of item that has been bought, represented as an integer number |
| Price | The price of the item represented as an integer number |
| Quantity | The quantity in this buying, represented as an integer number |

An example of the bought events data is shown in Figure 7.

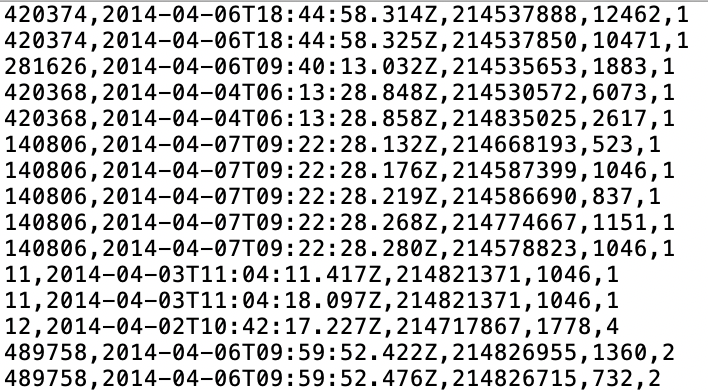


Figure 7: An example of the bought events data

The session id in the bought event dataset is linked to the same column in the clicked event dataset; that is, if a customer had bought an item, her click behavior can be found in the clicked event.

* 1. Mining potential customers with purchase intention

The first step is finding effective features that can be used for training machine learning classifiers. We implemented in this study two machine learning models: MLP neural network and KNN classifier. The features used in the models were 1) the day of the starting week, 2) the hour of the starting day, 3) the day of finishing week, 4) the hour of the finishing week, 5) maximum dwell time in the session, 6) numbers of clicks on the session, and 7) the number of items in the session. The feature vector for training is a vector with a length of seven. The description and type for each element are illustrated in Table 4.

Table 4: Features used in the purchase intention analysis

|  |  |
| --- | --- |
| Feature Description | Number/Type |
| Time features of the start/end of the session (day of week, hour) | 4 Category |
| Length of the session in seconds (dwell time) | 1 Number |
| Number of clicks | 1 Number |
| Number of items | 1 Number |

* + 1. Neural network method

We implemented the MLP neural network based on the structure of DeepFM. A neural network with 3 hidden layers is constructed with tanh activation function in each hidden layer. The first and second hidden layers had 32 neurons and the third hidden layer had 16 neurons. The output layer had 2 neurons with a SoftMax activation function and cross-entropy loss function. The structure of the neural network is shown in Figure 8.

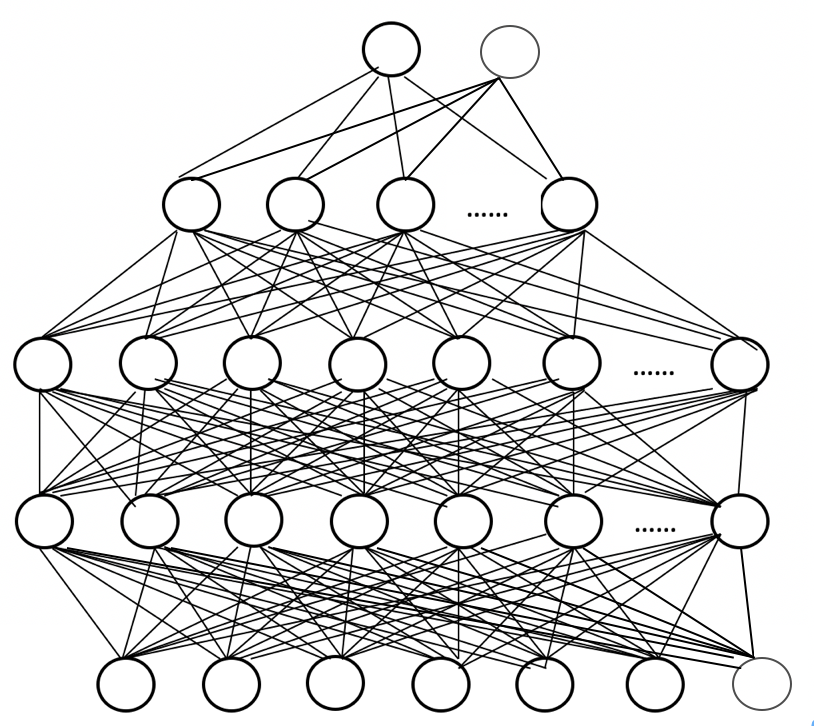


Figure 8: The structure of the neural network

* + 1. K-nearest neighbor method

The K-nearest neighbor (KNN) algorithm is one of the machine learning algorithms for classification. An object is classified by a plurality vote of its neighbors, with the object being assigned to the class most common among its k nearest neighbors. Details are illustrated in Figure 9.



Figure 9: Illustration of KNN

The green dot needs to be classified into either the blue or red group. In this example, if k=3, the green dot finds its closest three neighbors. Among the three neighbors, the majority number of the category is the green dot’s category, which is the triangle.

The k value determined for this study is illustrated in the next section.

* 1. Predicting products likely to be purchased by customers
     1. Sequential recommendation method based on Markov chain

We use the first-order Markov chain to predict products that customers may purchase based on their search sequences. There are four steps in this part. The first step is selecting the unique selected product sequence (USP) set for each customer. If the size of USP is less than 5, then all items in the USP are seen as the prediction items. Otherwise, the second step is calculating the item probability distribution and the transition matrix for the current state based on the current customer’s unique selected products. Third, the probability distribution of purchasing products is computed based on the inner product of the current probability distribution and the transition matrix. Lastly, the searched products are sorted by the decreasing order of the probability distribution of purchasing products, and the first half of items are treated as the final prediction.

The process for selecting the unique selected products (USP) sequence for each customer is described in Algorithm 1.

|  |
| --- |
| ALGORITHM 1: The process of selecting the unique selected products |
| for each customer:  queue=empty  for item in searched sequence based on time series:  if item not in queue:  enqueue(item) |

A file that recorded all customers and their unique selected products is generated from the clicked events dataset. An example of the file is shown in Table 5.

Table 5. An example of customers and their unique selected products

|  |  |
| --- | --- |
| Session ID | Unique selected products |
| 1 | 214536502, 214536500, 214536506, 214577561 |
| 2 | 214662742, 214825110, 214757390, 214757407, 214551617 |
| 3 | 214716935, 214774687, 214832672 |
| 4 | 214836765, 214706482 |

To get the probability distribution of the current state, the actual searched products (ASP) set is summarized for each customer. The process is described in Algorithm 2.

|  |
| --- |
| ALGORITHM 2: The process to get the actual searched products |
| for each customer:  queue=empty  for item in searched sequence based on time series:  enqueue(item) |

A file that recorded all customers and their actual selected products is generated from the clicked events dataset. An example of the file is shown in Table 6. The customer with session ID 2 searched product “214662742” appears once in Table 5 but twice in Table 6.

Table 6: An example of customers and their actual selected products

|  |  |
| --- | --- |
| Session ID | Actual selected products |
| 1 | 214536502, 214536500, 214536506, 214577561 |
| 2 | 214662742, 214662742, 214825110, 214757390, 214757407, 214551617 |
| 3 | 214716935, 214774687, 214832672 |
| 4 | 214836765, 214706482 |

In current state probability (CSP) distribution is defined as:

()

()

The transition matrix is the size of N × N matrix, where N is the size of USP set for each customer. Rows and columns are the element order in the USP set. The transition probability is defined as:

()

where means item x appeared before item y in the USP set.

The process of the first order Markov chain is described in Algorithm 3.

|  |
| --- |
| ALGORITHM 3: The process of the first order Markov chain |
| all\_customers\_USP=Algorithm1  all\_customers\_ASP=Algorithm 2  for current\_customer\_USP in All\_customers\_USP:  if size(current\_customer\_USP)<5:  prediction= current\_customer\_USP  else:  get transistion matrix for current\_customer\_USP  get initial item distribution for current\_customer\_USP  compute the buying probability distribution for current\_customer\_USP  sort current\_customer\_USP by decrease order  prediction=first half items from the sorted current\_customer\_USP  return prediction |

1. Results
   1. Mining customers with purchase intention

The clicked events dataset has 9,249,729 customers who are randomly shuffled and separated into the training, validation, and testing datasets. The validation and testing dataset have 10,000 customers each. They are the same for testing the neural network and KNN classifier. The remaining 9,229,729 customers are used for training the neural network. Within those customers, the top 1,000,000 customers are selected for training the KNN model. The detailed description of the dataset is shown in Table 7.

Table 7: Dataset description

|  |  |  |  |
| --- | --- | --- | --- |
|  | Total number of customers | Number of customers with purchase intention | Number of customers without purchase intention |
| Training MLP | 9,229,729 | 508,621 | 8,721,108 |
| Training KNN | 1,000,000 | 55,042 | 944,958 |
| Validation | 10000 | 529 | 9471 |
| Testing | 10000 | 546 | 9454 |
| Total | 9,249,729 | 509,696 | 8,740,033 |

The neural network was trained 100 epochs with a batch size 50,000 each. After the training process, the best result appeared on the initial state with the accuracy of 0.89, and the true positive rate of 0.03 for correctly predicting customers with purchase intention on the validation dataset. With the testing dataset, the respondent accuracy and true positive rate are 0.89 and 0.02. The confusion matrix based on the testing dataset is shown in Table 8.

Table 8: The confusion matrix of MLP neural network

|  |  |  |
| --- | --- | --- |
| Predict  Actual | Without purchase intend | With purchase intend |
| Without purchase intend | 8888 | 566 |
| With purchase intend | 537 | 9 |

The validation loss and accuracy during the 100 epochs are shown in Figure 10(a) and Figure 10(b).

|  |  |
| --- | --- |
| (a) | (b) |

Figure : The trend of loss and accuracy of the validation dataset

The loss dramatically decreased within the first 5 epochs and the accuracy increased dramatically beyond 0.94 after the first epoch. This phenomenon demonstrates the effectiveness of the extracted features in Table 4. However, due to the number of customers with purchase intention is lower than the number of customers without purchase intention, the neural network can only detect a few costumers with purchase intention.

In the KNN model, the k value was an odd number between 1 and 20. The best k value is 1 with an accuracy of 0.9 and the true positive rate is 0.12 in the validation dataset. The accuracy and true positive rate for the testing set are 0.9 and 0.13. The confusion matrix is shown in Table 9.

Table 9: Confusion matrix of KNN classifier

|  |  |  |
| --- | --- | --- |
| Predict  Actual | Without purchase intend | With purchase intend |
| Without purchase intend | 8948 | 506 |
| With purchase intend | 476 | 70 |

Comparing the neural network with the KNN model, the KNN model has better performance than the neural network because it used less training data and achieved better accuracy and true positive rate.

* + 1. Models comparison

Between the MLP and the KNN classifier, the accuracy of the testing dataset, true positive rate of the testing dataset, amount of training dataset, and the running time for the testing dataset are shown in Table 10.

Table . Comparison between MLP neural network and KNN classifier

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
|  | Accuracy | True positive rate | Amount of training dataset | Running time (s) |
| MLP neural network | 0.89 | 0.02 | 9,229,729 | 0.12 |
| KNN classifier | 0.9 | 0.13 | 1,000,000 | 1.15 |

From the application perspective, the KNN classifier is better than the MLP neural network in processing small amounts of data. There is no difference between the accuracy of the two models. The true positive rate of the KNN classifier is better than the MLP neural network.

From the perspective of algorithm performance, the MLP neural network is better than KNN. The neural network is capable to handle a much larger training dataset than the KNN classifier. Moreover, the MLP neural network spent less time than the KNN classifier in processing the data.

* 1. Predicting the products likely to be purchased by customers

The top 1,000 customers with purchasing behavior in the clicked dataset are selected. For each customer, the actual bought products (ABP) set is generated from the bought products dataset. The process is the same as Algorithm 1. The predicted purchasing products (PBP) set is computed by the first-order Markov chain model. The accuracy for each customer is defined as:

()

The average accuracy among the 1,000 customers is 0.97. The histogram of the accuracy is shown in Figure 11.

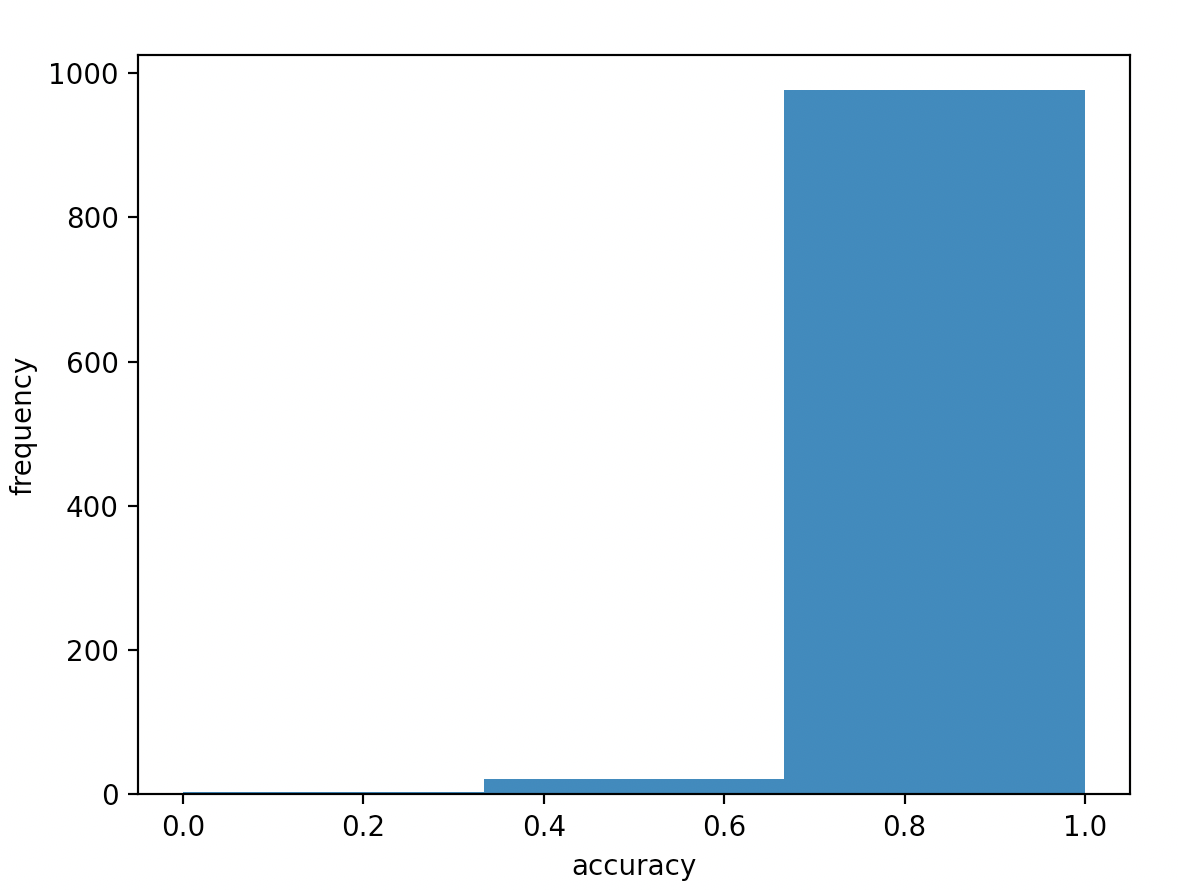


Figure : The histogram of accuracy

The histogram shows most customers have an accuracy between 0.65 and 1.0. A few customers have an accuracy between 0.35 and 0.65. The result indicates this model can apply to real-world situations.

1. Conclusion

This study proposed a two-step model for mining potential customers and predicting their interesting products. In the first step, the MLP neural network and KNN classifier can recognize customers with purchase intention with an accuracy of 90%. With the increase of the data size, the speed and the data processing capacity of the MLP neural network have a better performance than the KNN classifier. It implies that the MLP neural network has potential to deal with big data in real-world applications. In the second step, the first-order Markov chain shows an excellent performance in sequential analysis with an accuracy of 97%. It can also apply to real-world situations. In the future, high-performance models for mining potential customers can be explored. A recommendation system that utilizes the two-step model demonstrated in this study will be developed and applied to real e-commerce cases.

REFERENCES

[1] David Ben-Shimon, Alexander Tsikinovsky, Michael Friedmann, Bracha Shapira, Lior Rokach, and Johannes Hoerle. 2015. RecSys Challenge 2015 and the YOOCHOOSE Dataset. In *Proceedings of the 9th ACM Conference on Recommender Systems*, ACM, Vienna Austria, 357–358. DOI:https://doi.org/10.1145/2792838.2798723

[2] Wenliang Chen, Zhenghua Li, and Min Zhang. 2015. Linear and Non-Linear Models for Purchase Prediction. In *Proceedings of the 2015 International ACM Recommender Systems Challenge*, ACM, Vienna Austria, 1–4. DOI:https://doi.org/10.1145/2813448.2813518

[3] Huifeng Guo, Ruiming Tang, Yunming Ye, Zhenguo Li, and Xiuqiang He. 2017. DeepFM: A Factorization-Machine based Neural Network for CTR Prediction. *arXiv:1703.04247 [cs]* (March 2017). Retrieved March 1, 2021 from http://arxiv.org/abs/1703.04247

[4] Róbert Pálovics, Péter Szalai, Levente Kocsis, Adrienn Szabó, Erzsébet Frigó, Júlia Pap, Zsófia K. Nyikes, and András A. Benczúr. 2015. Solving RecSys Challenge 2015 by Linear Models, Gradient Boosted Trees and Metric Optimization. In *Proceedings of the 2015 International ACM Recommender Systems Challenge*, ACM, Vienna Austria, 1–4. DOI:https://doi.org/10.1145/2813448.2813513

[5] Peter Romov and Evgeny Sokolov. 2015. RecSys Challenge 2015: ensemble learning with categorical features. In *Proceedings of the 2015 International ACM Recommender Systems Challenge*, ACM, Vienna Austria, 1–4. DOI:https://doi.org/10.1145/2813448.2813510

[6] Y. Yang, H.-J. Jang, and B. Kim. 2020. A Hybrid Recommender System for Sequential Recommendation: Combining Similarity Models With Markov Chains. *IEEE Access* 8, (2020), 190136–190146. DOI:https://doi.org/10.1109/ACCESS.2020.3027380