Research on prediction of online purchasing behavior based on hybrid model

Xi Chen¹, Shichou Ding¹*, Yongsheng Xiang¹ and Lin Liu¹

¹ Department of Computer & Communication Engineering, Changsha University of Science & Technology, Changsha, Hunan, 410114, China;

*Corresponding author’s e-mail: dsc24@stu.csust.edu.cn

Abstract. The research on the potential purchase behavior of users can help merchants develop better marketing strategies. At present, many research methods of online purchasing behavior are based on simple rule prediction, and the prediction results are not satisfactory. We design a hybrid model of Gradient Boosting Decision Tree and logistic regression to accurately predict the purchase behavior of users, which combines the association characteristics between users and commodities. Firstly, clustering algorithm and association rules are used to solve the problem of data imbalance and mine more potential related features. This scheme not only improves the processing efficiency of large data, but also solves the problem of user cold start. Secondly, we construct a scalable tree enhancement system (XGBoost) to train the initial feature set, which is a strong classifier composed of several weak classifiers. A new training set combines the new features with the original features through feature reconstruction, and a hybrid machine learning system is constructed by logistic regression (LR) model. Finally, the LR model is trained by the new training set. Compared with the existing schemes, the integrated decision tree model can train more sample sets with less resources. The experimental results show that the accuracy of the hybrid model is better than single model, and the F1_score is higher.

1. Introduction
In recent years, with the rapid development of e-commerce and the continuous change of marketing strategy, China has entered a period of rapid development towards intelligence. With the increased application of machine learning, especially in the fields of e-commerce, the establishment of an effective prediction model and recommendation system has a crucial impact on the corporate profits [1]. In pace with the advent of the era of big data, the demand for data analysis in various industries continues to increase. Obtaining valuable information and data through data mining has gradually become the main driving propellant for the development of machine learning technology. It is vital to predict the purchase behavior of users according to their historical behavior data in recommendation system. The traditional collaborative filtering algorithm recommend you related products according to the browsing records of person similar to you, and recommend similar items according to your favorite items. It can be classified into two types: user-based collaborative filtering algorithm [2] and item-based collaborative filtering algorithm [3]. However, the traditional collaborative filtering algorithm lacks of sparsity and scalability in data processing with the non-ideal prediction results. Generalized linear model LR model [4] difficulty to express the characteristics of nonlinear relationship, the accuracy of the model might be unacceptable. In order to enhance the nonlinear relationship between the original features and the fitting targets, it is advisable to do some feature transformation for the new features. The common conversion
methods include: the discretization of continuous features, the cross between features, one-hot coding [5].

Ensemble learning has two series: one is Boosting [6] series, of which characters are the dependencies among weak learners; The other is Bagging series[7], of which characters are the parallel fitting of weak learners without dependence. RF (Random Forest) [8] model is an optimized version of Bagging algorithm. the random forest algorithm is an optimized version of Bagging algorithm, of which the idea is unique improvements of Bagging. First of all, RF uses the cart decision tree as weak learners, which is similar to GBDT. Second, based on the use of decision tree, RF improves the establishment of decision tree. Ordinary decision tree selects one of the best features among all n sample features on the node to divide the left and right subtrees of the decision tree, while RF randomly selects a part of the sample features on the node, assuming that it is m and m < n, and then one of the best features among these m sample features on the node to divide the left and right subtrees of the decision tree, which can further enhance the generalization of the model. However, for sample sets with strong noise, RF model can easily be over fitting, which is means that the features with more value division have a greater impact on the RF decision-making easily, leading to the influence on the fitting effect of the model.

Recent researches show that the users’ historical behavior data of most e-commerce websites are very irregular and scattered with the large amount of data [9]. However, not each of these information and data is valuable. The main function of recommendation systems is to analyze users’ behavior, find the users’ personalized needs, and then recommend the related products to interested users. Recommendation systems can find valuable data in massive data. for example, help users to seek out the products which are needed and difficult to find. Although the results of traditional recommendation algorithms based on the simple user behavior statistics are not promising, they are still widely used on the Internet, which might cause the omission of a lot of valuable data. In this paper, we find that there are less obvious rules and hidden association features in user behavior data through association rules, which can provide guidance for product design and bring a better experience to users.

Before building the model, we have to preprocess mass of data, while clustering and association rules are the main research contents within the existing data mining methods. Clustering is mainly to divide similar samples into the same cluster, and divide different samples into different clusters as far as possible. With the development of machine learning, many classical clustering algorithms have been generated. K-means algorithm [10] and K-medoids [11] algorithm are classical clustering algorithms based on partition. By comparison, the K-means algorithm was praised for low computational complexity, while the K-medoids has strong robustness and strong ability to deal with noise data. Boosting series algorithms, which are widely used in practical machine learning, train the interdependent weak classifiers in a serial way, of which basic idea is to stack the weak classifiers layer by layer, assign a higher weight to the misclassified samples in the previous layer during the training in each layer, and obtain the final results according to the weighted results of each layer by testing of strong classifiers. The main algorithms are Gradient Boosting Decision Tree (GBDT)[12] and Tree Boosting System (XGBoost)[13].

Under the existing conditions, the historical users’ behavior data of most e-commerce websites has a large time span. It is found that the influence of users’ behavior on purchase will weaken with time, hence the latest users’ behavior data might be more representative. Therefore, we have to set the weight as appropriate during the process of feature extraction. A hybrid model combining Gradient Boosting Decision Tree and logistic regression is proposed in [14], which is better than either of the two models in performance. It is well known that the relevant parameters of the model will affect the final prediction effect. As a result, the feature engineering is very important in this paper. The influence of other factors will be negligible with a complete feature set and a correct hybrid model. To selecting the optimal data sampling method and collecting fresh data samples, mining valuable features, dealing with the weight of features in the training feature set and choosing the right training model are superior in performance. To sum up, the main contributions of this paper are as follows:
(1) We propose a data balancing method, in which the negative samples are clustered by K-means algorithm to make the feature data relatively balanced. By mining association features by association rules, the problem of cold start of many users is solved.

(2) We propose a hybrid model which combines XGBoost model and LR model, which solves the shortcomings of LR model in feature selection. A scalable tree enhancement system is constructed to train the initial feature set by forming a strong classifier from several weak classifiers.

(3) We through feature reconstruction, the new feature and the original feature are stitched together to get a new training set. Experiments show that the hybrid model has better predictive effect and higher accuracy than the single model.

The structure of this paper is organized as follows. In section 2, we describe the main methods of this paper, the construction of hybrid model and the training process of the proposed model. In Section 3, we introduce the source of the data set, feature extraction and data preprocessing, as well as the experimental comparison and results analysis. Finally, the limitations of the proposed method are discussed and the future research is summarized in section 4.

2. Methods

2.1. Construction of hybrid model

This paper combines XGBoost[15] and LR algorithm to predict the purchase behavior of users, and compares our proposed hybrid algorithm with the traditional GBDT + LR hybrid model and some single models. The rank_avg[16] method is applied in hybrid method:

\[ \sum \text{weight} / \text{rank} \]  

(1)

where weight represents the weight of the model, of which the value of 1 represents the average fusion, and rank represents the ascending order of samples, defining the sample weight according to the user's click, browse, add shopping cart and purchase. In other words, the more front the sample is, the more front the fusion sample is and the faster the difference fusion among several model by rank but not weighted probability value of samples. For model training, this paper extracts five data subsets from the training set randomly by the fixed-seed random-extraction method predicts the test set according to five models trains from XGBoost algorithm, and obtains the final initial training results by the weighted average method in the end. After several adjustments, the method of equal weight average outperforms others.

In general, the hybrid model in this paper is a hierarchical structure, including two basic models XGBoost and LR, defined as model1 and model2, respectively. First of all, Eq (2) trains the training set based on model model1 to obtain new class features \( y_{true} \), and train model2 by stitching the new features and the original features together via feature reconstruction to obtain the predicted value T training:

\[
\begin{pmatrix}
\cdot \\
\cdot \\
\cdot \\
\end{pmatrix}
\xrightarrow{\text{model1_train}}
\begin{pmatrix}
\cdot \\
\cdot \\
\end{pmatrix}
\]

(2)

\[
\begin{pmatrix}
\cdot \\
\cdot \\
\end{pmatrix}
\xrightarrow{\text{model2_train}}
\begin{pmatrix}
\cdot \\
\cdot \\
\end{pmatrix}
\]

(3)

Complete features can determine the effect of the model, while representing data as features in a better way is the key to improve the performance of the model. If the data is expressed as linearly separable data, the model can perform better. The main process of hybrid model method is shown in
Figure 2. Firstly, the XGBoost model is trained with the initial relative balanced features, where each weak learner represents a branch tree. Then, new features are constructed by the tree learned from XGBoost model. The new constructed eigenvector is binary, of which each element corresponds to the leaf node of the tree in XGBoost model. When a sample point passes through a certain tree and ends up on a leaf node of this tree, the element value corresponding to this leaf node in the new eigenvector is 1, while the element value of other leaf nodes of this tree is 0. The length of the new eigenvector is equal to the sum of the number of leaf nodes contained in all trees in XGBoost model. As shown in Figure 1, the structure diagram of the hybrid model, the input features are transformed by enhancing the decision tree. The output of each leaf node is regarded as the input feature of sparse linear classifier. However, it should be noted that the output of XGBoost model is the index of leaf knot, where the new feature \( y_{true} \) has to encoding in the one-hot manner and then all the one hot types are trained on the LR model after being spliced with the original class features.

Figure 1: hybrid model flow chart

The boosting tree model is a decision tree model. The purpose of the new tree is to fit the residual error of the previous tree. First of all, we get a relatively balanced initial training set by eliminating outliers, filling in missing values and processing by K-means algorithm. This paper consider a train set with about 20 million samples, and about 190 original features and extracted joint features. In this paper, the data set is set as \( \{x_i \in \mathbb{R}^n, \ i \in (1,n) \} \) and XGBoost is an additive model composed of K tree models. The predicted values of K weak learners are accumulated to fit the model. The specific predicted values are as follows Eq (4):

\[
\hat{y}_i = \sum_{k=1}^{n} f_k(x_i)
\]

(4)

where \( f_k \in \{f(x) = w_{q(x)} \} \) is the function space constructed by all regression models, \( q(x) \) is the function of the eigenvector \( x \) mapping the index of each leaf node for each parameter, \( T \) is the sum of the number of leaf nodes corresponding to one of the decision trees, and \( W \) is the weight vector corresponding to the leaf nodes. Therefore, the \( f_k \) of each tree corresponds to a tree structure eigenvector \( Q \) and a weight vector corresponding to leaf nodes \( W \). When generalizing the objective function, we know that \( y_{true}^{(t-1)} \) is the prediction value of the previous T-1 ensemble learner on the sample. Combining with Eq (4), the corresponding the t-th tree can be written as Eq (5):

\[
\hat{y}_i^{(t)} = \sum_{k=1}^{n} g_k(x_i) = \hat{y}_i^{(t-1)} + g_t(x_i)
\]

(5)
In order to prevent overfitting, this paper adopts the shrinkage method. The idea of this method is to add an attenuation factor to Eq (5), and the model formula of decision tree can be written as Eq (6):

\[
y^{(r)}_j = \hat{y}^{(r-1)}_j + \lambda f_r(x_j)
\]

2.2. Feature extraction

In this experiment, we adopt the behavior data of 20000 Taobao users in one month [17], consisting of two parts: the first part is about 20 million samples of mobile terminal behavior data of 20000 users in the commodity collection from November 19 to December 18. While the second part is Commodity subsets. As for the construction of the initial feature set, the prediction subject in this paper is to predict the purchase of all <user, commodity> pairs. In fact, some purchases in the original data set are made by users according to their needs on that day, for which there was no historical behavior data. These data will obviously reduce the prediction effect. Therefore, this paper only considers the <user, commodity> pairs with historical behavior data in this month. Because it is not sure whether the user buy the product, and the purchase time, it will not only lead to a huge price to pay in estimating the impact of the above factors by learning the user’s historical behavior data, but also affect the prediction accuracy. Therefore, in this paper, we focus on these <user, commodity> pairs with historical interactive information and predicts whether the user will purchase the commodity on the specified date. Further, this paper classifies the features into three categories: user features, commodity features and joint features. User features mainly reflect the purchase habits of users, such as the number of views, additional purchases, active days, etc., which are not related to specific commodities. Commodity features reflect the nature of the product itself, the quantity of sale, the frequency of browsing, and so on, which are not related to specific users. Joint feature reflects the users’ interest in commodities, which has a great impact on the prediction results and the accuracy of the model. The main idea and process of the feature construction are as follows:

1. According to the analysis of the original data, it is found that the influence of user behavior on the purchase will be weakened with time. Due to the small influence of user behavior before ten days on whether to purchase on the inspection day, we only consider the characteristic data within ten days from the survey date. At the same time, there is some abnormal data that some users have a large amount of browsing (more than 1 million), which has exceeded the normal browsing level of ordinary users. It is no doubt that these users will increase the amount of training and affect the training effect of the model. Therefore, this paper filters out these abnormal users directly.

2. Since the data sets in this paper come from Taobao e-commerce, which has the characteristics of online purchase and offline consumption, indicating that the purchase behavior of users has a certain periodicity, we set the initially duration as seven days in this paper. Further, since the expected target survey day is December 19, which is Friday, the period of November 19 to December 18 is divided into three sections, with Friday as the observation day and seven days as a duration. The details are as Table 1:

| Partition | Category | Date             |
|-----------|----------|------------------|
| Part1     | TrainSet | 11.22-11.27-11.28|
| Part2     | ValidationSet | 11.29-12.04-12.05|
| Part3     | TestSet  | 12.13-12.18-12.19|

3. According to the current training sets and feature sets, we executes the feature construction from three basic dimensions of category: user, item, item_category, which are referred to as U, I and C, and their combination. At the same time, U-I, U-C, I-C are constructed as the basic joint features, where U-I is the user-commodity pair, U-C is the user-category pair, and I-C is the commodity-category pair.

4. Because the final prediction results of this paper is whether the purchase of the user-item occurred and the classification of label (0,1), the final feature data have to be transformed into the sample set with U-I as index. Due to taking all possible U-I combinations into consideration will lead to insufficient memory and data explosion, this paper only considers the U-I combination sets occurred within seven
days from the investigation day. The same is true for the construction process of U-C and I-C. With the large sample size, in order to fully consider the correlation among samples, this paper designs more than 100 features for data preprocessing.

3. Comparative analysis of experiments
The experiments is carried out on a computer based on Python 3.7. In this paper, we adopt the classical Precision, Recall and F1_score are used in this paper [29] as the evaluation indicators. The specific formula is as follows:

\[
\text{Precision} = \frac{|\text{PredictionSet} \cap \text{ReferenceSet}|}{|\text{PredictionSet}|}
\]

\[
\text{Recall}_\text{rate} = \frac{|\text{PredictionSet} \cap \text{ReferenceSet}|}{|\text{ReferenceSet}|}
\]

\[
F1\_score = 2 \times \frac{\text{Precision} \times \text{Recall}_\text{rate}}{\text{Precision} + \text{Recall}_\text{rate}}
\]

where the PredictionSet is the users purchase data set predicted by the model, the ReferenceSet is the user’s real answer purchase data set, F1_score is the final evaluation standard, and AUC (area under the ROC curve) is the only evaluation standard of the model. In the beginning, single model LR was used. Because of the sensitivity of LR model for the balance of positive and negative samples, the negative samples are clustered in K-means, by sampling the subsamples on each cluster to obtain comprehensive negative samplings. Then, the optimal ratio of positive and negative samples (n/p\_ratio) can be selected after adjusting parameters by applying down sampling on the basis of K-means. Figure 2 shows f1_score versus n/p\_ratio of LR model during training:

![Figure 2: F1\_Score chart of LR model](image)

As shown in Figure 2, F1\_Score is used in this paper to measure the prediction effect, and the results of LR model in some N/P\_ ratio are relatively good. In this paper, we adopt N/P\_ratio as 55 and 75 respectively. Setting the threshold parameter of the predicted sigmoid function cut\_off as variables, the changes of F1\_score of LR model on verification sets are shown in Figure 3.
As shown in Figures 5, the optimal value of cut_off varies with N/P_ratio. When the N/P_ratio is small, the feature space of the positive examples is relatively small. As the results, cut_off should be set as a smaller value, which can compress the prediction space of positive samples and reduce the deviation to achieve better results; vice versa. However, due to the randomness of data, nonlinearity and missing values and other related factors, the LR single model is difficult to achieve better results, of which the results are not satisfactory.

GBDT is a boosting model based on decision regression tree, while XGBoost is an improvement of GBDT, which the core idea is to fit the boosting process based on the residual of previous gradient regression, and reduce the deviation iteratively. XGBoost needs to adjust a lot of parameters during training in order to obtain the best parameter model suitable for the current feature set. In this paper, parameters are classified into two categories: process parameters and basic learner parameters. This paper adopted the heuristic greedy parameter adjustment method, alternative adjusting process parameters (learning rate, number of basic learning units, etc.) and parameters of basic learner (tree depth, leaf split sample number, etc.) iteratively to get better parameter combinations. In addition, the parameter search function only searches one parameter with the fixed other parameters, and adjusts the next parameter in the same way after getting the optimal value of one parameter. Figure 4 shows that the change curve of the verification set F1_score versus n_estimators with different learning_rate during parameter adjustment. It can be seen from the Figure 3 that too large learning rate leads to poor fitting effect or even divergence, while too small learning rate leads to too slow fitting; in addition, one can see in this paper that when the number of iterations increases to a certain number, the effect of continuous iterations have no significant improvement for the model. As mentioned, we can select a suitable n_estimators to obtain the optimal values of all parameters.
After fitting the XGBoost model well, the new features can be obtained by combining the new features from training with the original features. Through the model training, this paper draws the F1_score of XGBoost+LR model and XGBoost model versus cut_off with N/P_ratio=55, as shown in Figure 5.

![F1_score as function of n_estimators and learning rate of GBDT model](image1)

**Figure 4:** The change of F1_score with n_estimators under different learning_rates of GBDT model

From the results shown in Figure 5, the overall F1_score of the hybrid model is better than single model. Although the effect of the hybrid model may not be better even if the XGBoost model is well fitted originally, the hybrid model of XGBoost + LR is relatively good with the relatively large training samples, especially in the recommendation and prediction of e-commerce. The comparison of experimental data is shown as follow:

![F1_score changes for XGBoost and XGBoost&LR](image2)

**Figure 5:** The F1_score changes for XGBoost and XGBoost&LR

| Model      | AUC   | F1-score |
|------------|-------|----------|
| LR         | 0.9097| 0.08304  |
| RF         | 0.8977| 0.06195  |
| GBDT       | 0.9259| 0.08183  |
| XGBoost    | 0.9354| 0.08985  |

**Table 2:** Comparison of experimental results
After fitting the XGBoost model well, the effect after adopting the fusion of XGBoost + LR model might not be better. On the contrary, the effect of XGBoost + LR model tends to get worse easily while the data sample is not large enough. Cause the original Ali data set used in this paper is relatively large with many features, the fusion model outperforms the single model. From Figure 6, we can see that the AUC of the mixed model is better than the single model generally. There are many advantages to adopt XGBoost + LR fusion model.

Compared with the traditional GBDT, considering the low efficiency of the traditional greedy algorithm, XGBoost adopts an approximate tree greedy algorithm to find the optimal break point, which can speed up and reduce memory consumption. In addition, faced with the processing of the sparse data sets and missing values, XGBoost model can find the direction of node splitting for the samples with missing eigenvalues. The main function of the XGBoost model is to explore distinguishable features and feature combinations, as well as construct the new cross features by combining the effective features with relevant information, to improve the effect of the model. While nodes splitting, each sample feature and cross feature fall on the specified leaf node of the trained model.

However, as a linear classifier, LR model cannot achieve feature combination, which is very important in model training. In this paper, the numerical scale of the original features is different and have to be normalized before training; while some features are discrete features (sorting features) or missing values (time difference features), which have to be preprocessed by XGBoost model.

### 4. Conclusion

According to the online and offline historical behavior data of e-commerce users and the combination analysis of various related characteristics, this paper establishes a prediction model of user online purchase behavior by combining market demand and business analysis. In real business scenarios, in order to increase the turnover, we build a personalized forecasting model for a subset of all goods. In the process of building the model, this paper needs to use not only the user behavior data on the complete set of commodities, but also more abundant user behavior data. This paper obtains the behavior data of 10 million users within a month (November 18 - December 18), and predicts the products that users will buy on December 19. For offline simulation, this paper divides the data into three parts (training sets, verification sets and test sets) in the feature construction phase, and extracts four segments of behavior.
data from November 18 to December 18 for feature extraction, setting the Friday of each segment as the evaluation day. To mark label for the feature set according to the real situation of the evaluation day, for a \(\text{<user, commodity>} \) pair, we set the field of label for the feature table as 1 if the user actually purchased the product on the inspection day, otherwise set it as 0. Validation sets are executed in the same manner. In order to ensure the stability of the model, using the model obtained by the training sets, the maximum difference of AUC between the test sets and the verification sets is 0.04, and the F1 value is about the same (the difference is about 0.02), which shows that the model is reliable and the feature sets is effective.

From the actual training situation of each model, the division of data sets has a great impact on the results. In this paper, we use the training model of feature data of 1 day before, 3 days before and 6 days before to predict the situation of the evaluation day. The impacts turn out to be small with the low feature dimension. However, when the feature dimension is increased to 300 dimensions, the difference of F1 tends to be quite large, which may be caused by different the time span of feature extraction. Furthermore, the added features have a large relationship with the time span, which leads to the large deviation of model prediction.

Although the work of this paper has achieved some advantages in feature extraction and model construction, it only predicts the purchase habits of users in a short time. For the large span training sets with high probability uncertainty, we need to mine more collaborative features in the process of feature extraction, which are directly connected with the test set and play a decisive role in the prediction results. The GBDT+ LR and XGBoost + LR hybrid models used in this paper both are, the LR model is used for training after feature transformation and original feature splicing by training new features. However, the improvement of XGBoost on the basis of GBDT leads to the improvement of the stability of the whole hybrid model in some degree. For the work of this paper, it is indispensable to be patience and careful for data mining. The problem occurred at any step in the process, such as the data preprocessing, the feature extraction and connection, and the offline training prediction, may lead to unexpected results. Take it into consideration, we determine a framework in advance to prevent the premature and blind adjustment of model parameters, and understand the learning and prediction of the model to prevent the waste of time and energy for debugging parameters, which have been improved, to some extent, by the parameter search function in this paper. The feature extraction determines the theoretical upper bound of final results, while the process of parameter adjustment can only help to approach the upper bound. Therefore, it is necessary to keep the distribution of features on the training sets and prediction sets consistent, otherwise it may cause a large deviation between online and offline. At the same time, we should make good use of the advantages of each model, which can save a lot of unnecessary work. The factors and views mentioned above could be taken into account in future work.

References
[1] Cai, X.; Hu, Z.; Zhao, P.; Zhang, W.S.; Chen, J. (2020) A hybrid recommendation system with many-objective evolutionary algorithm. Expert Systems with Applications , 159, 113648.
[2] Zhao, Z.D.; Shang, M.S. (2010) User-Based Collaborative-Filtering Recommendation Algorithms on Hadoop. Third International Conference on Knowledge Discovery and Data Mining, WKDD 2010, Phuket, Thailand, 9-10 January 2010.
[3] Sarwar, B.; Karypis, G.; Konstan, J.; Riedl, J. (2001) Item-based collaborative filtering recommendation algorithms. Proceedings of the 10th international conference on World Wide Web, pp. 285–295.
[4] Mcculloch, C.E.; Neuhaus, J.M. (2005) Generalized Linear Mixed Models; John Wiley & Sons, Ltd.
[5] Buckman, J.; Roy, A.; Raffel, C.; Goodfellow, I. (2018) Thermometer encoding: One hot way to resist adversarial examples. International Conference on Learning Representations.
[6] Xu, J.; Li, H. (2007) AdaRank: a boosting algorithm for information retrieval. International Acm Sigir Conference on Research & Development in Information Retrieval.
[7] Dietterich, T.G. (2000) An Experimental Comparison of Three Methods for Constructing Ensembles of Decision Trees: Bagging, Boosting, and Randomization. Machine Learning, 40, 139–157.

[8] Liaw, A.; Wiener, M. (2002) Classification and Regression by randomForest. R News 23.

[9] Silva, L.R.; Corrêa, E.C.; BrandO, J.R.; de ávila, R.F. (2013) Environmentally friendly manufacturing: Behavior analysis of minimum quantity of lubricant - MQL in grinding process. Journal of Cleaner Production, p. S0959652613000383.

[10] Chakraborty, S.; Paul, D.; Das, S.; Xu, J. (2020) Entropy weighted power k-means clustering. International Conference on Artificial Intelligence and Statistics. PMLR, pp. 691–701.

[11] Yu, X.; Sun, X.; Zhao, Y.; Liu, J.; Li, H. (2020) Fault detection of continuous glucose measurements based on modified k-medoids clustering algorithm. Neural Computing and Applications, pp. 1–10.

[12] Ke, G.; Xu, Z.; Zhang, J.; Bian, J.; Liu, T.Y. (2019) DeepGBM: A deep learning framework distilled by GBDT for online prediction tasks. Proceedings of the 25th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining, pp. 384–394.

[13] Chen, T.; Guestrin, C. (2016) XGBoost: A Scalable Tree Boosting System.

[14] Bukhari, T.; Khan, A.; Shah, H.; Bhagat, B. (2014) FACEBOOK USER ADVERTISEMENT CLICK PREDICTION.

[15] Yu, B.; Qiu, W.; Chen, C.; Ma, A.; Jiang, J.; Zhou, H.; Ma, Q. (2020) SubMito-XGBoost: predicting protein sub mitochondrial localization by fusing multiple feature information and eXtreme gradient boosting. Bioinformatics, 36, 1074–1081.

[16] Ghedini, D.F.; Castilho, A.M.; Resende, R.L.C.D.; Leal, J.S.; de Andrade, M.A.P. (2016) Full-endoscopic lumbar discectomy. Coluna/columna.