Tech Science Press

Improving Recommendation for Effective Personalization in Context-Aware Data Using Novel Neural Network

R. Sujatha[1,*] and T. Abirami[2]

[1]Department of Information Technology, M. Kumarasamy College of Engineering, Karur, Tamilnadu, India
[2]Department of Information Technology, Kongu Engineering College, Perundurai, Tamilnadu, India
*Corresponding Author: R. Sujatha. Email: vasishtsarathi@gmail.com
Received: 21 April 2022; Accepted: 03 November 2022

Abstract: The digital technologies that run based on users' content provide a platform for users to help air their opinions on various aspects of a particular subject or product. The recommendation agents play a crucial role in personalizing the needs of individual users. Therefore, it is essential to improve the user experience. The recommender system focuses on recommending a set of items to a user to help the decision-making process and is prevalent across e-commerce and media websites. In Context-Aware Recommender Systems (CARS), several influential and contextual variables are identified to provide an effective recommendation. A substantial trade-off is applied in context to achieve the proper accuracy and coverage required for a collaborative recommendation. The CARS will generate more recommendations utilizing adapting them to a certain contextual situation of users. However, the key issue is how contextual information is used to create good and intelligent recommender systems. This paper proposes an Artificial Neural Network (ANN) to achieve contextual recommendations based on user-generated reviews. The ability of ANNs to learn events and make decisions based on similar events makes it effective for personalized recommendations in CARS. Thus, the most appropriate contexts in which a user should choose an item or service are achieved. This work converts every label set into a Multi-Label Classification (MLC) problem to enhance recommendations. Experimental results show that the proposed ANN performs better in the Binary Relevance (BR) Instance-Based Classifier, the BR Decision Tree, and the Multi-label SVM for Trip Advisor and LDOS-CoMoDa Dataset. Furthermore, the accuracy of the proposed ANN achieves better results by 1.1% to 6.1% compared to other existing methods.

Keywords: Recommendation agents; context-aware recommender systems; collaborative recommendation; personalization systems; optimized neural network-based contextual recommendation algorithm

1 Introduction

The recommender systems have now attained acceptance and attracted the attention of the masses for more than a decade. It can ease the complexities of the products and the various service selection tasks

and helps overcome information overload issues [1]. The recommender systems also collect information on user preferences and sift through large volumes of information scattered over the internet to choose the information best suited to the user preferences. Normally, these recommender systems tend to get information from users either in an explicit manner or implicitly. All extracted information using user ratings of different items is taken to be explicit extraction of information. The information obtained by user behaviour observation when interacting with recommender systems is known as the implicit extraction of information. This ubiquitous information processing can demand recommender systems with scalable information retrieval techniques and context-aware recommendations.

The CARS are nothing but the traditional recommenders' extensions that consider the contextual condition of users to whom such recommendations are made. The recommendation problem is still focused on recommending certain items to the target user. Most CARS research focuses on how this issue can be addressed, especially while integrating contextual factors into the recommendation process. Several other context-aware recommendation algorithms were developed in various domains of applications. It was demonstrated that effective recommendations were made when the context was considered during the recommendation. This can be in the domain of mobile apps, restaurants, tourism, music, or movies. However, as in the case of the other traditional recommender systems, the primary goal of the CARS is recommending an item (or user for the social recommendation) to the target user. There are different situations wherein the user's final decision is on whether the decision to choose the item depends on the context of its consumption. Therefore, identifying a suitable contextual situation for users in connection to a certain item may give important information for selecting.

The concept of multi-label learning concerns the problems in which every example has been represented using one single instance associated with several class labels simultaneously. The learning goal of multi-label is to induce a new multi-label predictor to assign a certain set of relevant labels to the unseen instance. To achieve this, an intuitive solution was to learn a binary classifier for every such class label, with the relevance of every class label determined by predicting the corresponding binary classifier [2]. BR can be the most intuitive among the solutions used for learning from examples. This works by decomposing multi-label learning tasks into different tasks of independent binary learning (one for each class label). The easiest approach for solving the MLC problem was the decomposition into different sets of classification problems for every label. However, the solution has a major disadvantage: it does not consider dependencies between various categories. Therefore, a different approach is used. Contrary to the conventional classification (for instance, the binary-class or Multi-Class Classification), the MLC will facilitate multiple predictions on various classes at the same time–the examples have an association with a set of labels $Y \subseteq L$, in which L will be a set of all possible labels, and also will be composed of at least two labels. It has extensive applicability in annotation applications, for instance, image annotation and text categorization. It is currently employed for tagging recommendations to acquire accurate predictions with a reasonable degree of personalization.

Support Vector Machines (SVM) is a machine learning method used to classify data. The SVMs provide a good performance of generalization on various problems. The ANN is a good example of supervised learning. The ANN acquires knowledge as a connected network unit. It can be challenging for humans to extract such knowledge. This has motivated the extraction of the classification rule for data mining. The process of classification begins with the dataset. The data is grouped into two, which are the training sample and the test sample. The former is used to learn the network, and the latter is used to measure classifier accuracy. Dividing the dataset is made by different methods like the hold-out method, random sampling, and cross-validation. ANNs were a universal approximation in numerical paradigms owing to their properties of advancement in input-to-output mapping, nonlinearity, fault tolerance, adaptive nature, and self-learning [3]. Context-aware data for the neural network is proposed in this paper.

The contributions of this work are:

- The CARS is considered to identify several influential and contextual variables to be effective in their recommendation.
- Classifiers such as BR Instance-Based Classifier, BR Decision Tree, Multi-label SVM, and ANN are considered.
- An ANN based on the contextual recommendation algorithm is proposed.

The remainder of the paper is organized thus. Section 2 reviewed the literature. Section 3 explained the techniques used. Then, the results are explained in Section 4. Finally, the conclusion of the study was completed in Section 5.

2 Related Works

Zhang et al. primarily aimed at reviewing BR for this based on three perspectives. Firstly, the basic settings for this multi-label learning with BR solutions were summarized. Next, representative strategies for providing BR with the label correlation abilities of exploitation were discussed. Thirdly, recent studies on BR were introduced on issues aside from label correlation exploitation. Finally, Haruna et al. [4] reviewed certain developmental processes in a fountainhead for research used in CARS. The work contributes to this by using an integrated approach for completing the process of CARS development, unlike the other review papers that address a certain aspect of CARS. Firstly, an in-depth review of classified and state-of-the-art literature was presented that considers the different application models, evaluation approaches, extraction, and filters. Next, the viewpoints are presented for literature extraction, analyzing their merits and demerits. Lastly, the outstanding challenges and opportunities used for future research were highlighted.

Aliannejadi et al. [5] introduced a dataset on the contextual suitability of locations and demonstrated the usefulness of predicting the locations and their contextual reference. Four different approaches were investigated to address the problem of data sparsity. One model reduced the location and its dimensionality for taste keywords, with three models for predicting the user tags for the new location. Furthermore, different scores were computed from many Location-Based Social Networks (LBSNs) that show how new information from mapping into the Points of Interest (POI) approach of recommendation is made. The computed scores are obtained and then integrated with the learning of rank techniques. These experiments were made on two different TREC datasets to prove their effectiveness. Zheng et al. [6] considered the problem of recommending suitable contexts where users choose the items. Context recommenders were used in the form of other tools that help users in decision-making. The context recommendation problem is formulated to discuss the motivation behind the applications of this concept. Two general classes of algorithms were identified to solve this: direct context prediction and indirect context recommendation. Also, other direct prediction algorithms were recommended using the Multi-Label Classification. The experiments demonstrated that the method outperformed other baseline methods, and to enhance this, personalization was needed for context recommenders.

Knoch et al. [7] proposed a mobile Running Route Recommender system (RRR) to support users planning a running route. Both the gathering and modelling of routes are discussed based on computational performance. A four-dimensional ranking function was based on the plugin that established the location-time, content-time, and community-specific routes for covering the proposed database. In addition, there was a conceptual model to depict the runner's physical condition by predicting the heart rate for the routes. Thus, ANNs are selected as the data mining methodology for the currently existing recommender systems.

Tanaka et al. [8] suggested a new adaptation for a BR algorithm considering the relations among labels that focus on the model and its interpretability. Experiments were conducted to compare, and this approach was found in the literature and duly applied to the genomic datasets. The experiment results proved that the proposed method performed better than other methods and provided another interpretable model from multi-label.

Luaces et al. [9] discussed interesting BR properties that produced optimal models for the multi-label loss functions. In addition, another analytical study for the multi-label benchmarks was used to point out the shortcomings. The paper also proposed using synthetic datasets for analyzing the multi-label methods and their behaviour using other characteristics. To support the claim, some experiments with synthetic data were conducted related to complex and difficult problems with different labels not mentioned in earlier studies.

Zhang et al. [10] proposed a recommender algorithm based on Instance-based Learning (IBL), which effectively handled an imbalanced dataset. Firstly, a meta-knowledge database was created using binary relation data traits that measure the ranks of candidate imbalance for datasets. Next, the traits are extracted and compared to the instance-based k-nearest Neighbour's algorithm whenever there is a new dataset. Lastly, an appropriate imbalance method of handling was derived through a combination of the recommended rank and individual bias. The experiment results were based on 80 datasets of public binary imbalance for a recommendation algorithm for an appropriate method of handling a given imbalance learning problem. This was with a rate of recommendation of 95%.

Wang et al. [11] proposed a CARS based on an end-to-end memory network for representations of papers and citation contexts. A Bidirectional-Long Short-Term Memory (Bi-LSTM) was used to find the relevance of the citations. A vector representing the author's information and citation contexts is created. The relevance between them is computed continuously using a multilayer memory network of Bi-LSTM.

Thaipisutikul et al. [12] proposed deep learning CARS for capturing the dynamic preferences of the users. First, hierarchical relationships between contexts and items in the short and long term are identified. Then a neural attentive bi-directional GRU network identifies highly related items in the short-term session. Finally, both short and long-term relevant items are considered to form the final user representation.

Mei et al. [13] proposed an Attentive Interaction Network (AIN) to enhance recommendations using CARS. The proposed method captures the relations between the user and context adaptively. The AIN is made up of Interaction-Centric, User-Centric and Item-Centric Modules for capturing the interaction between users and context. Based on the interactions, the recommendation scores are predicted.

Wang et al. [14] presented Content and context Aware Music Embedding (CAME) for obtaining the low-dimension dense real-valued feature representations of music pieces from HIN. The users' general music favorites and their contextual preferences for music were obtained, and CAME gave recommendations.

Paliwal et al. [15] designed a recommendation system based on the fusion of sentiment analysis and gradient boosting. The user reviews are used for finding the polarity of the sentiments, which is then used as input for gradient boost to generate the drug recommendation.

It is observed from the reviews that deep learning methods are effective in finding recommendations. Thus, in this work, the efficacy of the ANN is used to build the recommender system. The influential variables and context are identified by the proposed method to give recommendations.

3 Methodology

For context prediction, the data is modelled as: <User × Item × Rating → Contexts>. The contexts are labels that correspond to a set of contextual conditions and are directly predicted based on the rating profile tuples {user, item, rating} using classification techniques. It is assumed that the recommendations for the

given context are taken from the predicted context labels. MLC is accounted for as a method of mapping from a single sample to multiple labels. These multiple labels will be part of the same label set, wherein the labels are inconsistent. The MLC's objective is to construct a classification model for unseen samples. The primary goal of multi-label classification was inducing models to tag objects with labels to describe them. The BR was a decomposition method based on the assumption that all labels were independent.

3.1 Dataset

TripAdvisor data is an enriched hotel rating data [16,17], with rating profiles scripted from the hotel reviews. The data was sparse since few users were in the hotel for different contexts and didn't always leave ratings. Some instances of the reviews used in this work are:

"great stay great stay, went seahawk game awesome, downfall view building didn't complain, room huge".

"nice hotel not nice staff hotel lovely staff quite rude, bellhop desk clerk going way make things difficult".

"disappointed say anticipating stay hotel monaco based reviews seen tripadvisor, definitely disappointing".

The LDOS-CoMoDa [18] refers to a context-rich dataset of movie recommenders. It is a context-aware movie dataset obtained from surveys. It has movie ratings with twelve different pieces of contextual information for the situation of the movies. Certain important properties of datasets are contextual information and ratings explicitly from users when the user views and contextual information describing the situation. Based on past interactions, all ratings and contextual information were not from a hypothetical situation or the user's memory. The users could rate this item several times if they could view it several times. A dataset acquired for the proposed research was based on CARS.

Some instances of the reviews used in this work are:

"It's a funny, fantastic movie with the Avengers dealing with internal conflict (not least the fact that Tony and Banner basically created said enemy). Also, it is a film which will have lasting and meaningful consequences on the MCU".

"Big, brash, and overcrowded, but entertaining enough to satiate fans".

3.2 Classifiers

3.2.1 Binary Relevance Instance-Based Classifier

BR is referred to as an approach that handles the task of MLC. Compared to the new multi-label methods, BR is normally employed as a baseline method. It is a very simple strategy that is very effective. BR can induce certain optimal models when the target loss function is a macro-averaged measure. Furthermore, it can support the hypothesis that a BR is quite competitive in the complex approaches where the tasks of MLC are challenging in domains with several labels and dependencies.

The BR will decompose the learning of h into tasks of binary classification for each label in which every single model h_j. will be independently learned with information on that label by ignoring information on all the other labels. For the symbols, $h_j: \chi \rightarrow \{0, 1\}$. The BR's primary drawback was that it considered label dependency and failed to predict certain combinations where dependence was present. However, the BR has various other advantages (i) a binary learning method is considered as the base learner; (ii) it has a linear complexity for the labels, and (iii) it is parallelized very easily [19]. However, the main advantage of the BR was that it could optimize various loss functions. With a multi-label task of classification D (1), if M is the performance measure for a pair of subsets of labels: $M(y_1, \ldots, y_n), (\hat{y}_1, \ldots, \hat{y}_n) = M(Y, \hat{Y})$, wherein Y,

and $\hat{Y}$ lists of the subsets of both actual and predicted labels. If the higher values of M are preferred, optimal predictions for the input lists X will be given using a hypothesis.

$$h\sum\nolimits_{M}^{*}(X) = \underset{\hat{Y}}{argmax}\sum\nolimits_{Y} Pr(Y|X).M\left(Y,\hat{Y}\right) \tag{1}$$

Optimization of the measures that are macro-averaged can be equal to the optimization of measures for subordinate BR classifiers. If M is a micro-averaged measure, optimization of (2) is decomposed with a set of labels. Therefore,

$$h_{M}^{*}(X) = \underset{\hat{Y}}{argmax}\frac{1}{L}\left(\sum\nolimits_{j=1}^{L} M\left(Y[j],\hat{Y}[j]\right)\right) = \underset{\hat{Y}}{argmax}\sum\nolimits_{Y}\sum\nolimits_{j=1}^{L} Pr(Y|X)M\left(Y[j],\hat{Y}[j]\right) \tag{2}$$

where Y[j] is the jth column of matrix Y representing its corresponding label, l_j.

3.2.2 Binary Relevance Decision Tree

The traditional algorithms cannot always handle multi-label instances, as these algorithms had been designed to predict a single label. A solution for this was to transform its original dataset into different datasets in which all the sets can contain attributes, and only a single label is predicted. The algorithm is called BR. These studies showed that the approach was not a proper solution as every label has been individually treated, thus generating a classifier for every label. The algorithm will identify another classifier for every label in an intuitive manner capturing correlations among them for which a simpler classifier may be identified (one that has a smaller number of rules). Under such circumstances, it is crucial to develop techniques using the BR algorithm that extends it to get relations among the labels [20].

Another new adaptation of this BR algorithm that uses Decision Trees was presented for treating multi-label problems. The decision trees were symbolic learning models analyzed to set rules for improving understanding by human experts. For the same reason, the proposed algorithm w designed to capture relations among labels; a feature of the original BR algorithm will not be considered for upgrading generalization ability. Also, as the current study has model interpretability and not just performance, the proposed approach reduced the induced trees for its expert interpretation: in this scenario, only one model (tree) will classify the labels [21].

The steps for a BR with Decision Trees are as below:

1. It performs induction of c decision trees considering the labels alone.
2. For every label, the decision tree will be induced.
3. All C trees induced are converted into G, the initially empty graph structure. Let C (G) be the number of connected components of graph G.
4. If C(G) is the connected component of graph G, the first step attempts to find the related labels represented by G, the connected component. This will not consider the directionality of the edges, and an undirected graph will be built. Finally, three possible solutions may occur:
   a. C(G) = 1, all the labels are related to one another, so only a single connected component for G with all labels is available.
   b. C(G)-c, no other label will be related to the other, then G will consist of c connected components; and
   c. 1 < C(G) < c, related to the relationships among certain labels.
5. The final induction of the tree will be conducted considering all the attributes done one label at one time.

6. Once this is done, all decision trees are extended, and the process will be performed so that every tree can produce more than one label. The component that is connected to G with the node is considered. The results for new trees will have a list of labels for every leaf node.

7. In case all the labels are correlated (the first situation), the tree is extended by including the labels on the leaves. Suppose there are more than two connected components (the third situation). In that case, the tree will be extended for the labels that belong to G. In case the second situation occurs, the un-extended tree c is considered for each label.

8. The final step permits choosing the best tree for every component, thereby decreasing the number of trees that have to be analyzed. An algorithm outputs C(G) trees, one for every best tree by the component.

3.2.3 Multi-Label SVM

The SVM is used widely in different research areas, designed in the form of a binary classifier. But there was a set of binary SVMs employed for performing an MLC. Binary SVM classification refers to a problem of optimization that is based on the identification of a hyperplane. An optimal hyperplane can simultaneously reduce the distance among training samples that are part of the same class, increasing the distance between two different classes, known as the margin, with the close samples known as support vectors [22]. A hyperplane is defined in Eq. (3):

$$w \, . \, x + b = 0 \tag{3}$$

Here, w and b are the parameters of the SVM model, and x is the input training data.

When considering an MLC, it is considered an independent binary classification problem via the one-*vs*.-one-all approach. The number of binary SVMs required for MLC will depend on the number of classes and is computed as in (4):

$$(\text{NumClasses} * (\text{NumClasses} - 1))/2 \tag{4}$$

Each binary SVM will predict one class label, and the one-*vs*.-one strategy predicts the model with the most predictions or votes. For a multi-label training set, $D = \{(x_i, y_i)\}$, with x_i being the i-th input feature vector and y_i being the label vector. The y_i indicates the class to which the input is assigned. While defining ξ_{il} to be the non-negative slack variable, c as the trade-off parameter, a multi-label-SVM classifier is written as the optimization for a class as per Eq. (5):

$$\underset{w_l}{\textbf{min}} \frac{\mathbf{1}}{\mathbf{2}} \parallel w_l \parallel^2 + c \sum_{i=1}^{N} \xi_{il}^2 \tag{5}$$

This is subject to $y_{il}(w_1^T x_i + b_i) \geq 1 - \xi_{il}$, $1 = 1, \dots.L$.

N and L refer to the training data with the number of classes [23].

3.2.4 Proposed Artificial Neural Network (ANN)

The primary advantage of the ANNs is that they can make using models easy and accurate using large inputs [24]. ANN is quite novel and useful when applied to machine learning and problem-solving. The elementary building blocks for the ANN were the artificial neurons. The ANNs contain a new collection of neurons that are information processing units that are interconnected. These are organized into the input layer, one or more hidden layers, and finally, an output layer. The input layer neurons will pass the input signal to the hidden layer using synaptic weights. Hidden layer neurons computed a weighted summation of the inputs by applying an activation function to determine if it can be passed to the subsequent layer. So, the process of learning this neuron network will continue by adjusting the weights using the Backpropagation algorithm.

In the artificial neuron, the inputs will be weighted by means of multiplying every input by its weight. For the internal neuron, a function sums the weighted bias and inputs. The output neuron and its summary for the earlier weighted bias and their contributions will be passed over in the form of an activation function. The ANN products with K elements have been depicted in Eq. (6):

$$y(x) = \sum_{i=1}^{k} w_i y_i(x) \tag{6}$$

In this y_i and w_i denote the output and the weights at layer i.

The different contextual attributes are given as the input. Each label and its combination are used during the training of ANN. The optimal set of labels of the context attributes will be identified, and recommendations based on the accuracy are given as output.

4 Results and Discussion

The steps of model deployment in the research are as follows: load an example data set into MATLAB. The various classifiers BR Instance, BR Decision Tree, Multi-label Support Vectors Machine, and ANN are loaded. Then, 10-fold cross-validation is carried out on both TripAdvisor and LDOS-CoMoDa. The training is carried out, and new instances are used for validation. The performance is evaluated using accuracy, recall, and precision metrics.

Accuracy is the ratio of the number of correctly classified instances to the total number of instances in (7):

$$Accuracy = \frac{TP + TN}{TP + TN + FP + FN} \tag{7}$$

where True Positive (TP), True Negative (TN), False Positive (FP), and False Negative (FN).

The recall is defined as the proportion of observed correct instances predicted as correct by a classifier in (8):

$$Recall = \frac{TP}{TP + FN} \tag{8}$$

Precision is the proportion of instances predicted as incorrect that actually are incorrect in (9),

$$Precision = \frac{TP}{TP + FP} \tag{9}$$

Users and their context-dependent preferences were inferred using various classification algorithms. Tables 1, 2 and Figs. 1 to 6 shows the accuracy, precision, and recall for dataset such as Trip Advisor & LDOS-CoMoDa, respectively.

Table 1: Summary of results

Trip Advisor	Accuracy (%)	Precision	Recall
BR Instance-based classifier	70.2	0.7242	0.6844
BR Decision tree	71.5	0.7484	0.7087
Multi-label SVM	73.8	0.7643	0.7344
ANN	74.6	0.7893	0.7594

Table 2: Summary of results for LDOS-CoMoDa

| LDOS-CoMoDa	Accuracy (%)	Precision	Recall
BR Instance-based classifier	71.4	0.7461	0.7095
BR Decision tree	72.2	0.7759	0.7252
Multi-label SVM	74.2	0.8033	0.7534
ANN	75.4	0.8244	0.7792

4.1 Results for Trip Advisor Dataset

Fig. 1 shows that the accuracy of the proposed ANN performs better by 6.1%, 4.2%, and 1.1% than BR Instance-Based Classifier, BR Decision Tree, and Multi-label SVM, respectively, using Trip Advisor Dataset. The results confirm that proposed ANN models can be effectively improved to generate better recommendations related to the user's context.
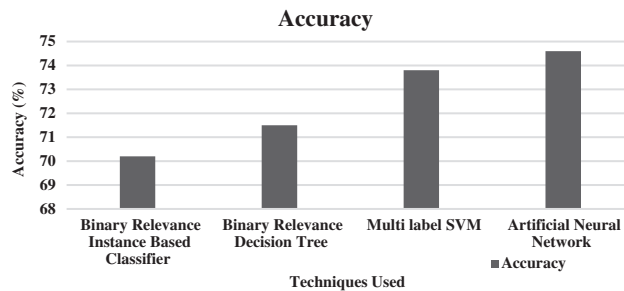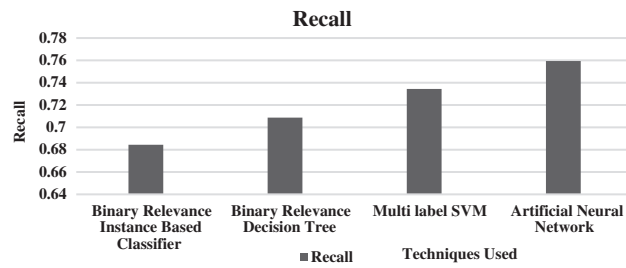

Figure 1: Accuracy for proposed classifier

Fig. 2 shows that the precision of the proposed ANN performs better by 8.6%, 5.3%, and 3.2% than BR Instance-Based Classifier, BR Decision Tree, and Multi-label SVM, respectively, using Trip Advisor Dataset. These results further prove the effectiveness of the proposed strategy.
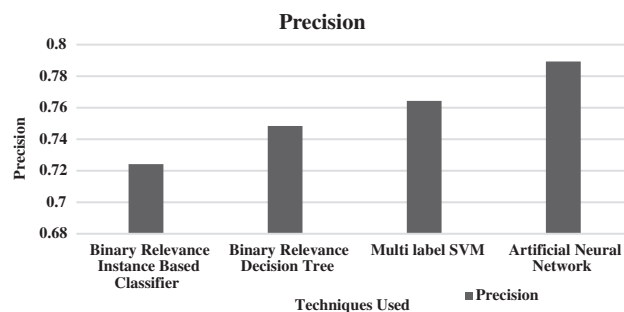
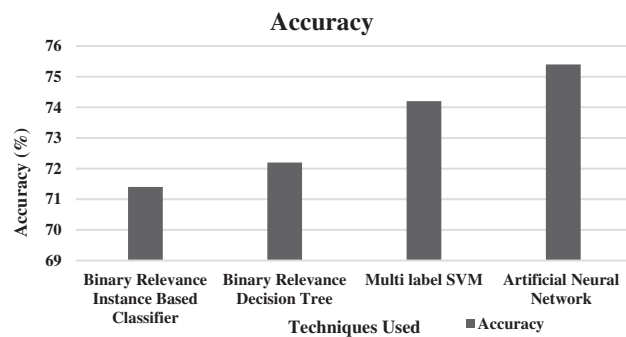**Figure 2:** Precision for proposed classifier

Fig. 3 shows that the recall of the proposed ANN performs better by 10.4%, 6.9%, and 3.3% than BR Instance-Based Classifier, BR Decision Tree, and Multi-label SVM, respectively, using Trip Advisor Dataset.

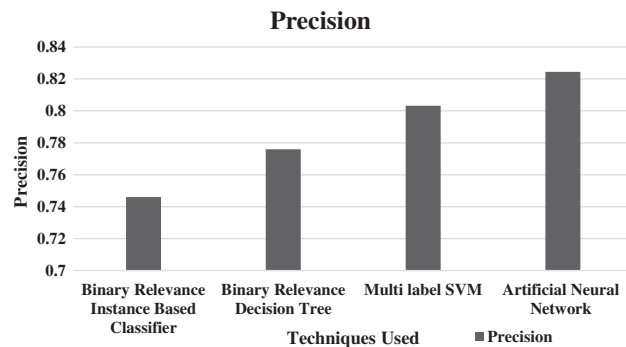**Figure 3:** Recall for proposed classifier

4.2 Results for LDOS-CoMoDa

Fig. 4 shows that the accuracy of the proposed ANN performs better by 5.4%, 4.3%, and 1.6% than BR Instance-Based Classifier, BR Decision Tree, and Multi-label SVM, respectively, using LDOS-CoMoDa Dataset. That is, the proposed ANN outperforms BR Instance-Based Classifier, BR Decision Tree, and Multi-label SVM, indicating that the proposed ANN can effectively utilize the user's history and strive to recommend similar moves without compromising on the accuracy aspect.
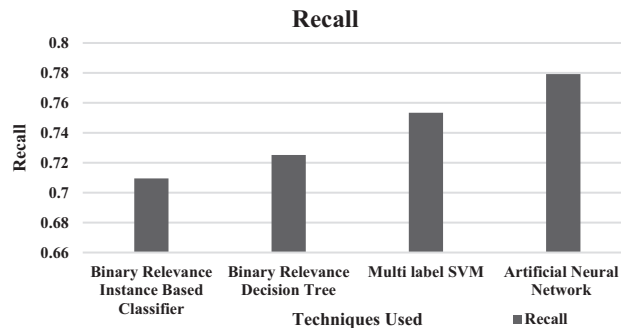

Figure 4: Accuracy for proposed classifier

Fig. 5 shows that the precision of the proposed ANN performs better by 9.9%, 6.1%, and 2.6% than BR Instance-Based Classifier, BR Decision Tree, and Multi-label SVM, respectively, using LDOS-CoMoDa Dataset. This indicates that the proposed ANN remains effective and robust.

Figure 5: Precision for proposed classifier

Fig. 6 shows that the recall of the proposed ANN performs better by 9.4%, 7.2%, and 3.4% than BR Instance-Based Classifier, BR Decision Tree, and Multi-label SVM, respectively using LDOS-CoMoDa Dataset. Based on the user profile and context, the proposed method has a better recall in giving recommendations.

Figure 6: Recall for proposed classifier

The proposed ANN model outperformed the various baseline models like the BR Instance-Based Classifier, BR Decision Tree, and Multi-label SVM on all of the measures (accuracy, precision, and recall). These results indicate that adding context information to the neural model improves recommendation recall. The proposed ANN enhances the accuracy when additional context information is included in the input.

5 Conclusion

The recommender systems are at the leading edge of the systems for leveraging all relevant information, which emerges as an appropriate tool to increase the speed of the information-seeking process. Multi-label learning was a learning framework for real-world model objects with many semantic meanings. The recommender systems are now extensively used to recommend various domains like e-learning, news, and so on. The ANNs were a model for machine learning that was competitive with the statistical models and conventional regression. In this work, additional context information is included in the input, enhancing the recommendations' accuracy. The results proved that the proposed ANN was able to perform better by about 6.1%, by 4.2%, and further by 1.1% compared to the BR Instance-Based Classifier, the BR Decision Tree, and the Multi-label SVM using the Trip Advisor Dataset. Further, the accuracy of the proposed ANN has performed better by about 5.4%, 4.3%, and 1.6% than the BR Instance-Based Classifier, the BR Decision Tree, and the Multi-label SVM, respectively, making use of the LDOS-CoMoDa Dataset.

In future work, we aim to include more available context information in the model learning process. Investigations to improve the structure of the model to make the model less complex and more efficient need to be carried out. The use of metaheuristic methods to improve the ANN structure for enhancing the recommendations needs to be investigated.

Conflicts of Interest: The authors declare that they have no conflicts of interest to report regarding the present study.

References

[1] A. Abbas, L. Zhang and S. U. Khan, "A survey on context-aware recommender systems based on computational intelligence techniques," *Computing*, vol. 97, no. 7, pp. 667–690, 2015.

[2] M. L. Zhang, Y. K. Li, X. Y. Liu and X. Geng, "Binary relevance for multi-label learning: An overview," *Frontiers of Computer Science*, vol. 12, no. 2, pp. 191–202, 2018.

[3] K. Saravanan and S. Sasithra, "Review on classification based on artificial neural networks," *International Journal Ambient System Application (IJASA)*, vol. 2, no. 4, pp. 358–372, 2014.

[4] K. Haruna, M. Akmar Ismail, S. Suhendroyono, D. Damiasih and A. C. Pierewan, "Context-aware recommender system: A review of recent developmental process and future research direction," *Applied Sciences*, vol. 7, no. 12, pp. 1211–1234, 2017.

[5] M. Aliannejadi and F. Crestani, "Personalized context-aware point of interest recommendation," *ACM Transactions on Information Systems (TOIS)*, vol. 36, no. 4, pp. 1–28, 2018.

[6] Y. Zheng, B. Mobasher and R. Burke, "Context recommendation using multi-label classification," *2014 IEEE/WIC/ACM International Joint Conferences on Web Intelligence (WI) and Intelligent Agent Technologies (IAT)*, vol. 2, pp. 288–295, 2014.

[7] S. Knoch, A. Chapko, A. Emrich, D. Werth and P. Loos, "A context-aware running route recommender learning from user histories using artificial neural networks," in *23rd Int. Workshop on Database and Expert Systems Applications*, Vienna, Austria, vol. 1, pp. 106–110, 2012.

[8] E. A. Tanaka, S. R. Nozawa, A. A. Macedo and J. A. Baranauskas, "A multi-label approach using binary relevance and decision trees applied to functional genomics," *Journal of Biomedical Informatics*, vol. 54, no. 1, pp. 85–95, 2015.

[9] O. Luaces, J. Díez, J. Barranquero, J. J. del Coz and A. Bahamonde, "Binary relevance efficacy for multi-label classification," *Progress in Artificial Intelligence*, vol. 1, no. 4, pp. 303–313, 2012.

[10] X. Zhang, R. Li, B. Zhang, Y. Yang and J. Guo, "An instance-based learning recommendation algorithm of imbalance handling methods," *Applied Mathematics and Computation*, vol. 351, no. 2–3, pp. 204–218, 2019.

[11] J. Wang, L. Zhu, T. Dai and Y. Wang, "Deep memory network with Bi-LSTM for personalized context-aware citation recommendation," *Neurocomputing*, vol. 410, no. 9, pp. 103–113, 2020.

[12] T. Thaipisutikul and T. K. Shih, "A novel context-aware recommender system based on a deep sequential learning approach (CReS)," *Neural Computing and Applications*, vol. 1, no. 17, pp. 1–24, 2021.

[13] L. Mei, P. Ren, Z. Chen, L. Nie, J. Ma *et al.,* "An attentive interaction network for context-aware recommendations," in *Proc. of the 27th ACM Int. Conf. on Information and Knowledge Management*, Torino, Italy, pp. 157–166, 2018.

[14] D. Wang, X. Zhang, D. Yu, G. Xu and S. Deng, "Came: Content-and context-aware music embedding for recommendation," *IEEE Transactions on Neural Networks and Learning Systems*, vol. 32, no. 3, pp. 1375–1388, 2020.

[15] S. Paliwal, A. K. Mishra, R. K. Mishra, N. Nawaz and M. Senthilkumar, "XGBRS framework integrated with word2vec sentiment analysis for augmented drug recommendation," *Computers, Materials & Continua*, vol. 72, no. 3, pp. 5345–5362, 2022.

[16] Z. Xiang, Q. Du, Y. Ma and W. Fan, "Assessing reliability of social media data: Lessons from mining TripAdvisor hotel reviews," *Information Technology & Tourism*, vol. 18, no. 1, pp. 43–59, 2018.

[17] Y. Zheng, R. Burke and B. Mobasher, "Differential context relaxation for context-aware travel recommendation," *13th Int. Conf. on Electronic Commerce and Web Technologies*, vol. 1, pp. 88–99, 2012.

[18] A. Ko, A. Odic, M. Kunaver, M. Tkalcic and J. F. Tasic, "Database for contextual personalization," *Elektrotehniski Vestnik*, vol. 78, no. 5, pp. 270–274, 2011.

[19] O. Luaces, J. Díez, J. Barranquero, J. J. del Coz and A. Bahamonde, "Binary relevance efficacy for multilabel classification," *Progress in Artificial Intelligence*, vol. 1, no. 4, pp. 303–313, 2012.

[20] T. Senthil Kumar and S. Pandey, "Customization of recommendation system using collaborative filtering algorithm on cloud using mahout," *Advances in Intelligent Systems and Computing*, vol. 321, pp. 548–559, 2015.

[21] S. M. Basha, D. S. Rajput and V. Vandhan, "Impact of gradient ascent and boosting algorithm in classification," *International Journal of Intelligent Engineering and Systems*, vol. 11, no. 1, pp. 41–49, 2018. ISSN: 2185-3118.

[22] Z. Sun, K. Hu, T. Hu, J. Liu and K. Zhu, "Fast multi-label low-rank linearized SVM classification algorithm based on approximate extreme points," *IEEE Access*, vol. 6, pp. 42319–42326, 2018.

[23] Z. Hanifelou, P. Adibi, S. A. Monadjemi and H. Karshenas, "KNN-based multi-label twin support vector machine with priority of labels," *Neurocomputing*, vol. 322, no. 3, pp. 177–186, 2018.

[24] A. Pal, M. Selvakumar and M. Sankarasubbu, "Multi-label text classification using attention-based graph neural network," in *Proc. of the 12th Int. Conf. on Agents and Artificial Intelligence (ICAART 2020), arXiv preprint arXiv*, pp. 1–12, 2020.