Open Science Index, Computer and Information Engineering Vol:17, No:5, 2023 publications.waset.org/10013104.pdf

Design and Implementation of a Software Platform Based on Artificial Intelligence for Product Recommendation

G. Settanni, A. Panarese, R. Vaira, A. Galiano

***Abstract*—**Nowadays, artificial intelligence is used successfully in the field of e-commerce for its ability to learn from a large amount of data. In this research study, a prototype software platform was designed and implemented in order to suggest to users the most suitable products for their needs. The platform includes a recommender system based on artificial intelligence algorithms that provide suggestions and decision support to the customer. Specifically, support vector machine algorithms have been implemented combined with natural language processing techniques that allow the user to interact with the system, express their requests and receive suggestions. The interested user can access the web platform on the internet using a computer, tablet or mobile phone, register, provide the necessary information and view the products that the system deems them the most appropriate. The platform also integrates a dashboard that allows the use of the various functions, which the platform is equipped with, in an intuitive and simple way. Also, Long Short-Term Memory algorithms have been implemented and trained on historical data in order to predict customer scores of the different items. Items with the highest scores are recommended to customers.

***Keywords*—**Deep Learning, Long Short-Term Memory, Machine Learning, Recommender Systems, Support Vector Machine.

I. Introduction

IN recent years the use of machine learning algorithms has spread worldwide both in academia and industry. In particular, deep learning algorithms make it possible to implement recommender systems capable of automatically filtering and personalizing information in order to propose options that are both relevant for customers and advantageous for the service provider.

A Deep Learning-Based Recommender System (DLRS) is capable of learning representative features in the input data. The availability of apps and various services on the internet provides companies that use commercial platforms to collect a large amount of data, such as text, images, video and audio, in which valuable information on users and items is hidden. Deep learning algorithms allow the company to extract information from these heterogeneous data and combine them with each other in order to recommend specific items to users in a personalized way [1].

DLRSs facilitate the information search process as they can learn the latent characteristics of users from huge volumes of data. The recommender systems generate a list of recommendations that is very effective for the e-commerce user [2].

In recent times, deep learning algorithms have been used successfully to obtain highly accurate recommender systems. Compared to traditional recommender systems, DLRSs are now applied in different areas as they have considerable potential. The first strength of DLRS consists in the ability to model the nonlinearities that can characterize the data, thanks to the use of non-linear activation functions, such as sigmoid and hyperbolic tangent [3]. This nonlinear modeling allows to overcome the limitations of linear models that oversimplify the relationship between features [4]. In detail, models based on deep learning are able to effectively capture the complex relationships between users and items. The DLRS are also able to identify the latent characteristics of users and items. Since deep learning algorithms have the ability to selectively capture relationships between users and items, these are able to extract more complex information [5]. In fact, the higher layers allow to analyze in depth the complex characteristics of the users and to extrapolate their preferences. The hidden layers of the deep learning architectures give a DLRS better coding of the abstractions and accuracy of the recommendations.

The main deep learning technologies that are used for implementing high-performance recommender systems are RNN (Recurrent Neural Network), LSTM (Long Short-Term Memory), CNN (Convolutional Neural Network) and AE (Autoencoder) [2]. Moreover, XGBoost (eXtreme Gradient Boosting) is a powerful algorithm gaining popularity in the field of product prediction and suggestions [5]. The advantages of this machine learning algorithm are related to the use of optimization and regularization techniques and to the possibility of setting a large number of hyperparameters.

The architecture of RNN is characterized by connections between neurons, called feedback loops, which allow the output of the relative nodes to influence the subsequent input to the same nodes. Due to this mechanism, RNNs remember previous calculations and are very efficient for sequential data processing. Therefore, RNNs are used in recommender systems to address temporal dynamics related to user behavior [6].

LSTMs are a particular type of RNN as they are equipped

G. Settanni was with Department of Mathematics and Computer Science, University of Bari, Bari, Italy. She is now with the Dyrecta Lab srl, MIUR Research Institute, Conversano (BA), Italy (corresponding author, e-mail: giuseppina.settanni@dyrecta.com).

A. Panarese was with Department of Chemistry, University of Bari, Bari, Italy. He is now with the Dyrecta Lab srl, MIUR Research Institute, Conversano (BA), Italy (e-mail: antonio.panarese@dyrecta.com).

R. Vaira and A. Galiano are with the Dyrecta Lab srl, MIUR Research Institute, Conversano (BA), Italy (e-mail: raffaele.vaira@dyrecta.com, maurizio.galiano@dyrecta.com).

with cells that make the algorithm capable of judging whether information is useful or not. A cell is a kind of "processor" in which there are three gates, called input gate, forget gate and output gate [7]. The LSTM retains only the useful information, which satisfies the conditions of the algorithm, while it forgets, at the level of the forget gate, the superfluous information.

CNN is a feedforward neural network with convolution layers and pooling operations. The basic structure of a CNN is made up of input layer, convolutional layer, subsampling layer (pool layer), fully connected layer and output layer [8]. CNN can capture global and local characteristics and significantly improve efficiency and accuracy. The shared-weights architecture of CNN allows to improve the generalization capacity of the model and to decrease the number of the parameters used by the model and, therefore, its complexity. CNNs are used in the implementation of recommender systems to perform the feature extraction operation [2].

AE is an artificial neural network (ANN) whose goal is to reconstruct its input data in the output layer. In general, the intermediate layer, called hidden layer, is used to filter the salient features of the input data [2]. AE can be schematized by means of an encoder, which compresses the input data by selecting the characteristics of interest, and a decoder, which restores the data to their original structure while preserving the selected characteristics [9]. The encoding and subsequent decoding of the input parameters are aimed at learning the latent characteristics of the data and filtering them [10]. For this reason, AE finds application in recommender systems to filter information of interest.

The recommender systems perform the important function of automatically filtering and personalizing information, thus allowing to manage with the IT overload to which the user is exposed on a daily basis. Recently, international research has experimented with the use of deep learning technologies with the aim to increase the potential of traditional recommender systems [5], [12], [13].

Modern platforms for developing machine learning and deep learning algorithms take advantage of powerful tools, such as TensorFlow, Keras and Pytorch, which allow developers to create highly flexible recommendation models that can be integrated into e-commerce platforms.

In developing a software platform, it is of fundamental importance to evaluate the user experience following the interaction via app or website. The ultimate goal is to determine positive feelings in the user during navigation that prompt the user to make a purchase and use the platform again. The user must be considered as the real protagonist of the entire development process of a platform integrating a recommender system and not just as the end user. Browsing a site that is not suited to the user's needs can give him negative feelings, frustration, high cognitive effort and consequently push him to abandon the product.

Artificial intelligence algorithms make it possible to process the data collected on user browsing taking into consideration the emotional state and preferences of the customer by analyzing the time spent on a webpage, the clicks made and other behaviors related to the navigation.

The research [11] proposes an approach based on the TF-IDF (Term Frequency – Inverse Document Frequency) technique and an SVM (Support Vector Machine) algorithm for the classification of journalistic news starting from the news headlines on different websites and blogs. The developed methodology is based on three consecutive steps: text preprocessing, feature extraction by means of TF-IDF and text classification by means of SVM. The purpose of this approach is to classify news into various groups to make it easier for users to choose news of interest among the most popular news.

Authors of [12] present a hybrid approach that integrates different methodologies, such as TF-IDF and Singular Value Decomposition (SVD), in order to combine different types of recommender systems. Furthermore, Self-Organizing Map (SOM) neural network and K-means clustering are used which are artificial intelligence techniques that allow to improve performance.

The authors of [13] discuss and compare various solutions used to implement recommender systems. They conclude that the most precise systems that best solve the cold start problem are those that combine the use of deep learning algorithms (CNN, RNN, LSTM) and sentiment analysis, also known as opinion mining.

In the present paper we design and propose an original recommender system model based on different artificial intelligence technologies. In detail, NLP, machine learning and deep learning tools have been used to implement a recommender system that incorporates the contents of the items (content based) on the characteristics of the users and, above all, on their preferences (collaborative filtering). This hybrid approach allows to overcome the disadvantages that characterize the two types of systems.

II. Methodology

Using an e-commerce site and social media allows companies to collect a large volume of data that contain valuable information. A recommender system based on deep learning algorithms offers e-commerce companies the opportunity to process data, obtain organizational and economic benefits, increase customer satisfaction and retain them.

Fig. 1 shows the flowchart of the implemented recommender system. The first step is data collection. These are data of a different nature as they relate to the characteristics of the items and customers. In turn, customer data refer to previous purchases, navigation on the e-commerce site and customer feedback. In the second step, an initial processing of the raw data takes place in order to extract the features hidden in them and make the data suitable for processing by deep learning algorithms. In fact, this phase of data modeling has as output the U matrix containing the customer features, the V matrix containing the item features and the R utility matrix whose elements contain the customer rating of the items they have judged, directly or indirectly. The next step is represented by data processing using deep learning algorithms. In particular, LSTM algorithms were chosen due to their large capacity to process time series [14]. This feature allows LSTMs to seize changes in customer preferences as soon as they occur. LSTM predicts the score of customers about items with which they have not yet interacted and therefore allows to obtain a ranking of items for each customer. The final recommendation is based on the ranking of items as each customer is suggested the items with the highest score.

Due to the ability of the discussed deep learning algorithms to self-learn, the proposed model is able to exploit the useful information hidden in data, for instance preferences of the customer starting from the web pages consulted, the time spent on a page, the clicks made and other behaviors related to navigation. Furthermore, the combination of LSTM with NPL methods, such as TF-IDF and SVM, allows to extrapolate the user emotional state and correlate it to other variables.

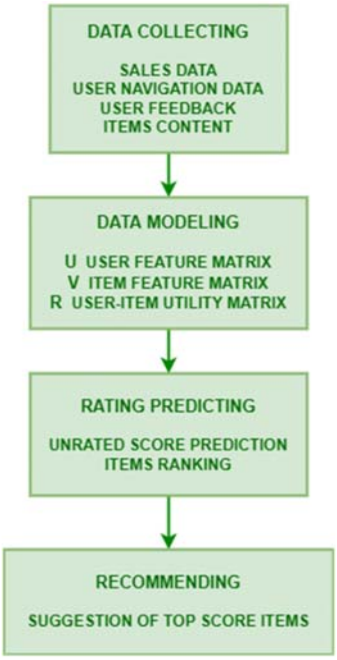

Fig. 1 General flowchart of the proposed recommender system

Fig. 2 depicts the architecture of the data modeling stage. Data modeling begins with the preprocessing of text related to item content, customer judgment and sales data. These are unstructured data that contain both useful and useless data, in fact they have been collected from different sources and must be cleaned, for example from punctuation. Text tokenization, namely dividing the text into single words, allows to improve the result of the subsequent processing steps [11]. Word filtering eliminates unnecessary objects such as articles or words that are not important for the specific application. These preprocessing operations also determine the decrease in complexity and computational cost.

The extraction of the text features is carried out using the TF-IDF technique, widely used in the field of text mining and NLP, because it allows to emphasize very significant words. This technique uses the TF-IDF weighting function to calculate the ratio of the local frequency of a word to its global frequency. This assigns a weight w_{ij} to the $i-th$ word in the $j-th$ document in order to measure the relevance of the words rather than just their frequency:

$$w_{ij} = \frac{n_{ij}}{d_j} \log \frac{T}{N_i}$$

where T is the number of texts, N_i is the number of documents containing the $i-th$ term, n_{ij} is the number of times that the $i-th$ term is present in the document $j-th$ which is made up of d_j terms.

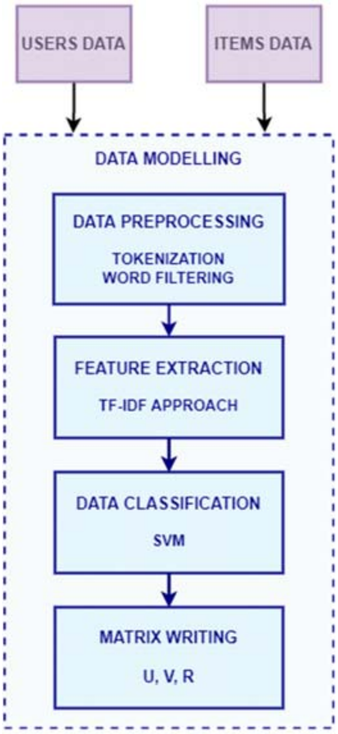

Fig. 2 Architecture of the data modeling step

Data classification is performed using SVM that is an algorithm that has a low computational cost, but is very efficient in NLP field [15]. SVM is a supervised learning model that uses a hyperplane to properly separate data and classify it [16]. In detail, it was decided to use a multiclass SVM algorithm which is a generalization of SVM that allows to separate the data points into a number of classes greater than 2 [17], [18]. Ultimately, multiclass SVM classifier allows users and items to be classified in different classes according to their characteristics. Typical user features are age, interests, web pages visited, number of clicks, frequency of purchase, items purchased, product categories, etc. Similarly, a typical example of item feature is the product category. The processing by means of SVM, which is performed separately for customer and item data, permits to write the matrices $U \in \mathbb{R}^{N \times A}$ and $V \in \mathbb{R}^{M \times B}$, respectively. N is the number of customers, A is the dimension of customers features, M is the number of items and B is the dimension of item features. The vector U_i represents the $i-th$ row of U and contains the hidden features of the $i-th$ user. Instead, the vector V_j coincides with the $j-th$ row of

V and provides information about the $j-th$ item.

Finally, the knowledge of U and V allows to obtain the user-item matrix $R \in \mathbb{R}^{N\times M}$. In fact, R is the synthesis of the information contained in the customer historical data and in the item data. R_{ij} represents the score that the $i-th$ customer has attributed to the $j-th$ item. If the $i-th$ customer has not expressed any opinion on the $i-th$ item and has not read any information about it, R_{ij} is null. The final output of the phase of data modeling phase consists of the normalized matrices U, V and R, which contain the information about customers and items in a hybrid perspective. The synthesis of this information makes it possible to achieve greater accuracy and greatly alleviate the cold start problem.

As shown in Fig. 3, the matrices thus obtained are supplied as input to the LSTM network which processes the latent features and the relationships between the variables. The final output is the prediction of the coefficient values of the rating matrix $S \in \mathbb{R}^{N\times M}$. Fig. 3 shows the architecture of the LSTM designed for the prediction of S_{ij}. Before being used, the network is trained so that it can learn the correlation between the various variables involved and optimize the model parameters. This phase is essential for the model to capture the non-linear relationship between the various quantities that characterize users and objects.

During the training, the dropout technique is used to prevent overfitting. The embedding input layer reshapes the input variables so that they can be processed by the LSTM layers, which form the core of the neural network architecture. Instead, the fully connected layer calculates the output of the parameter, namely the values of the coefficients S_{ij} of the ranking matrix. This level allows to change the dimensionality of the output from the previous level so that the model can provide consistent results based on the relationships it is implementing.

Model accuracy is measured using the following metrics:

$$\text{RMSE} = \sqrt{\frac{1}{N_S}\sum_{i,j}\left(S_{ij} - \hat{S}_{ij}\right)^2}$$

$$\text{MAE} = \frac{1}{N_S}\sum_{i,j}\left|S_{ij} - \hat{S}_{ij}\right|$$

where $\hat{S}_{ij}$ is the predicted matrix coefficient and N_S is the number of training or testing data samples, that is the sample size.

It is very important to note that the proposed model, based on the algorithms discussed, is able to calculate the rating of the various items for each customer in real time. As new data become available, LSTMs can be retrained and provide updated predictions.

The implementation of both the SVM and LSTM models is done using python, a high-level language that has very powerful open-source libraries, such as TensorFlow, Keras and Scikit-Learn.

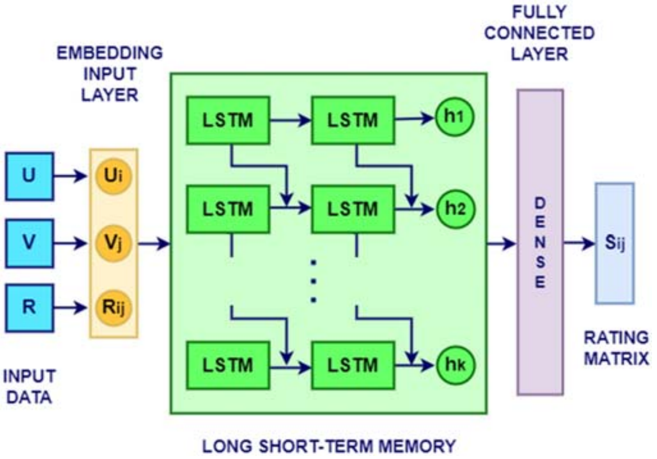

Fig. 3 LSTM architecture proposed for predicting the rating scores

Acknowledgment

The proposed work has been developed within the framework of the industry project titled: "Design and implementation of tools based on artificial intelligence for the configuration of software systems".

References

[1] Z. Batmaz, A. Yurekli, A. Bilge, and C. Kaleli, "A review on deep learning for recommender systems: challenges and remedies," in *Artif Intell Rev* 52, 1–37, 2019.

[2] R. Mu, "A Survey of Recommender Systems Based on Deep Learning," *IEEE Access*, vol. 6, pp. 69009-69022, 2018. doi: 10.1109/ACCESS.2018.2880197.

[3] S. Zhang, L. Yao, A. Sun, and Y. Tay, "Deep Learning Based Recommender System: A Survey and New Perspectives," *ACM Comput. Surv.*, vol. 52, 1, Article 5 (February 2019), 38 pages. https://doi.org/10.1145/3285029

[4] W. Serrano, "Intelligent Recommender System for Big Data Applications Based on the Random Neural Network," in *Big Data Cogn. Comput.*, vol. 3, 15, 2019. https://doi.org/10.3390/bdcc3010015

[5] A. Panarese, G. Settanni, V. Vitti, and A. Galiano. "Developing and Preliminary Testing of a Machine Learning-Based Platform for Sales Forecasting Using a Gradient Boosting Approach," *Applied Sciences* vol. 12, no. 21, pp. 11054, 2022. https://doi.org/10.3390/app122111054

[6] C. Chen, X. Meng, Z. Xu, T. Lukasiewicz, "Location-Aware Personalized News Recommendation with Deep Semantic Analysis," *IEEE Access*, vol. 5, pp. 1624-1638, 2017. doi: 10.1109/ACCESS.2017.2655150.

[7] M. Yuan, Y. Wu, L. Lin, "Fault diagnosis and remaining useful life estimation of aero engine using LSTM neural network," in *2016 IEEE International Conference on Aircraft Utility Systems (AUS)*, pp. 135-140, 2016 doi: 10.1109/AUS.2016.7748035.

[8] A. Massaro, A. Panarese, S. Selicato, A. Galiano, "CNN-LSTM Neural Network Applied for Thermal Infrared Underground Water Leakage," in 2021 IEEE International Workshop on Metrology for Industry 4.0 & IoT (MetroInd4.0&IoT), pp. 219-224, doi: 10.1109/MetroInd4.0IoT51437.2021.9488536.

[9] J. Naranjo-Alcazar, S. Perez-Castanos, P. Zuccarello, F. Antonacci, M. Cobos, "Open Set Audio Classification Using Autoencoders Trained on Few Data," *Sensors*, vol. 20, no. 13, 3741, 2020. doi:10.3390/s20133741

[10] A. Massaro, A. Panarese, A. Galiano, "Technological Platform for Hydrogeological Risk Computation and Water Leakage Detection based on a Convolutional Neural Network," in *2021 IEEE International Workshop on Metrology for Industry 4.0 & IoT (MetroInd4.0&IoT)*, pp. 225-230, 2021, doi: 10.1109/MetroInd4.0IoT51437.2021.9488561.

[11] S. M. H. Dadgar, M. S. Araghi, and M. M. Farahani, "A novel text mining approach based on TF-IDF and Support Vector Machine for news classification," in *Proc. 2016 IEEE International Conference on Engineering and Technology (ICETECH)*, 2016, pp. 112-116, doi: 10.1109/ICETECH.2016.7569223.

[12] Y. Afoudi, M. Lazaar, and M. Al Achhab. "Hybrid recommender system combined content-based filtering and collaborative prediction using artificial neural network," *Simulation Modelling Practice and Theory*. vol. 113, 2021, 102375, ISSN 1569-190X, https://doi.org/10.1016/j.simpat.2021.102375.

[13] R.J.K. Almahmood, and A. Tekerek, "Issues and Solutions in Deep Learning-Enabled Recommender systems within the E-Commerce Field," *Appl. Sci.*, vol. 12, 11256, 2022. https://doi.org/10.3390/app122111256

[14] A. Massaro, A. Panarese, M. Gargaro, A. Colonna, A. Galiano. "A Case Study of Innovation in the Implementation of a DSS System for Intelligent Insurance Hub Services," *Computer Science and Information Technology*, vol. 9(1), pp. 14-23, 2021. DOI: 10.13189/csit.2021.090102.

[15] M. Wankhade, A.C.S. Rao, C. Kulkarni, "A survey on sentiment analysis methods, applications, and challenges," *Artif Intell Rev 55*, 5731–5780 2022. https://doi.org/10.1007/s10462-022-10144-1

[16] Vapnik, VN., (1995). The Nature of Statistical Learning Theory (New York, Springer-Verlag.

[17] G.J.J. Van Den Burg, and P.J.F. Groenen. "GenSVM: a generalized multiclass support vector machine," *Journal of Machine Learning Research*, vol. 17, 1, pp 7964–8005, 2016

[18] G. Pouliot. "Equivalence of Multicategory SVM and Simplex Cone SVM: Fast Computations and Statistical Theory," in *Proceedings of the 35th International Conference on Machine Learning, in Proceedings of Machine Learning Research* 80:4133-4140 Available from https://proceedings.mlr.press/v80/pouliot18a.html.