

Predicting Shopping Intent of e-Commerce Users using LSTM Recurrent Neural Networks

Konstantinos Diamantaras^a, Michail Salampasis^b, Alkiviadis Katsalis^c
and Konstantinos Christantonis^d

*Intelligent Systems Laboratory, Department of Information and Electronic Engineering,
International Hellenic University, Sindos, Thessaloniki, Greece*

Keywords: Purchase Intent, e-Commerce, LSTM-RNN, Web Usage Mining.

Abstract: An e-commerce web site is effective if it turns visitors into buyers achieving a high conversion rate. To this realm, it is useful to predict each user's purchase intent and understand their navigation behavior. Such predictions may be utilized to improve web design and to personalize shopper's experience, hopefully leading to increased conversion rates. Additionally, if such predictions can be done in real-time, during the ongoing navigation of an e-commerce user, the e-commerce application can take proactive stimuli actions to offer incentives with a view to increase the probability that a user will finally make a purchase. This paper presents a method for predicting in real-time the shopping intent of e-commerce users using LSTM recurrent neural networks. We test several variants of our method in a dataset created from the processing of Web server logs of an industry e-commerce web application, dividing user sessions in three different classes: browsing, cart abandonment, purchase. The best classifier achieves a predictive accuracy of almost 98%. This result is competitive with other state-of-the-art methods, which affirms that accurate and scalable purchasing intention prediction for e-commerce, using only session-based data, is feasible without any intense feature engineering.

1 INTRODUCTION

For many businesses, a significant part of their retail sales comes from their e-commerce website. Furthermore, e-commerce has ramped up during the pandemic around the world, something which is expected to become a longer-lasting effect. Subsequently, the effectiveness of e-commerce solutions, as it is expressed by increased purchase conversion rates, has become an important challenge for successful e-businesses.

We can categorize two different approaches for increasing purchase conversion rates of an e-commerce web application. The first aims to mine and analyze the usage of an e-commerce application (Moe, 2003) to improve its design (Carmona et al, 2012) or, to personalize web page content or the web site structure that is rendered to different types of online shoppers (Ding et al, 2015). The second

approach aims to create in *real time* a model of user purchasing behavior, and to predict the purchase intent and probability, cart abandonment likelihood etc. If such predictions would be done accurately, then conversion rates could be improved by applying different marketing and web stimuli, such as offers, coupons, time-limited price discounts and other purchase incentives. In real settings, the two different strategies can co-exist implementing two different pathways to the same objective. The general assumption underlying both approaches is that the collective patterns hidden in users' navigation paths on e-commerce applications can be analyzed to make effective predictions about the next actions and behavior of the current user and her/his purchase intent.

In this paper we present a real-time e-commerce analysis component that makes purchase intent predictions using only the *short-term browsing*

^a <https://orcid.org/0000-0003-1373-4022>

^b <https://orcid.org/0000-0003-4087-125X>

^c <https://orcid.org/0000-0002-4416-4790>

^d <https://orcid.org/0000-0001-9847-7410>

pattern of a user. The motivation for using short-term browsing patterns only, is that in many e-commerce applications long-term user profiles do not exist, because users are first-time visitors, or the application does not require visitors to login, in order to conform with users' privacy concerns or other policies such as the General Data Protection Regulation (GDPR).

In our method, we use Long Short-Term Memory (LSTM) Recurrent Neural Networks (RNNs) to process online shoppers' behavioral data. RNNs possess several properties that make them suitable for sequence learning of user sessions. The last few years a significant line of research has emerged to explore whether recommendations in e-commerce web sites can be viewed as a sequence prediction problem using RNNs. This research was mostly driven by the findings that RNN produces models with higher learning capacity and generalization ability than Hidden Markov Model (HMM).

In fact, several studies have recently used RNN for session-based recommendation (SBRs) (e.g. Hidasi and Karatzoglou, 2018; Salampasis et al, 2021). SBRs are recommendation techniques which rely on the user's recent actions, the observed community buying behavior and, other session-specific data. Generally, in the last five years RNNs have been widely used in the e-commerce domain to incorporate temporal features and models for analyzing sequences of user actions. The main task was to improve recommendations or, to predict future behavioral directions. Though, they have been barely used in the task of predicting purchase intent. From that perspective, our research work contributes to better understanding of the RNN-LSTM method for predicting purchase intent. More specifically we wished to determine, if and under what e-commerce scenarios, RNN-LSTM can achieve comparable results with more traditional ML methods that have been the state-of-the-art for this task, but they do require extensive feature engineering processes.

Sakar et al (2019) presented a study which is much more relevant to our purchase intent task. In this work they use LSTM-RNN to predict the probability that the user will leave the site within a certain time. In our study we use LSTM-RNNs to predict in real-time the purchase probability, i.e. if the user will add an item to his or her cart, and if s/he will finally make a purchase or will abandon the cart. Also, our approach on creating a model for predicting purchase intent is different. We utilize all $\tau\pi\epsilon$ of actions from previous anonymous user sessions, and we create a model that in production settings will be normally used for making predictions after each user action. If a certain output for the current session is

predicted, for example cart-abandonment, then the e-commerce application can proactively offer incentives (e.g. discount coupons, or any other enticement based on a business or marketing rule) with a view to increase the probability that a user will finally make a purchase. This is precisely how we envisage that an e-commerce system would integrate our purchase intent component.

To evaluate our method, we use an industry dataset that has log data from a medium size e-commerce web application (leather apparel). We also considered using standard datasets such as the YooChoose, which was first presented in the RecSys Challenge 2015. This dataset provides sequences of click events, product view and buying events, and the goal is to predict whether the user (a session) is going to buy something or not, and if s/he is buying, what would be the items s/he is going to buy. However, this dataset is not suitable, because our problem definition and prediction task are different. First, our aim was not to consider only the product view and buying events, but the entire spectrum of many different user action types. This is done in the same sense that it is studied in the behavior modelling and evaluation of web-based information seeking systems (e.g. Salampasis and Diamantaras, 2002). Second, our task is not to predict if a session will be a buying session or not *as a whole*. Instead, our approach is more dynamic. Specifically, during a user session, and for each user action, we predict for the *remaining session section*, what would be the final outcome (i.e. browse only, cart abandonment, purchase).

Besides the basic user action data that our e-commerce Web site stores in its log files, in our research work, we additionally calculate and utilize some extra features. These are: the time spent in each action (in seconds), how the user has landed in the e-commerce site (origin: referred by an online advertisement or not), season (autumn/winter or not), day (Weekend or not) and working hours (yes/no). We do not include more features because are potentially difficult to maintain and recalculate in a realistic production setting, or features relying on user profiles. Generally, our feature selection process is driven by the idea to use only features that are compatible with a strictly session-based operational scenario. Specifically, we select features that do not require any user profile data, and additionally are straightforward to implement in a real-business environment, that would require frequent (even daily) re-training and update of the prediction models.

The rest of the paper is organized in the following manner. Section 2 presents prior work. In Section 3 we discuss our dataset, how we model the

shopping intent problem, and finally the method that we used to provide a solution. In Section 4 we describe the experiments conducted and we report and discuss the results. Section 5 concludes the paper, summarizing the findings and presenting ideas for future development.

2 PRIOR WORK

A large range of statistical, machine learning and neural network methods have been applied to predict the purchase probability in e-commerce applications. For example, simple Bayes and multilayer perceptron classifiers have been used to predict whether an e-commerce visitor is likely to make a purchase or not (Budnikas, 2015).

Suchacka et al (2015) collected data from an online bookstore and have applied SVM using many different variables (23) to classify user sessions as either browsing or buying sessions. The best SVM classifier proved to be very effective, with an overall predictive accuracy of 99% and the probability of predicting a buying session of almost 95%.

Suchacka and Chodak (2017) continued this work using association rules and a k-nearest neighbor (k-NN) classifier to assess the purchase probability in the online bookstore. They analyzed the Web server log and extracted core user action data (action type and time spent) as well as other features such as session length, total session duration in seconds, average action time in seconds, origin representing how the user was referred to the bookstore site, product categories viewed during the session. They used simple association rule mining to predict with good accuracy the purchase probability of the users and other behavior knowledge for two customer groups: traditional customers (accuracy 90%) and innovative customers (accuracy 88%). k-NN method was equally effective, however it is deemed not suitable for real-time prediction since it is a lazy-learning algorithm.

The prediction of an e-commerce user purchase intent requires the predictions of her/his next actions and it is naturally a sequence-based problem. For that reason, Hidden Markov Model (HMM) was used as a promising solution. Generally, there are a lot of Web usage mining works that use HMM to predict the next action of a user or infer about her/his overall navigation behavior. These works have applied and tested HMM in several contexts and for different tasks such as predicting web search success (Ageev et al, 2011), recommender systems (Aghdam et al, 2015), tourism web sites (Yifan et al, 2013).

A research work more related to our study, is presented in Ding et al (2015) which uses HMM to learn real-time shopper intent for optimal page adaptation. To capture shopper's behavior in real-time they model and monitor several cart choices (exit, no change, remove item, add item, purchase). Their proposed a model that effectively differentiates each user according to her or his real-time intent. In a simulated test it manages to reduce shopping cart abandonment by 32,4% and improves purchase conversion by 6,9%, if the e-commerce Web site initiates optimal page adaptation.

The review in this Section shows that the problem of purchase intent or session classification in e-commerce applications has been studied with a large variety of methods. The RecSys2015 challenge asked the contenders to predict the set of items purchased in a session based on the user clicks. The challenge winners (Romov and Sokolov, 2015) proposed a two-stage classifier trained by a version of gradient boosting which is still regarded as the State-of-the-Art (SotA) for this problem. The first classifier makes a binary prediction whether there will be at least one buy in the session or not, while the second classifier predicts the purchased item set. The method used many categorical features of the sessions and the items, such as the time and date of the session or the click, the number of clicks for each item, etc.

However, this approach makes purchase intent prediction using all click events of a session. Very few works have attempted to make dynamic predictions of user purchase intent during the session where the next user actions are not yet available. Sheil et al (2018) used recurrent neural networks (RNN) to capture both session and dataset-global event dependencies and relationships for user sessions of any length. Results on benchmark datasets from the RecSys15 challenge show that their work performed very close to the RecSys15 challenge winner. The main difference is that their method does not require any domain or dataset specific feature engineering for all types of sessions.

Sakar et al (2019) present another online shopper behavior analysis system based on RNN-LSTM. It has two modules operating in parallel and the experiment they report used a non-public dataset. The first module predicts shopping intent, but this module uses ML classifiers such as Random Forest, support vector machines and multilayer perceptron. Only the second module uses RNN-LSTM and predicts the Web site abandonment likelihood without making a purchase. Their shopping intent module performs significantly lower than the SotA. The second module which predicts Web site abandonment after a short

action window (3 moves) achieves an accuracy of almost 75%.

Ling et al (2019) use a full connected long short-term network (FC-LSTM) for modeling the interactions between customers, and promotion channels, as well as the nonlinear sequence correlations and cumulative effects between customer's browsing behavior. However, to improve the performance of the prediction they incorporate more features of customer profile including purchase history and demographics.

Considering the realities of the e-commerce domain and the dynamic nature of buying behavior, we believe that if comparable performance can be attained using RNN-LSTM and with less demanding feature engineering, this advantage should be considered important. The reason is that in e-commerce applications any prediction model would need very frequent updates to address the dynamic nature of online shopping. For example, in a realistic business environment, a prediction model should be frequently updated/re-trained, in order to capture temporal changes in the shoppers buying behavior and generally accommodate any temporal aspect of the e-commerce application.

3 PREDICTING PURCHASE INTENT USING LSTM-RNN

3.1 Dataset

The raw data for the experiments reported in this paper are taken from the web server logs of an e-

commerce application for a relatively long period of time (several months). Those kinds of datasets that are extracted through real applications, entrain all possible malfunctions. In our dataset, there were sessions with hundreds of events which admittedly belong on the maintenance staff and should be discarded, while there were events that were spanning long periods or actions and therefore were not considered reliable. For that reason, we introduced some limit criteria on our sessions and events and discarded everything above those. Indeed, sessions with more than 250 user actions (events) were removed and any single action lasting more than 10 minutes was limited to 600 seconds.

After we setup some basic data curation, the log data were analyzed to identify sessions, session length, user actions in each session, actions' related items, item categories, and time spent in each action. However, not all these data are used in this study, e.g. item and item categories are not used because our task was not to predict the item that will be purchased, but what is the purchase intent. The user-agent cookie of each log line was used to identify unique sessions. Each sequence identified by a unique cookie is a user's session and consists of all the actions that the user has made during a session. The above dataset was further processed to obtain only the sessions that contain at least 3 user action sequences. As a result of this preprocess the final dataset consists of 21.896 sessions that altogether count 258.101 user actions. The average session size is 11,7 actions and the Median is 8. The average size of Browsing, Cart Abandonment and Purchase sessions are 11,05, 18,8 and 19,54 respectively.

Table 1: Types of user actions extracted from the Web server log file of our e-commerce application and their frequencies in the dataset and in each of the three different session types.

	All sessions (21.896)		Purchase sessions (689)		Browse sessions (19.902)		Cart Abandonment sessions (1305)	
CATEGORY	141561	54,8%	2760	20,5%	128845	58,5%	9956	40,6%
VIEW PRODUCT	83299	32,3%	3149	23,4%	73639	33,5%	6511	26,5%
HOME	6139	2,4%	316	2,3%	5463	2,5%	360	1,5%
ASK QUESTION	4027	1,6%	150	1,1%	3398	1,5%	479	2,0%
ORDER	730	0,3%	730	5,4%	0	0,0%	0	0,0%
CONTACT	4264	1,7%	42	0,3%	4089	1,9%	133	0,5%
ADD CART	3380	1,3%	1113	8,3%	0	0,0%	2267	9,2%
VIEW CART	10722	4,2%	5011	37,2%	1228	0,6%	4483	18,3%
SEARCH	69	0,0%	0	0,0%	69	0,0%	0	0,0%
CONCERNED	1172	0,5%	115	0,9%	874	0,4%	183	0,7%
ACCOUNT	2068	0,8%	75	0,6%	1838	0,8%	155	0,6%
RECOMMEND	670	0,3%	5	0,0%	656	0,3%	9	0,0%
	258.101	100%	13.466	100%	220.099	100%	24.536	100%

The complete set of actions were retrieved for the website’s user’s behavior analysis (click-stream analysis) and all user actions (web page visits) were classified in twelve basic action types. We believe that these twelve action types represent virtually all actions that a user could perform in any e-commerce application, hence they can be used as a “standard” set of action types for future studies in the e-commerce domain that consider a broad spectrum of potential user actions, i.e. not only product views and purchase actions that are typically considered in Recommender Systems research. Table 1 shows all user action types modelled in our system and what is their appearance frequency in the dataset categorized in each of the three different session types. Most of the action types shown in Table 1 are self-explained and their semantics are apparent. Action type CONCEREND means that a user has visited a web page related to privacy policy, payment security or product shipping & returns. ASK_QUESTION is triggered when a user makes an enquire during his session. Finally, action RECOMMEND represents the action of a user recommending a product to another person (by sending an email notification).

The 689 sessions that ended in purchases represent a conversion rate of 3,14%, quite an average rate for apparel e-commerce applications. In the 90.9% sessions (19.902), users did not have any items in their shopping cart when they exited, which implies sessions which were pure browsing. The rest of the sessions (1305) had items in their shopping cart when they finished, but never turned into purchases representing the “cart-abandonment” sessions.

3.2 Method

Our method models, for each user session, all user

actions as a sequence representing the user navigation during the entire session. Each user action belongs to one action type from those presented in Table 1. Additionally, each action has a duration which is calculated in seconds and is the time that user stays in this web page before a new user action occurs. For each user action k during a session, the input sample for the model training are the N events/actions that precede event k , with N being a parameter of the training method (action history window).

It is important to clarify that despite an entire session can be considered as browse-only, cart-abandonment and purchase, the training of the prediction model is not done by taking these session labels to statically determine the output of all session steps/actions during a session. Instead, for every sequenced user action in each session, the output is calculated based on the *remaining actions* until the session ends. So, it is possible, during a session, each session segment to be labelled with a different output. In other words, a single event (e.g. make order) which occurs anywhere in the session, will not uniformly determine the label of *all* user actions of the *entire* session. This is an important modeling detail which differentiates our method to other research works on purchase intent reported in the literature.

To prepare the training data in a suitable and easy to process format, we use the time (in seconds) that the user spent in this action, to denote which user action has occurred in each session step as shown in Table 2. All the other user actions are denoted with 0. Additionally, we use four extra bits to represent the extra features that our model uses (origin, season, day, working hours). These extra features were selected due to characteristics of the specific e-commerce application and because previous studies have shown that buying rate changes over time

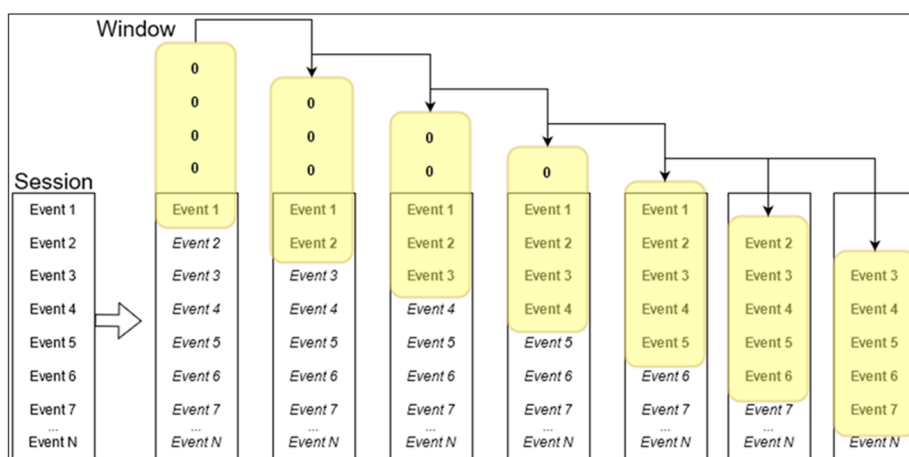


Figure 1: Sliding window running from the start to the end of each session to obtain all user’s session instances for training-prediction purposes.

(Romov and Sokolov, 2015). The feature Season indicates a high-season (autumn/winter for leather apparel) or not. We know that a shopper makes a purchase with a higher probability on the weekend than on the working day. We also know that visiting the e-commerce site during midday leads to purchases several times more often than in the night hours.

In this way, each session is modeled as a sequence of events and each event is represented with a vector like the ones presented in Table 2. Each item in this vector, in the first twelve positions, represents one action type from those illustrated in Table 1, while the rest four positions correspond to the four extra features. The action that occurs in each session step is signified with its duration in seconds (>0) and all other events are marked with 0 value. For example, in the data that are presented in Table 2, the user action (Event 1) was a View-Product action that lasted 265 seconds, the action 3 was again a View-Product that lasted 270 seconds, and event k was an ASK_QUESTION user action that lasted 45 seconds.

Using this modeling of the session data, the training is straightforward. For each session sequence a sliding window starting from the first session action, signifies an instance in the e-commerce user navigation that is used as a sample for training (Figure 1). The length of the window is fixed (N) and it is a parameter of the method. In cases when the front of the sliding window lies within the first $N-1$ events, i.e. the available user navigation history steps are still fewer than the size of window, the extra slots are padded with fully zeroed events.

Table 2: Modeling the user actions sequence and the extra features (origin, season, day, working hours). Each training line corresponds to a single session event (user actions) and has 12 numbers (each number indicating one user action illustrated in Table 1), plus 4 numbers representing the extra features.

Event 1	0 ; 265 ; 0 ; 0 ; 0 ; 0 ; 0 ; 0 ; 0 ; 0 ; 0 ; 1 ; 1 ; 0 ; 1
...	
Event 3	0 ; 270 ; 0 ; 0 ; 0 ; 0 ; 0 ; 0 ; 0 ; 0 ; 0 ; 1 ; 1 ; 0 ; 1
...	
Event k	0 ; 0 ; 0 ; 45 ; 0 ; 0 ; 0 ; 0 ; 0 ; 0 ; 0 ; 1 ; 1 ; 0 ; 1

For each input instance that our method prepares for training, the corresponding target is computed based on the remaining part of the session (i.e. the remaining events). Specifically, for every event E_i the method calculates what the target would be, considering all the remaining session events, starting from E_{i+1} until the last event of the session. The structure of the target is a bit-alike 2-digit formation. Each digit represents one of the two actions of interest (i.e. add cart and make order). That means that there

are 4 possible output statuses. Two of them, (1,0) and (0,1) signify the existence of at least one *add-cart* or one *purchase* event, and a third one the existence of both events (1,1). In case of the complete absence of these two events in the rest of the remaining session segment, the target becomes (0,0) which means we have a browse-only remaining session. Figure 2 shows an overview of the prediction model architecture.

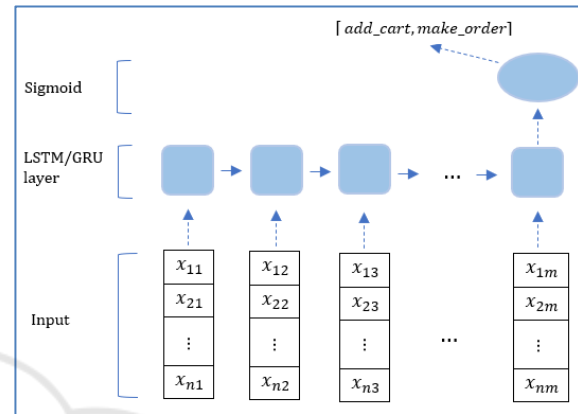


Figure 2: An illustration of the model architecture. In the input layer, n denotes the number of features/actions and m denotes the window (sequence) length. Also, x_{ij} denotes the action length in seconds or 1 or 0 for the binary extra features. After passing the LSTM/GRU and Sigmoid layer the output is a vector with two binary elements (0 or 1) each element representing the *add_cart* event and *make_order* events respectively.

To summarize, in our study, the purchasing intention model is technically designed as a two-label classification problem, each label representing the presence or the absence of the *add_cart* and/or *make_order* events, in the session section which follows immediately after the input session window. From an application/task point of view, the final classification can be summarized towards a single purchase intent scenario outcome. The following four scenarios describe the four different outcomes representing the four different targets in the training phase. Obviously, the same targets are used in the testing/prediction phase:

- The user will add item(s) to the cart but will not make a purchase. This target represents the cart abandonment scenario.
- The user will add item(s) to the cart and will make a purchase (purchase scenario).
- Neither of the events *add-item* and *make-order* will occur (browse only).
- The user will make a purchase (event). Note that this last scenario is possible when an *add-cart*

event has already occurred in the user navigation history. Subsequently, in the remaining events of the session, an order is completed without an add-cart event.

4 RESULTS

We conducted several tests to find the best hyperparameter tuning. We used 10-fold stratified cross-validation and we concluded that using 0.2 Dropout rate, Adam Optimizer and 500 LSTM units produces the best performance. The results using this specific parameter tuning and window size N=10 are shown in Table 3. Binary accuracy is the evaluation metric we use. We notice that LSTM combined with a GRU layer give the best performance compared to only one LSTM layer. This finding is in line with other research for sequence modeling (Chung et al, 2014) variants such as LSTM and GRU regularly outperform standard recurrent units. Both models end up to a Dense Layer with sigmoid function to give the final probabilities for multilabel prediction.

Table 3: Accuracy Results (Window size=10).

Model	Units	Extra features used	Binary Accuracy
LSTM	500	NO	97%
LSTM	500	YES	97.3%
LSTM+GRU	500 X 300	NO	97.2%
LSTM+GRU	500 X 300	YES	97.6%

The results obtained from our method are better with other research works using RNN-LSTM. Also they are very much comparable to the accuracy results that other SotA methods have achieved in similar tasks. However, these methods were tested in other datasets, therefore a direct comparison cannot be done.

We conducted more experiments to test our model using various window sizes. We also wanted to experiment particularly with the cart abandonment sessions. These sessions are of particular interest for e-commerce applications because the user adds item(s) in the cart, but s/he does not make a purchase. Apparently, any e-commerce application would benefit, if these sessions can be effectively predicted, as soon as possible during a session. Table 4 presents the Binary Accuracy results using various window sizes when: a) all sessions are included, and b) when only the cart-abandonment sessions are considered.

We can see in Table 4 that the classification of cart abandonment sessions is significantly less

effective, in comparison our method when all sessions are considered. Similar findings have been seen in other studies of the same task due to the nature of the dataset which contains many Browse only sessions and very few cart abandonment and purchase session.

Table 4: Accuracy results as a function of Window size.

Window size	Binary Accuracy (all sessions)	Binary Accuracy (Cart Abandonment sessions)
4	94.64	56.40
8	96.35	71.00
11	97.45	78.60
14	97.82	81.17
20	98.15	83.16
25	98.14	82.65
30	98.12	83.30

5 CONCLUSIONS

We presented an RNN-LSTM classification method and model for predicting users' purchase intent in e-commerce applications. Our work on this model was mainly driven by the need to produce a system for effective predictions that is easy to maintain, re-train and update. Our model produces very good results, achieving performance very close to the SotA methods reported in the literature, although a direct comparison cannot be made because our task is not modeled in the same way. However, it is important to point out that we manage this performance without using any application explicit features, or excessive feature engineering.

The model is rather straightforward to implement, and there is nothing foreseeable to prevent it from easily generalizing to different datasets and e-commerce applications with similar performance. Also, it can be trained with modest hardware resource requirements and can also provide predictions in real-time.

Our work was mainly inspired by the idea that e-commerce web applications should have components for continuously monitor users during their navigation. We believe that the primary features such components should incorporate into the e-commerce application are: customer short history, collective experience from community purchase behavior and, a set of proactive stimuli actions offering buying incentives to the user. All these features should be integrated into an end-to-end framework that can be deployed cost-effectively in small and medium size e-commerce applications.

We believe that we have demonstrated the feasibility of producing such a framework. However, there are aspects of the framework that we wish to explore and develop further.

In conclusion then we feel that in this paper we have already demonstrated a dynamic method based on RNN-LSTM for effectively predicting purchase behavior in e-commerce. This method could become the starting point for developing more complex frameworks for e-commerce applications that will aim at higher conversion rates and better profitability.

ACKNOWLEDGEMENTS

This research has been co-financed by the European Regional Development Fund of the European Union and Greek national funds through the Operational Program Co, under the call RESEARCH-CREATE-INNOVATE (project code: T1EDK-01776).

REFERENCES

- Mikhail Ageev, Qi Guo, Dmitry Lagun, and Eugene Agichtein. 2011. Find it if you can: a game for modeling different types of web search success using interaction data. In Proceedings of the 34th international ACM SIGIR conference on Research and development in Information Retrieval (SIGIR '11). Association for Computing Machinery, New York, NY, USA, 345–354.
- Mehdi Hosseinzadeh Aghdam, Negar Hariri, Bamshad Mobasher, and Robin Burke. 2015. Adapting Recommendations to Contextual Changes Using Hierarchical Hidden Markov Models. In Proceedings of the 9th ACM Conference on Recommender Systems (RecSys '15). Association for Computing Machinery, New York, NY, USA, 241–244.
- Hidasi Balázs and Karatzoglou Alexandros. Recurrent Neural Networks with Top-k Gains for Session-based Recommendations. In Proceedings of the 27th ACM International Conference on Information and Knowledge Management (CIKM '18). Association for Computing Machinery, New York, NY, USA, 2018, 843–852 (2018).
- Carmona CJ, Ramirez-Gallego S, Torres F, Bernal E, del Jesus MJ, Garcia S (2012) Web usage mining to improve the design of an e-commerce website: OrOliveSur. com. Expert System with Applications 39(12):11243–11249
- Ding AW, Li S, Chatterjee P (2015) Learning user real-time intent for optimal dynamic web page transformation. Inf Syst Res 26(2):339–359.
- C. Ling, T. Zhang and Y. Chen, "Customer Purchase Intent Prediction Under Online Multi-Channel Promotion: A Feature-Combined Deep Learning Framework," in IEEE Access, vol. 7, pp. 112963-112976, 2019
- Wendy W. Moe, Buying, Searching, or Browsing: Differentiating Between Online Shoppers Using In-Store Navigational Clickstream, Journal of Consumer Psychology, Volume 13, Issues 1–2, Pages 29-39, 2003.
- C Okan Sakar; Polat, S Olcay; Katircioglu, Mete; Kastro, Yomi. Neural Computing & Applications; Heidelberg Vol. 31, Iss. 10, (Oct 2019): 6893-6908.
- Grażyna Suchacka and Grzegorz Chodak. 2017. Using association rules to assess purchase probability in online stores. Inf. Syst. E-bus. Manag. 15, 3 (August 2017), 751–780.
- Suchacka, G., M. Skolimowska-Kulig and A. Potempa. "Classification Of E-Customer Sessions Based On Support Vector Machine." ECMS (2015).
- Germanas Budnikas. "Computerised Recommendations On E-Transaction Finalisation By Means Of Machine Learning." Statistics in Transition New Series, Polish Statistical Association, vol. 16(2), pages 309-322, June, 2015.
- Peter Romov and Evgeny Sokolov. 2015. RecSys Challenge 2015: ensemble learning with categorical features. In Proceedings of the 2015 International ACM Recommender Systems Challenge (RecSys '15 Challenge). Association for Computing Machinery, New York, NY, USA, Article 1, 1–4.
- Michail Salampasis, Theodosios Siomos, Alkiviadis Katsalis, Konstantinos Diamantaras, Konstantinos Christantonis, Marina Delianidi and Pantelis Kaplanoglou. Comparison of RNN and Embeddings Methods for Next-item and Last-basket Session-based Recommendations. ICMLC 2021, Shenzhen, China, February 2021.
- Michail Salampasis & Konstantinos Diamantaras. Rich interactions in digital libraries: Short review and an experimental user-centered evaluation of an Open Hypermedia System and a World Wide Web information-seeking environment. Journal of Digital Information, BCS & Oxford University Press, ISSN 1368-7506, May 2002.
- Humphrey Sheil, Omer Rana, Ronan Reilly. Predicting Purchasing Intent: Automatic Feature Learning using Recurrent Neural Networks. eCOM@SIGIR 2018.
- Junyoung Chung, Çağlar Gülçehre, KyungHyun Cho, and Yoshua Bengio. 2014. Empirical Evaluation of Gated Recurrent Neural Networks on Sequence Modeling. CoRR abs/1412.3555 (2014). arXiv:1412.3555 <http://arxiv.org/abs/1412.3555>
- Amy Wenxuan Ding, Shibo Li, and Patrali Chatterjee. 2015. Learning User Real-Time Intent for Optimal Dynamic Web Page Transformation. Info. Sys. Research 26, 2 (June 2015), 339–359.
- Shi, Yifan & Wen, Yimin & Fan, Zhigang & Miao, Yuqing. (2013). Predicting the Next Scenic Spot a User Will Browse on a Tourism Website Based on Markov Prediction Model. Proceedings - International Conference on Tools with Artificial Intelligence, ICTAI. 195-200. 10.1109/ICTAI.2013.38.