# Recommendation Systems in a Conversational Web

Konstantinos N. Vavliakis[1,2], Maria Th. Kotouza[1], Andreas L. Symeonidis[1] and Pericles A. Mitkas[1]
[1]*Department of Electrical and Computer Engineering, Aristotle University of Thessaloniki, Thessaloniki, Greece*
[2]*Pharm24.gr, Greece*

Keywords: Personalization, Recommendation, Conversational Web, e-Commerce, RFM, Recurrent Neural Networks.

Abstract: In this paper we redefine the concept of Conversation Web in the context of hyper-personalization. We argue that hyper-personalization in the WWW is only possible within a conversational web where websites and users continuously *"discuss"* (interact in any way). We present a modular system architecture for the conversational WWW, given that adapting to various user profiles and multivariate websites in terms of size and user traffic is necessary, especially in e-commerce. Obviously there cannot be a unique fit-to-all algorithm, but numerous complementary personalization algorithms and techniques are needed. In this context, we propose PRCW, a novel hybrid approach combining offline and online recommendations using RFMG, an extension of RFM modeling. We evaluate our approach against the results of a deep neural network in two datasets coming from different online retailers. Our evaluation indicates that a) the proposed approach outperforms current state-of-art methods in small-medium datasets and can improve performance in large datasets when combined with other methods, b) results can greatly vary in different datasets, depending on size and characteristics, thus locating the proper method for each dataset can be a rather complex task, and c) offline algorithms should be combined with online methods in order to get optimal results since offline algorithms tend to offer better performance but online algorithms are necessary for exploiting new users and trends that turn up.

## 1 INTRODUCTION

Personalization (or customization) systems focus on tailoring a service or a product to accommodate specific individuals, sometimes tied to groups of individuals. They have become increasingly popular in recent years and are considered key elements in a variety of areas including e-commerce, movies, music, news, research articles, search queries and social tags. They are broadly used for improving customer satisfaction, sales conversion and marketing results. In specific, web personalization dynamically serves the most relevant content, call-to-action elements, and messaging for stakeholders' unique interests. The importance of personalization in e-commerce has been undisputed, already from 1998 in an interview with the Washington Post; Jeff Bezos made a visionary statement about the web: *"If we have 4.5 million customers, we shouldn't have one store. We should have 4.5 million stores"*.

Nowadays it is state of practice for web companies some sort of personalization in their websites, based on IP address, browser language, the referral link and user's history. The vast majority of them employ customer profiles which sometimes may be dynamic. This means that personalization targeted to segments of users rather than individuals requires specific steps in order to launch an effective strategy, such as to identify audience, understand visitors, plan and create different experience for each audience.

Recently a new term has appeared, *"hyper personalization"*, defined as: *"the use of data to provide more personalized and targeted products, services and content"*. Hyper-Personalization means to rethink customer interaction on a one-to-one basis, where we treat each and every customer uniquely and design a customized experience for each one. The key element for hyper-personalization is interacting one-to-one with individuals, not the customer segments they fall in. To anticipate an individual's desires at any point in time, however, requires having deep customer insight, which comes from analyzing granular and big data.

Hyper-personalization is the next era of digital marketing; emails that change content based on where a customer is and when the email is opened. Context-aware messages and segments that are build for more relevant communications with your customer, pushing only those messages he/she should like to receive, this way targeting to increased revenue. Although of



Vavliakis, K., Kotouza, M., Symeonidis, A. and Mitkas, P.
Recommendation Systems in a Conversational Web.
DOI: 10.5220/0006935300680077
In *Proceedings of the 14th International Conference on Web Information Systems and Technologies (WEBIST 2018)*, pages 68-77
ISBN: 978-989-758-324-7

added value, there are numerous reasons why hyper-personalization has not yet been adopted by the majority of websites. Some of these are: a) the over-abundance of non-actionable data, as most companies have an abundance of data but cannot use it to personalize digital experiences, b) not knowing who to personalize first, as content is locked up in a content management system and controlled by developers, while visitor data is not available for targeting in real time, c) difficulties in measuring the impact of personalization, as companies often lack a direct way to measure the aggregate effect of that portfolio of customized content across their site over time.

Alongside *"hyper-personalization"* another term, *"conversational web"* has recently started to be used in the context of user interfaces, also known as chatbots or virtual assistants, as well as in the context of web services. Conversation interfaces interact with users combining chat, voice or any other natural language interface with graphical UI elements like buttons, images, menus, videos, etc. The new trend to evolve from NLP (natural language processing) to NLU (natural language understanding). On the other hand, conversational web services (CWS) refer to web services that communicate multiple times with a client to complete a single task. Conversations provide a straightforward way to keep track of data between calls and to ensure that the Web Service always responds to the correct client.

In this paper we redefine the term ***"Conversational Web"*** in the context of ***hyper-personalization***. Conversational Web refers to dynamic, multiple and asynchronous interactions (implicit conversations) between users and websites. These conversations allow both sides to understand each other and communicate efficiently. We argue that only in a truly conversational system is hyper-personalization possible, as in order to be able to create absolutely targeted messages, offers, interfaces, and recommendations that resonate and connect differently with each individual, one must first listen the needs and wills of each and every individual. This is only possible within a conversational web where websites and users continuously *"discuss"* (interact). This discussion takes place in the forms of clicks, mouse movement, scrolling, purchases, back or forward movements and time of each page on behalf of customers. On the other hand, websites *"hear"* customer's talking and respond in the form of personalized product recommendations, offers, coupons, order appearance in search, newsletters communications, popups and push notifications. Users in turn react to these responses and a new cycle of communication begins.

In order to produce accurate predictions and recommendations, big data analysis is necessary for identifying trends and patterns in data. This analysis can only take place in offline mode as it is both a time and resource consuming process. On the other hand new customers, products and trends continuously emerge, thus achieving hyper-personalization requires more than just analysis of historical data. The "discussion" between users and websites should continuously be analyzed for improving customer experience, and online analysis should also take place and complement the results of the offline processes.

In this context, we propose a modular architecture for conversational websites. We acknowledge that the conversational web needs to adapt to various user profiles and independent websites with varying context, size and user traffic, thus there cannot be a unique fit-to-all algorithm, but numerous complementary personalization algorithms and techniques are required, as well as a framework to decide when and where to use each algorithm. For this reason, we propose PRCW (Product Recommendations for Conversation Web), a novel hybrid approach combing offline and online recommendations using RFMG (Recency-Frequency-Monetary-Gender), an extension of the well-known RFM method. Through PRCW, modeling and partial matching recommendations can be combined with existing deep neural networks and provide improved results. We evaluate the proposed methodology on two discrete datasets, with different characteristics to test how the proposed method performs. Then we combine the proposed method with the deep neural network and we show that this combination leads to improved results.

The remainder of this paper is structured as follows. Related work on personalization and recommender systems, is discussed in Section 2. Section 3 describes in detail a framework for the *Conversational Web*, while Section 4 introduces a novel hybrid approach for recommendations, which is evaluated in Section 5. Section 6 summarizes work done, discusses future work and concludes the paper.

## 2 RELATED WORK

Web personalization implies tailoring a website to accommodate specific individuals or groups of individuals. Recommender systems are key elements in almost every personalization system and are divided in online and offline systems. Offline recommendation systems (Koren et al., 2009) either consisting of content-based recommendations (Pazzani and Billsus, 2007) or collaborative filtering (Sarwar et al., 2001), have weaknesses. They require significant training time; data updates usually require retraining the whole model and cannot take into account frequent changes in interests and profile of users. In more detail, the techniques used in the field of recommendation systems can be categorized into four general types: content-based filtering (CBF); collaborative filtering (CF); rule-based approaches; hybrid approaches (Ranjbar Kermany and Alizadeh, 2017). Collaborative Filtering techniques look for patterns in the overall user activity to produce recommendations, and can be further categorized into Neighborhood-based, Model-based, Clustering and Association Rules methods. In recent years, with the rise of big data, deep model-based approaches have been applied in this field with promising results.

Although session-based recommendation was until recently a relatively unappreciated problem, in the last few years it has attracted interest (Hidasi et al., 2015). This is because the behavior of users shows session-based traits, or users often have only one session. Recommendation systems widely use factor models (Koren et al., 2009) or neighborhood methods (Sarwar et al., 2001). Factor models are hard to apply in session-based recommendation due to the absence of user profiles, while neighborhood models, such as item to item similarity, ignore the information of the past clicks.

Drawbacks of offline recommender systems have been acknowledged and for that reason various online recommender methods (Ying et al., 2006) have been proposed, which need less processing power and do not require training. Nevertheless, online recommenders are less accurate than offline methods, thus hybrid approaches (Burke, 2002) have been proposed that combine the advantages of online and offline recommendation methods. Preference elicitation is also a popular personalization technique. In the context of preference elicitation, questionnaires, reviewing pre-selected items, dynamic learning (Rubens et al., 2011), entropy optimization (Salimans et al., 2012) and latent factor models (Huang, 2011) have been employed. Nevertheless, preference elicitation is not always efficient and it is recommended only in specific problems (Zhao et al., 2013). Interactive systems are another popular group of methods relative to our case. In interactive systems users play an active role, they are usually based on reviews (Chen and Pu, 2012), constrains (Felfernig et al., 2011), and questionnaires (Mahmood and Ricci, 2009). A common method used in interactive systems, is when users are asked to review a predefined selection of items, in order to cope with the cold-start problem. These requirements may frustrate users.

Recently deep learning and especially recurrent neural networks allow sequential data modeling and have shown remarkable results (Quadrana et al., 2017). Embedding deep learning techniques into recommender systems is gaining traction due to its state-of-the-art performances and high-quality recommendations. Deep learning (Zhang et al., 2017) provides a better understanding of user's demands, item's characteristics, historical interactions and relationships between them than traditional methods do. Recursive Neural Networks (RNNs) (Goodfellow et al., 2016) are a family of neural networks suitable for modeling variable-length sequential data and can scale to much longer sequences than other neural networks. Unlike feed-forward neural networks, RNNs use loops which allow information to be passed from one state of the network to the next one. This way former computations and previous states of the sequence are remembered and taken into account. Variants such as Long Short Term Memory (LSTM) and Gated Recurrent Unit (GRU) network are often deployed to overcome the vanishing gradient problem.

Obviously a lot of progress has been made in personalization and recommendation systems; however there is no integrated solution that can semantically understand user's intentions and dynamically evolve based on them. Advancing the state-of-the-art we propose the combination of state-of-the art deep learning techniques, which have shown impressive results, with online recommenders, that use partial matching. This combination not only achieves better results, but is also more adaptable.

## 3 A FRAMEWORK FOR THE CONVERSATIONAL WEB

Although from the system point of view, creating a truly conversational website involves a rather complex multi-step procedure; from a user point of view an e-commerce website supporting conversation web technologies is just an ordinary website. The only difference is that somehow it seems so much easier to use and find products and everything seems simple and intuitive both in terms of UX elements and product search.

### 3.1 A Use-case Scenario

Consider a customer, Irene that wants to buy the new brand X1 night face cream. Irene performs a web search and clicks the first result that redirects her to an e-commerce site she has never visited before. At this point the implicit conversation between the customer and the user has already begun. The website "listens" that a new customer landed from an organic source, searching for brand X1 night cream, so it responds with recommendations about other night creams that are popular among users coming from this source type, together with other Brand X1 products, as the user seems interested to this brand. In addition, the site recognizes that this is a new user, so it displays the "subscribe to our newsletter" banner in a more prominent location. Next, Irene adds the product to her basket and then hovers for some time over a shampoo for oily hair, but finally clicks on a brand X2 serum she noticed in a banner of the main page. These actions alone comprise four discrete messages: as the user has stated that she is actually a) very interested in the brand X1 night cream (with intent to buy), b) she is also interested in general for brand X2 and c) more specifically in serums, and d) she may need a shampoo for oily hair.

The website once again "listens" and responds with even more personalized results as it quickly learns the interests of the user, for example it recommends cheaper shampoos for oily hair as the ones displayed before are considered premium products and are probably too expensive. In case Irene clicks on a cheaper shampoo the website will classify Irene as a customer interested in mid-level products (at least until she starts showing interest for premium products). This is a continuous and everlasting process; the website not only adapts to better serve Irene's interest but also learns from her behavior and the behavior of other users, aggregating this collective wisdom into actionable insights for improving the overall e-commerce UX of the site.

### 3.2 The Proposed Framework

Next, we propose an integrated framework for creating a conversational website that consists of four pylons: a) behavior analysis; b) user experience analysis; c) big data warehousing and d) personalization. Figure 1 illustrates the overall proposed architecture, detailed at a module level, as well as the data exchange means between subsystems.

a) The Behavior analysis module is responsible for dynamically analyzing user behavior. Data from analytics tools (e.g. Google Analytics, Yandex, etc.) along with scroll maps and mouse gestures should be combined in order to effectively recognize different patterns and user groups, such as novice or experienced users, users that are just browsing or intent to buy, and categories, brands or specific products users are interested in. For this task, classification and support vector machines have provided eminent results in the past (Sun et al., 2002), while recently deep learning and especially recurrent neural networks have shown improved performance. Semantic analysis is also required, as topic modelling and latent dirichlet allocation are useful for analyzing user's interests

b) User experience analysis is necessary in order to better understand users and be able to adapt to their needs. User experience is a multifactor parameter, such as website structure, marketing, trust, interactive and information elements, colors and ease of use. All these factors are hard to be defined as they contain strongly subjective elements, key performance indicators, such as bounce rate, average time on site, conversion rate, and depth of search can provide accurate metrics for calculating user experience.

c) Considering special requirements for data warehousing is necessary in order to build a conversational system. Due to the nature of conversation, which is continuous, lengthy and heterogeneous, data warehousing should be able to cope with big data and extremely low response times in queries that will allow real time queries, as well as different type of information, including product data, user click history, mouse movements, scroll data, e-commerce data including buys, add to cart, and favorites, visual elements and statistics about their use. This information should be combined in offline operations where intelligent models will be trained, as well as in real-time situations for delivering personalized services and UI/UX. Luckily there are plenty of open source tools able to handle this type of data, including Apache Hadoop, Elastic Server, and MongoDB.

d) Finally, the key module of our proposed framework is the personalization module which is responsible for dynamically integrating information data and user actions originating from user experience and behavior and composing different recommendations, website user interfaces and content tailored to the individual needs of every visitor. Time performance is crucial for this step, as most operations are real-time. This step also includes feedback and dynamic learning using user-website conversations, thus it's a self-improving process.

## 4 A HYBRID APPROACH FOR RECOMMENDATIONS

In this Section we propose a hybrid approach for product recommendations in e-commerce sites that we call ***PRCW*** (Product Recommendations for the Conversational Web). As there is not a universally good solution that can fit all circumstances and solve any problem in product recommendation, different

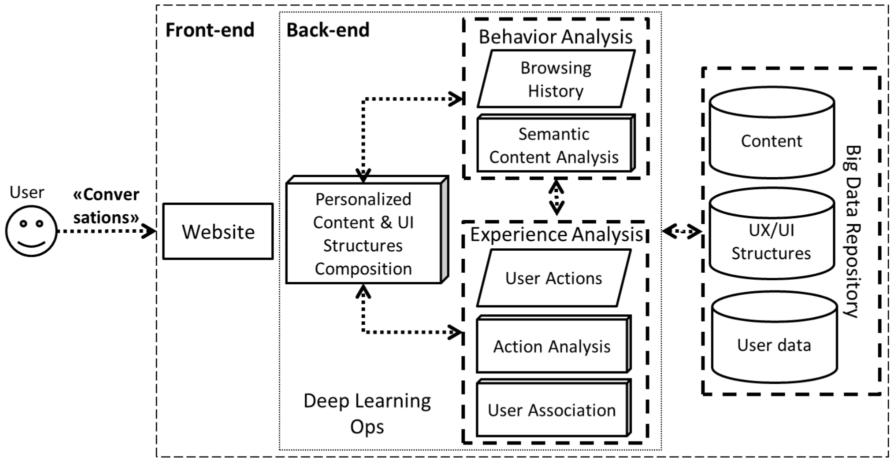


Figure 1: Overall system architecture in a conversational website (detailed at module level).

approaches have to be used depending on the dataset and the kind of the target e-commerce site. One of the main parameters that have to be taken into account is the size of the e-shop in terms of traffic, number of orders and available products. For this reason, we propose a hybrid approach and we apply a deep model to two different click stream datasets, one originating from a small-to-medium e-commerce site and another coming from a large European retailer's website.

Recommendations need to satisfy two fundamental principles in a conversational e-commerce site, a) must be relevant and b) must be provided in real-time. Thus, hybrid approaches are required, that can provide recommendation in online-mode and date processing for improving results in offline-mode. For this reason, we introduce a new hybrid approach using offline and online processing that combines a clustering algorithm with a rule-based method. Clustering is applied to perform consumer segmentation based on consuming behavior, using RFMG, a modified version of RFM modeling that combines recency, frequency and monetary with gender, whereas the proposed rule based approach that uses four different partial matching processes focuses on solving the problem of unknown user history.

## 4.1 Offline Phase

The offline phase consists of data preprocessing, clustering via RFMG analysis and post-processing analysis. Figure 2 depicts this phase. Data Preprocessing: Data preprocessing is necessary to make knowledge discovery easier and more accurate. In this step, data are processed in order to follow the desired format, attributes are selected, and auxiliary operations like outlier detection, normalization and discretization are performed. Users from whom information is not of adequate value (e.g. users that have only one or even no product views) are removed from the dataset.

**RFMG Analysis:** RFM (Recency, Frequency, Monetary) analysis is a marketing model that provides information about customers' consumption behavior and widely used for customer segmentation (Birant, 2011). The three variables are computed on the transaction history and measure how recently, how often and how much do the consumers buy. These three attributes are not only computed on the product orders (RO, FO, MO), but on the product views as well (RV, FV, MV), because of the more extensive amount of information that page views provide. In our work we extend the traditional RFM model to the RFMG model, by adding the Gender attribute, as gender is a major factor for decision making in almost every e-commerce environment. The six attributes are defined as follows:

- RO/RV: the number of days passed since the customer last viewed/purchased a product. Range [0,d]
- FO/FV: the number of purchases/product views made by the customer in the last d days. Range [0,f]
- MO/MV: the summary of the prices of the products that were ordered/viewed by the customer in the last d days. Range [0,m]

If an attribute value is higher than the maximum upper threshold, the maximum allowed value is used instead. Then, normalization is performed in the range [0,1].

**Clustering:** Consumer segmentation is an unsupervised machine learning process that allows

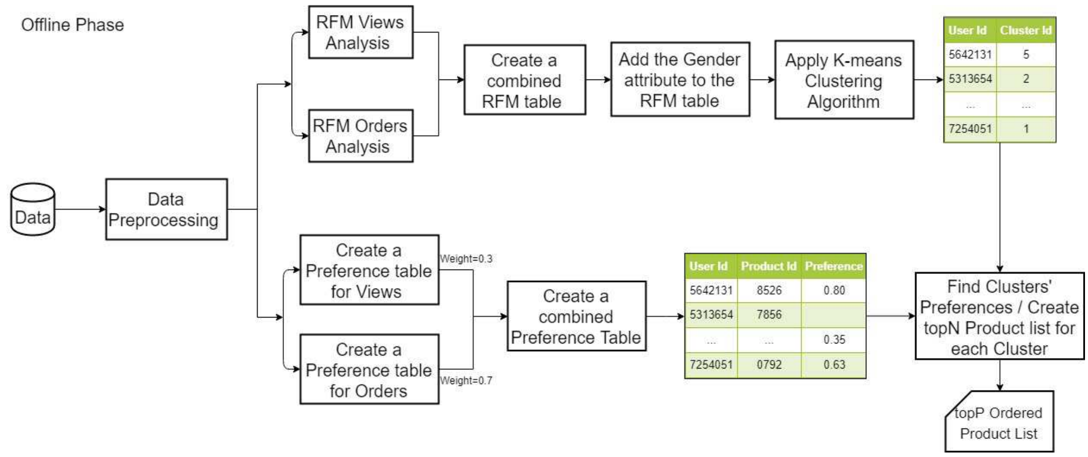


Figure 2: Offline phase of PRCW.

understanding consumer characteristics and grouping consumers according to their behavior. Thus, it's an excellent start for designing personalized solutions according to each customer group's needs.

**Preference Table Creation:** A user's preference for an item is obtained by the number of times the user has viewed or ordered the target product. Product list extraction: For each consumer segment, a top list of ordered products is extracted by finding the most preferred items of the users that belong to the particular cluster.

## 4.2 Online Phase

The proposed online phase of our approach is depicted in Figure 3. Prediction by Partial Matching (PM), or else known as Markov Model, is a method used to predict the next state of the model taking into account the n previous states (Gellert and Florea, 2016). This means that the current state depends on the previous n states. The number of previous states determines the order of the PM model. Assuming that $q_t$ is the state at time t, an R-order model is defined as in Eq 1.

$$P[q_t|q_{t-1},...,q_1] = P[q_t|t_{t-1},...,q_{t-R}] \quad (1)$$

In our case the states are represented by product views. When the target user views the product $q_t$, partial matching can be used to find the pattern $< q_{t-1}, q_t >$ within the history of all the users. The products found to follow the matched pattern are saved and their frequencies are computed in order to extract the topM products. Needless to say, when the order of the model, R, increases, the possibilities to find the desired pattern into the history becomes lower. So, in our situation the second ordered model is used.

However, because of the limited number of observations in smaller e-shops, the non-matching pattern possibility remains high. In order to address this problem, we introduce two elastic variants of the partial matching procedure:

- The first one is called PM by intervals and looks for the pattern $< q_{t-1}, ..., q_t >$ within the history, with the constriction that the time interval between the product views $q_{t-1}$ and $qt$ is less than a time period T. In this case, the topM list is computed using the products that were viewed within the time period T and after the product view $q_t$.
- The second one is called PM by session and looks for the pattern $< q_{t-1}, ..., q_t >$ within the history, with the constriction that the product views $q_{t-1}$ and $q_t$ occurred within the same session. The topM list is computed using the products that were viewed within the same session and after the product view $q_t$.

Assume that the target user views the sequence data $< i_9, i_1 >$. If sessions [Session1-Session5] have been extracted by the history, the topM recommendation list using the four algorithms is presented in Table 1.

Session1: <i3><i5>**<i1>**<i2>
Session2: <i4>**<i9><i1>**<i3>
Session3: <i6><i4><i9><i4>
Session4: <i4>**<i1>**<i2><i6>
Session5: **<i9>**<i2>**<i1>**<i4>

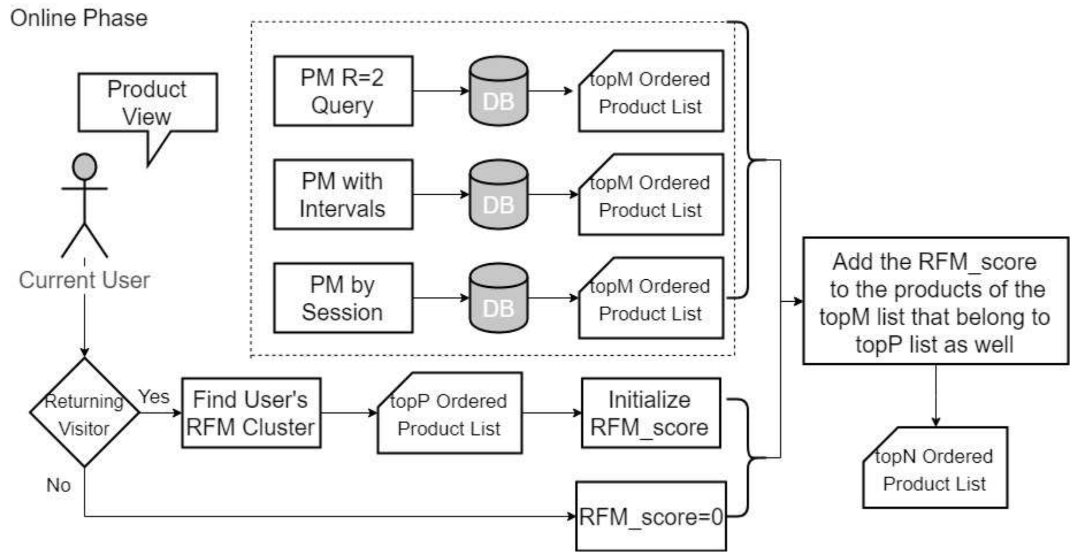


Figure 3: Online phase of PRCW.

Table 1: Example of topM recommendation list using the different PM algorithms.

| Method | Recall@1Next | Recall@AllNext | PrecisionR |
|---|---|---|---|
| **PM R=1** | i2 40% | i3 20% | i4 20% |
| **PM R=2** | i3 20% | - | - |
| **PM by intervals** | i3 20% | i4 30% | - |
| **PM by session** | i2 40% | i3 20% | i4 20% |

# 5 EXPERIMENTAL EVALUATION

## 5.1 Experimental Setup

We evaluate the proposed hybrid recommendation method on two different datasets. The first dataset originated from Pharm24.gr, a small-medium (in terms of traffic) retailer in Greece. The available click stream contained data from a period of 9 months. Data from the first 7 months were used as the training set whereas data from the last 2 months where used as the test set. Items with less than 5 views were filtered out from the training set, as well as sessions with less than two item views. Sessions with less than one item view were also removed from the test set, as well as item views that do not exist in the training set. After the preprocessing, the training set contained 53,071 sessions of 875,366 events and 9,733 items, whereas the test set contains 86 sessions of 585 events and 244 items.

The second dataset is the RecSys dataset that was provided for the RecSys Challenge 2015 (Ben-Shimon et al., 2015). This dataset contains click-streams of a big e-commerce site, organized in sessions. The training set contains all but the last 10 days of the dataset, whereas the test set contains the sessions of the last 10 days. After the same preprocessing phase, the training set contains 7,802,137 sessions of 30,958,148 events and 37,331 items, while the test set contains 71,060 sessions of 217,014 events and 10,829 items. The evaluation was performed by providing the events of each session of the test set one by one and making recommendations applying the proposed algorithm to the training set (Algorithm 1).

## 5.2 Evaluation Metrics

Precision and recall, two commonly used metrics in the field of recommender systems were employed for evaluating the performance of our algorithms. Suppose that U is the set of users that are examined, R(u) is the set of items recommended to user u, V(u) is the set of items viewed by user u after the recommendation and V(u,1) is the first product that user u viewed after the recommendation. We define *PrecisionR* (Eq. 2) as the percentage of recommended items viewed by the user over the number of recommended products and *PrecisionV* (Eq. 3) as the percentage of recommended items viewed by the user

$$PrecisionR = \frac{\sum_u |R(u) \cap V(u)|}{\sum_u |R(u)|} \quad (2)$$

Algorithm 1: Partial pseudo-code for "UserPersonomy".

```
Input: (test set: table Tx3 [user_id, timestamp, item])
Output: (topM1, topM2, topM3, topP, topN, next_views)
#Preprocessing
1. Filter out products with less than 5 views
2. Filter out sessions with less than 2 products
3. Separate the dataset into training and test set
4. Filter out products from the test set that do not belong to the training set
#Iteration
For each user_id-u:
   For each timestamp-t:
      Find current item i_c , previous item i_{c-1} and the next items i_{c+1}, .., i_n
      Perform PM R2 using i_c i_{c-1}: topM1 list is returned
      Perform PM by intervals using i_c i_{c-1}: topM2 list is returned
      Perform PM by session using i_c i_{c-1}: topM3 list is returned
      If user_id[u] belongs to any clusterRFM then
         Get the topP[u,t]  list from the corresponding element of clusterRFM
         Add a score to topM[u,t] belonging to topP[u,t] and (topM[u,t] & topP[u,t])
      End_if
      Merge topM1[u,t], topM2[u,t], topM3[u,t]  lists
      Update (topN[u,t] list, next_views[u,t] list< i_{c+1}, .., i_n >)
   End_For
End_For
Return (topN, next_views)
```

$$PrecisionV = \frac{\sum_u |R(u) \cap V(u)|}{\sum_u |V(u)|} \quad (3)$$

We define recall as the percentage of users that viewed recommended items at next timestamps. Three variants of recall are defined: *Recall@1Next* (Eq. 4), the strictest one, which determines only the first next view after recommendation, *Recall@AllNext* (Eq. 5), which determines all next views after recommendation, and *Recall@Positive* (Eq. 6), which considers only the cases the recommendation list has at least one item. The evaluation process is depicted in Algorithm 1.

$$Recall@1Next = \frac{\sum_u |R(u) \cap V(u,1)|}{|U|} \quad (4)$$

$$Recall@AllNext = \frac{\sum_u |R(u) \cap V(u)|}{|U|} \quad (5)$$

$$Recall@Positive = \frac{\sum_u |R(u) \cap V(u)|}{\sum_u |R(u) \neq 0|} \quad (6)$$

### 5.3 Results

Tables 2 and 3 present the results achieved by the proposed algorithm PRCW, the RNN and the combination of them using the Pharm24.gr and the RecSys dataset, accordingly. For deep model evaluation we used a GRU-based RNN model (Hidasi et al., 2015) for session-based recommendations. The input of the network was the actual state of the session represented by a 1-of-N encoding, where N is the number of items (a vector with 1 to the active items and 0 elsewhere), and the output was the likelihood for each item to be part of the next session. Session-parallel mini-batches and mini-batch based output sampling were used for the output.

Taking into account Tables 2 and 3 one can observe that the RNN model could not achieve good enough results in a smaller and sparse dataset, while the proposed approach not only demanded considerable less RAM and CPU recourses, but also performed better as PRCW achieved better results than RRN for the Pharm24 dataset, both in terms of Recall and Precision. On the other hand, the RNN has better performance in the RecSys dataset which contains more data both in terms of quantity and density. Nevertheless, the combination of both methods (PRCW+RNN) achieves improved performance in both datasets.

Comparing the results, one can better understand the difference between the algorithms and datasets. Bigger datasets have improved chances to get better recommendations, due to the larger amount of infor-

Table 2: Results of the Pharm24 dataset using the hybrid approach.

| Method | Recall@1Next | Recall@AllNext | Prec.R | Prec.V | Pos.Recall |
|---|---|---|---|---|---|
| **PRCW** | 0.2880 | 0.5247 | 0.0518 | 0.1414 | 0.5247 |
| **RNN** | 0.1993 | 0.3101 | 0.0348 | 0.0936 | 0.3101 |
| **PRCW+ RNN** | 0.3901 | 0.6065 | 0.0734 | 0.1737 | 0.6065 |

Table 3: Results of the RecSys dataset using the hybrid approach.

| Method | Recall@1Next | Recall@AllNext | Prec.R | Prec.V | Pos.Recall |
|---|---|---|---|---|---|
| **PRCW** | 0.0868 | 0.1711 | 0.0273 | 0.0229 | 0.1711 |
| **RNN** | 0.8120 | 0.8886 | 0.0998 | 0.6380 | 0.8886 |
| **PRCW+RNN** | 0.8366 | 0.9037 | 0.1139 | 0.7069 | 0.9037 |

mation that contain, and achieve worse results at the PrecisionR metric, as there are too many products in the dataset. On the other hand, smaller datasets have shorter sessions and achieve worse results at the PrecisionV metric. Deep learning can perform exceptionally well, as long as there are enough data and processing power to feed the neural network. On the other hand, the proposed method PRCW works better on smaller datasets. In any case combining both PRCW and RNN delivers the best results in both datasets, which leads us to the conclusion that both methods deliver useful results that should be combined for optimal performance.

## 6 CONCLUSION

In this paper we redefined the concept of Conversation Web in the context of hyper-personalization. We proposed a generic design for conversational web that may be expanded in terms of hyper-personalization, such as product recommendation, UI/UX personalization, as well as individual messages and promos per customer. We argued that in a high-demanding and versatile environment, such as the WWW there is not a unique fit-to-all solution, thus various solutions have to be evaluated and blended in order to provide relevant results in live environments. For this reason, we proposed a novel hybrid method that extends the RFM model by introducing the Gender factor, combined with Partial Matching. The method provided improved results for small-to-medium datasets. In addition, we combined the proposed algorithm with a deep learning method and we showed that they can work complementary as improved results are achieved when combined of both methods.

Future work includes working on the decision module for applying the optimum algorithm based on the dataset characteristics. In addition, we plan to explore the possibility of further integrating our hybrid approach with RNNs. Another issue for future consideration is privacy concerns that may arise and how to tackle them.

## ACKNOWLEDGEMENTS

This work was partially funded by an IKY scholarship funded by the “Strengthening of Post-Academic Researchers” Act from the resources of the OP “Human Resources Development, Education and Lifelong Learning” with Priority Axes 6,8,9 and co-funded by the European Social Fund ECB and the Greek government.



## REFERENCES

Ben-Shimon, D., Tsikinovsky, A., Friedmann, M., Shapira, B., Rokach, L., and Hoerle, J. (2015). Recsys challenge 2015 and the yoochoose dataset. In *Proceedings of the 9th ACM Conference on Recommender Systems*, RecSys ’15, pages 357–358, New York, NY, USA. ACM.

Birant, D. (2011). Data mining using rfm analysis. In Funatsu, K., editor, *Knowledge-Oriented Applications in Data Mining*, chapter 6. Rijeka.

Burke, R. (2002). Hybrid recommender systems: Survey and experiments. *User Modeling and User-Adapted Interaction*, 12(4):331–370.

Chen, L. and Pu, P. (2012). Critiquing-based recommenders: survey and emerging trends. *User Modeling and User-Adapted Interaction*, 22(1):125–150.

Felfernig, A., Friedrich, G., Jannach, D., and Zanker, M. (2011). *Developing Constraint-based Recommenders*, pages 187–215. Springer US, Boston.

Gellert, A. and Florea, A. (2016). Web prefetching through efficient prediction by partial matching. *World Wide Web*, 19(5):921–932.

Goodfellow, I., Bengio, Y., and Courville, A. (2016). *Deep Learning*. MIT Press.

Hidasi, B., Karatzoglou, A., Baltrunas, L., and Tikk, D. (2015). Session-based recommendations with recurrent neural networks. *CoRR*, abs/1511.06939.

Huang, S.-L. (2011). Designing utility-based recommender systems for e-commerce: Evaluation of preference-elicitation methods. *Electron. Commer. Rec. Appl.*, 10(4):398–407.

Koren, Y., Bell, R., and Volinsky, C. (2009). Matrix factorization techniques for recommender systems. *Computer*, 42(8):30–37.

Mahmood, T. and Ricci, F. (2009). Improving recommender systems with adaptive conversational strategies. In *Proceedings of the 20th ACM Conference on Hypertext and Hypermedia*, HT '09, pages 73–82, New York, NY, USA. ACM.

Pazzani, M. J. and Billsus, D. (2007). *Content-Based Recommendation Systems*, pages 325–341. Springer Berlin Heidelberg, Berlin, Heidelberg.

Quadrana, M., Karatzoglou, A., Hidasi, B., and Cremonesi, P. (2017). Personalizing session-based recommendations with hierarchical recurrent neural networks. In *Proceedings of the Eleventh ACM Conference on Recommender Systems, RecSys 2017, Como, Italy, August 27-31, 2017*, pages 130–137.

Ranjbar Kermany, N. and Alizadeh, S. H. (2017). A hybrid multi-criteria recommender system using ontology and neuro-fuzzy techniques. *Electron. Commer. Rec. Appl.*, 21(C):50–64.

Rubens, N., Kaplan, D., and Sugiyama, M. (2011). *Active Learning in Recommender Systems*, pages 735–767. Springer US, Boston.

Salimans, T., Paquet, U., and Graepel, T. (2012). Collaborative learning of preference rankings. In *Proceedings of the Sixth ACM Conference on Recommender Systems*, RecSys '12, pages 261–264, New York, NY, USA. ACM.

Sarwar, B., Karypis, G., Konstan, J., and Riedl, J. (2001). Item-based collaborative filtering recommendation algorithms. In *Proceedings of the 10th International Conference on World Wide Web*, WWW '01, pages 285–295, New York, NY, USA. ACM.

Sun, A., Lim, E.-P., and Ng, W.-K. (2002). Web classification using support vector machine. In *Proceedings of the 4th International Workshop on Web Information and Data Management*, WIDM '02, pages 96–99, New York, NY, USA. ACM.

Ying, Y., Feinberg, F., and Wedel, M. (2006). Leveraging missing ratings to improve online recommendation systems. *Journal of Marketing Research*, 43(3):355–365.

Zhang, S., Yao, L., and Sun, A. (2017). Deep learning based recommender system: A survey and new perspectives. *CoRR*, abs/1707.07435.

Zhao, X., Zhang, W., and Wang, J. (2013). Interactive collaborative filtering. In *Proceedings of the 22Nd ACM International Conference on Information & Knowledge Management*, CIKM '13, pages 1411–1420, New York, NY, USA. ACM.