Exploiting Additional Dimensions as Virtual Items on Top-N Recommender Systems

Marcos Aurélio Domingues
INESC Porto
Faculty of Science, UP
Porto, Portugal
marcos.a.domingues@inescporto.pt

Alípio Mário Jorge
LIAAD-INESC Porto L.A.
DCC-FCUP, UP
Porto, Portugal
amjorge@fc.up.pt

Carlos Soares
LIAAD-INESC Porto L.A.
Faculty of Economics, UP
Porto, Portugal
csoares@fep.up.pt

***Abstract*—Traditionally, recommender systems for the web deal with applications that have two dimensions, users and items. Based on access data that relate these dimensions, a recommendation model can be built and used to identify a set of N items that will be of interest to a certain user. In this paper we propose a multidimensional approach, called DaVI (*Dimensions as Virtual Items*), that enables the use of common two-dimensional top-N recommender algorithms for the generation of recommendations using additional dimensions (e.g., contextual or background information). We empirically evaluate our approach with two different top-N recommender algorithms, Item-based Collaborative Filtering and Association Rules based, on two real world data sets. The empirical results demonstrate that DaVI enables the application of existing two-dimensional recommendation algorithms to exploit the useful information in multidimensional data.**

***Keywords*-Recommender systems; multidimensional recommender systems; multidimensional data**

I. Introduction

Most web sites offer a large number of information resources to their users. Finding relevant content has, thus, become a challenge for users. Recommender systems have emerged in response to this problem. A recommender system for the web is an information filtering technology which can be used to output an ordered list of items/recommendations that are likely to be of interest to the user [1].

Traditionally, the data that are most often available for recommender systems are web access data that represent accesses from users to pages. Therefore, the most common recommender systems focus on these two dimensions. Based on access data that relate these dimensions, a recommendation model can be built and used to identify a set of N pages that are expected to be of interest to a certain user. However, other dimensions, such as time and type of content (e.g., genre of music the user is listening to a music portal) of the accesses, can be used as additional information. They may capture the context or background information in which recommendations are being made to improve their performance. For example, a system may recommend different vacation packages in summer or in winter.

In this paper we present a multidimensional approach, called **DaVI** (*Dimensions as Virtual Items*), that enables the application of traditional two-dimensional recommender algorithms for the generation of recommendations using additional dimensions. The **DaVI** approach was introduced in an earlier workshop paper [2], where preliminary ideas of the approach were described. In this paper we formalize the **DaVI** approach and describe an algorithm that instantiates it. We also evaluate the approach using two real world data sets and compare it against to the previously introduced combined reduction approach [3].

II. Dimensions as Virtual Items

The idea behind **DaVI** is to treat additional dimensions as virtual items, using them together with the regular items in a recommender system. Here, we assume that virtual items are only used to build the recommendation model and not recommended. On the other hand, regular items are used to build the model and they can also be recommended.

Let p be the number of users $U = \{u_1, u_2, ..., u_p\}$ and q the number of all possible items that can be recommended $I = \{i_1, i_2, ..., i_q\}$. In addition, we have other dimensions (e.g., contextual or background information), $\mathcal{D} = \{D_1, D_2, ..., D_t\}$, where each dimension D comprehends a set of values, i.e., $D = \{d_1, d_2, ..., d_f\}$. For example, the dimension $Hour$ can define a set of integer values from 1 to 24. Now, let j be the number of multidimensional sessions in a web site $S' = \{s'_1, s'_2, ..., s'_j\}$. Each session s' is a tuple defined by a user $u \in U$, a set of accessed items $I_{s'} \subseteq I$ and a set $D_{s'} \subseteq D_1 \cup D_2 \cup ... \cup D_t$ containing all the dimension values associated with the session s', i.e., $s' = \langle u, I_{s'}, D_{s'} \rangle$.

A multidimensional session can have two types of dimensions in terms of granularity: session-based dimensions and item-based dimensions. If a single dimension D is session-based, a session $s' = \langle u, I_{s'}, D_{s'} \rangle$ has a single dimension value (virtual item) $d \in D_{s'}$ associated to the session s'. Here, the dimension value d can represent, for example, the hour or location from where the session is accessed. On the other hand, if the dimension D is item-based, a session $s' = \langle u, I_{s'}, D_{s'} \rangle = \langle u, \{i_1, ..., i_q\}, \{d_1, ..., d_q\} \rangle$ has the dimension values (virtual items) $d_1, ..., d_q$ associated to respective items $i_1, ..., i_q$ in the session s'. For example, if the dimension values $d_1, ..., d_q$ represent the genre of songs in a music web site, we will have the values associated to

songs (items) in the session and not directly to the session as presented in the first case.

The **DaVI** approach consists in converting each multidimensional session $s' = \langle u, I_{s'}, D_{s'} \rangle$ into an extended two-dimensional session $s'' = \langle u, I_{s''} \cup D_{s''} \rangle$, where the values of the additional dimensions in $D_{s''}$ are used as virtual items together with the regular items in $I_{s''}$. The **DaVI** approach can also be applied to a subset of dimensions or even to a single dimension. For example, a multidimensional session $s' = \langle u, I_{s'}, D_{s'} \rangle = \langle u, \{i_1, ..., i_q\}, \{d_1, ..., d_q\} \rangle$, with a single dimension $D_{s'} \subseteq D_1$, can be converted into an extended two-dimensional session $s'' = \langle u, I_{s''} \cup D_{s''} \rangle = \langle u, \{i_1, ..., i_q, d_1, ..., d_q\} \rangle$. Thus, we define the **DaVI** approach as an operator that converts a set of multidimensional sessions into a set of extended two-dimensional sessions,

$$S'' = \textbf{DaVI}(S', \widehat{D}), \tag{1}$$

where S'' is the set of extended two-dimensional sessions, S' is the set of multidimensional sessions and $\widehat{D} \subseteq \mathcal{D}$ is a set indicating which dimension values in S' must be converted to virtual items.

Once we have a set of extended two-dimensional sessions S'', building/learning a multidimensional recommendation model M' consist in applying a two-dimensional recommender algorithm on S''. Then, to generate the recommendations, we use the multidimensional model M' providing to it with the items and additional dimensions (transformed in virtual items by the **DaVI** approach) from the active user session $s''_a = \langle u_a, I_{s''_a} \cup D_{s''_a} \rangle$ as follows:

$$R = M'(I_{s''_a} \cup D_{s''_a}), \tag{2}$$

where $I_{s''_a} \cup D_{s''_a}$ is referred to a set of observable items O, and it contains the items ($I_{s''_a}$) and dimension values ($D_{s''_a}$) which are, respectively, the regular and virtual items from the active user session s''_a. R is a set of items/recommendations, such that $R \subset I$ and $R \cap O = \varnothing$, that are the most relevant/interesting for the user u_a according to the model M'. As stated before, virtual items can not be recommended. Thus, we apply a filter on the recommendations generated by the model M' in order to guarantee that the model will never recommend virtual items.

One important advantage of our approach is that it can be combined with different recommendation methods. This means that **DaVI** makes it easy to apply existing recommender algorithms to multidimensional data and obtain multidimensional models without changing the algorithms.

III. DaVI-BEST Algorithm

An important issue with respect to the **DaVI** approach is to determine which dimensions should be considered in the recommendation model, given that some dimensions are more informative than others. In this section, we address this problem by proposing the **DaVI**-BEST algorithm that evaluates and selects the best dimension in a data set to build the multidimensional recommendation model.

To determine the best dimension for a given top-N recommender algorithm A, the **DaVI**-BEST algorithm first applies the **DaVI** approach on each candidate dimension and builds its respective multidimensional recommendation model. Then, it evaluates each model and selects the best dimension, i.e., the one whose recommendation model presents the best performance. The **DaVI**-BEST algorithm is presented in Algorithm 1.

Algorithm 1 DaVI-BEST algorithm

Input: A set of multidimensional sessions $S' = \{s'_1, s'_2, ..., s'_j\}$, where each session s' is a tuple defined by a user $u \in U$, a set of accessed items $I_{s'} \subseteq I$ and a set of dimension values $D_{s'} \subseteq D_1 \cup D_2 \cup ... \cup D_t$; A, a top-N recommender algorithm; n, the number of folds which are used to evaluate the multidimensional models; N, the number of recommendations generated during the evaluation of the models.

Output: $\overline{M}$, an object containing the final two-dimensional or multidimensional recommendation model.

1: $\mu_{\mathcal{F}} := \varnothing$; {F1 values, for each fold, calculated using the two-dimensional models}
2: $\mu'_{\mathcal{F},\mathcal{D}} := \varnothing$; {F1 values, for each fold and dimension, calculated using the multidimensional models}
3: $\mathcal{D}^+ := \varnothing$; {Set of pairs $< dimension, F1\ values >$ for informative dimensions}
4: $\mathcal{F} :=$ *create-folds*(S', n);
5: **for all** folds $F \in \mathcal{F}$ **do**
6: $\quad M_F := A(\mathcal{F} - F)$;
7: $\quad \mu_F :=$ *eval*(M_F, F);
8: **end for**
9: **for all** dimensions $D \in \mathcal{D}$ **do**
10: $\quad$ **for all** folds $F \in \mathcal{F}$ **do**
11: $\quad\quad M'_{F,D} := A($ **DaVI**$(\mathcal{F} - F, D)$ $)$;
12: $\quad\quad \mu'_{F,D} :=$ *eval*$(M'_{F,D},$ **DaVI**(F, D) $)$;
13: $\quad$ **end for**
14: $\quad$ **if** *t-test*$(\mu'_{\mathcal{F},D} > \mu_{\mathcal{F}}, \alpha = 0.05)$ **then**
15: $\quad\quad \mathcal{D}^+ := \mathcal{D}^+ \cup < D, \mu'_{\mathcal{F},D} >$;
16: $\quad$ **end if**
17: **end for**
18: **if** $\mathcal{D}^+ \neq \varnothing$ **then**
19: $\quad D^+ := argmax_{D^+ \in \mathcal{D}^+}[F1(D^+)]$;
20: $\quad \overline{M} := A($ **DaVI**(S', D^+) $)$;
21: **else**
22: $\quad \overline{M} := A(S')$;
23: **end if**
24: **return** $\overline{M}$;

Firstly, the algorithm sets to "$\varnothing$" (empty set) the variables $\mu_{\mathcal{F}}$, $\mu'_{\mathcal{F},\mathcal{D}}$ and $\mathcal{D}^+$. The variables $\mu_{\mathcal{F}}$ and $\mu'_{\mathcal{F},\mathcal{D}}$ denote the values of the evaluation measure (F1, in this case) for each fold in $\mathcal{F}$ and dimension in $\mathcal{D}$. The variable $\mathcal{D}^+$ stores the informative dimensions and respective F1 values in the form $< dimension, F1\ values >$. Here, a dimension is informative if its respective multidimensional model presents an F1 value significantly higher than the F1 value of the two-dimensional model. In line 4, the function *create-folds*

partitions the sessions into n folds which are used to evaluate the dimensions through their respective multidimensional recommendation models. The evaluation is carried out using the All But One protocol [4] with the n-fold cross validation technique [5].

The next step (lines 5-8) consists of building n two-dimensional models from all folds but one and evaluating them on the remaining fold. To build a model, we can use any two-dimensional top-N recommender algorithm. The function A builds a model and the function *eval* evaluates it against the validation set, calculating the F1 metric. These F1 values will be used as reference to evaluate the performance of the multidimensional models in lines 14-16.

The evaluation and selection of informative dimensions are performed in the lines 9-17 of the Algorithm 1. Lines 10-13 build the multidimensional models for each fold and evaluate them on the corresponding validation data (using the function *eval* to compute the F1 metric). The function **DaVI** converts a set of multidimensional sessions into a set of extended two-dimensional sessions, as described in Section II. Lines 14-16 analyze whether the F1 values of the multidimensional models are significantly higher than the F1 values of their respective pure two-dimensional models (without additional dimensions) or not. The analysis is performed using the function *t-test* that computes a one-sided paired t-test with a 95% confidence level (significance level $\alpha = 0.05$). In line 15, the dimensions that are informative are stored in the set D^+ with their respective F1 values.

Finally, we test whether the set $\mathcal{D}^+$ is empty or not. If it is not, we use the function *argmax* to return the dimension $D^+ \in \mathcal{D}^+$ which provides the highest F1 value (line 19). Then, using the dimension D^+, we apply the **DaVI** approach on the whole set of sessions S' to build the final multidimensional recommendation model (line 20). If the set $\mathcal{D}^+$ is empty, we use the whole set of sessions S' to build the pure two-dimensional model (line 22). The final multidimensional or two-dimensional recommendation model is returned in line 24 of the Algorithm 1.

Once we have a recommender model in $\overline{M}$, we can generate the recommendations. If $\overline{M}$ contains a pure two-dimensional model M, we generate the set of N recommendations R as usual. On the other hand, if $\overline{M}$ contains a multidimensional model M', we generate the recommendations following the equation 2.

IV. Empirical Evaluation

In this section, we evaluate our **DaVI** approach with two top-N recommender algorithms, Item-based Collaborative Filtering (CF) [6] and Association Rules based (AR) [7].

The evaluation is carried out on two different data sets from Palco Principal[1], a web site of Portuguese music. The first one, called *Listener*, contains accesses to music tracks of the site. The data set has 62,208 accesses, 6,428 different items (music tracks) and 9,740 sessions. The second data set, called *Playlist*, represents the set of music tracks explicitly selected by users to include in their individual playlist. The data set has 37,022 accesses, 5,428 different items (music tracks) and 4,417 sessions.

The additional dimensions for both data sets are presented in Table I. The first group of dimensions is related to the time and location of the accesses and it is obtained by pre-processing web access data. The second one consists of domain specific information and it is collected from the content management system (CMS) of the web site.

Table I
Additional dimensions for the *Listener* and *Playlist* data sets

Dimension	Description
day	Day of each access (from 01 to 31).
month	Month of each access (from 01 to 12).
week_day	Week day of each access (from Monday to Sunday).
work_day	If the accesses were made during the week (Monday to Friday) or weekend (Saturday or Sunday).
hour	Hour of each access (from 01 to 24).
work_hour	If the accesses were made during working hours (from 8 a.m. to 6 p.m.) or not.
location	Location where the accesses were made (country/city).
band	The band which plays a music track.
music_genre	The genre of a music track (pop, rock, jazz, etc).
instrumental	If a music track is instrumental or not.

To measure the predictive ability of the recommender systems, we calculate the Precision, Recall and F1 metrics using the All But One protocol with 10-fold cross validation. Then, for each metric, the 10 global values are summarized using mean and standard deviation. To compare two recommendation algorithms, we apply the two-sided paired t-test, with a 95% confidence level, on the 10 global values of each metric. We run the experiments for N equal $1, 2, 3, 5$ and 10, where N is the number of items to be recommended by the top-N recommender systems.

We start the evaluation by comparing the **DaVI**-BEST recommender algorithm against the traditional two-dimensional version ($user \times item$). We have summarized the results (i.e., F1 values) of our experiments in Table II. The values which are statistically significant are indicated in boldface. The character "*" indicates the highest value in each set of experiment (i.e., data set $\times$ base recommender $\times$ number of recommendations).

In Table II, we can see that the **DaVI**-BEST algorithm provides a great potential to improve the predictive ability of top-N recommender systems (with 15 of the highest values). When the base recommender is CF, the **DaVI**-BEST algorithm presents F1 gains ranging from 11.9% to 33.7% (*Listener*), and from 7.6% to 25.4% (*Playlist*). For AR models, we have F1 average gains of 26.16% (*Listener*) and 13.66% (*Playlist*).

Additionally, in Table II, we also present results for

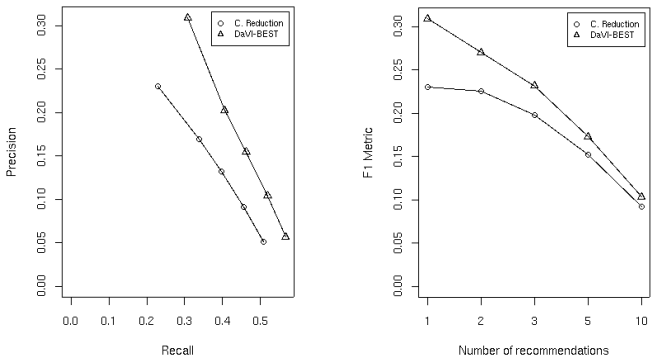
[1] http://www.palcoprincipal.pt/

the combined reduction-based approach (C. Reduction) [3]. Briefly, this approach uses the values of the dimensions as labels for segmenting web accesses. It consists of the following two phases. First, using the training data, a recommendation method is run for each segment (e.g., accesses on Mondays would be a segment) to determine which ones outperform the traditional two-dimensional model (using only user-item information). Second, taking into account the dimension of the active session, we choose the best segment to build a model and make the recommendations. The best segment is the one whose model has the highest F1 value. To the best of our knowledge, this approach is considered the first one for multidimensional recommender systems.

Table II
COMPARING THE F1 VALUES FOR **DaVI**-BEST AND C. REDUCTION ALGORITHMS AGAINST THE VALUE FOR $user \times item$

		CF		**AR**	
Algorithm	N	Listener	Playlist	Listener	Playlist
$user \times item$	1	0.231	0.342	0.175	0.225
DaVI-BEST	1	**0.309***	**0.429***	**0.207***	**0.255**
C. Reduction	1	0.231	0.342	**0.197**	**0.262***
$user \times item$	2	0.226	0.293	0.149	0.176
DaVI-BEST	2	**0.27***	**0.337***	**0.181***	**0.201**
C. Reduction	2	0.226	0.293	**0.17**	**0.203***
$user \times item$	3	0.198	0.242	0.122	0.142
DaVI-BEST	3	**0.231***	**0.271***	**0.153***	**0.16**
C. Reduction	3	0.198	0.242	**0.143**	**0.164***
$user \times item$	5	0.152	0.178	0.088	0.103
DaVI-BEST	5	**0.173***	**0.193***	**0.115***	**0.117**
C. Reduction	5	0.152	0.178	**0.106**	**0.119***
$user \times item$	10	0.092	0.104	0.051	0.061
DaVI-BEST	10	**0.103***	**0.112***	**0.069***	**0.07**
C. Reduction	10	0.092	0.104	**0.062**	**0.071***

Our results also demonstrate that the **DaVI**-BEST algorithm presents better results (i.e., Precision, Recall and F1 metric) than the combined reduction-based algorithm. In Figure 1, we show the results for the *Listener* data set. There, we have Precision gains ranging from 11.7% to 33.7%, Recall gains from 11.3% to 33.7%, and F1 gains from 11.9% to 33.7%.



(a) Precision $\times$ Recall curve. (b) F1 metric.

Figure 1. Comparing the **DaVI**-BEST algorithm against the Combined Reduction-based algorithm using the CF technique in the *Listener* data set.

V. CONCLUSION

In this paper, we described a multidimensional recommendation approach called **DaVI** (*Dimensions as Virtual Items*). It simply consists in using the values of the additional dimensions as (virtual) items. It is combined with traditional two-dimensional recommender algorithms to generate recommendations using additional dimensions (e.g., contextual or background information). The main advantage of this approach is that it can be combined with different existing two-dimensional recommender algorithms. The results of our empirical evaluation have showed that the **DaVI**-BEST algorithm is able to take advantage of the information in multidimensional data to improve the predictive ability of top-N recommender systems.

There are several directions to be explored in a future research. For example, the **DaVI** approach can be tried with other recommender algorithms, such as Markov Models. It would also be important to further challenge the **DaVI** approach with new data sets, in order to validate the conclusions of this paper.

ACKNOWLEDGMENT

Fundação para a Ciência e a Tecnologia (FCT) under the PhD grant SFRH/BD/22516/2005, FCT project Rank! (PTDC/EIA/81178/2006) and QREN-AdI Palco3.0/3121 PONORTE.

REFERENCES

[1] P. Resnick and H. R. Varian, "Recommender systems," *Communications of the ACM*, vol. 40, no. 3, pp. 56–58, 1997.

[2] M. A. Domingues, A. M. Jorge, and C. Soares, "Using contextual information as virtual items on top-n recommender systems," in *ACM RecSys'09 Workshop on Context-Aware Recommender Systems (CARS-2009)*, 2009.

[3] G. Adomavicius, R. Sankaranarayanan, S. Sen, and A. Tuzhilin, "Incorporating contextual information in recommender systems using a multidimensional approach," *ACM Transactions on Information Systems*, vol. 23, no. 1, pp. 103–145, 2005.

[4] J. S. Breese, D. Heckerman, and C. M. Kadie, "Empirical analysis of predictive algorithms for collaborative filtering." in *Proceedings of the Fourteenth Conference on Uncertainty in Artificial Intelligence*, 1998, pp. 43–52.

[5] T. M. Mitchell, *Machine Learning*. New York: McGraw-Hill, 1997.

[6] M. Deshpande and G. Karypis, "Item-based top-n recommendation algorithms," *ACM Transaction on Information System*, vol. 22, no. 1, pp. 143–177, 2004.

[7] A. M. Jorge, M. A. Alves, and P. J. Azevedo, "Recommendation with association rules: A web mining application," in *Proceedings of Information Society (IS-2002): Data Mining and Warehouses*, Ljubljana, Slovenia, 2002.