

An Effective Hybrid Recommender Using Metadata-based Conceptualization and Temporal Semantics

<https://doi.org/10.3991/ijes.v4i3.5943>

M. Venu Gopalachari¹, Porika Sammulal²

¹Chaitanya Bharathi Institute of Technology, Gandipet, Hyderabad, India

²JNTUH College of Engineering Jagtial, Karimnagar, Telanagana, India

Abstract—Modern recommender systems target the satisfaction of the end user through the personalization techniques that collect the history of the user’s navigation. But the sole dependency on user profile based on navigation alone cannot promise the quality of recommendations because of the lack of semantics of various aspect such as demographics of the user, time of usage, concept of need etc in the processing. Though the literature provides many techniques to conceptualize the process makes high computational complexity because of the content data considered as input information. In this paper a hybrid recommender framework is developed that considers Meta data based conceptual semantics and the temporal patterns on top of the history of the usage. This framework also includes an online process that identifies the conceptual drift of the usage dynamically. The experimental results shown the effectiveness of the proposed framework when compared to the existing modern recommenders also indicate that the proposed model can resolve a cold start problem yet accurate suggestions reducing computational complexity.

Index Terms—Concept drift, Domain ontology, Recommender, Sequential patterns, temporal semantics.

I. INTRODUCTION

Now a day, recommender system became an important part in all web services and playing a key role of all e-commerce sites such as flipkart, Amazon, Snapdeal etc. Recommender Systems expanded their scope for many topics such as tourism [5], movies [1], songs [3], web search [6], books [2], jokes [4] etc. Content based recommenders and collaborative filtering recommenders are the two major approaches in recommending web pages having their own merits and demerits. However, hybrid recommenders could able to grab the optimal by minimizing the disadvantages of these approaches by combining the features from both the kinds. Generally the quality of recommenders is measured by the user’s response towards the recommendation such as the time spending on a suggested web page, buying the product recommended, watching the recommended video etc. Traditional approaches of recommender systems suffer from the issues such as cold start and data sparsity. Data sparsity in ratings of the products considerably affects the quality of recommendations, for example, in an e-commerce website the user will not get recommendations of a product either if that user does not have history about that product or if the product did not have support of enough number of accesses. Cold start problem occurs when trying to sug-

gest the new user who does not have much access log so far. These issues arise with lack of semantics and conceptualization in recommendation process. The incorporation of semantics into the recommender process needs the construction of knowledge from the domain of the corresponding web application. The input source of the these semantics in many recommenders is the content of the web pages treated as domain leads to the high computational complexity of the web service. An alternative is much needed to ensure the quality of recommendations and also leads to the light weight application with respect to the complexity.

In this paper a novel hybrid recommender framework is proposed by combining content based and collaborative filtering techniques using conceptualization and also incorporates the temporal semantics in order to achieve the qualitative recommendations. The following questions are answered through this framework; first one is how to construct domain ontology by making use of the meta data of the web domain and incorporate into web usage mining as a light weight application, second one is how to find the temporal constructs in the web log and how to map them to the concepts of the domain, third one is how to handle the dynamics in the concept of the user’s navigation. Experimental results clearly shown the improvement in the performance of the proposed framework and also proved the importance of analyzing the interest drift of the user when predicting recommendations in the temporal scale.

The rest of the paper is organized as some related work is presented then the architecture of the proposed hybrid recommender model is explained and there after results and analysis shows the experimentation part and finally conclusion and references are given.

II. RELATED WORK

In this section some of the existing methodologies presented which are grouped as history based that considers only web log and ontology based that considers the user’s context knowledge as input information. The history-based methods focus on predicting the user intentions by analyzing sequential patterns from the user’s action history information. In [13], Liang analyzed the user behavior and interest towards the web page based on the time spent on that page with respect to a threshold. This strategy might remove the user preferred page if it is accessed short time because of lack of semantics. In [14, 10], the association rules among the sequences of web page navigations were defined known as sequential patterns. The

transitions among the web pages in a user session are modeled using tree structures, and in those models, Pre order linked WAP tree mining [15] algorithm performs better than other sequence learning models.

Ontology based methods finds the behavior patterns integrating the knowledge base of the user or products in addition to the usage data. Georges Gardarin proposed a SEWISE system [16], which extracts the domain specific information from the contents of the web site and generates domain specific ontology. It integrates the wrapper and text mining technology on the web pages and the resulting knowledge is stored in XML database. Also in [17], a recommender system that uses both content and usage knowledge was explained, which is the knowledge of the current user and the user's interests. Quickstep is the recommender of online academic research articles domain uses the research paper topic ontology [18]. However these ontology based approaches were limited to content based recommendations and also has a drawback of computing consumption in ontology construction.

Collaborative filtering in recommender system tries to find the neighbors of similar users or items in history by using typical traditional clustering techniques. Also in literature, there were several approaches like [8, 9, 11] used for collaborative filtering through clustering. In Bian and Holtzman [7] and Kwon and Kim [12], collaborative filtering friend recommendation systems were proposed based on social and physical contexts, for which, the procedure of extracting these contexts information is not explained. However these systems are lack of temporal semantics which are essential in social networks to share interests. Hybrid recommenders integrate different strategies to compensate the limitations of individual methodol-

ogies. These hybrid approaches defined with different strategies in [19] such as switching [21], weighting [20], feature augmentation [22] etc. Weighting strategy gives ranking to the contributed methods; switching strategy selects the method based on the confidence of the approach, feature augmentation make use of the features provided about items by each participated recommender.

However the mentioned recommenders are different from our work which exploit and analyze not only the context of similar interests but also the similar temporal behavior for individual domain in personalized space considering the dynamics with concept drift.

III. PROPOSED FRAMEWORK

In this section the proposed hybrid recommender framework with temporal and conceptual constructs are explained as shown in figure 1. The domain ontology constructed from the web domain considering the Meta data of the web pages and the initial raw input for the model is the web usage log that contains details such as session id, web page title, URL, starting and ending time of the page access etc. Then the log will undergo with preprocessing so that the records give information suitable for the analysis such as removing irrelevant fields and records etc. Further step process the log to derive and add some semantics such as time duration accessed on a page and time spent on a concept etc results the conceptual and temporal data of a user. There after the mining algorithm on the context table provides the patterns of the user which will be used in content based and collaborative filtering strategies and finally hybrid approach combines the suggestions from both ways.

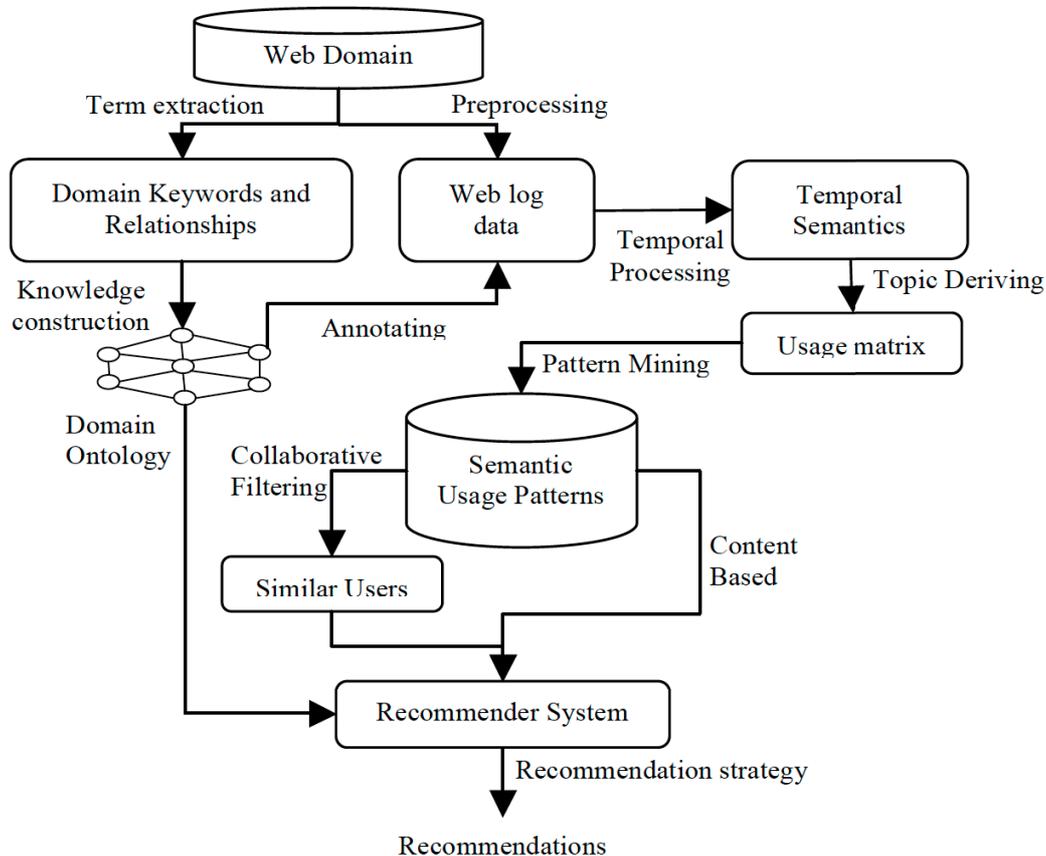


Figure 1. Architecture of proposed hybrid recommender model

A. Domain Ontology Construction

In the proposed system ontology is built for a web domain that targets personalizing the user’s navigation with the concepts extracted from the web pages. The ontology constructed that represents domain knowledge of web documents proposed by Liang [26, 13] is such an example to built. The system needs domain ontology to cluster the web pages in the domain as such in categorization. The similarity among items or web pages has to be measured through their concepts derived from the ontology generated. However ontology construction from the content of the web is reasonable if an offline process is chosen in the application sacrificing the scalability, dynamics and maintenance of the ontology. In order to make the process light weight in terms of computational complexity the proposed model uses the title of the page and the tags provided for a web page are used as sources to derive concepts to generate ontology. Generally the tags and title of the web page contains key terms that represent the content of the page. The idea is to define the concepts from these key terms and their combinations depending on the number of occurrences. This task can be accomplished by following steps that involve a defining concepts and relationships among concepts.

Input: T-Set of titles of web pages $\{T_1, T_2 \dots T_n\}$

Output: C-Set of Concepts $\{C_1, C_2 \dots C_m\}$

Start.
 Initialize word set $W \leftarrow \{\phi\}$
 For each $T_i \in T$ where i is 1 to n
 Apply stop word removal technique.
 Split as g individual words where $g \geq 1$
 For each $\{w_j | j=1 \text{ to } g\}$
 $W \leftarrow W \cup w_j$
 End for
 End for
 Apply association analysis on W
 Apply support threshold on Word sets.
 Add all frequent word sets as concept C_k to C .
 Return C .
 Stop.

B. Semantic Preprocessing

The web usage log has to be preprocessed so that the irrelevant attributes such as protocol name, browser name are removed and constructs the needed attributes such as concept of the web page title, calculating the duration of the time spent on that particular web page. The proposed framework applies data reduction technique to remove irrelevant attributes from the web log and also applies data extraction so that the records about the current user from the log will be retrieved. There after Web log gets annotated with the semantics for each URL with a set of the entities from the domain ontology which is part task in profiling the user. Web usage logs will be semantically enriched by associating each requested URL with one or more ontological entities such as concepts, relations, and attributes to better describe the web navigation. Following steps explains the semantic processing of the log.

Input: Web Server Log having record set $R = \{r_1, r_2 \dots r_{tot}\}$

Output: semantic log with result set $RS = \{RS_1, RS_2 \dots RS_n\}$

Process:

Start.

For each record r_i in the log
 Check if the record belongs to set U_j then
 Add to the set $RS: RS \leftarrow RS \cup r_i$

End for

For each record r_i in RS

 Parse the ontology O with the title and tags of the page in the record.

 Annotate with the concept matched in ontology O .

End for

Stop.

C. Temporal Preprocessing

Temporal processing is making calculation of the duration of time spent on each page by a particular user in order to identify the interests of the user w.r.t. time towards a web page. The preprocessing step provides set of log records related to a user with their time stamps and semantic concepts. Now in order to retrieve temporal semantics about concepts the log records $\{r_1, r_2 \dots r_n\}$ of a web session WS in a sequence have to be processed with their timestamps $\{t_1, t_2 \dots t_n\}$. The timestamps in date-time format are applied to transformation to the scale of $[0, 24]$ as shown in the table 1. Further the transformed time values are used to find the time spent TS on a concept C . The time spent on each web page is calculated by the difference between transformed times of the current page (t_i) and the consecutive page (t_{i+1}). Then the topic name for each record is derived by accessing the ontology so that each web page is represented with a concept that suits exactly. This information is further used to find the user’s context table about time periods and concepts.

TABLE I.
TYPE SIZES FOR RANGE TRANSFORMED TIMESTAMP

IP address	Time Stamp	Transformed Time (T) [0,24]	Time spent (TS) [1,n]
1.2.3.4	01.09.16 am	01.15	1
1.2.3.4	01.09.40 am	01.16	2
1.2.3.4	01.10.55 am	01.18	1
1.2.3.4	01.11.29 am	01.19	1

D. Context table Construction

The context table can be derived by aggregating the time spent of a particular concept C_i and particular time period TP_i . Here for instance three time periods are considered as “ TP_1 (Morning: 07 am – 03 pm)”, “ TP_2 (Evening: 03 pm – 09 pm)” and “ TP_3 (Night: 09 pm – 07 am)” and also three context attributes “ C_1 (Search Engine)”, “ C_2 (Games)” and “ C_3 (Paid Games)” are considered in constructing the table. The aggregation of time spent is converted to the weightage of the user towards that concept or time period in the range of $[0, 1]$.

The total time spent on a concept C_i in a Session WS_k can be found by the sum of time spent of the pages in that session which belongs to C_i .

$$TS(WS_k, C_i) = \sum_{i=1}^n TS(r_i) \quad \forall r_i . C \in C_i$$

$$TS(r_i) = \begin{cases} 0 & \text{if } r_i . C \neq C_i \\ T(r_{i+1}) - T(r_i) & \text{if } 1 \leq i < n \\ \frac{\sum_{i=1}^{n-1} T(r_{i+1}) - T(r_i)}{n-1} & \text{if } i = n \end{cases}$$

The aggregation must also be applied for the predefined time periods TP₁, TP₂ ... TP_n as such morning, afternoon, evening or night etc., depending on the predefined time range.

$$TS(WS_k, TP_i) = \sum_{i=1}^n TS(r_i) \quad \forall r_i . T \in TP_i$$

$$TS(r_i) = \begin{cases} 0 & \text{if } r_i . T \neq TP_i \\ T(r_{i+1}) - T(r_i) & \text{if } 1 \leq i < n \\ \frac{\sum_{i=1}^{n-1} T(r_{i+1}) - T(r_i)}{n-1} & \text{if } i = n \end{cases}$$

Thereafter with the aggregated time spent of each concept(C) and time period (TP) of a user in each session (WS) the weightage (W) given by the user in that session have to be found. This weightage can resemble the user's interest towards a concept or in a time period in that session.

$$W(WS_k, C_i) = \frac{TS(WS_k, C_i)}{\sum_{i=1}^n TS(r_i)}$$

$$W(WS_k, TP_i) = \frac{TS(WS_k, TP_i)}{\sum_{i=1}^n TS(r_i)}$$

Where 'n' is the number of records in a web session WS. So as to apply aggregation in such a way the total time spent on each concept in each time period is found and tabulated as in table II. This contextual table is used for pattern mining in the proposed model that aggregates all the sessions for the participated contexts.

TABLE II.
CONTEXT TABLE OF A USER FOR FIVE WEB SESSIONS

Web Session Id	TP ₁	TP ₂	TP ₃	C ₁	C ₂	C ₃
WS ₁	0	0	1	0.11	0.81	0.04
WS ₂	0	0.49	0.40	0.09	0.28	0.63
WS ₃	0.89	0.11	0	0	0.35	0.65
WS ₄	0.73	.27	0	0	0.42	0.58
WS ₅	0	0	1	0.10	0.05	0.85

E. Pattern Extraction

The proposed algorithm finds the association of concepts in sequence for each time period. This process uses the support and confidence measures which were redefined for the current information constructs. The algorithm returns set of patterns in the order of priority predicted for a particular user based on the context table as shown in the following algorithm.

Input: TP, C, context table, processed usage log.

Output: patterns in the set P.

Process:

Initialize

$G^1 \leftarrow \{TP_1, TP_2 \dots TP_n, C_1, C_2 \dots C_m\}$;

```

P ← {φ};
For all {g|g ∈ G1}
    g.s ← support(g) //Calculate support of all contexts
    If g.s < τ then // τ is the minimum support threshold
        G1 ← G1 - {g}; // if any context is less than τ
        remove from G
    End if
End for
// construct next level combination Gk+1 where k ≥ 1
next:
For all {g|g ∈ Gk ∧ (g ∈ TP)}
    M ← {g}; //selects each time period
    For all {u|u ∈ Gk and u ∈ C} // selects each concept
        context combination for time period.
        M ← M ∪ {u}
        If M.s ≥ τ then
            //checks the combination of TPi and Cj satisfying with
            minimum support threshold.
            Gk+1 ← Gk+1 ∪ M;
        End if
    End for
    M ← {φ};
End for
If Gk+1 ≠ {φ} then
    k = k + 1;
    Goto next;
End if
//extracts all patterns satisfying confidence measure.
For all Gl where 1 ≤ l ≤ k
    For all {g | g ∈ Gl}
        g.c ← confidence(g);
        If g.c ≥ δ then //where δ is the minimum confi-
            dence threshold
            P ← P ∪ {g};
        End if
    End for
End for
    
```

This pattern mining algorithm makes use of support and confidence measures represented as 'g.s' and 'g.c' respectively for 'g' which is the sequence of temporal and concept contexts. The support measure for the combination of the Time Period TP_i and Concept C_j is defined as

$$support(TP_i, C_j) = \frac{\sum_{k=1}^v TP_i(ws_k) \times C_j(ws_k)}{v}$$

Where {ws₁, ws₂ ... ws_v} is the set of sessions participated by the user. The confidence measure for the combination of the Time Period TP_i and Concept C_j is defined as

$$confidence(TP_i, C_j) = \frac{support(TP_i, C_j)}{\sum_{j=1}^m support(TP_i, C_j)}$$

$$= \frac{\sum_{k=1}^v TP_i(ws_k) \times C_j(ws_k)}{\sum_{j=1}^m \sum_{k=1}^v TP_i(ws_k) \times C_j(ws_k)}$$

Here 'm' represents the number of concepts accessed by the user in all the participated sessions of the log. For instance pattern table of one of the user in the data set is shown in table III which also summarizes the support and confidence values of each pattern.

TABLE III.
USAGE PATTERNS OF A USER

Pattern	Support	Confidence
[TP3, C3]	0.23	0.49
[TP1, C3]	0.20	0.62
[TP3, C2]	0.19	0.4
[TP1, C2]	0.12	0.38
[TP2, C3]	0.11	0.61

Now these identified patterns acts as usage profile of the user in the descending order of the confidence measure. That is these usage patterns will be the basic source of input for making recommendations with any approach.

F. Pattern ExtractionContent Based Recommender Strategy

The content based recommender strategy makes use of the user's patterns (P_u) derived from the base of domain knowledge. Following steps defines the strategy of the content based recommender approach.

- Upon the login of the user the recommender has to identify the time period (TP_u) of the user by making use of the timestamp of the session.
- The pattern set (P_u) of the user matching with the current time period TP_u .
- The retrieved patterns have to be arranged in the descending order of the confidence and return the list.

This strategy suggests pages from the retrieved concept sequence of the user if it is purely a content based recommender. For instance, if the user logged in at 8:10PM then the time period of the user falls in TP_3 . Then according to the example pattern set the concepts C_3 then C_2 will be the identified by the recommender strategy. User will be recommended some initial portion of the recommendations from C_3 and later from C_2 . This clearly proves the strategy is ensuring the diversity in suggestions by providing the recommendations from different concepts.

G. Collaborative Filtering

The access patterns of the user are applied for CF technique that gives the neighborhood of the user that is similar users. The attributes that participates in CF are the temporal and conceptual contexts that are identified as user profile. Here in this proposed model kNN (k-nearest neighbors) technique is used to find the neighborhood of the user which can be vulnerable for the sparsity of the data being used, where k is predefined. The similarity measure to find the similarity between two users' u_i and u_j is defined as match ratio (MR) which depends on the access sequences of the user.

$$MR(u_i, u_j) = \frac{|(p_i \cap p_j) \forall p_i, p_j \in P \cap (p_i \in P_{u_i}) \cap (p_j \in P_{u_j})|}{\min\{|P_{u_i}|, |P_{u_j}|\}}$$

The inputs for the collaborative filtering are $U = \{u_1, u_2 \dots u_n\}$, denoting set of n users, $P = \{p_1, p_2 \dots p_k\}$, denoting set of patterns representing the combination of TP and C, $P_{u_i} = \{p_1, p_2 \dots p_m\}$ representing the pattern set of user u_i . Steps for the neighbor selection based on semantic usage patterns.

Step 1: Construct $U \times P$ similarity matrix for the access patterns.

For each u_i in U where $1 \leq i \leq n$

For each u_j in U where $i < j \leq n$

Calculate similarity of u_i and u_j i.e.,

$MR(u_i, u_j)$

$MR(u_j, u_i) \leftarrow MR(u_i, u_j)$

End For

End For

Step 2: Apply K-NN clustering technique for the active user u .

Extract k users having highest similarity MR.

Step 3: Aggregate the ratings from all the neighbors.

The output of the algorithm gives k neighbors and their matched access patterns. From this pattern the ratings are aggregated and the recommendations will be populated.

H. Collaborative Filtering Recommender strategy

Here the input information is the set of k users who are neighbors according to derived patterns. The following steps will recommend the web pages according to the collaborative filtering.

- Upon the login of the user the recommender has to identify the time period (TP_u) of the user by making use of the timestamp of the session.
- The pattern set (P) containing patterns matching with TP_u from all the neighbors has to be retrieved.
- Arrange the concepts in descending order from the retrieved patterns according to the frequency of the occurrences of concepts in the patterns.
- Return the set of concepts in sequence to e recommended.

This strategy suggests pages from the retrieved concept sequence of the user if it is purely a collaborative filtering recommender. For example if the logs at 8:10PM for which the time period falls in TP_3 . If there were 4 neighbors out of which C_3 occurred 5 times and C_2 occurred 3 times and C_1 occurred 2 times then the recommendations will be from the concept sequence C_3, C_2 and C_1 . However the proposed model will take this sequence and hybrids with content based recommender.

I. Hybrid Recommender Strategy

As the both approaches list the concept sequences from their side now hybrid recommendation has to combine those sequences and have to recommend pages to the user. Here in the proposed model the weighted average is the strategy used to combine the concept sequences. For example, If the user logs at 8:10PM where time period TP falls in TP_3 then the average rating of the concepts will be in the C_3, C_2, C_1 .

- Identify the Time Period of the user TP_u with the timestamp of log in of the user.
- Combine the concept sequences from CB and CF approaches according to the weightage of the recommenders.
- Arrange the concepts in the descending order according to the weightage derived in all the patterns.
- Recommend the web pages or items according to the concept sequence arranged.

Thus the proposed hybrid recommender will recommend pages according to the concept sequence.

IV. ONLINE CONCEPT DRIFT IDENTIFICATION

The recommendation engine is more affective if the process also involves the online information provided by the user. This particular task can limit the disadvantages of the offline recommenders and leads to more satisfaction of the user. As the previous offline part of RS already identified user profile with patterns, it search for the pattern that matches the Time period and consequently returns the set of concepts with highest confidence. On the other hand CF technique returns the similar patterns from the neighbors. The recommender will aggregate the identified set of concepts from user's profile and as well CF. However if the user is with other concepts or in the middle of the sequence found then the offline process cannot be succeeded. For instance the user is logged and accessing C_2 and the concept sequence proposed by the proposed model suggests C_3 would not possibly affective for the user's satisfaction. In order to handle this concept interest drift by the user the proposed model also applies online methodology as shown in figure 2. If the proposed hybrid recommender is treated as layered process then the process can be divided into three layers. First layer is about the construction of domain knowledge by making use of metadata. Second layer is finding sequential access patterns from the usage log using the domain knowledge. These two layers are the offline processes involved without intervening the user's current session. Finally the third layer is an online process that handles the current session data of the user. The initial task of this layer is to identify the current active user pattern which can be identified by the current session log that gives the time period of the session. The primary task thereafter is to identify the concept of the session and identification of the pattern sequence matching that concept.

Afterwards the web pages from the mentioned concepts will be listed and suggested however the number of recommendations is according the application's requirement.

V. RESULTS AND DISCUSSIONS

The proposed model used different datasets to analyze the performance out of which some were benchmark da-

tasets and one is raw log dataset. One of the benchmark dataset used in this analysis is from Last.fm database (extracted from "<http://www.dtic.upf.edu/~ocelma/MusicRecommendationDataset/lastfm-360K.html>") that has the log of music tracks listened by individual users together with a time stamp. The dataset contains the records of 990 users with 19,150k records. The log then preprocessed with the earlier said and aggregated on the temporal constructs that are predefined. Here the time constructs are pre defined as three splits known as three time periods Morning (00:00-11:59), Afternoon (12:00-17:59), Night (18:00- 23:59).

Another benchmark dataset used is Movie lens dataset provided by www.grouplens.org in different variants w.r.t. number of ratings and number of users. The dataset is collected by group lens research project team at university of Minnesota. One variant used is Movielens100k data set that consists of 1,00,000 ratings by 943 users for 1682 items scaling from 1 to 5 where 1 represent low negative and 5 represents high positive opinion of the user. The statistics of the movie lens data set is shown in table V.

TABLE IV. STATISTICS OF MOVIE LENS 100K DATA SET

Maximum Ratings per item	583
Minimum Ratings per item	1
Average Ratings per item	59.453
Maximum Ratings per user	737
Minimum Ratings per user	20
Average Ratings per user	106.045
Sparsity	0.063
Frequency of rating value	1(6110), 2(11370), 3(27145), 4(34174), 5(21201)

TABLE V. NUMBER OF USERS IN EACH CONTEXT

Time Period	Number of Users	Number of records
TP1(Morning)	182	130519
TP2(Afternoon)	184	79231
TP3(Night)	187	98387

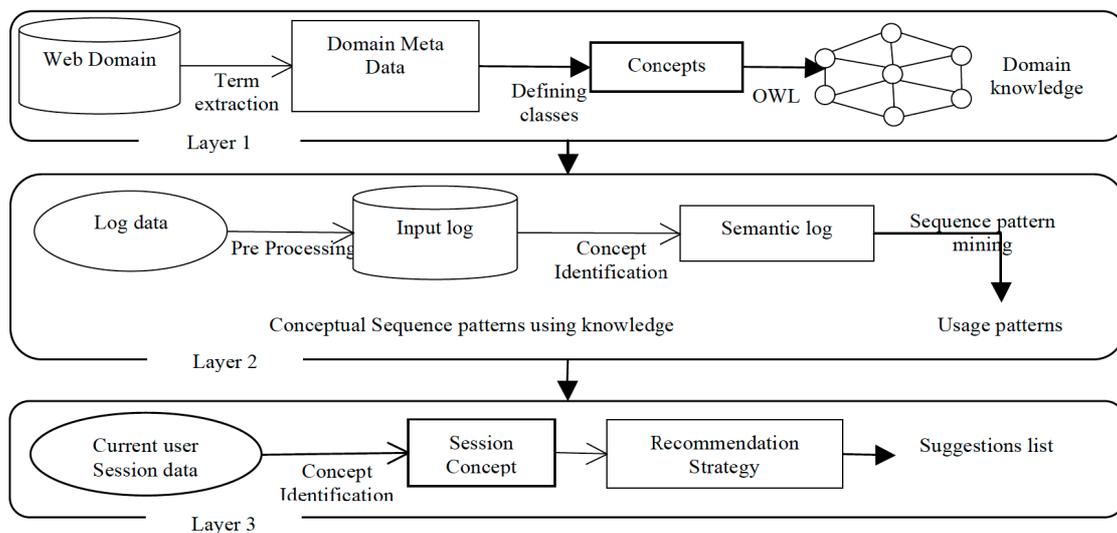


Figure 2. Layered Architecture of the Proposed Recommender system

For the evaluation two evaluation metrics called Mean Absolute Error (MAE) and Root Mean Squared Error (RMSE) are used because these can measure the quality of recommendation accurately. RMSE gives the deviation between actual and predicted ratings and MAE gives the difference between actual and predicted ratings. The less RMSE and MAE represent high quality for recommendations and if the actual ratings are $\{ar_1, ar_2, \dots, ar_n\}$ and predicted ratings are $\{pr_1, pr_2, \dots, pr_n\}$ then MAE and RMSE are defined as

$$RMSE = \sqrt{\frac{\sum_{i=1}^n (ar_i - pr_i)^2}{n}}$$

$$MAE = \frac{\sum_{i=1}^n |ar_i - pr_i|}{n}$$

The results are analyzed by comparing the performance of the proposed model with the popular existing model named PLWAP-Mine [15] SVD++ [23], CBF [24], slope one predictor [25]. PLWAP-Mine is the technique that uses sequential patterns on the usage data with tree representation. SVD++ is the single value decomposition strategy for collaborative filtering approach and Content Based Filtering is a popular approach that does not use any knowledge apart from navigation data. Slope one predictor predicts the rating by using baseline of CF technique which computes the average difference between users rated items. Table 6 shows the values of the MAE and RMSE measures for the comparative analysis of the proposed recommender model.

TABLE VI.
MAE AND RMSE VALUES OF THE PLWAP AND PROPOSED METHODS

SNo	Method	Last.fm data set		Movielens100k data set	
		MAE	RMSE	MAE	RMSE
1	PLWAP-mine	0.7988	0.8275	0.8003	0.8452
2	SVD++	0.7135	0.9065	0.72	0.919
3	CBF	0.7741	0.7694	0.7433	0.8127
4	Slope one predictor	0.7225	0.9143	0.738	0.938
5	Proposed model	0.7103	0.7652	0.7103	0.7652

As shown in figure 3 it is observable that the proposed method gives lowest errors when compared to the existing methods. The performance of CBF is near by the proposed system because CBF is using the ontology with user's log and able to handle cold items problem because of using the concept of the item instead of the name of the item.

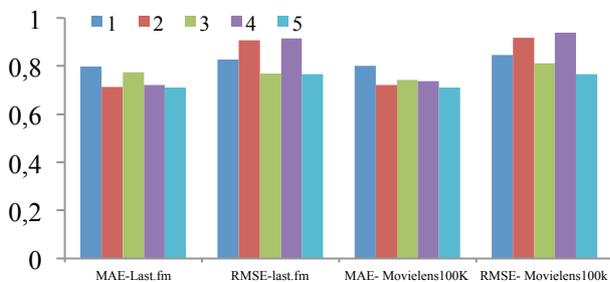


Figure 3. Comparison for MAE and RMSE values with existing methods on last.fm and Movie lens data sets

This proves that one can combine sequential patterns with concept and Time Contexts to improve satisfaction of

the user by means of predicting user's preference on top of domain ontology.

VI. CONCLUSION

In this paper a hybrid recommender model is proposed which is the combination of content based technique and collaborative filtering. The content based part considers the conceptual and temporal contexts of the user access, for which the preprocessing is applied on the log using domain knowledge and then the access patterns are extracted. In addition collaborative filtering applied on user profile that identifies the users with similar temporal and conceptual contexts in their access. The recommender strategy aggregates the ratings from both techniques and suggests the items or pages accordingly. The experiments conducted on benchmark datasets shown the significance of the proposed model in improving the satisfaction of the user towards recommendations.

REFERENCES

- [1] P. Winoto and T. Y. Tang, "The role of user mood in movie recommendations", *Expert System Applications*, volume 37, pp. 6086–6092, 2010. <http://dx.doi.org/10.1016/j.eswa.2010.02.117>
- [2] R. G. Crespo, O. S. Martinez et. al., "Recommendation system based on user interaction data applied to intelligent electronic books," *Computer Human Behavior*, volume 27, pp. 1445–1449, July 2011. <http://dx.doi.org/10.1016/j.chb.2010.09.012>
- [3] S. K. Lee, Y. H. Cho and S. H. Kim, "Collaborative filtering with ordinal scale-based implicit ratings for mobile music recommendations," *Inf. Sci.*, volume 180, pp. 2142–2155, June 2010. <http://dx.doi.org/10.1016/j.ins.2010.02.004>
- [4] Ken Goldberg, T. Roeder et. al., "Eigentaste: A constant time collaborative filtering algorithm", *Information Retrieval Journal*, volume 4, Pp. 133–151, July 2001. <http://dx.doi.org/10.1023/A:1011419012209>
- [5] S.T.Cheng, G.J.Horng and C.L.Chou, "The adaptive recommendation mechanism for distributed group in mobile environments", *IEEE Transactions on Systems, Man and Cybernetics*, volume 42, pp. 1081–1092, November 2012. <http://dx.doi.org/10.1109/TSMCC.2012.2196271>
- [6] S. H. Ha, "Helping on line customers decide through Web personalization", *IEEE Intelligent Systems*, volume 17, pp. 34–43, November 2002.
- [7] Li Bian, Holtzman et. al, "MatchMaker: A friend recommendation system through TV character matching", 2012 IEEE international conference on Consumer Communications and Networking Conference (CCNC), January 2012, Pp.714 – 718.
- [8] J. L. Herlocker, J. A. Konstan and J. Riedl, "Explaining Collaborative Filtering Recommendations", in *Proceedings of ACM Conference on Computer Supported Cooperative Work*, pp. 241–250, 2000.
- [9] J. L. Herlocker, J. A. Konstan, A. Borchers and John Riedl, "An Algorithmic Framework for Performing Collaborative Filtering", in *Proceedings of 22nd ACM SIGIR Conference on Information Retrieval*, pp. 230–237, 1999.
- [10] R. Agrawal and R. Srikant, "Mining Sequential Patterns", in *Proceedings of 11th International Conference on Data Engineering*, pp. 3-14, March 1995. <http://dx.doi.org/10.1109/icde.1995.380415>
- [11] T. Hofmann, "Latent Semantic Models for Collaborative Filtering", *ACM Transactions on Information Systems*, volume 22, pp. 89–115, 2004. <http://dx.doi.org/10.1145/963770.963774>
- [12] J. Kwon and S. Kim, "Friend Recommendation Method using Physical and Social Context", *International Journal of Computer Science and Network Security*, volume 10, November 2010.
- [13] Liang T.P. and H.-J. Lai, "Discovering user interests from web browsing behavior: An application to internet news services", in *Proceedings of the 35th Annual Hawaii International Conference on System Science*, Washington, DC, USA: IEEE Computer Society, 2002, pp. 2718 – 2727.

- [14] M.N. Moreno, F.J. Garcia, M.J. Polo, and V.F. Lopez, "Using Association Analysis of Web Data in Recommender Systems", in Proceedings of 5th International Conference on E-Commerce and Web Technologies, August 2004. http://dx.doi.org/10.1007/978-3-540-30077-9_2
- [15] C. I. Ezeife and Y. Lu, "Mining Web Log Sequential Patterns with Position Coded Pre-Order Linked WAP-Tree," Data Mining and Knowledge Discovery, vol. 10, pp. 5-38, 2005. <http://dx.doi.org/10.1007/s10618-005-0248-3>
- [16] Georges Gardarin, Huaizhong KOU, Karine Zetourni et. al, "SEWISE: An Ontology – based Web Information Search Engine", 2003.
- [17] Fong A.C.M, Baoyao Zhou et.al., "Web Content Recommender System based on Consumer Behavior Modeling," IEEE Transactions on Consumer Electronics, Volume 57, May 2011. <http://dx.doi.org/10.1109/TCE.2011.5955246>
- [18] S. E. Middleton, N. R. Shadbolt, et D. C. De Roure, "Ontological user profiling in recommender systems", ACM Trans. Inf. Syst. TOIS, vol. 22, no 1, p. 54–88, 2004. On, vol. 54, no 2, Pp. 727–735, 2008.
- [19] R. Burke, "Hybrid recommender systems: Survey and experiments," User Model. User-Adapted Int., vol. 12, no. 4, pp. 331–370, Nov. 2002. <http://dx.doi.org/10.1023/A:1021240730564>
- [20] T. Hornung, C.-N. Ziegler, S. Franz, M. Przyjaciel-Zablocki, A. Schatzle, and G. Lausen, "Evaluating hybrid music recommender systems," in Proc. IEEE/WIC/ACM Int. Joint Conf. Web Intelligent Agent Technologies, Nov. 2013, pp. 57–64. <http://dx.doi.org/10.1109/wi-iat.2013.9>
- [21] M. Van Setten, "Supporting people in finding information", Ph.D. dissertation, Telematica Instituut, Enschede, Univ. Enschede, the Netherlands, 2005.
- [22] P. Forbes and M. Zhu, "Content-boosted matrix factorization for recommender systems," in Proc. 5th ACM Conf. Recommender Systems 2011, p. 261. <http://dx.doi.org/10.1145/2043932.2043979>
- [23] Yehuda Koren. "Factorization meets the neighborhood: a multifaceted collaborative filtering model", In KDD'08, pages 426–34. <http://dx.doi.org/10.1145/1401890.1401944>
- [24] Martin Saveski and Amin Mantrach. "Item cold-start recommendations: learning local collective embeddings", In ACM Conference on Recommender Systems, pages 89–96. ACM, 2014.
- [25] Daniel Lemire and Anna Maclachlan. Slope one predictors for online rating-based collaborative filtering. In SDM, volume 5, pages 1–5. SIAM, 2005.
- [26] Liang Gou et. al., "SFViz: interest-based friends exploration and recommendation in social networks", in Proceedings of international symposium on the 2011 Visual Information Communication, ISBN: 978-1-4503-0786-4, ACM New York, USA, 2011. <http://dx.doi.org/10.1145/2016656.2016671>

AUTHORS

M. Venu Gopalachari is with Chaitanya Bharathi Institute of Technology, Gandipet, Hyderabad, India (e-mail: venugopal.m07@gmail.com).

Porika Sammulal is with JNTUH College of Engineering Jagtial, Karimnagar, Telanagana, India (e-mail: sam@jntuh.ac.in).

Submitted 12 June 2016. Published as resubmitted by the authors 26 August 2016.