International Journal of Research
Available at https://edupediapublications.org/journals

e-ISSN: 2348-6848
p-ISSN: 2348-795X
Volume 05 Issue-04
February-2018

Leveraging Social Network Data to Alleviate Cold-Start Problem in Recommender Systems

Srilatha Sahukara
M.Tech,Dept. of Computer Science and Systems Engineering,
AU College of Engineering(A), Visakhapatnam

Abstract-*Now-a-days, the gap between e-commerce and social networking became progressively diminished. Several e-commerce websites and mobile applications support the mechanism of social login wherever users will sign in the websites using their social network identities like their Facebook or Twitter accounts.Users can also post their newly purchased products on microblogs with links to the e-commerce product web pages. In this paper,we represent a novel solution for cross-site cold-start product recommendation, which aims to recommend products from e-commerce websites to users at social networking sites in "cold-start" situations, a problem which has rarely been explored before. A noteworthy issue is how to leverage knowledge extracted from social networking sites for cross-site cold-start product recommendation. In particular, we proposed the solution for cold-start recommendation by linking the users to social networking sites and ecommercewebsites i.e. customers who have social network identities and have purchased on e-commerce websites as a bridge to map users social networking features into another feature representation which can be easier for a product recommendation. Here we propose to learn by using recurrent neural networks both user's and product's feature representations called user embedding and product embedding from the data collected from e-commerce website and then apply a modified gradient boosting trees method to transform user's social networking features into user embeddings. Once obtained, then we will develop a feature-based matrix factorization approach which will use the user and product features for the cold-start product recommendation. Experimental results show that our approach effectively works and gives the best-recommended results in cold start situations.*

Keywords:e-commerce, product recommender, product demographic, micro-blogs, Cold-Start, Recommendation.

1. INTRODUCTION

Today's world is becoming fully automatic through Internet. Internet provides the most needed information. The access to Internet creates large amount of data day by day. E-commerce websites such as eBay features many of the characteristics of social networks, including real-time status updates and interactions between its buyers and sellers. Some e-commerce websites also support the mechanism of social login, which allows new users to sign in with their existing login information from social networking services such as Facebook, Twitter or Google+. Both Facebook and Twitter which has introduced a new feature last year had attracted more buyers which allowed more number of users to buy products directly from their websites by clicking a "buy" button to purchase items based on some adverts or other posts. In China, the e-commerce company ALIBABA has made a strategic investment in SINA WEIBO where ALIBABA product adverts can be directly delivered to SINA WEIBO users. With the new trend of conducting e-commerce activities on social networking sites- the reviews, leveraging product adopter information, extracted from e-commerce and profile details of social networking sites used for the development of the cold start product recommendation systems. In this, Recommendation plays a important role in many fields and has attracted a lot of research interest. For example, Netflix has released an interesting fact that about 75% of its subscribers watch are from recommendations. In a recommender system such as Netflix and Amazon, e-bay, Flipkart, users can browse items and choose those items they are interested in, the advertisement also plays a major role were in the system also recommend the product to the users. Then the items that the system thought as a best one will be the best match of preference to the product recommendation. Afterward, the user may provide feedback (such as rating, usually represented as a score between, for example, 1 and 5, also the reviews make a huge decision in the product purchase) on how the user thinks about an item after she/he has experienced the item. One important task for the recommendation engine is to understand users' personalized preferences from their historic rating behaviours. In this paper, we study an interesting problem of recommending products from e-commerce websites to users at social networking sites who do not have historical purchase records, i.e., in "cold-start" situations. We called it cross-site cold-start product recommendation. Most studies only focus on constructing solutions within certain e- commerce websites and mainly utilise user's historical transaction records. To the best of our knowledge, cross-site cold-start product recommendation has been rarely studied before. Another challenging task is how to improve the recommendation accuracy for the new (or rarely rated) items and the new (or inactive) users. Comparing to the popular items, for the newly released ones and the old items that are rarely rated by users, it is difficult for the standard recommendation approaches such as collaborative filtering approach to provide high-quality recommendations. In our problem setting here, only the users' social networking information is available and it is a challenging task to transform the social networking information into latent user features which can be effectively used for product recommendation. To address this threat, we represent to use the linked users across social networking sites and e-commerce websites (users who have social networking accounts and have made purchases on e-commerce websites) as a bridge to map users' social networking features to latent features for product recommendation. In specific, we represent learning both users' and products' feature representations (called user embeddings and product embeddings, respectively) from data collected from ecommerce websites using recurrent neural networks and then apply a modified gradient boosting trees method to transform users' social networking features into user embeddings. We then develop a featurebased matrix factorization approach which can leverage the learnt user embeddings for cold-start product recommendation. We built our dataset from the largest Chinese microblogging service SINA WEIBO2 and the largest Chinese B2C e-commerce website JINGDONG3, containing a total of 20,638 linked users. The experimental results on the dataset have shown the feasibility and the effectiveness of our proposed framework.

2. LITRATURE SURVEY

1] Opportunity model for e-commerce recommendation: Right product; right time

Author:-J. Wang and Y. Zhang

Description: Most of existing e-commerce suggester systems aim to recommend the proper product to a user, supportedwhether or not the user is probably going to buy or sort of a product. On the opposite hand, the effectiveness ofrecommendations conjointly depends on the time of the advice. Allow us to take a user World Health Organizationsimply purchased a laptop computer as an example. She might purchase a replacement battery in a pair of years(assuming that the laptop computer's original battery typically fails to figure around that time) and get a brand newlaptop in another a pair of years. During this case, it's not a decent plan to suggest a brand new laptop computer or areplacement battery right when the user purchased the new laptop computer. It may hurt the user's satisfaction of therecommender system if she receives a doubtless right product recommendation at the incorrect time. We have atendency to argue that a system mustn't solely suggest the foremost relevant item, however conjointly suggest at theproper time.

2] Retail sales prediction and item recommendations using customer demographics at store level

Author:-M. Giering

Description: This paper outlines a retail sales prediction and products recommendation system that was enforced for asequence of retail stores. The relative importance of client demographic characteristics for accurately modeling thesales of every client kind square measure derived and enforced within the model. Knowledge onsisted of daily salesdata for 600 products at the shop level, broken out over a collection of non-overlapping client varieties. A recommendersystem was designed supported a quick on-line skinny Singular worth Decomposition. It's shown that modelling knowledge at a finer level of detail by clump across client varieties and demographics yields improved performancecompared to one mixture model designed for the complete dataset. Details of the system implementation squaremeasure represented and sensible problems that arise in such real-world applications square measure mentioned.

3] Amazon.com recommendations: Item-to-item collaborative filtering

Author:-G. Linden, B. Smith, and J. York

Description:Recommendation algorithms area unit best glorious for his or her use on e-commerce internet sites,wherever they use input a couple of customer's interests to come up with an inventory of suggested things. Severalapplications use solely the things that customers purchase and expressly rate to represent their interests, however they'lladditionally use alternative attributes, together with things viewed, demographic information, subject interests, andfavourite artists. At Amazon.com, we tend to use recommendation algorithms to change the web store for every client.the shop radically changes supported client interests, showing programming titles to a engineer and baby toys to areplacement mother. There area unit 3 common approaches to resolution the advice problem: ancient cooperativefiltering, cluster models, and search-based strategies. Here, we tend to compare these strategies with our algorithmicprogram, that we tend to decision item-to-item cooperative filtering.

4] The new demographics and market fragmentation

Author:-V. A. Zeithaml

Description: The underlying premise of this text is that dynamic demographics can result in a breakage of the massmarkets for grocery product and supermarkets. A field study investigated the relationships between five demographicfactors-sex, feminine operating standing, age, income, and matrimonial status-and a large vary of variables related topreparation for and execution of food market looking. Results indicate that the demographic teams dissent in importantways that from the standard food market shopper. Discussion centres on the ways in which dynamic demographics andfamily roles might have an effect on retailers and makers of grocery product.

5. We know what you want to buy: a demographic-based system for product recommendation on microblogs

Author:- W. X. Zhao, Y. Guo, Y. He, H. Jiang, Y. Wu, and X. Li

Description: Product recommender systems square measure usually deployed by e-commerce websites to boost userexpertise and increase sales. However, recommendation is proscribed by the merchandise data hosted in those ecommerce sites and is barely triggered once users square measure playing e-commerce activities. During this paper, wetend to develop a completely unique product recommender system known as breed, a merchandiser Intelligencerecommender System, that detects users' purchase intents from their microblogs in close to time period and makesproduct recommendation supported matching the users' demographic data extracted from their public profiles withproduct demographics learned from microblogs and on-line reviews. Breed distinguishes itself from ancient productrecommender systems within the following aspects:

1) breed was developed supported a microblogging serviceplatform. As such, it's not restricted by the knowledge obtainable in any specific e-commerce web site. Additionally,breed is in a position to trace users' purchase intents in close to time period and build recommendations consequently.

2) In breed, product recommendation is framed as a learning to rank drawback. Users' characteristics extracted fromtheir public profiles in microblogs and products' demographics learned from each on-line product reviews andmicroblogs square measure fed into learning to rank algorithms for product recommendation.

3. METHOD OF DATA EXTRACTION

EXTRACTING AND REPRESENTING MICROBLOGGING ATTRIBUTES

Our solution to microblogging feature learning consists of three steps: Prepare a list of potentially useful microblogging attributes and construct the microblogging feature vector a_u for each linked user ; Generate distributed feature representations v_u using the information from all the users U on the ecommerce website through deep learning; Learn the mapping function, $f(a_u)\text{->}v_u$, which transforms the microblogging attribute information auto the distributed feature representations v_u in the second step. It utilizes the feature representation pairs $\{a_u,v_u\}$ of all the linked users u € U^L as training data.

MICROBLOGGING FEATURE SELECTION

In this section, we study how to extract rich user information from microblogs to construct a_u for a microblogging user. We consider four groups of attributes.

DEMOGRAPHIC ATTRIBUTES

A demographic profile (often shortened as “a demographic”) of a user such as sex, age and education can be used by ecommerce companies to provide better personalized services. We extract users’ demographic attributes from their public profiles on SINA WEIBO. Demographic attributes have been shown to be very important in marketing, especially in product adoption for consumers. Following our previous study, we identify six major demographic attributes: gender, age, marital status, education, career and interests.

TEXT ATTRIBUTES

Recent studies have revealed that microblogs contain rich commercial intents of users. Also, users’ microblogs often reflect their opinions and interests towards certain topics. As such, we expect a potential correlation between text attributes and users’ purchase preferences. We perform Chinese word segmentation and stop word removal before extracting two types of text attributes below. Topic distributions. Seroussi et al. proposed to extract topics

p-ISSN: 2348-6848
e-ISSN: 2348-795X
Volume 03 Issue 12
August 2016

from usergenerated text using the Latent Dirichlet Allocation (LDA) model for recommendation tasks. Follow the same idea, we first aggregate all the microblogs by a user into a document, and then run the standard LDA to obtain the topic distributions for each user. The benefits of topics distributions over keywords are twofold. First, the number of topics is usually set to 50 200 in practice, which largely reduces the number of dimensions to work with. Second, topic models generate condense and meaningful semantic units, which are easier to interpret and understand than keywords. Word embeddings. Standard topic models assume individual words are exchangeable, which is essentially the same as the bag-of-words model assumption. Word representations or embeddings learned to use neural language models help addressing the problem of traditional bag-of-word approaches which fail to capture words' contextual semantics. In word embeddings, each dimension represents a latent feature of the word and semantically similar words are close in the latent space. We employ the Skip-gram model implemented by the tool word2vec4 to learn distributed representations of words. Finally, we average the word vectors of all the tokens in a user's published document as the user's embedding vector.

NETWORK ATTRIBUTES

In the online social media space, it is often observed that users connected with each other (e.g., through following links) are likely to share similar interests. As such, we can parse out latent user groups by the users' following patterns assuming that users in the same group share similar purchase preferences. Latent group preference. Since it is infeasible to consider all users on WEIBO and only keeping the top users with the most followers would potentially miss interesting information, we propose to use topic models to learn latent groups of following sasin [10].Wetreatafollowinguserasatokenandaggregateallthefoll owingsofauserasanindividualdocument.In this way, we can extract latent user groups sharing similar interests (called "following topics"), and we represent each user as a preference distribution over these latent groups

TABLE 1
Categorization of the Microblogging Features

Categories	Features
Demographic Attributes	Gender (2), Age (6), Marital status (10), Education (7), Career (9), Interests (6)
Text Attributes	Topic distributions (50), Word embeddings (50)
Network Attributes	Latent group preference (50)
Temporal Attributes	Daily activity distribution (24), Weekly activity distribution (7)

The number of feature dimensions are shown in parentheses.

TEMPORAL ATTRIBUTES

Temporal activity patterns are also considered since they reflect the living habits and lifestyles of the microblogging users to some extent. As such, there might exist correlations between temporal activities patterns and users' purchase preferences. Temporal activity distributions. We consider two types of temporal activity distributions, namely daily activity distributions and weekly activity distributions. The daily activity distribution of a user is characterized by a distribution of 24 ratios, and the ith ratio indicates the average proportion of tweets published within the ith hour of a day by the user; similarly weekly activity distribution of a user is characterized by a distribution of seven ratios, and the ith ratio indicates the average proportion of tweets published within the ith day of a week by the user. We summarize all types of features in above table.

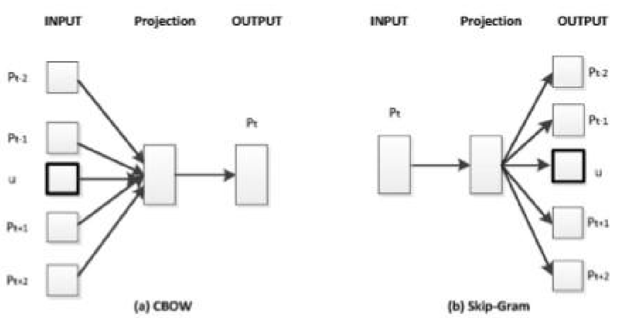
DISTRIBUTED REPRESENTATION LEARNING WITH RECURRENT NEUTRAL NETWORKS

We use recently proposed methods in learning word embeddings using recurrent neutral networks to learn user embeddings or distributed representation of user. We first discuss how to learn product embeddings and in the later part the word embeddings. There are two simple recurrent neutral architectures to train product embeddings, the Continuous Bag-Of-Words model (CBOW) and the Skip-gram model [1]. The major difference between these two architectures is in the direction of prediction: CBOW predicts the current product using the surrounding context, while Skip-gram predicts the context with the current product. In our evaluations, the context is defined as a window of size 4 surrounding a target product which contains two products purchased before and two after. With product embeddings, if we can learn user embeddings in a similar way, then we can explore the related representations of a user and products for product recommendation. The purchase history of a user is like a

p-ISSN: 2348-6848
e-ISSN: 2348-795X
Volume 03 Issue 12
August 2016

"sentence" having of a sequence of product IDs as word tokens. A user ID is placed at the beginning of each sentence, and both user IDs and product IDs are treated as word tokens in the learning process. During training, for each sentence, the sliding context window will always include the first word (i.e., user ID) in the sentence. In this way, a user ID is essentially always associated with a set of her purchase records (of 4 products at a time). Advantages:

Gain customer information like what they are, what they like, etc. which can transform our business. Increase brand awareness i.e. targets more people to our e-commerce. Run customer targeted ads with real time results. Generate valuable leads i.e. transform ad viewer to a customer. Increase website traffic and search ranking. Find out information about how competitor is performing and change ourselves according to that. Share content faster and easier.



Two architectures to learn both product and user embeddings. Here u denote a user ID. The major difference between para2vec and word2vec lies in the incorporation of userID as additional context

FUNCTIONAL MATRIX FACTORISATION

Now we consider constructing the interview process for cold-start collaborative filtering. Assume that a new user registers at the recommendation system and nothing is known about her. To capture the preferences of the user, the system initiates several interview questions to query the responses from the user. Based on the responses, the system constructs a profile for the user and provides recommendations accordingly. In the plain matrix factorization model described in Section 3.1, the user profile ui is estimated by optimizing the ℓ2 loss on the history ratings rij. This model does not directly apply to cold-start settings because no rating is observed for the new user prior to the interview process. To build user profiles adaptively according to the user's responses in the course of the interview process, we propose to parameterize the user profile ui in such a way that the profile ui is tied to user i's responses in the form of a function, thus the name functional matrix factorization (FMF**).** More precisely, assume there is P possible interview questions. We assume that an answer to a question takes value in the finite set {0,1, Unknown}, representing "Dislike", "Like" and "Unknown", respectively. Furthermore, let a_i denote the P dimensional vector representing the answers of user i to the P questions. And we tie the profile to the answers by assuming $u_i = T(a_i)$, where T is a function that maps the responses a_i to the user profile u_iRk. To make recommendations for user i, we simply use $r_{ij} = vT_j\ T(a_i)$. Our goal is to learn both T and v_j from the observed ratings K. To this end, substituting $u_i = T(a_i)$ into the low-rank matrix factorization model, we have the following optimization problem:

$$T, V = \underset{T \in \mathcal{H}, V}{\operatorname{argmin}} \sum_{(i,j) \in O} (r_{ij} - v_j^T T(a_i))^2 + \lambda \|V\|^2, \quad (1)$$

where V = (v1, ...,vM) is the matrix of all item profiles, H is the space from which the function T(a) is selected and the second term is the regularization term. Several issues need to be addressed in order to construct the interview process by the above functional matrix factorization. First, the number of all possible interview questions can be quite large (e.g. up to millions of items in movie recommendation); yet a user is only patient enough to answer a few interview questions. Second, the interview process should be adaptive to user's responses, in other words, a follow-up question should be selected based on the user's responses to the previous questions. Therefore, the selection process should be efficient to generate interview questions in real time after the function T(a) is constructed. In addition, since we allow a user to choose "Unknown"to the interview questions, we need to deal with such missing values as well. Following prior works of [8,20], we use a ternary decision tree to represent T(a). Specifically, each node of the decision tree corresponds to an interview question and has three child nodes. When the user answers the interview question, the user is directed to one of its three child nodes according to her answer. As a result, each user follows a path from the root node to a leaf node during the interview process. A user profile is estimated at each leaf node based on the users' responses, i.e., T(a). The number of interview questions presented to any user is bounded by the depth of the decision tree, generally a small number determined by the system. Also, non-responses to a question can be handled easily in the decision tree with the introduction of a "Unknown" branch.

4. RELATED WORK

A.OSN System Construction Module

In the first module, we develop the Online Social Networking (OSN) system module. We buildup the system with the feature of Online Social Networking. Where, this module is used for new user registrations and afterregistrations the users can login with their authentication. Where after the existing users can send messages to privately andpublicly, options are built. Users can also share post with others. The user can able to search the other user profiles and public posts.

In this module users can also accept and send friend requests. With all the basic feature of Online Social Networking Systemmodules is build up in the initial module, to prove and evaluate our system features.Given an e-commerce website, with a set of itsusers, a set of products and purchase record matrix, each entry of which is a binary value indicating whether has purchased product. Each user is associated with a set of purchased products with the purchase timestamps. Furthermore, a small subset of users can belinked to their microblogging accounts (or other social network accounts).

B. Microblogging Feature Selection

In this module, we develop the Microblogging Feature Selection. Prepare a list of potentially useful microblogging attributes andconstruct the microblogging feature vector for each linked user. Generate distributed feature representations using the informationfrom all the users on the ecommerce website through deep learning. Learn the mapping function, which transforms themicroblogging attribute information au to the distributed feature representations in the second step. It utilises the featurerepresentation pairs of all the linked users as training data.Ademographic profile (often shortened as "a demographic") of a usersuch as sex, age and education can be used by ecommerce companies to provide better personalised services. We extract users'demographic attributes from their public profiles. Demographic attributes have been shown to be very important in marketing,especially in product adoption for consumers.

C.Learning Product Embedding

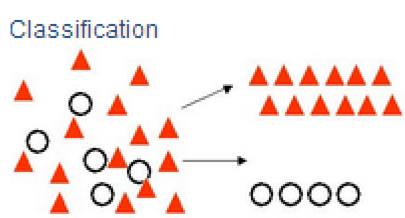
In the previous module, we develop the feature selection, but it is not straightforward to establish connections between users andproducts. Intuitively, users and products should be represented in the same feature space so that a user is closer to the products thathe/she has purchased compared to those he/she has not. Inspired by the recently proposed methods in learning word embeddings, wepropose to learn user embeddings or distributed representation of user in a similar way. Given a set of symbol sequences, a fixed-lengthvector representation for each symbol can be learned in a latent space by exploiting the context information among symbols,in which "similar" symbols will be mapped to nearby positions. If we treat each product ID as a word token, and convert the historical purchase records of a user into a timestamped sequence, we can then use the same methods to learn product embeddings.Unlike matrix factorization, the order of historical purchases from a user can be naturally captured.

D.CART and MART Algorithm

CART(Classification and Regression Tree)

Decision Trees are commonly used in data mining with the objective of creating a model that predicts the value of a target (or dependent variable) based on the values of several input (or independent variables). In today's post, we discuss the CART decision tree methodology. The CART or Classification & Regression Trees methodology was introduced in 1984 by Leo Breiman, Jerome Friedman, Richard Olshen and Charles Stone as an umbrella term to refer to the following types of decision trees:

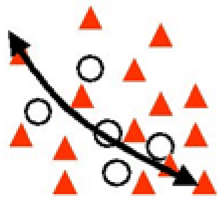
Classification Trees: where the target variable is categorical and the tree is used to identify the "class" within which a target variable would likely fall into.



Regression Trees: where the target variable is continuous and tree is used to predict it's value.

International Journal of Research
International JournalofResearch
Available at https://edupediapublications.org/journals
p-ISSN: 2348-6848
e-ISSN: 2348-795X
Volume 03 Issue 12
August 2016

Regression



The CART algorithm is structured as a sequence of questions, the answers to which determine what the next question, if any should be. The result of these questions is a tree like structure where the ends are terminal nodes at which point there are no more questions.

MART(Multiple Additive Regression Tree)

MART is an implementation of the gradient boosting methods for predictive data mining (Classification and Regression)

Algorithm:

Input: a random output variable 'y', a set of random input variables x={x_1,x_2….x_n}, a training sample$\{yi,xi\}_1^N$ of known (y,x) - values

1. $F_0(\mathbf{x}) = \arg\min_\rho \sum_{i=1}^{N} L(y_i, \rho)$
2. For $m = 1$ to M do:
3. $\tilde{y}_i = -\left[\frac{\partial L(y_i, F(\mathbf{x}_i))}{\partial F(\mathbf{x}_i)}\right]_{F(\mathbf{x})=F_{m-1}(\mathbf{x})}, \; i = 1, N$
4. $\mathbf{a}_m = \arg\min_{\mathbf{a},\beta} \sum_{i=1}^{N} [\tilde{y}_i - \beta h(\mathbf{x}_i; \mathbf{a})]^2$
5. $\rho_m = \arg\min_\rho \sum_{i=1}^{N} L(y_i, F_{m-1}(\mathbf{x}_i) + \rho h(\mathbf{x}_i; \mathbf{a}_m))$
6. $F_m(\mathbf{x}) = F_{m-1}(\mathbf{x}) + \rho_m h(\mathbf{x}; \mathbf{a}_m)$

Output: F_M(x)

E. Similarity Computation

In this module, we are applying cosine similarity for identifying similarity between users and items. Cosine similarity is a measure of similarity between two vectors of an inner product space that measures the cosine of the angle between them. Cosine similarity is particularly used in positive space, where the outcome is neatly bounded in [0,1].

F. Cold-Start Product Recommendation

We used a local host based e-commerce dataset, which contains some user transaction records. Each transaction record consists of auser ID, a product ID and the purchase timestamp. Products are recommendedto a user who is new to e-commerce website and linked to social networking website based on the micro-blogging features such as his interests , occupation of that user.

5. CONCLUSION

In this paper, we've studied a unique hassle, cross site cold-start product recommendation, i.e., recommending products from e-commercewebsites to microblogging customers without historical buy facts. Our essential idea is that at the e-trade websites, users andmerchandise may be represented inside the identical latent characteristic area via characteristic getting to know with the recurrentneural networks. Our main idea isthat on the e-commerce websites, users and products can be represented in the same latent feature space throughfeature learning with the recurrent neural networks. Using a set of linked users across both e-commerce websites andsocial networking sites as a bridge, we can learn feature mapping functions using a modified gradient boosting treesmethod, which maps users' attributes extracted from social networking sites onto feature representations learnedfrom e-commerce websites. The mapped user features can be effectively incorporated into a feature-based matrixfactorization approach for cold start product recommendation. We have constructed a large dataset from WEIBOand JINGDONG. The results show that our proposed framework is indeed effective in addressing the cross-sitecold-start product recommendation problem. We believe that our study will have profound impact on both researchand industry communities. Currently, only a simple neutral network architecture has been employed for user andproduct embeddings learning. We will also consider improving the current feature mappingmethod through ideas in transferring learning [30].

References

[1]J. Wang and Y. Zhang, "Opportunity model for e-commerce recommendation: Right product; right time," inSIGIR, 2013.

[2] M. Gyring, "Retail sales prediction and item recommendations using customer demographics at store level,"SIGKDD Explor. Newsl., vol. 10, no. 2, Dec. 2008.

[3] G. Linden, B. Smith, and J. York, "Amazon.com recommendations: Item-to-item collaborative filtering," IEEEInternet Computing, vol. 7, no. 1, Jan. 2003.

[4] V. A. Zeithaml, Sarwar et al "The new demographics and market fragmentation," Journal of Marketing, vol. 49,pp. 64–75, 1985.

International JournalofResearch
Available at https://edupediapublications.org/journals
p-ISSN: 2348-6848
e-ISSN: 2348-795X
Volume 03 Issue 12
August 2016

[5] W. X. Zhao, Y. Guo, Y. He, H. Jiang, Y. Wu, and X. Li, Miyahara and Pazzani "We know what you want tobuy: a demographic-based system for product recommendation on microblogs," in SIGKDD, 2014.
[6] J. Wang, W. X. Zhao, Y. He, and X. Li, "Leveraging product adopter information from online reviews forproduct recommendation," in ICWSM, 2015.
[7] Y. Seroussi, F. Bohnert, and I. Zukerman, "Personalised rating prediction for new users using latent factormodels," in ACM HH, 2011.
[8] T. Mikolov, I. Sutskever, K. Chen, G. S. Corrado, and J. Dean, "Distributed representations of words and phrasesand their compositionality," in NIPS, 2013.
[9] Q. V. Le and T. Mikolov, "Distributed representations of sentences and documents," CoRR, vol. abs/1405.4053,2014.
[10] J. Lin, K. Sugiyama, M. Kan, and T. Chua, "Addressing coldstart in app recommendation: latent user modelsconstructed from twitter followers," in SIGIR, 2013.
[11] T. Mikolov, K. Chen, G. Corrado, and J. Dean, "Efficient estimation of word representations in vector space,"CoRR, vol. abs/1301.3781, 2013.
[12] RSs. Lenuget al. ,Y. Koren, R. Bell, and C. Volinsky, "Matrix factorization techniques for recommendersystems," Computer, vol. 42, no. 8, pp. 30–37, Aug. 2009.
[13] J. H. Friedman, "Greedy function approximation: A gradient boosting machine," Annals of Statistics, vol. 29,pp. 1189–1232, 2000.
[14] L. Breiman, J. Friedman, R. Olshen, and C. Stone, Classification and Regression Trees. Monterey, CA: Wadsworth and Brooks, 1984.
[15] L. Breiman, "Random forests," Mach. Learn., vol. 45, no. 1, Oct. 2001.
[16] K. Zhou, S. Yang, and H. Zha, "Functional matrix factorizations for cold-start recommendation," in SIGIR,2011.

AUTHORS

Srilath
PG Scholar
Department of CSE,
Andhra University, Visakhapatnam