International Journal of Trend in Scientific Research and Development (IJTSRD)
International Open Access Journal
ISSN No: 2456 - 6470 | www.ijtsrd.com | Volume - 2 | Issue – 3

Survey Kernel Optimized Regression Model for Product Recommendation

Lija John
PG Scholar, Computer Science and Engineering,
Sree Buddha College of Engineering,
Elavumthitta, Kerala, India

Vani V Prakash
Assistant Professor, Computer Science and Engineering, Sree Buddha College of Engineering,
Elavumthitta, Kerala, India

ABSTRACT

The boundaries between e-commerce and social networks have become increasingly blurred. Many e-commerce websites support social login mechanisms, and users can use their social network identities (such as Facebook or Twitter accounts) to log in on websites. Users can also post new purchased products on Weibo and link to e-commerce product pages. It proposes a novel solution for cross-site cold start product recommendation, which aims at recommending e-commerce site products to social networking site users in the context of "cold start", which is a problem rarely discussed before. One of the major challenges is how to use cross-site cold start product recommendations using knowledge extracted from social networking sites. It proposes to use linked users as bridges across social networking sites and e-commerce sites (users who have social network accounts and who have already shopped on e-commerce sites), mapping the user's social networking capabilities to another functional representation of product recommendations. Specifically, it is recommended to learn the user's and product's characteristic representation (referred to as user embedding and product embedding, respectively) from data collected from e-commerce websites that use recursive neural networks, and then apply the modified gradient-enhanced tree method to change the user's social network. Feature embedded user. Then develop a feature-based matrix decomposition method that can use learning user embedding for cold start product recommendation [1].

Keywords: *Recommender systems, Cross-domain recommendation, Social network mining*

I. INTRODUCTION

In the past two decades, the amount of data stored in databases and the number of database applications in business and science have increased dramatically. The success of relational data storage and the development and maturation of data recovery and operating technologies have accelerated the dramatic increase in the amount of electronic storage data. A large amount of stored data contains knowledge of several aspects of their business, which are waiting to be used and for more effective business decision support. The database management system used to manage these data sets currently only allows users to access explicit information that exists in the database, i.e. data [1].

1.1 Recommendation System

The recommender system usually generates a list of recommendations in one of two ways - through collaboration and content-based filtering or personality-based approaches. Collaborative filtering methods build models based on the user's past behavior (previously purchased or selected items and / or numerical scores given to these items) and similar decisions made by other users. The model is then used to predict the items (or item ratings) that the user may be interested in. Content-based filtering uses a series of discrete features of a project to recommend other items with similar attributes. These methods are often combined (Hybrid Recommendation System).

2. LITERATURE SURVEY

The project work is mainly related to three lines of research:

- Recommender systems
- Cross-domain recommendation
- Social network mining

3.1 Opportunity Models for E-commerce Recommendation: Right Product, Right Time [2].

Most existing e-commerce recommendation systems are intended to recommend the correct product to users based on whether they are likely to buy or like the product. On the other hand, the validity of the advice also depends on the proposed time. Let's take the example of a user who just bought a laptop. She may buy replacement batteries within two years (assuming the laptop's original batteries often fail to work during the time) and buy new laptops for another two years. In this situation, it is not a good idea to recommend a new laptop or replace the battery immediately after the user purchases a new one. If she receives a potentially correct product recommendation at the wrong time, she may compromise the user's satisfaction with the referral system. The system should not only recommend the most relevant items, but also recommend it at the right time [2].

How to recommend the right product at the right time? It applies the proportional risk modeling approach in survival analysis to recommended research areas and proposes a new opportunity model that incorporates time explicitly into the e-commerce recommendation system. The new model estimates the joint probability that a user will later purchase a particular product at a particular time. Such co-purchase probabilities may be leveraged by the recommender system in a variety of scenarios, including recommended scenarios pulled based on zero queries (example, recommendations on e-commerce sites) and proactive push campaigns (example, based on e-mail or text message marketing). Evaluate Opportunity Modeling with Multiple Indicators.

The major contribution of this paper includes:

- When do you recommend the right product in the field of e-commerce?
- In order to solve this problem, it proposes a principled approach (example, a chance model) to predict the joint probability of buying products and the time of the event. As part of the solution, it extends the proportional hazards model in hierarchical Bayesian statistics and derives detailed reasoning steps based on the varying Bayesian algorithm.
- Joint probabilities are valid in both zero-query-based referrals and proactive push emails / messages. In particular, Probability allows an active recommendation agent to decide whether to send recommendations to users for certain items at a particular time based on the Fixed Utility Optimization Framework. Opportunity modeling methods significantly improve user satisfaction and system conversion rates.

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

Recommended algorithms are well-known for their use on e-commerce websites, where they generate a list of recommended items using input on customer's interests. Many applications use only items that customers purchase and explicitly evaluate their interests, but they also have access to other attributes, including items viewed, demographics, topic interests, and favourite artists. On the Amazon website, it uses a recommended algorithm to personalize an online store for each customer. The store changes fundamentally based on customer interest, shows the software engineer program titles, and shows new mothers the baby toys.

Instead of matching users with similar customers, collaborative filtering of items to items matches the items purchased and rated by each user to similar items, and then groups the similar items into a list of recommendations. In order to determine the most similar match for a given item, the algorithm builds a similar item list by finding items that customers tend to buy together. It builds a product-to-product matrix by traversing all pairs of items and calculating the similarity measure for each pair. However, many products do not have a common client and therefore are inefficient in processing time and memory usage. The following iterative algorithm provides a better method of calculating the similarity between a single product and all related products:

```
For each item in product catalog, $I_1$
For each customer $C$ who purchased $I_1$
For each item $I_2$ purchased by customer $C$
Record that a customer purchased $I_1$ and $I_2$
For each item $I_2$
Compute the similarity between $I_1$ and $I_2$
```

In a similar project table, the algorithm finds items that are similar to the user's purchase and rating, summarizes the items, and then recommends the most popular or related items. This calculation is very fast and depends only on the number of items purchased or evaluated by the user [3].

3.3 Leveraging product adopter information from online reviews for product recommendation [4].

This article presents a new method of extracting product references from online reviews. The extracted product adopters are then sorted into many different demographic groups. The general population demographics of many product adopters can be used to characterize products and users and can be incorporated into recommended methods using weighted regularization matrix decomposition [4].

A novel bootstrapping method for extracting product references from review documents and then discussing how to classify extracted product adopters into six user categories based on the idea of demographic segmentation in market research. The product consumer categories are then used to characterize products and users in the form of product consumer distributions or user preference distributions. A preliminary data analysis is conducted to determine if a large number of product or user product user distributions or user preference distributions are observed to peak in the minority user categories and whether users prefer to purchase products with similar demographic characteristics.

3.4 Addressing Cold-Start in App Recommendation: Latent User Models Constructed from Twitter Followers [5].

It describes a way to explain emerging information from Twitter in order to provide advice on this cold-start scenario. The Twitter handle is used to access the app's Twitter account and extract the ID of their Twitter follower. A fake document is created that contains the ID of the Twitter user who is interested in the application and then applies potential Dirichlet assignments to generate potential groups. At test time, target users seeking referrals are mapped to these potential groups. By using the relationship between potential groups and applications, we estimate the probability that users prefer applications [5].From the handle, it identifies the ID of the follower who followed the app. By focusing on the application's Twitter processing, Twitter followers can subscribe to tweets related to a particular application, which can be viewed as an indication of interest. Figure 3.1 illustrates the relationship between users, applications, and Twitter followers.

By using information from app Twitter followers, it can build "potential people" from two data sources: the app store and Twitter. Using these potential personalities, our algorithm can recommend newly released apps without rating.

Potential Dirichlet Assignment (LDA) - A probabilistic model of probabilities used to discover potential semantics, primarily for text corpora. Given a set of text documents, LDA generates a probability distribution for the underlying "topics" for each document in the corpus, each of which is textually distributed. Documents with similar topics share the same potential topic distribution. Adjust LDA for collaborative filtering. Users download apps and apps may have Twitter followers. Therefore, user u and Twitter user (the user has already downloaded the application) are similar to the text in the document and the document [5].

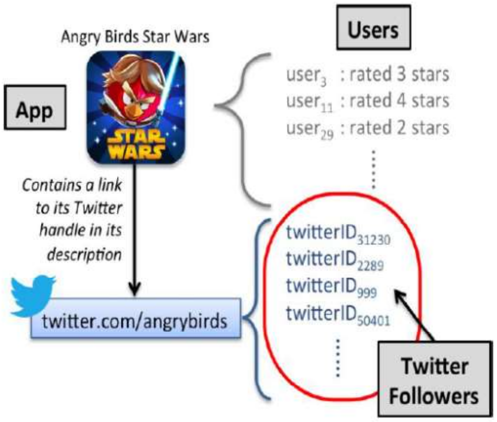

Figure 3.1 [5]: Instead of solely relying on the ratings of users, approach also makes use of the Twitter IDs that follow the apps (red oval).

3.5 Personalized Rating Prediction for New Users Using Latent Factor Models [6].

Personalized advice has played an important role in helping users deal with large amounts of online information. Personalized advice is usually based on rating forecasts, so accurate rating forecasts are crucial to generating useful recommendations. Recently, the matrix-based rating prediction algorithm has become increasingly popular due to its high precision and scalability. However, these algorithms still provide inaccurate rating forecasts for new users who submit only a small number of ratings. It addresses new user issues by introducing several extensions to the basic matrix factorization algorithm that considers user attributes in generating rating predictions. It takes into account the demographic properties explicitly provided by the user and the inferred properties of the user-generated text [6].

The MFUA model is defined for binary user attributes. Some user characteristics (such as gender) are naturally binary, but others (such as age) are defined as discrete or continuous proportions. Binaryizing these features can result in a large number of binary attributes, causing the user to sparsely represent. For population attributes, it uses principal component analysis (PCA), whereas for textual attributes and uses potential Dirichlet allocation (LDA). PCA is a well-dimensioned technique. The main idea of PCA is that variables are often not independent. Therefore, PCA transforms the original variables into new, irrelevant variables (main components) that are ordered so that the first major component interprets the fact that the original data contains Most of the changes. It translates the original user attributes into the main component space, retaining only the first few components for a compact representation of the user. The transformed attribute values are binarized by dividing the values into quartiles and defining one attribute for each value falling within each quartile (ie, four attributes for each of the principal components).

3.6 Addressing Cold Start in Recommender Systems: A Semi-supervised Co-training Algorithm [7].

Cold start is one of the most challenging issues in the recommended system. It solves the cold-start problem by proposing a context-aware, semi-supervised co-training method called CSEL. Specifically, the decomposition model is used to capture fine-grained user project contexts. Then, in order to build a model that can improve the recommended performance by using context, a semi-supervised integrated learning algorithm is proposed. The algorithm constructs different (weak) prediction models using examples with different contexts and then employs a co-training strategy to allow each (weak) prediction model to learn from the other prediction models. This method has several significant advantages over standard recommended methods for solving cold-start problems. First, it defines a more precise fine-grained context for modeling user project preferences. Second, this method naturally supports supervised learning and semi-supervised learning, which provides a flexible way to incorporate unlabeled data [7].

To summarize, contributions of the work include:

- Through in-depth analysis of the existing algorithms, a fine-grained modeling method is proposed to capture the user project context.
- It proposed a semi-supervised collaborative training named CSEL to take advantage of unlabeled examples. With a priori context-aware model, CSELs can construct different regressors by merging different contexts, which ideally helps to capture data features from different views.
- The empirical study of the actual data set verifies the validity of the proposed semi-supervised cooperative training algorithm.

Based on the learning context, it proposes a semi-supervised Collaborative Training (CSEL) framework to deal with cold-start problems in the proposed system. The CSEL aims to build a semi-supervised learning process by assembling two models generated using the context-aware model described above to provide more accurate predictions. Specifically, CSEL contains three main steps.

- Constructing multiple regressors. Addressing a regression problem, the first step is to construct the two regressors, i.e. h_1 and $h_2$from L, each of which is then refined with the unlabeled examples that are labeled by the latest version of their peer regressors.
- Co-Training. In the co-training, each regressor can learn from each other. More accurately, those examples with high confidences are selected for the regressor and being labeled (predicted) by it, and later to be used to “teach” the other regressors.
- Assembling the results. In the end the results obtained by the individual regressors are assembled to form the final solution.

3.7 Methods and Metrics for Cold-Start Recommendations [8].

Developed a method for recommending items that combine content and collaboration data in a single probability framework. The naive Bayesian classifier algorithm for cold-start problems would like to recommend items that are still unmanned in the community. It systematically explores three test methods that use publicly available datasets and explains how these methods are applied to a particular practical application.

It promotes heuristic recommendations in benchmarks to provide effective benchmarking performance. It introduces a new CROC curve of performance metrics and empirically demonstrates that the various components of our test strategy combine to provide a deeper understanding of the performance characteristics of a proposed system. Although our test focus is on cold start recommendations, our recommendations and evaluation methods are general [8].

It identifies three test patterns on a dataset corresponding to different real-world applications. Statements on recommended system performance should be based on the recommended tasks simulated in the test. Differences in test patterns and evaluation and purchase data in the role of recommended issues. It may want to predict the customer will buy an item, or the customer will buy and like items. The final task is to evaluate the purchased item (guess) the customer's rating. Implicit rating forecasting refers to the forecast of purchasing history and other data; purchase does not necessarily mean satisfaction, but the purchase of an item that represents an implied need or desire.

The GROC and CROC charts compare hedonic / actor models, naïve Bayesian recommenders, and heuristic recommenders on the implicit rating prediction mission (for the GROC case). The heuristic recommender for the GROC graph is created by replacing the recommender output with the total number of movies seen by person i. CROC Graphs Comparing People / Actors and Naïve Bayes Approach Implicit Rating Prediction Tasks There are no heuristic recommendations. Due to the cold-start issue, there are no obvious heuristic recommender programs available except for the random recommendation. Note that both machine learning methods performed significantly better than the random predicted area of 0.5 as dictated by the CROC and GROC indicators [8].

3.8 Wisdom of the Better Few: Cold Start Recommendation via Representative based Rating Elicitation [9].

Because new users and / or items are always present, the recommender system must deal with cold-start issues. Rating Inspiration is a common way of dealing with cold starts. However, there is still a lack of a guiding principle on how to choose the most useful rating.

Use a principled approach to identifying representative users and projects using representative matrix decomposition. Selected delegates outperformed other competing approaches in achieving a good balance between coverage and diversity, but it also showed that rating of the selected delegates was more useful (some 10% better than competing approaches) in making recommendations [9].

The representative group should include active users who can well represent the entire population but have less overlap. It follows that this guide has been designed to represent the pursuit of algorithms. It consists of two main steps:

- Dimension Reduction: When attempting to preserve the relationships between users to the maximum, reduce the dimensionality of Y's column space from m to k,
- Basic Choices: Choose k representative users to form a conditional basis in reduced space.

The Representative Matrix Decomposition (RBMF) model allows us to identify the most representative set of users and projects based on past ratings and provide recommendations to existing users. It also provides an intuitive rating heuristic for new users and projects. In most real-world systems, new projects and new users are constantly added to the system.

The recommendation system should be able to quickly adapt its model so that new users and new projects can be advised as soon as possible. This requires technology to learn parameters related to new users and new projects based on incremental new data without the need to completely retrain the entire model. This type of technique is also called folding. With the RBMF model, folding is effortless.

In particular, it only needs to score a new item from k representative users in order to recommend it to other users.

3.9 We Know What You Want to Buy: A Demographic-based System for Product Recommendation on Microblogs [10].

Product recommendation systems are often deployed by e-commerce sites to improve user experience and increase sales. However, advice is limited by the product information hosted on these e-commerce sites and is only triggered when users perform e-commerce activities.

It has developed a new product recommendation system called METIS, an intelligent recommendation system that detects users' buying intentions in real time, based on user demographics and product demographics extracted from the user's public profile Learn from Weibo and online reviews.

The difference between METIS and traditional product recommendation system is: 1) METIS was developed based on Weibo service platform. Therefore, it is not limited by the information on any particular ecommerce website.

In addition, METIS tracks users' buying intentions in near real time and makes recommendations accordingly. 2) In METIS, product recommendations are defined as rankings of learning. User characteristics extracted from user features published on Weibo and demographics of products learned from online product reviews and Weibo are put on the learning list to rank product recommendation algorithms [10].

The METIS consists of three major components:

- Intention to buy test. This component is designed to detect the user's purchase intention in near real time. To reduce noise, you first filter out irrelevant tweets using manually-constructed keyword lists. A classification-based approach is then used to identify tweets that contain the purchase intentions. Although only text features are used to learn classifiers, it suggests considering the user's demographic information as well as vocabulary and grammar information from tweets [10].
- Demographic information extraction. The component is divided into two parts: user demographic data extraction and product demographic data extraction. On the user side, it extracts user demographics from the public profile of Weibo sites; and on the product side, it proposes two ways to leverage social media information by extracting online product reviews on e-commerce sites and The following/ mentioned information on Weibo. For users and products, map their demographic information to the same demographic attribute feature space.
- Products Recommended. This is the core component of the system, which returns a list of recommended products to users. It proposes a novel demographic-based recommendation algorithm in which similarity measures are performed between users and products based on features from their demographic information, which are then combined in learning to be highly accurate for product recommendation rankings Framework [10].

3.10 Towards Linking Buyers and Sellers: Detecting Commercial Intent on Twitter [11].

As more and more people use the Twitter Weibo platform to communicate their needs and aspirations, it has become a particularly interesting medium for identifying business activities. Potential buyers and sellers can connect directly, opening up new perspectives and economic possibilities. By examining the commercial intentions, this work is seen as a first step in bringing together buyers and sellers.

Twitter and other microblogging platform for the distribution of personal information provide the appropriate means, thus creating an unprecedented economic opportunity. People often push their needs and desires. They will also post what they want to get out of. From an economic point of view, it would be valuable, for example, to provide the appropriate product information or purchase interest. As a precondition for "linking buyers and sellers," tweets that contain commercial intentions need to be tested [11].

A classification method that is designed to automate tasks that separate tweets into two categories (with / without commercial intentions). Use annotated tweets that contain implicit and explicit business intent for the class that contains "Commercial Intentions." As an attribute type, it uses words and part-of-speech 3 n-gram. Extensive preprocessing and filtering are required to apply Stanford's POS tagger 4 and WEKA machine learning kits.

Filtering includes the following steps: (i) Delete all characters except AZ, az, 0-9 and spaces, (ii) make each character a lower case, and (iii) if not present, append to each tweet Time, and (iv) or more consecutive spaces exactly one space. To generate word attributes, it uses WEKA's preprocessing suite, which includes token differentiation and n-gram creation, setting the n-gram parameter to a range of 2 to 5 (n other parameters do not improve the result).

WEKA provides us with a series of different classifiers that allow us to compare classifiers, for example, about linear and nonlinear decision boundaries. Tweets for comments show implicit and explicit business intent for positive categories (i.e., 120 tweets). The remaining 1215 tweets are negative class representatives. As a property, it uses words and part-of-speech n-gram. Accuracy and recall scores are calculated by 10-fold cross-validation. Because they are primarily interested in getting the high value of a positive category, that is, a tweet that contains a commercial intent, it only reports the accuracy and reviews the positive category scores [11].

REFERENCES

1. Wayne Xin Zhao, Sui Li, Yulan He, “Connecting Social Media to E-Commerce: Cold-Start Product Recommendation Using Microblogging Information”, IEEE Transactions on Knowledge And Data Engineering, VOL. 28, NO. 5, MAY 2016,pp.1147-1158.
2. J. Wang and Y. Zhang, “Opportunity model for E-commerce recommendation: Right product; right time,” in Proc. 36th Int. ACM SIGIR Conf. Res. Develop. Inf. Retrieval, 2013, pp. 303–312.
3. G. Linden, B. Smith, and J. York, “Amazon.com recommendations: Item-to-item collaborative filtering,” IEEE Internet Comput., vol. 7, no. 1, pp. 76–80, Jan./Feb. 2003.
4. J. Wang, W. X. Zhao, Y. He, and X. Li, “Leveraging product adopter information from online reviews for product recommendation,” in Proc. 9th Int. AAAI Conf. Web Social Media, 2015, pp. 464–472.
5. J. Lin, K. Sugiyama, M. Kan, and T. Chua, “Addressing cold-start in app recommendation: Latent user models constructed from twitter followers,” in Proc. 36th Annu. Int. ACM SIGIR Conf. Res. Develop. Inf. Retrieval, 2013, pp. 283–292.
6. Y. Seroussi, F. Bohnert, and I. Zukerman, “Personalised rating prediction for new users using latent factor models,” in Proc. 22nd ACM Conf. Hypertext Hypermedia, 2011, pp. 47–56.
7. M. Zhang, J. Tang, X. Zhang, and X. Xue, “Addressing cold start in recommender systems: A Semi-supervised Co-training algorithm,” in Proc. 37th Int. ACM SIGIR Conf. Res. Develop. Inf. Retrieval, 2014, pp. 73–82.
8. N. N. Liu, X. Meng, C. Liu, and Q. Yang, “Wisdom of the better few: Cold start recommendation via representative based rating elicitation,” in Proc. 5th ACM Conf. Recommender Syst., 2011, pp. 37–44.
9. W. X. Zhao, Y. Guo, Y. He, H. Jiang, Y. Wu, and X. Li, “We know what you want to buy: A demographic-based system for product recommendation on microblogs,” in Proc. 20th ACM SIGKDD Int. Conf. Knowl. Discovery Data Mining, 2014, pp. 1935–1944.
10. B. Hollerit, M. Kr€oll, and M. Strohmaier, “Towards linking buyers and sellers: Detecting commercial intent on twitter,” in Proc. 22nd Int. Conf. World Wide Web, 2013, pp. 629–632.
11. W. Pan, E. W. Xiang, N. N. Liu, and Q. Yang, “Transfer learning in collaborative filtering for sparsity reduction,” in Proc. AAAI, 2010, pp. 230–235.