ISSN (e): 2250 – 3005 || Volume, 14 || Issue, 4|| Jul. - Aug. – 2024 ||
International Journal of Computational Engineering Research (IJCER)

Detecting Deceptive Reviews on Yelp Using Machine Learning Classification Techniques

CHALLA SAI CHARISHMA[1], K.V. DURGA DEVI[2]
#1. M.Tech Scholar in Department of Artificial Intelligence & Data Science,
#2. Assistant Professor, Department of Artificial Intelligence & Data Science, Kakinada Institute Of Engineering & Technology or Women, AP, India.

Abstract –
Presently a day's online shopping has accomplished an enormous disrepute privileged less measure of time. As of late few ecommerce sites has been created their functionalities to a point with the end goal that they suggest the product for their clients alluding to the availability of the clients to the social media and give coordinate login from such social media, (for example, facebook, Google+ ,and so forth). For suggesting the clients that are absolutely new to the sites, we utilize novel answer for cross-webpage cold-start product recommendation that goes for prescribing products from online business sites. In particular, we propose learning the two clients and products include portrayals from information gathered from internet business sites utilizing repetitive Matrix Factorization to change client's social networking highlights into client embeddings. We at that point build up a feature-based matrix factorization approach which can control the learnt client embedding for cold-start product recommendation.
Keywords – *social networks, e-commerce, product recommendation, Microblogging.*

I. Introduction

The present world is ending up completely programmed through Internet. Web gives the most required data. The entrance to Internet makes vast measure of information step by step. Web based business sites, for example, eBay highlights a large number of the attributes of social networks, including continuous notices and assistance between its purchasers and dealers. Some web based business sites likewise bolster the instrument of social login, which enables new clients to sign in with their current login data from social networking administrations, for example, Facebook, Twitter or Google+. Both Facebook and Twitter which has presented another component a year ago had pulled in more purchasers which enabled more number of clients to purchase products straightforwardly from their sites by clicking a "purchase" to buy things in view of a few adverts or different posts. In China, the web based business organization ALIBABA has made a vital interest in SINA WEIBO1 where ALIBABA product adverts can be specifically conveyed to SINA WEIBO clients. With the new pattern of transmitting internet business exercises on social networking destinations the audits, utilizing product adopter data, separated from web based business and profile subtle elements of social networking locales utilized for the advancement of the cold start product recommendation frameworks .In this Recommendation assumes a critical part in numerous fields and has pulled in a ton of research intrigue. For instance, Netflix has discharged a fascinating truth that around 75% of its supporters watch are from recommendations. In a recommender framework, for example, Netflix and Amazon, e-straight, Flipkart, clients can peruse things and pick those things they are occupied with, the ad likewise assumes a noteworthy part were in the framework additionally prescribe the product to the clients. At that point the things that the framework thought as a best one will be the best match of inclination to the product recommendation. A while later, the client may give criticism, (for example, rating, normally spoke to as a score between, for instance, 1 and 5, additionally the surveys settle on a tremendous choice in the product buy) on how the client considers a thing after she/he has encountered the thing. One essential assignment for the recommendation motor is to comprehend clients' customized inclinations from their notable rating practices. In this paper, we contemplate a fascinating issue of prescribing products from web based business sites to clients at social networking locales who don't have authentic buy records, i.e., in "cold-start" circumstances. We called it cross-site cold-start product recommendation. Albeit online product recommendation has been broadly. Most investigations just spotlight on developing arrangements inside certain web based business sites and for the most part use client's authentic exchange records. To the best of our insight, cross-site cold-start product recommendation has been seldom examined previously. Another testing undertaking is the way to enhance the recommendation exactness for the new (or once in a while evaluated) things and the new (or latent) clients. Contrasting with the mainstream things, for the recently discharged ones and the old things that are once

in a while appraised by clients, it is troublesome for the standard recommendation methodologies, for example, synergistic sifting way to deal with give fantastic recommendations.

II. Related work

Opportunity Models for E-business Recommendation: Right Product, Right Time Author: Jian Wang, Yi Zhang This paper contemplates the new issue: how to suggest the correct product at the perfect time? We adjust the corresponding risks displaying approach in survival examination to the recommendation inquire about field and propose another open door model to expressly fuse time in an internet business recommender framework. The new model gauges the joint likelihood of a client influencing a followup to buy of a specific product at a specific time. This joint buy likelihood can be utilized by recommender frameworks in different situations, including the zero-inquiry pull-based recommendation situation (e.g. recommendation on an online business site) and a proactive push-based advancement situation (e.g. email or instant message based blemish keting). We assess the open door demonstrating approach with numerous measurements. Trial comes about on an information gathered by a realworld ecommerce website(shop.com) demonstrate that it can anticipate a client's subsequent buy conduct at once with drop exactness. What's more, the open door show fundamentally enhances the change rate in pull-based frameworks and the client fulfillment/utility in push-based frameworks 2.2 We Know What You Want to Buy: A Demographic-construct System for Product Recommendation In light of Microblogs Author: Wayne Xin Zhao1, YanweiGuo Product recommender frameworks are frequently sent by ecommerce sites to enhance client experience and increment deals. Be that as it may, recommendation is restricted by the product data facilitated in those web based business destinations and is just activated when clients are performing ecommerce exercises. In this paper, we build up a novel product recommender framework called METIS, a Merchant Intelligence Recommender System, which identifies clients' buy aims from their microblogs in close continuous and makes product recommendation in light of coordinating the clients' statistic data separated from their open profiles with product socioeconomics gained from microblogs and online surveys. METIS separates itself from customary product recommender frameworks in the accompanying angles: 1) METIS was created in view of a microblogging administration stage. All things considered, it isn't constrained by the data accessible in a particular online business site. What's more, METIS can track clients' buy purposes in close realtime and make recommendations likewise. 2) In METIS, product recommendation is confined as a figuring out how to rank issue. Clients' attributes removed from their open profiles in microblogs and products' socioeconomics gained from both online product audits and microblogs are bolstered into figuring out how to rank calculations for product recommendation. We have assessed our framework in a substantial dataset crept from Sina Weibo. The test comes about have checked the possibility and adequacy of our framework. We have likewise made a demo variant of our framework freely accessible and have executed a live framework which enables enrolled clients to get recommendations progressively. Retail Sales Prediction and Item Recommendations Using Customer Demographics at Store Level Author: Michael Giering This paper traces a retail deals expectation and product recommendation framework that was actualized for a chain of retail locations. The relative significance of buyer statistic qualities for precisely demonstrating the offers of every client write are determined and executed in the model. Information comprised of every day deals data for 600 products at the store level, broken out finished an arrangement of non over lapping client composes. A recommender framework was constructed in view of a quick online thin Singular Value Decomposition. It is demonstrated that displaying information at a better level of detail by bunching crosswise over client writes and socioeconomics yields enhanced execution contrasted with a solitary total model worked for the whole dataset. Points of interest of the framework usage are portrayed and functional issues that emerge in such genuine applications are examined. Preparatory outcomes from test stores over a one year time span show that the framework brought about fundamentally expanded deals and enhanced efficiencies. A concise diagram of how the essential strategies talked about here were reached out to a significantly bigger informational collection is given to affirm and represent the versatility of this approach

III. MICROBLOGGING SERVICES

Microblogging Feature selection Methods are:

- Demographic Attributes
- Text Attributes
- Network Attributes
- Temporal Attributes

Table I. Categarisation of the Microblogging Features

Categories	Features
Demographic Attributes	Gender, age, marital status, education, career, etc.
Text Attributes	Topic distribution, Word embedding
Network Attributes	Latent group preferences
Temporal Attributes	Daily activity distribution, weekly activity distribution

IV. Proposed Methodology

Social networks offer new ways to reach first-time customers, engage and reward existing customers, and showcase the best your brand has to offer. Your social network profiles and the content you share are as important as a business' storefront signage and displays in the 1950s. Businesses that integrate social media into their marketing strategy – from customer acquisition, to sales, to re-engagement campaigns – will benefit. Marketers can see in real-time what your audience cares about most, their interests, the conversations they're having and what they like. Use your social networks to better segment audience and understand your target demographics. This will help you optimize your campaigns and deliver more targeted messaging. Immediacy is big in social media; we want information and we want it now. That's why social networks are so great for customer service. They enable businesses to quickly respond to customer inquiries. Plus, social media makes it easier to spot and respond to unpleasant customer experiences. Develop a strategy for responding to customer inquiries via social media. Gathering the details from the user's social network profile and creating product recommendations. The profile details will match with the news server for gathering recommended news information. According to the news three news channels will be consolidated for getting the confident information. Recommended product information can be changed according to the user's public chat and user's updated profile details. News details available in search mode also, news can be search area wise, city wise and state wise.

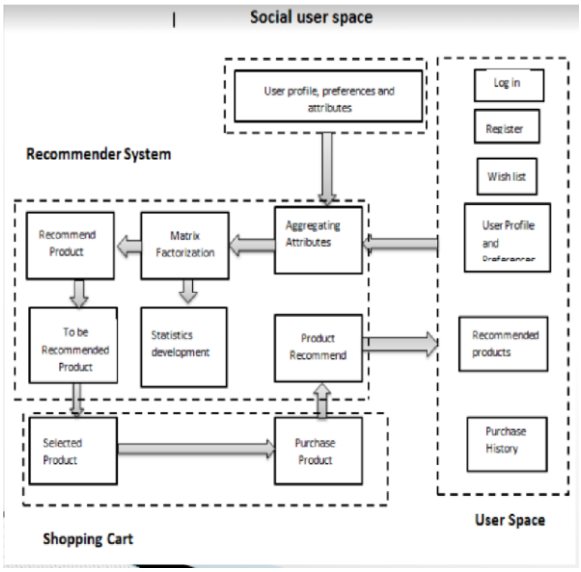

Fig -1: Proposed Architecture

In the above fig-1, Architecture is shown. The system architecture user space is allocated to user with the availability of full access to his/her account including the purchases made. For aggregating the attributed/information of the user is fetched from the social user space of user's social network. The recommender system in turn process the algorithm in order to predict the products to be recommended.

System architecture describes the overview and exact flow of working. In this system, first we collect the dataset as an input. The dataset may contain the user's information and list of products. Feature extraction process is done for both the sites i.e. social networking sites and e-commerce websites. These extracted features are mapped for product recommendation using content propagation and matrix factorization algorithm. The flow of the system is sequencial. Matrix factorization technique is used for the representation of the user-item rating matrix.

V. Performance Evolution

Our task requires data from both an e-commerce website and an online social networking site. E-commerce data we used a large e-commerce dataset shared by [7], which has 138.9 million transaction records from 12 million users on 0.2 million products. Each transaction record consists of a user ID, a product ID and the purchase timestamp. We first group transaction records by user IDs and then get a list of purchased products for each user.

Micro blogging data: We used our previous data [6] collected from the largest Chinese micro blogging site SINA WEIBO, in which we have retrieved a total of 1.7 billion tweets from five million active users within a half-year time span from January 2013 to June 2013.

User linkage: We have found that WEIBO users sometimes shared their purchase record on their micro blogs via a systemgenerated short URL, which links to the corresponding product entry on JINGDONG. By following the URL link, we can meet the JINGDONG history of the WEIBO user. We

TABLE 1
Statistics of Our Linked User Datasets

Datasets	#users	#products	Average #products	Average #tweets
$\mathcal{D}_{dense}$	15,853	98,900	52.0	41.0
$\mathcal{D}_{sparse}$	4,785	6,699	2.6	35.7

TABLE 2
Performance Comparisons of MAE Results for Fitting User Embedding on D_{dense}

$\frac{\#train}{\#test}$	CART	$MART_{old}$	$MART_{sample}$	$MART_{both}$
1/1	0.557	**0.515**	**0.515**	**0.515**
1/4	0.557	0.522	**0.521**	**0.521**
1/9	0.564	0.589	0.558	**0.529**

Smaller is better.

Classified (23,917) associated users out of five million active users by scanning tweets in this way. We first filter out 3,279 users with too little information on their WEIBO public profiles. Next, we further divide users into two groups. The first group has users with more than five product purchases, denote as D_{dense}. The second group has the remaining users, denoted as D_{sparse}. The statistics of these linked users explain in Table 1. For privacy consideration, all the WEIBO IDs and JINGDONG IDs of all linked users explain by unique IDs, and all their textual information and buying information is encoded with numeric symbols.

Evaluation on User Embedded Fitting

Given a linked user $u \in U^L$, we have the micro blogging feature vector a_u extracted from WEIBO and the user embedding v_u learnt based on her JINGDONG purchase record. We use a regression-based approach to fit v_u with a_u for heterogeneous feature mapping, and the fitted vector is denoted as $\hat{v}_u$. To check the effectiveness of the regression performance, the Mean Absolute Error (MAE) is used as the evaluation metric

$$MAE = \frac{1}{|\tau|}\{\sum_{u\in\tau}\frac{\sum_{k=1}^{K}|v_{u,k}-\hat{v}_{u,k}|}{K}\} \quad \textbf{(1)}$$

where $|T|$ is the number of test users. We consider three different comparison methods: (1) CART, (2) $MART_{old}$, which is the original implementation (3) $MART_{sample}$, which is our modified implementation with feature sampling; (4) $MART_{both}$, which is our modified application with feature sampling and fitting refinement.

For user embedding fitting, we use D_{dense} for evaluation, since the users in D_{dense} have a considerable number of purchases for learning the ground truth user embedding using our modified parallel method, which are more reliable for evaluation. The dataset $_{Ddense}$ is split by users into training set and test set with three different $\frac{\#train}{\#test}$ ratios, namely 1:1, 1:4 and 1:9. We use a similar evaluation method as N-fold cross validation. Given the $\frac{\#train}{\#test}$ ratio of 1: *N*, each fold will be treated as the training data exactly once and the rest *N*-1 folds are treated as the test data, the process will be repeated *N* times and the last results are averaged over *N* such runs the number of boosting

iterations for all MART variants and the values of μ_1 and μ_2 for MARTboth are optimized by *N*-fold cross validation.

In Table 2, we can see that when the training data is relatively large (ratio 1:1), all the MART variants give similar results and they do consistently better than the simple CART. Interestingly, when the size of training data becomes

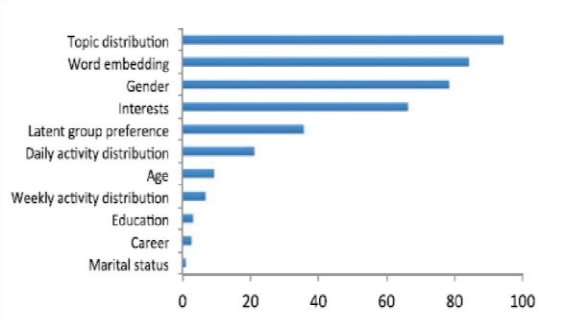

Fig.2. Relative attribute importance ranking (corresponding to the features)

Smaller, MART_{sample} and MART_{both} outperforms MART_{old}. In specific, the performance gain achieved by MART_{both} over the other two MART variants is more significant with smaller set of training data. These results show that our changes of feature sampling and fitting refinement are very effective.

Relative attribute importance. Tree-based methods offer other feasibility to learn relative importance of each attribute. Inspired by the method introduced in, we calculate a statistic of the relative importance of each attribute for MART based on the training data. Recall that in MART, each feature corresponds to an attribute value. First, we traverse through all the regression trees, and calculate for each feature its contribution to the cost function by adding up the contributions of all the nodes that are split by this feature. Here we define feature contribution explain of the squared error in the loss function. For each attribute, we can sum up the contributions of all of its possible attribute values as its overall contribution.

The results are shown in Fig. 2. We have the following observations: 1) The text attributes occupy the top two rank positions; 2) Within the demographic group, Gender and Interests are more important than the others. 3) The social based attributes are ranked relatively lower compared to the other two categories. It seems that demographic attributes are less important than text attributes in our dataset. One possible reason is that many demographic attribute values are missing in users' public profiles on WEIBO. still, the ranking of relative attention of attributes does not entirely confine on their completeness proportion. For example, Interests is more important than Latent group preference even though the later has a larger completeness proportion. Another possible reason is that the feature dimension for text attributes is larger than that of demographic attributes, e.g., Topic Distribution has fifty feature dimensions while Gender only has two feature dimensions. We can also test the importance of each attribute by conducting experiments on the traditional product recommended task. We use the standard MF approach as a baseline and add attributes one at a time using the SVD Feature framework discussed, and then check the performance improvement yielded by the added attribute. The attribute ranking obtained in this way is similar to the ranking in Fig. 1, but the gap between text attributes and demographic attributes becomes smaller.

Evaluation on Cold-Start Product Recommended

For cold-start product recommended, we aim to recommended products to micro blog users without their knowledge of their factual purchase records.

Construction of the Evaluation Set

The evaluation set splits users into training set and test set. For the training set, we sample negative products with a ratio of 1:1 for each user, i.e., we have the same number of negative and positive products. For the test set, we randomly sample negative products with a ratio of 1:50 for each user, i.e., each positive product would involve 50 negative products. All negative products are sampled from the same product class as the corresponding positive one. For example, for "iPhone 6", we can sample "Samsung Galaxy S5" from the "Mobile Phones" category as a negative product. Given a user, we can generate a list of candidate products consisting of both positive and negative products. On average, a user has about 52 positive products and 2,600 negative products in our experimental dataset, which is indeed a challenging task. Similar to the evaluation scenario in Information Retrieval, we would like to look at the performance that a system ranks positive products over negative products.

Methods to Compare

We consider the following methods for performance comparison:

Popularity (Pop): products are ranked by their historical sale volumes.

Popularity with Semantic Similarity (Pop++): the ranking score is a combination of two scores: (1) the popularity score S_1, (2) the cosine similarity S_2 between product description and user text information, including profile, tweets and tags. The two scores are combined by log $(1+S_1) \times$ log $(1+ S_2)$.

Embedding Similarities (ES): Similarity scores $\hat{v}_u^T.v_p$ between a user embedding $\hat{v}_u$ and a list of product embeddings v_p are used to rank products.

MF with user attributes (MFUA): User attributes (including user profile and topic distributions) are incorporated into the basic matrix factorization algorithm for product rating prediction [8]. For fairness, we also use the pair wise loss function to train the model.

FM without User Interactions (FMUI): Rendle applied the Factorization Machines for "follow" recommended in KDDCup 2012. It has been found that similar performance was obtained with or without the interactions of user features. FM without user feature interactions is equivalent to SVD Feature. We re-implement this method in the SVD Feature groundwork with our derived micro blogging features.

Cold$_E$: Our proposed approach which uses the fitted user embedding features and product embedded features.

Cold$_{D+E}$: Our proposed approach which uses the micro blogging features, the product embedding features and the fitted user embedding features. Especially, we only use analytical attributes here, since they have been displayed important to product recommended [6].

Cold$_{++}$: Since the user and product ingrained can be learned for all the users and products appropriately in the e-commerce website, we can convoy Cold$_E$ with all the users in U, not limited to the linked users U^L. This variant is called Cold$_{enhanced}$.

We set the regularize coordinate to a 0.00400, the emphasis number to 50.00 and the factor number to 32.00 for all the approach. We use the CBOW building to learn the embedding vectors based on the purchase records from all the non-linked users and the partial purchase records from linked users in our training set. The number of dimensions of embedding vectors is set to 50. The user embedding features in the test sets for different $\frac{\#training}{\#test}$ settings are set to the values fitted using MART$_{both}$. For Cold enhance, we add additional 10,000 randomly selected non-linked users from U into the training set.

Evaluation Metrics for Product Recommended

Five widely used metrics are used for the evaluation of product recommended results, including Precision@k, Recall@k, the Mean Average Precision (MAP), the Mean Reciprocal Rank (MRR) and the area under the ROC Curve (AUC).

Experimental Results on D_{dense}

We first test the performance of product recommended on Ddense, where d percent linked users are used as the training data, and the remaining (100-δ) percent linked users as the test data. To test the performance with varying amount of training data, we set d to 80, 50, 20 and 10, which correspond to the $\frac{\#training}{\#test}$ Split Ratios (SR) of 4:1, 1:1, 1:4 and 1:9.

The results of different methods for overall product recommended are presented in Table 3. It can be observed that:

 Apart from the simple baseline Popularity, which does not rely on any training data, the performance of all other methods improves with the increasing size of the training data. Popularity appears to be a competitive baseline for cold-start recommended due to the fact that negative products are selected from the same product categories as the positive ones. By

TABLE 3
Performance Comparisons of Different Methods on Cold-Start Product Recommended

SR	Methods	P@10	R@50	MAP	MRR	AUC
4:1	Pop	0.175	0.215	0.120	0.380	0.669
	Pop_{++}	0.175	0.215	0.120	0.380	0.669
	ES	0.117	0.195	0.115	0.267	0.653
	MFUA	0.212	0.245	0.136	0.495	0.701
	FMUI	0.226	0.253	0.145	0.502	0.730
	$Cold_E$	0.237	0.265	0.155	0.512	0.751
	$Cold_{D+E}$	**0.243***	**0.270***	**0.159***	**0.527***	**0.771***
	$Cold_{++}$	0.239	0.261	0.157	0.517	0.763
1:1	Pop	0.175	0.215	0.120	0.380	0.669
	Pop_{++}	0.175	0.215	0.120	0.380	0.669
	ES	0.117	0.195	0.115	0.267	0.653
	MFUA	0.210	0.240	0.130	0.469	0.681
	FMUI	0.215	0.241	0.125	0.481	0.687
	$Cold_E$	0.222	0.251	0.142	0.484	0.724
	$Cold_{D+E}$	**0.229***	**0.257***	**0.146***	**0.508***	**0.734***
	$Cold_{++}$	0.226	0.255	0.146	0.497	0.730
1:4	Pop	0.175	0.215	0.120	0.380	0.669
	Pop_{++}	0.175	0.215	0.120	0.380	0.669
	ES	0.117	0.195	0.115	0.267	0.653
	MFUA	0.202	0.231	0.126	0.449	0.693
	FMUI	0.186	0.225	0.131	0.389	0.670
	$Cold_E$	0.216	0.243	0.137	0.475	0.700
	$Cold_{D+E}$	0.218	0.248	0.137	0.477	0.705
	$Cold_{++}$	**0.220***	**0.249***	**0.140***	**0.484***	**0.715***
1:9	Pop	0.175	0.215	0.120	0.380	0.669
	Pop_{++}	0.175	0.215	0.120	0.380	0.669
	ES	0.117	0.195	0.115	0.267	0.653
	MFUA	0.193	0.230	0.118	0.439	0.678
	FMUI	0.172	0.225	0.117	0.411	0.668
	$Cold_E$	0.205	0.234	0.128	0.461	0.683
	$Cold_{D+E}$	0.206	0.238	0.129	0.473	0.685
	$Cold_{++}$	**0.217***	**0.245***	**0.138***	**0.482***	**0.695***

* *indicates that our Cold method is significantly better than the best baseline at the level of 0.01.*

Incorporate the semantic similarity between users and products it leads to negligible performance change, which indicates the simple surface similarity cannot well capture the purchase preferences.

- FMUI performs better than MFUA on the dataset with the split ratios of 1:1 and 4:1, but is worse with the other two ratios. A possible reason is that FMUI involves all the micro blogging attributes and thus potentially requires more training data for a better performance. When the training data is finite, FMUI cannot collect ample statistics for some micro blogging attributes due to data sparsity.
- Our proposed Cold variations are often better than the baselines. Interestingly, $Cold_{enhanced}$ is not sensitive to the amount of training data, which gives rather stable performance across all the three ratios. By integrating other demographic attributes, $Cold_{D+E}$ is often better than $Cold_E$, and the improvement seems more significant when the training data is abundant (at the ratio of 1:1). When the training data is limited, $Cold_{++}$ outperforms all the other methods. But with more training data, it performs slightly worse than $Cold_{D+E}$.

TABLE 4
Performance Comparisons of Different Methods on Cold-Start Product Recommended on D_{sparse}

Methods	MAP	MRR	R@10	AUC
Pop	0.175	0.125	0.120	0.684
Pop_{++}	0.175	0.175	0.120	0.684
MFUA	0.251	0.337	0.419	0.718
FMUI	0.252	0.337	0.421	0.720
$Cold_E$	**0.275***	**0.363***	**0.458***	**0.757***

* *indicates that $Cold_E$ is significantly better than the best baseline at the level of 0.01.*

Experimental Results on D_{sparse}

We have examined the performance of product recommended on frequent buyers above. In real-world applications, "longtail" users (i.e., those with few purchases) are prevalent in e-commerce Websites. Therefore, an effective recommending system should also be capable of generating recommending to the users. We use the users in D_{dense} as the coaching data for both user inlay fitting and matrix factorization learning, and consider the users in D_{sparse} as the test data for product recommended. Since the users in D_{sparse} have fewer than five purchases, we only report the performance of Recall@k but not Precision@k. We also use MRR and AUC as appraisal metrics. We can see that from the Table 4 that our expected approach $Cold_E$ is often superior than all the baselines, which indicates that the effectiveness of recommended for long-tail users.

Scalability Analysis

We present the scalable scrutiny for our model Cold$_E$. We first check the time complexity for both offline parameter guidance and online product recommended. For offline factor criterion, the cost of training the MART models is $N_{tree} \times \bar{C}_{tree}$, where Ntree is the number of trees and $\bar{C}_{tree}$is the average cost for generating a decision regression tree. Then, the SGD method to train Cold$_E$ has the computational complexity of O ($nL\bar{|}D|$), where n is the iteration number, L is the number of latent factors, $\bar{}$ is the average number of non-zero features for a training instance and $|D|$ is the training data size. In practice, we have found that SGD converges quickly and usually converges in 30-50 iterations on our training set. For online product recommended, when a new user arrives, we first generate the fitted user embedding features, at most incurring a cost of $h_{max} \times N_{tree}$, where h_{max} is the largest tree height. When making recommended, we use to score each candidate product. In, a user incurs a cost of $K \times L$ additions and K multiplications to derive$\sum_{k=1}^{K} \hat{v}_{u,k,x_k}$ and a cost of L multiplications and L additions for dot product, while y_p + $\sum_{k=1}^{K} \hat{v}_{p,k,x_k} y_k$ for all the products are pre-computed. To generate recommended, we further need a cost of $N_{list} \times \log N_{list}$ for ranking candidate products for a user, where N_{list} is the length of candidate product list.

TABLE 5

Running Time and Memory Costs for Our Approach on D_{dense} with the Split $\frac{\#train}{\#test}$ Ratio of 1:1

Phases	#users	Time (sec.)	Space (MB)
Training	7,927	563 (MART) 304 ($Cold_E$)	4.67 (MART) 15.72 ($Cold_E$)
Test	7,926	13.8 (MART) 5.1 ($Cold_E$)	4.67 (MART) 15.72 ($Cold_E$)

While for space complexity, our major cost consists of space for MART models and latent factors. MART models take up a cost of O ($\bar{N}_{node} \times \bar{C}_{node} \times N_{tree}$), where $\bar{N}_{node}$ and C_{node} denotes the average number of nodes in a MART tree and the average space cost for a single node respectively. We have a cost of $(|U| + |P| + K) \times L$ to store latent factors. Compared to traditional matrix factorization, it incurs an additional cost of $K \times L$. In practice, K is usually set to 50~200. We summarize the time and space cost for Cold$_E$ in Table 5. It can be observed that our method is very efficient in online recommended. When dealing with extremely large datasets, the training process can be performed in a distributed way by using SGD, and the test process can still be efficient since it only involves the MART tree traversal and latent vector operations.

Parameter Analysis

For our methods, an important part is the embedding models, which can be set to two simple architectures, namely CBOW and Skip-gram. We analytically compare the results of our approach Cold$_E$ using these two architectures, and find that the performance of using Skip-gram is slightly worse than that of using CBOW. We also check how the performance varied with different some embedded dimension from 50 to 150 with a gap of 25. We inspect that the work is relatively balanced with the shifting number of embedding dimensions. This is not surprising since the MART models fitting each dimension independently. The optimal performance of ColdE was obtained when the dimension number is 100, which is only slight better than that of 50 than real. Thus, using 50 enclosing dimensions would be ample for our recommended tasks seeing the trade-off between performance and computational complexity. For matrix factorization methods, an important factor to set is the number of unrealized factors. We use Cold$_E$ and MFUA as a contrast and vary the number of latent factors from 16 to 80 with a gap of 16. The performance of two methods is relatively stable with different numbers of latent factors, and Cold$_E$ is consistently better than MFUA.

VI. Conclusion

In this paper we conclude that a novel problem, cross-site cold-start product recommendation, i.e., recommending products from ecommerce websites to Micro blogging users without historical purchase records is studied. The main idea is that on the e-commerce websites, users and products can be represented in the same latent feature space through feature learning with the recurrent neural networks. Using a set of linked users across both ecommerce websites and social networking sites as a bridge, feature mapping functions can be learned using a modified gradient boosting trees method, which maps users' attributes extracted from social networking sites onto feature representations learned from e-commerce websites. The mapped user features can be effectively incorporated into a feature-based matrix factorization approach for cloud start product recommendation. A large dataset is constructed from WEIBO and JINGDONG. The results will show that the proposed framework is indeed

effective in addressing the cross-site cold-start product recommendation problem. Currently, only simple neutral network architecture has been employed for user and product embedding learning. In the future, more advanced deep learning models such as Convolution Neural Networks13 can be study for learning.

References

[1]. Wayne Xin Zhao,Sui Li,Yulan he"connecting social media to e-commerce;cold start product recommendation using microblogging information"vol x,No.x,xxx2016

[2]. CHAMSI ABU QUBA Rana,HASSAS Salima,"From a "cold" to a "warm" start in recommender systems" 2014 IEEE 23rd International WETICE conference

[3]. Vibhu Jawa,Varun Hasija,"A sentiment and Interest Based Approach for product recommendation"2015 17th UKSIM-AMSS International Conference on Modelling and Simulation

[4]. Bharat singh,Sanjoy Das,"Issues and challenges of online user generated reviews across social media and e-commrece website"International Conference on computing, communication and automation(ICCCA2015)

[5]. R.Nithya,Dr.D.Maheswari,"Correlation of feature score to overall sentiment score for identifying the promising features"(ICCCI-2016),Jan.07-09,2016,Coimbatore,INDIA

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

[7]. 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 SIGKDD, 2014.

[8]. J. Wang, W. X. Zhao, Y. He, and X. Li, "Leveraging product adopter information from online reviews for product recom- mendation," in ICWSM, 2015.

[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 cold- start in app recommendation: latent user models constructed from twitter followers," in SIGIR, 2013.

[11]. A.Mislove, B. Viswanath, K. P. Gummadi, and P. Druschel, "You are who you know: Inferring user profiles in online social networks," in WSDM, 2010.