A Novel Product Recommendation Approach with Modified Self-Attention Mechanism and Neural Network

[1]Prakruti Patel,[2]Abhishek Patel

[1]ME Student,[2]Assistant Professor
[1]Department of Computer Engineering,
[1]Silver Oak College of Engineering & Technology, Ahmedabad, India.

Abstract: Product recommendation is a technique that is being applied to most of the ecommerce platforms and for different businesses to analyze buyer's behavior periodically. Unsupervised Learning finds a natural fit with unlabeled data. This leads to efficient principal component analysis and in such way we can understand the data and its properties. Moving beyond the age-old techniques, if we use latest machine learning with deep learning algorithms, we are able to calculate right products and the high-dimensional space based recommendation techniques. This enables us to use different clustering and classification methods, such as k-means and k-medians and hierarchical clustering and SVM, NB and random forest classification. We also use Shannon entropy and mean squared error to find optima. If the input data are labelled, then an exponential speedup is feasible over classical algorithms with the proposed algorithm, which is capable of their self-analysis for better results.

***IndexTerms*–Recommendation System, LDA, Factorization, NN Classifier.**

I. INTRODUCTION

Recommendation System [1] is termed info filtering system. For instance, you looked for an item online scarcely any days prior and afterward you continue getting messages for shopping recommendations for different items. Other than this, you may have seen that the shopping site or the application proposes a few things to you that matches with your taste. For instance, when you are purchasing a portable on the web, at that point that webpage will propose some other pertinent things like versatile spread, charger, earphone and so forth. Absolutely, this makes shopping experience simple yet did you realize that it's called appropriate recommendation? The product recommendations are made based on your behavior with the site/application, past buys, things preferred or disdained or added to truck, inclinations for brand and so forth. In the event that you've as of late viewed YouTube recordings about vehicles, at that point YouTube is going to begin indicating you a great deal of vehicle related recordings with comparable titles and subjects! Proposal frameworks utilize a lot of procedures and calculations which can recommend "significant" things to clients. For the most part, the recommended things are as important to the client as could be expected under the circumstances, with the goal that the client can have those things: YouTube recordings, news stories, online items, etc. Recommendation System works in three steps: Data Collection, Data Storage, Filtering the Data. Data Collection is the first and most vital advance to construct a Product Recommendation. The information can be gathered by two strategies: explicitly and implicitly. Explicit information will be data that is given purposefully, for example, input from the users such as movie ratings. Implicit data is information that can't deliberately yet accumulated from accessible information streams like hunt history, clicks, request history, and so forth. The amount of data shows how good the recommendation model will function. For instance, in a film recommendation system, the more ratings users give to films, the better the recommendations get for other users. After gathering and putting away the information, we need to filter it to extract the applicable data required to make the final recommendations.

In the literature, there are basically three algorithms to produce a list of recommendation: collaborative filtering [2], content-based recommendation algorithms [3] and hybrid recommendation algorithms [4]. In Collaborative Filtering User is in focus. This filtering system utilizes information on user's past buys or comparable choices by different users to recommend items. This is called user based recommendation. Collaborative filtering depends on the supposition that individuals who concurred in the past will concur later on, and that they will like comparative sorts of things as they loved before. The thought behind Collaborative Filtering is that the authentic information of the users is sufficient to make a prediction. The main advantage of Collaborative Filtering is that it is not dependent on item information. The disadvantage of using Collaborative Filtering is that it can't prescribe items if no user surveys/connections accessible. So it is hard to make recommendations for new users. There are basically two types of collaborative filtering: User-User Collaborative Filtering and Item-Item Collaborative Filtering. In Content-based Filtering item is in focus. This filtering method utilizes information on every item to suggest a comparable item. This method is called item based recommendation. Content-based filtering techniques are based with respect to a detailed description of the item. Fundamentally, these techniques utilize an item profile to characterize the item within the system. The main advantage of Content-based Filtering is that it works without client audits. The drawback of Content-based Filtering is that it is hard to execute for enormous items as it requires descriptive information for each item. Let's have a case of Netflix. All the data identified with every client is spared in a vector structure that contains past conduct of the client, for example the movies enjoyed/detested and the evaluations. This vector is known as the profile vector. All the data identified with movies is put away in another vector considered the item vector that contains the subtleties of every movie, similar to sort, cast, executive, and so forth. The content-based filtering algorithm finds the comparability between the profile vector and item vector. Most recommendation system uses hybrid approach that mixes cooperative filtering, content-based filtering, or both. Many studies that by trial and error compare the performance of the hybrid with the pure collaborative and content-based ways and demonstrated that the hybrid ways will give a lot of correct recommendations than pure approaches. Netflix may be an ideal of the use of hybrid recommender systems. the web site makes recommendations by comparing the observation and searching habits of

comparable users (i.e., collaborative filtering) likewise as by giving movies that share characteristics with films that a user has rated extremely (content-based filtering).

The rest of the paper is organized as follows: In Section II, the motivation of this work is presented. In Section III, the related research in recent years on the recommendation algorithms is introduced. The explanation of terminologies, methodologies and proposed algorithm methods are presented and analyzed in Section IV. Section V presents algorithm of our proposed method. In Section VI, we verify result with experiments. Finally, the conclusion of the paper is shown in Section VII.

II. MOTIVATION

The explanation of utilizing a recommendation system these days [1] is that individuals have an excessive amount of choices to use from because of the notoriety of Internet. Previously, individuals used to shop from a physical store, in which the accessibility of things is constrained. For example, the quantity of films that can be put in a Blockbuster store relies upon the size of that store. On the other hand, these days, the Internet permits individuals to get to numerous assets on the web. Netflix, for instance, has a tremendous assortment of films. Instead of a physical store to purchase items, online locales contain more items to purchase. In spite of the fact that the accessibility of data expanded, another issue emerged as individuals made some hard memories choosing the things they really need to see. This is the point at which the Recommendation System is utilized. Recommendation System spares users' time by giving them their preferred best results and builds the deal and benefit of the business.

The sparsity of the rating matrix, cold start-up means a situation where a recommender does not have enough information about a user or an item to make relevant predictions. This is one of the serious issues that lessen the performance of recommendation system. The profile of such new user or item will be empty since he has not rated any item: consequently, his taste is not known to the system. Most recommendation algorithms just think about the users while ignoring the relationship between the items, all of what limit the viability of the recommendation algorithms.

III. RELEVANT WORK

In the previous not many decades, researchers have proposed different recommendation algorithms which are generally applied in recommendation systems. For example, in this paper [5], visual recommendation is used because the visual appearance of products has a strong impact on consumer's decisions. With CNN based image processing, the feature maps of different styles are generated. Style features play a vital role in the visual recommendation as a user's decision depends largely on whether the product fits his/her style. However, the representation of the conventional image features fails in capturing the styles of a product. To bridge this gap, style feature modeling which is highly relevant with user preference, into the visual recommendation model is introduced in this paper. Other than this, they are adding style features into collaborative learning to create awareness to the preferences of users. In this paper, Style-aware Bayesian Personalized Ranking (SBPR) framework is used. In addition to the style features, they employed a style representation which includes correlation between different feature spaces. This is successful in understanding the style preferences of users. Experiments conducted on two challenging datasets named Amazon and Tradesy show effectiveness of proposed method. Their proposed model has the stronger ability to sense various styles of products automatically. Here we are learning visual recommendation for feature extraction in image structure.

E-Commerce is the easiest, most commodious way of organizing business over the internet for business experts and individuals. Managing business over the internet is simply surfing specific website for shopping products online or business related matters. Consumer reviews and product rating are the main parameters that companies used in the e-commerce sites in order to strategize the analysis. These reviews provide a crucial role in users to decide about buying an item or not. Thus, examining consumer feedbacks help shopping companies and manufacturers who can identify specific areas of improvement in their products. It is very difficult to read through each individual review of different items and make a good decision for an individual customer and also it is difficult to identify the important features of a product which cannot be identified by looking at the reviews. This approach is very useful for those customers who target at specific features in a product. For example, a photographer looking for an excellent camera features and can compromise on other features of the mobile. This system [6] helps users like the above mentioned to choose a product based on their specific requirement. The purpose is to help the users in purchasing the desired product and also help the manufacturers to identify the buying experience of their products. This system extracts customer reviews and specification list for the user selected product. The Ajax Google API service is used to search the product in the Flipkart and Amazon website. The alchemy API is used for sentiment analysis in order to determine the polarity of individual features in the review. It uses machine learning algorithms to extract semantic meta-data from text content. The sentiment of the individual feature contained in the review is identified. The system will calculate score for each feature based on its polarity in the extracted review and overall rating of a product is calculated by aggregating individual feature scores. Finally, the system will output the pros and cons of the user selected product in terms of score.

This paper [7] proposes a knowledge recommendation approach that integrates the degree of correlation between knowledge and tasks. A good knowledge recommendation approach should recommend the right knowledge (what to recommend) to the right person (who to recommend) at the right time (when to recommend) in the right way (how to recommend). This paper proposes a correlation-experience-demand (CED) integrated knowledge recommendation approach to solve the above four problems. This approach can effectively shorten the time for designers to acquire knowledge by recommending applicable knowledge to assist designers in completing design tasks with high quality and efficiency. The relationships between tasks, knowledge, and designers are studied in this paper [7]. The correlation between tasks and knowledge and the demand of designers for knowledge are quantized.

In this paper [8], recommendation system will be implemented on e-commerce platforms and is expected to help users and sellers. The results of recommendations provided with the ontology approach not only provide recommendations for specific products, but also provide recommendations on categories that may be of interest to the users. Thus, the recommendations will be more varied and are expected to be more in line with user interests. In computer science, ontology [9] is a formal representation of the knowledge by a set of concepts within a domain and the relationships between those concepts. An ontology represents concepts and relationships in a particular domain of interest. In this study, the Slope One algorithm [10] is used where the input rating is given based on the domain ontology of the product. Domain ontology is used to represent relationships between products. Thus, the product recommendations are expected to be in accordance with the user's interest. So that product sales are

right on target and users get products that suit their needs. The Slope One Algorithm performs calculations based on a linear relationship of preference or weight values for each item compared.

This paper [11] presents a method to consider the diverse needs with varying level of competence. This paper presents the case study performed on the recommender system implementation in college campus which will result a recommendation in placement of students (employee) to companies (employer) as per their requirements in shortest possible time. This paper presents a model to generate recommendation based on marks of student. It discovers best solutions which would have otherwise remained hidden. The steps involved in applying algorithm to find results are: Preprocessing, Extraction, Filtering, Clustering, Identification of Users, Content Based filtering in Recommendation system. Basically this is used for Categorizing students based on their credentials. Using these soft computing techniques, the student can be referred to the job profile which is not used as reference for the placement otherwise.

In this work, HARSAM [12] - a new hybrid recommendation Algorithm is designed, which is based on combining the ability of deep neural network model to extract deep features from complex data with that of using self-attention mechanism to extract internal relations between data. This algorithm measures the user preference of an item based on two factors: One is the similarity between the user and the items to be recommended, and the other is the similarity between the items to be predicted and items the user prefers. Our main idea is that, to improve the accuracy of recommendation, it is necessary to consider the relationships between items as well as their mutability, because of external influences. For example, during the World Cup, users will generally care about football-related goods. Taking this into consideration, therefore, in this paper, user interaction data at different time intervals are modeled in the way of combining deep neural networks with attention mechanisms. Therefore, the latent preferences of users and the latent representation of items are also learned. In the meantime, considering the problem of poor computability between items, the SDAE model is used to learn latent representations of items from the rating data, combined with latent representations extracted from item features, to collectively describe an item. SDAE, the Stacked Denoising AutoEncoder [12], is an improved AutoEncoder (AE) which is a simple three-layer neural network structure with an input layer, a hidden layer, and an output layer. SDAE, a combination of deep learning model and DAE, not only retains the advantages of the DAE model, but also has the ability of extracting deep features of data. Deep learning is good at extracting the deep features of data. In real life, user preferences are not fixed, for example, when users focus on some certain items, they will inevitably ignore the others. Due to this, in this work, HARSAM model is proposed, which uses the self-attention to model the internal relationships among items in each period. HARSAM model consists of two parts: the left part and the right part. The left part is used to learn user's latent preference representation Su, and the right part is response for extracting the representation of item features Sv, with a fully connected neural network. There are four stages of learning the representation of user's latent preferences, including embedding the item data, extracting item features, modeling with self-attention, and learning the representation of user's latent preferences.

IV. PRELIMINARIES

This section explains the methods required in our work. In our work we are taking into consideration the amazon review dataset for Clothes, shoes and jewelleries and Beauty products. We are considering the reviews and ratings given by the user to different products as well as his/her reviews about his/her experience with the product(s).

A. Denoising Auto-encoder

Denoising Auto-encoder is widely used in Neural Network. Denoising autoencoders are an extension of the basic autoencoder, and represent a stochastic version of it. Denoising autoencoders attempt to address identity-function risk by randomly corrupting input (i.e. introducing noise) that the autoencoder must then reconstruct, or denoise. AE is a basic three-layer neural network structure, and is made out of an input layer, a hidden layer, and an output layer. The training purpose of AE model is to make input and output as similar as possible. What's more, the output of hidden layer after training is the latent representation of input.

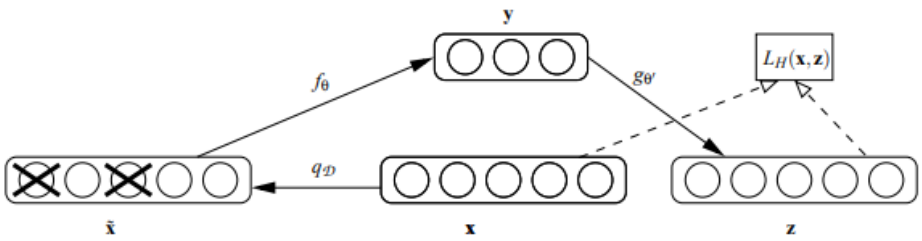


Fig. 1The denoising autoencoder architecture

A common method of improving robustness of the model is to add noise into the input data. As in Denoising AutoEncoder (DAE), which is different from AE, noise is added into the input before training. Obviously, DAE must learn to denoise the data and obtain original input during training. Briefly, the process of training the model is also a process of learning how to eliminate noise. Fig. 1 shows the denoising autoencoder architecture **[13]**. An example x is stochastically corrupted (via q_D) to x˜. The autoencoder then maps it to y (via encoder f_θ) and attempts to reconstruct x via decoder $g_{\theta'}$, producing reconstruction z. Reconstruction error is measured by loss$L_H(X,Z)$.

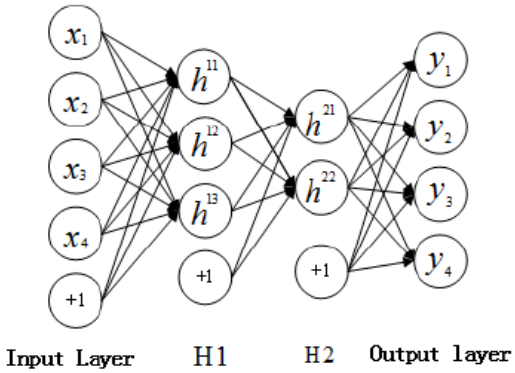


Fig. 2 Stacked Denoising Autoencoder

Past research recommends that deep learning is good at extracting the deep features of data. SDAE, the Stacked Denoising AutoEncoder [14], is an improved AutoEncoder [15] (AE). A stacked denoising autoencoder is simply many denoising autoencoders strung together. SDAE, a combination of deep learning model and DAE, not only retains the advantages of the DAE model, but also has the ability of extracting deep features of data.

$$h^1 = f(W^1 X^+ + b^1)$$
$$h^2 = f(W^2 h^1 + b^2)$$
$$X^\wedge = f(W^3 h^2 + b^3)$$

Above formulas are for learning model of SDAE. In above formulas, X^+ is the noise-incremented representation of the input data X. h^1, h^2 and X^ denote the output of Hidden Layers h^1, h^2 and Output Layer, respectively. f (.) is the activation function, in which, W and b are weight matrix and bias used in the neural network.

B. Factorization

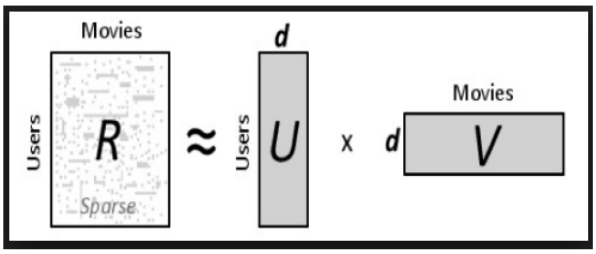


Fig. 3 Matrix Factorization

Matrix factorization [16], [21] is the collaborative based filtering method where matrix m*n is decomposed into m*k and k*n. It is fundamentally utilized for calculation of complex matrix operation since this method performs on the decomposed matrix rather than the original matrix. Division of matrix is such that if we multiply factorized matrix we will get original matrix as appeared in Fig. 3. Matrix Factorization algorithms work by decomposing the user-item interaction matrix into the product of two lower dimensionality rectangular matrices. There are various applications for matrix factorization such as Dimensionality reduction.

C. Feature Extraction - Linear Discriminant Analysis

Feature extraction [17], [22] is for making a new, smaller set of features that stills captures the greater part of the helpful data. Feature Extraction is used to reduce the number of features in a dataset by creating new features from the existing ones and it discards the original features. This new reduced set of features is able to summarize most of the information contained in the original set of features. As the dimensionality increases the computational cost also increases, usually exponentially. To overcome this problem, it is necessary to find a way to reduce the number of features in consideration. The process of feature extraction can be done by several methods, one of which is Linear Discriminant Analysis [18]. Linear discriminant analysis is also called latent Dirichlet allocation. Linear discriminant analysis (LDA) is a supervised method that can only be used with labeled data. The weakness of LDA is that LDA requires labeled data, which makes it more situational. LDA is better than PCA in recognition rate(accuracy) but in term of time taken, PCA is better than LDA [22].

D. Feature Classification

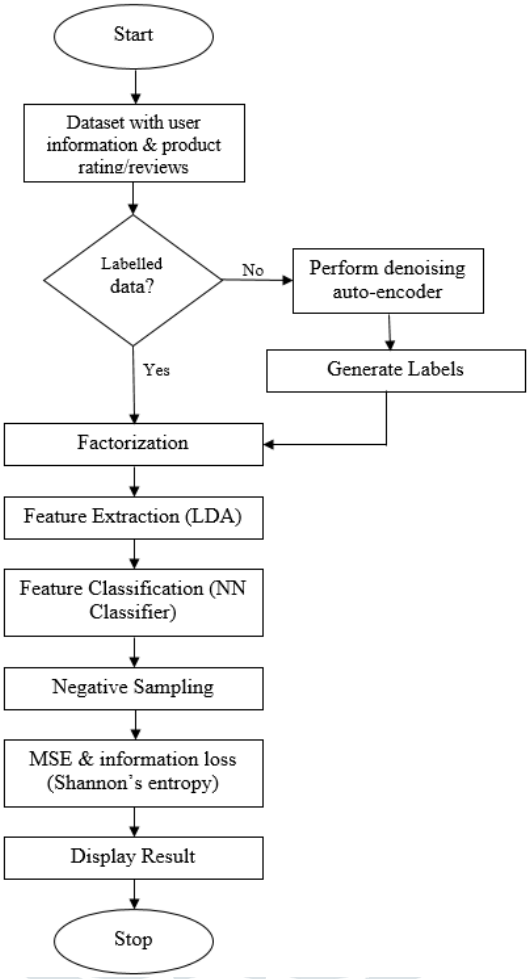
Recommendation systems apply machine learning and data mining feature classification [19] techniques for filtering unseen information and can predict whether a user would like a given resource. Machine learning classifiers can be used for recommendation by training them on content information. Classification involves predicting which class an item belongs to. Some classifiers are binary, resulting in a yes/no decision. Others are multi-class, able to categorize an item into one of several categories. There are different types of classification algorithms [23]: Logistic Regression, Decision Tree Algorithm, Random Forest Algorithm, Naive Bayes Classifier, k-Nearest Neighbor (KNN), Artificial Neural Networks and Deep Neural Networks.

E. Negative Sampling

The key challenge in learning from implicit feedback lies in the natural scarcity of negative signal, known as one-class problem. To address this issue, negative sampling [20] has been widely adopted, where the common approach is to uniformly sample negative instances from the missing data (i.e., the unobserved interactions). Preferences normally have two measurements: one is positive and the other is negative/neutral (essentially what user hates or has no uncommon enthusiasm for). The two are similarly significant if your point is to prescribe something to the user which he/she may like. The explanation behind thinking about negative/ neutral preferences while recommending is that they give you "negative" data. You would need to improve the recommendation of positive preferences of users and avoid recommending negative/neutral preferences of users. As a result, negative sampling increases computational speed and decreases the number of training examples.

V. PROPOSED WORK

In this section, proposed work flow is discussed in detail.



Proposed algorithm works on input dataset with user information and product ratings and reviews. Then if unlabeled data is given then denoising auto encoder is used to generate labels on unlabeled data. During this step features(labels) are generated from user information and product ratings & reviews. After generating labels factorization is performed using stated formula. If labelled data is given, then directly we are performing factorization. During this step matrix are generated. Cold start-up problem can be solved using factorization as we are generating features here. After that, feature extraction is performed using Linear Discriminant Analysis. Dimension Reduction is done during this step. When we have high dimensional data to work with, this technique is very useful. Here we are working with labelled data, so we are using Linear Discriminant Analysis (LDA) technique which is supervised learning technique. It will benefit in accuracy improvement, overfitting risk reduction, speed up in training, improved data visualization and improve in explain ability of our model. After extracting features of data, feature classification is performed with NN classifier. When we are working with complex data deep NN classification techniques works good. This improves prediction results. It is used to solve classification problem and work for better selecting and aggregating the class. By using this, computational speed may down but it results in high accuracy. After that negative sampling is performed to increase computational speed. It will help in decreasing the number of training examples. Then Mean Square Error and Information Loss with Shannon's Entropy is calculated. Mean Square Error is an estimator of a procedure that measures the average of the squares of the errors. It is always near to zero means positive. This will give how accurate your prediction result. So this is one parameter for analysis of work. It will calculate information gain means how trained the data are?

---

algorithm

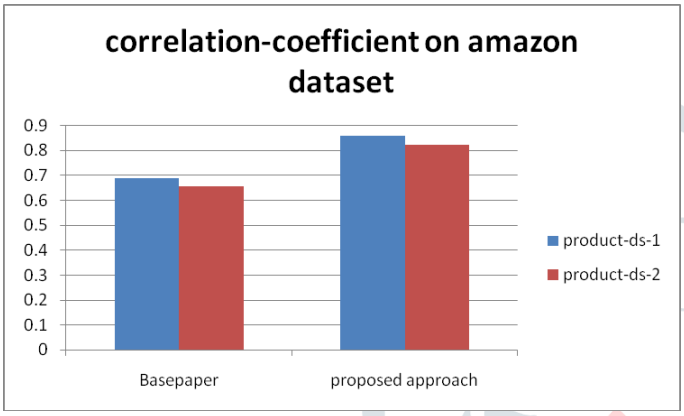
Input: Dataset with user information and product ratings and reviews.
Output: List of items for recommendation.

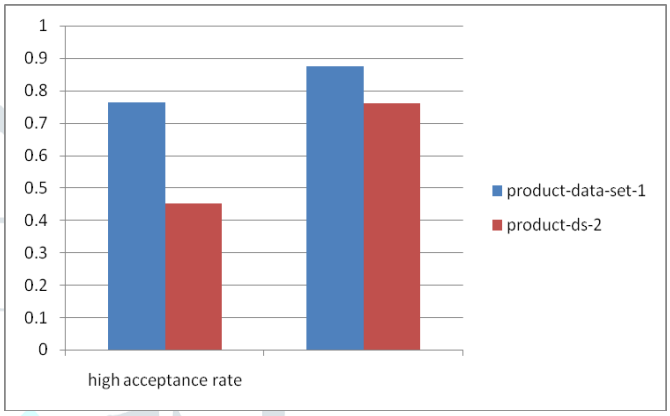
1. **if** unlabeled data **then** perform denoising auto-encoder to generate labels (on unlabeled data)
2. Perform factorization with the given formula.

$$L_H(X,Z) = \sum_{k=1}^{d} [X_k \log Z_k + (1 - X_k) \log(1 - Z_k)]$$

3. Perform feature extraction with LDA.
4. Perform feature classification with NN classifier.
5. Perform negative sampling.
6. Calculate MSE & information loss with Shannon's entropy.
7. Display result.

VI. RESULTS AND DISCUSSION





The above results show that coefficient correlation based various conditions with given customized dataset of Amazon products classification results. We have tested the results with no recommendation ns, and high acceptance rate where the products are likely to buy with good ratings and positive remarks. We also have analyzed the results are 57.8% better than the base paper approach in no recommendation scenario and approx. 30% better in high acceptance rate.

VII. CONCLUSION

The product recommendation system can be as useful as the market survey. As there are many approaches researched with data mining algorithms and text mining techniques with sentiment analysis, we have proposed to use artificial intelligence branches like machine learning and deep learning. The results show remarkable accuracy as compared the earlier one. So we can say that the methods proposed in this paper makes more improvement to the effectiveness of recommendations. In future, this research can be further analyzed for symbolic recommendation and sign language like emojis.

REFERENCES

[1] H.-L. Xu, X. Wu, X.-D. Li, and B. P. Yan, ``Comparison study of Internet recommendation system," J. Softw., vol. 20, no. 2, pp. 350_362, 2009.

[2] R. Chen, Q. Hua, Y.-S. Chang, B. Wei, L. Zhang, and X. Kong, ``A survey of collaborative filtering-based recommender systems: From traditional methods to hybrid methods based on social networks," IEEE Access, vol. 6, pp. 64301_64320, 2018.

[3] Kim, BD. & Kim, "A new recommender system to combine content-based and collaborative filtering systems" SO. J Database Mark Cust Strategy Manag (2001) 8: 244.

[4] W. Haiming, Z. Peng, T. Lu, H. Gu, and N. Gu, ``Hybrid recommendation model based on incremental collaborative filtering and content-based algorithms," in Proc. CSCWD, Apr. 2017, pp. 337_342.

[5] Ming He , Shaozong Zhang , and Qian Meng, "Learning to Style-Aware Bayesian PersonalizedRanking for Visual Recommendation", IEEE 2019

[6] *Vamsee Krishna Kiran M, Archanaa R, Vimalkumar K, Vinodhini R E, "*User specific product recommendation andrating system by performing sentiment analysison product reviews*", ICACCS* -2017

[7] XIYAN YIN , BUYUN SHENG, FEIYU ZHAO, XINGGANG WANG, ZHENG XIAO, AND HUI WANG, "A Correlation-Experience-Demand Based Personalized Knowledge Recommendation Approach", IEEE 2019

[8] Ni Made Satvika Iswari, Wella, Andre Rusli, "Product Recommendation for e-Commerce System based on Ontology", 1st International Conference on Cybernetics and Intelligent System (ICORIS - 2019), IEEE

[9] A. Sieg, B. Mobasher, and R. Burke, "Ontology-based collaborative recommendation," CEUR Workshop Proc., vol. 606, pp. 20–31, 2010.

[10] D. Lemire and A. Maclachlan, "Slope One Predictors for Online Rating-Based Collaborative Filtering," pp. 471–475, 2007.

[11] Ms. Nishigandha Karbhari, Prof. Asmita Deshmukh, Dr. Vinayak D. Shinde, "Recommendation System using Content Filtering - A case study for College Campus Placement", ICECDS-2017

[12] DUNLU PENG, WEIWEI YUAN, AND CONG LIU, "HARSAM: A Hybrid Model for Recommendation Supported by Self-Attention Mechanism", IEEE-2019

[13] Pascal Vincent, Hugo Larochelle, Isabelle Lajoie, Yoshua Bengio, Pierre-Antoine Manzagol, "Stacked Denoising Autoencoders: Learning Useful Representations in a Deep Network with a Local Denoising Criterion", Journal of Machine Learning Research 11 (2010)

[14] Y. Suzuki and T. Ozaki, ``Stacked denoising autoencoder-based deep collaborative filtering using the change of similarity," in Proc. Int. Conf. Adv. Inf. Netw. Appl. Workshops, Mar. 2017, pp. 498-502.

[15] M. Angshul and J. Anant, ``Cold-start, warm-start and everything in between: An autoencoder based approach to recommendation," in Proc. IJCNN, 2017, pp. 3656-3663.

[16] Kharitonov E. (2011) Empirical Study of Matrix Factorization Methods for Collaborative Filtering. In: Kuznetsov S.O., Mandal D.P., Kundu M.K., Pal S.K. (eds) Pattern Recognition and Machine Intelligence. PReMI 2011. Lecture Notes in Computer Science, vol 6744. Springer, Berlin, Heidelberg

[17] Ervin Hidayat, Fajrian Nur A., Azah Kamilah Muda, Choo Yun Huoy, Sabrina Ahmad, "A comparative Study of Feature Extraction using PCA and LDA for Face Recognition", IEEE 2011

[18] P M Prihatini, I K Suryawan and IN Mandia, "Feature Extraction for document text using Latent Dirichlet Allocation", IJCST-2017

[19] M. Ghazanfar and A. Prügel-Bennett, "Building switching hybrid recommender system using machine learning classifiers and collaborative filtering," IAENG International Journal of Computer Science, vol. 37, no. 3, 2010

[20] Jingtao Ding, Yuhan Quan, Xiangnan He, Yong Li, Depeng Jin, "Reinforced Negative Sampling for Recommendation with Exposure Data", IJCAI-2019

[21] https://medium.com/@paritosh_30025/recommendation-using-matrix-factorization - 5223a8ee1f4

[22] https://towardsdatascience.com/feature-extraction-techniques-d619b56e31be

[23] https://missinglink.ai/guides/neural-network-concepts/classification-neural-networks-neural-network-right-choice/