



HAL
open science

A Survey on Techniques and Methods of Recommender System

Aanal Raval, Komal Borisagar

► **To cite this version:**

Aanal Raval, Komal Borisagar. A Survey on Techniques and Methods of Recommender System. 5th International Conference on Computational Intelligence in Data Science (ICCIDS), Mar 2022, Virtual, India. pp.97-114, 10.1007/978-3-031-16364-7_8. hal-04381289

HAL Id: hal-04381289

<https://inria.hal.science/hal-04381289v1>

Submitted on 9 Jan 2024

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.



Distributed under a Creative Commons Attribution 4.0 International License



This document is the original author manuscript of a paper submitted to an IFIP conference proceedings or other IFIP publication by Springer Nature. As such, there may be some differences in the official published version of the paper. Such differences, if any, are usually due to reformatting during preparation for publication or minor corrections made by the author(s) during final proofreading of the publication manuscript.

A Survey on Techniques and Methods of Recommender System

Aanal Raval¹[0000-0003-3848-0882], Dr. Komal Borisagar²[0000-0002-1047-6091]

¹Computer Engg. Dept, PhD student GSET,
GTU- Ahmedabad, India
aanalraval@gtu.edu.in

²E.C. Engg. Dept., Associate Professor,
GSET, GTU- Ahmedabad, India
asso_komal_borisagar@gtu.edu.in

Abstract: As prevalence is growing for social media, the value of its content is becoming paramounting. This data can reveal about a person's personal and professional life. The behaviors done on social media either frequent or periodic can comprehend fondness and attentiveness of users on certain matters. But accompanying it, diverse data from multiple sources with high volumes sets its foot in. Here becomes operational the usage of recommender systems. These are capable of providing customized assistance to users based on respective quondam behavior and preferences. Machine Learning and deep learning methods have proven to be a boon in these tasks of predictions with notable accuracy. This paper discusses existing techniques with its fors and againsts, concerns and issues, extrapolating the results and solutions, which in turn can help in better interpretation of current developments to pave a path for pioneering researches.

Keywords: *Machine Learning, Recommender systems, deep learning, reinforcement learning, soft computing techniques.*

I. INTRODUCTION:

It's an arduous task for business firms to recommend products and services according to their respective needs, withstanding the competition. Customized services satisfying individual demands, bearing vast and diverse information, can provide an enriched user experience. Recommender systems gratifies these needs with preeminent results. Different ML techniques can be used to forge flashing presciences of correlation between users and items, introducing tricky data representations, as well as deducing compendious facts regarding demographical, contextual, and geographical data along with those in patterns. This paper sets sight on to unearth different existing AI/ML techniques in recommender system, with reigning state of art methods and pinpointing current issues with the system with its futuristic evolution. The central concept of these system is to obtain a list or set of items according to utility (in terms of user ratings) of items that user hasn't consumed, by defining the utility of particular item to user. The goal is to maximize this function. Predictions of these systems depend on different ML algorithms selected.

II. TRADITIONAL METHODS

Recommender System resolves the prediction issue by filtering and assigning priorities to information according to their relevancy. This paper presents a survey of recommender systems using different available methods.

2.1 Content-Based

Content- based filtering [1] uses similarities in features to make decisions, comparing user interests to product features. The products recommended are the ones which have most of features overlapping. Here, two methods (or can be in combination) can be used. Firstly, two users will have same taste in the future, given they had similar preference in past. There are two types: 1. Model-based systems deploy ML models to understand and predict ratings. Ratings could be customers are regularly given any stock of functions out of which they'll pick out anything they perceive with the foremost. Secondly, the set of regulations can preserve report of the products the individual has determined on in advance than together with those abilities to the customers' data. Users can be asked about the intuition of features they mostly identify with the products. First of all binary value or an arbitrary number is allotted to both user interest and product features. Then needs a formula to identify similarities between products and user interests. This task can be implemented through dot product:

$\sum_{i=1}^d p_{ui}$ (where p_i is the product feature value and u_i user interest value in column i).

This model becomes easily extensible because of low bulk of data. Moreover, unlike other models, this doesn't require to match with data of another users', it nonpareil niche results particular to the current user.

However, this model requires enough domain knowledge from the people attributing product features. So, its accuracy is totally hooked in to that knowledge being precise. Cold start problem is a great issue.

2.2 Collaborative Filter Based

Collaborative filtering applications apply two types of techniques to show fresh products to the users.[2]

The elemental perspective is to show new recommendations to the user one at a time, simultaneously ratings will predict the user's interest within the items. For a given active user $user_a$,

$$Score_a = 1/k_a \sum V_{aj} \in P_a | \hat{v}_{aj} - v_{aj} |$$

where \hat{v}_{aj} of V_{aj} is the predicted value, v_{aj} is its attested value, and k_a specifies the amount of predicted votes. Then score's average is taken. Lower scores are more preferable. Evaluation of a model with how well it is consistent with given criterion is named average absolute deviation scoring.

The second perspective is to assign currently operative users with an ordered list of recommended items. This list is named a ranked list of things. Now evaluating a model and checking its consistency with this approach is exerted and is called ranked scoring. Sum over all items of utility merchandise of product is the expected utility of ranked list. Probability of that item being viewed by the user is given by,

$$R_a \equiv EU(\text{list}) = \sum_{k=0}^n \max(v_{aj} - d_j, 0) P(\text{item}_j \text{ will be viewed})$$

Where number of predicted voted for each user $user_a$ is denoted by k_a , index j is sorting criteria for the votes, consistent with declining values of predicted votes v_{aj} . The first item will be viewed for certain. Probability of items being viewed in the list declines exponentially down.

Collaborative filtering hinges on two presumptions:

- Assumption of user preferences remain constant over a given period of time.
 - The chances that two users will have similar preferences in the future, given they had same taste in past.
- There are two types:
1. **Model-based systems** use ML models to understand in order to estimate ratings. Ratings can be either a real valued number or a binary value. Hence, when we are given with a user and an item, the system estimates the rating that user will allocate to product. A diverse class of algorithms are employed to perform these tasks of predictions, such as neural networks, logistic regression, Bayesian networks or SVMs. The recognized state of art technique in context of model-based recommender systems are matrix factorization methods, that deals with what matrix factorization outputs in due course harmonizes given assortment of latent features.
 2. **Memory-based systems:** Memory-based systems use user ratings to calculate the similarity amongst users or products and generate predictions. As examples of this approach, item-based/user-based top-N recommendations and neighborhood-based CF can be considered. In phrases of consumer desire, it generally expressed via way of means of categories. Explicit Rating, is a fee given via way of means of a consumer to an object on a sliding scale, Implicit Rating, indicates customers desire indirectly, together with web views and clicks of a page, buy records, willingness to a music track, and so on.[3]
 - a) **User Based CF:** Here we have a rating matrix, $n \times m$, where u_i , ($i = 1, \dots, n$) is the user and p_j , ($j=1, \dots, m$) is the item. Now the requirement is to expect the score r_{ij} which depicts if goal person i did now no longer watch/charge an object j . The core idea is the calculation of the similarities among goal person i and all different users, then pick out the pinnacle X comparable users, and take the weighted average of ratings from these X users with similarities as weights. Some of similarity measures, generally used are Pearson, Jaccard, Spearman rank, Mean Squared and Proximity impact popularity.

$$r_{ij} = R_i + [\sum_k \text{similarities}(u_i, u_k)(r_{kj} - R_k) / \text{number of ratings}]$$
 - b) **Item Based:**[3] It can be said that two products are identical in case they receive close ratings from a same user. Then, this information can be used to predict a goal user associated with any product by taking a weighted average of ratings for most X similar products from this user. One key advantage of Item-based CF is unlike the preferences of human beings, firmness of ratings on a specific product will not change significantly overtime.

Matrix Factorization starts with SVD, creating three matrices, $n \times r$ user feature, $m \times r$ product feature, and $r \times r$ Σ -diagonal matrix accommodating the singular value of original matrix. Now to reduce matrix Σ to k dimensions, The selection of k should be done such that A is apt of grasping the most of variance within the original matrix R , so that A is the approximation of R , $A \approx R$. Error between A and R has to be minimized. When matrix R is dense, there are chances of factorization of U and V analytically. But in case it's sparse, missing values and its factors are to be found. So instead of using SVD, U and V directly can be directly found with the goal that when U and V are multiplied back together, the result matrix is the nearest approximation of R (R') instead of a sparse matrix. This numerical approximation is generally achieved using Non-Negative

Matrix Factorization. The anticipated score for particular person and object, object i is mentioned as a vector q_i , and person u is mentioned as a vector p_u such that the dot Product of those vectors is the expected score for consumer u on object i . This value is presented in the matrix R' at row u and column i . [3]

$$\text{Predicted ratings: } r'_{ui} = p_u^T q_i$$

To find optimal q_i and p_u , a loss function is defined to minimize the cost of errors. r_{ui} is the genuine ratings from authentic user-item matrix. Optimization procedure is to find the optimal matrix P composed via way of means of vector p_u and matrix Q composed via way of means of vector q_i so as to reduce the sum rectangular blunders between predicted ratings r'_{ui} and the true ratings r_{ui} . Also, L2 regularization has been delivered to save you overfitting of consumer and object vectors. [3] Its additionally a popular method to feature a bias term which usually has 3 major components: average rating of item i minus μ (noted as b_u), average rating of all items μ , average rating given by user u minus u (noted as b_i).

$$\text{Min}_{q,p} \sum (r_{ui} - p_u^T q_i)^2 + \lambda (\|p_u\|^2 + \|q_i\|^2)$$

Optimization of non-negative matrix factorization includes Alternative Least Square. Due to loss function being convex, global minimum can't be found, but can reach to an approximation by finding local minimum. Alternative Least Square is to confine user factor matrix constant, adjust item factor matrix, taking derivatives of loss function and initializing it equal to 0, and then moving on, setting item factor matrix constant along with adjusting user factor matrix. Repeat the process, until convergence, by switching and adjusting matrices back and forth [3].

2.3 Demographic-based Recommender System:

This technique categorizes the users taking attributes into account and makes predictions to recommend considering demographic classes [5]. Appropriate market analysis in the peculiar region escorted with a short survey is a must to accumulate data for creating categories. Demographic strategies form "people-to-people" correlations for example collaborative ones, however use different kinds of data. For this technique, a history of person scores like what during collaborative and content is not needed.

2.4 Utility-based Recommender System:

Utility-based recommender device makes recommendations primarily based on calculations of usage of every item for the person. The significant trouble for this sort of device is a way to create an application for person users. In application primarily based totally device, each enterprise can have a one-of-a-kind approach for arriving at a person particular application feature and making use of it to the items below consideration. The principal benefit of the usage of a application primarily based totally recommender device is that it could thing non-product attributes, inclusive of product availability and supplier reliability, into the application calculation. This makes it convenient to test the actual-time stock of the item and proclaim the same to the person [5].

2.5 Knowledge-based Recommender System:

Knowledge-based advice works on useful understanding: they have got understanding approximately how a specific object meets a specific person's want, and may consequently purpose approximately the connection between a want and a piece of likely advice [5].

2.6 Hybrid Recommender System [5]:

It mingles the power of more than two Recommender machine and along with it eradicates any weak point which exist whilst simplest one recommender machine is used. Variety of methods wherein the structures may be combined, such as:

- 1 Weighted Hybrid Recommender: The rating of an endorsed object is computed from the outcomes of all that need advice strategies present inside the system.
- 2 Switching Hybrid Recommender: witches between the recommendation techniques when one doesn't work.
- 3 Mixed Hybrid Recommender: It is used where a large number of recommendations are to be made. Here pointers from multiple approach are supplied together, so the person can pick from a huge variety of recommended items.

III. REVIEW WORK

Traditional systems were not competent to unstructured data and to preserve robustness [38]. In recent years, many techniques of computer vision, reinforcement learning, and many more are rapidly deployed for recommendation systems that yield better results than traditional ones.

3.1 Based on Deep Learning

The first application of deep learning was seen in [39] where restricted Boltzmann machines (RBM) were used along with collaborative filtering [38]. Deep learning models that have been used are with two-layers cramped by Boltzmann machine (RBM) to search for the ordinal property of ratings [22] and an extension of mentioned work by the parameterization options [23]. Fact that recommender system includes all data like text, image, videos etc. motivated use of deep learning in recommendation systems [24]. MLP (multi-layer perceptron) [27] models both linear and nonlinear relationships with machine factorization for feature engineering [25]. Neural collaborative filtering can prototype the non-linear relationship between users and products in co-occurrence with matrix factorization to model the linear relationship [26]. With the purpose of learning non-linear latent representations of users or products, AutoRec combines an autoencoder and matrix factorization [28]. AutoSVD++ is a hybrid for generation of item feature representations from item content method that hybrids a matrix factorization and contractive autoencoder [29]. Long term and short term user preferences are combined, along with user preference drift is taken to consideration and are finely tuned by a hierarchical attention network [30]. [31] applied perturbations on the user-item embedding with notion of Bayesian personalized ranking as an adversarial regularizer. User-item preferences are learnt from interactions, knowledge graphs [32], tags and images [33]. Apart from these, CNN based models like DeepConn, ConvMF, RNN, LSTM, GNN and GAN have also been applied [7]. Using computer vision, popular models are VGG-16, VGG-19, Inceptionv3, Xception and resnet-50 [8]. Another common example can be seen of YouTube, Yahoo, Twitter and eBay prefer deep neural networks (DNNs), while Spotify choose convolutional neural networks (CNNs) [34].

3.2 Based on Transfer Learning:

Transfer learning is a system mastering technique in which a version evolved for a project is reused considering it as the start line for a version on a 2d project. Common examples of transfer learning in deep learning [6]. To unsheathe knowledge from a number of data sources to help the task of learning for the target domain. Here are its classifications: (1) Inductive transfer learning. The goal task differs from the initiating task. When data with labeling is present in the goal area and inductive transfer learning is the same as multi task learning, besides if there is no labeled data, it is known as self-taught learning [7]. (2) Transductive transfer learning. The initiating and goal actions are the same, however, their domain names are different. Transductive transfer learning is commonly used for area adjustment. For this transfer learning technique, the mismatch among the supply area and the goal area may be as a result of the characteristics of variety of feature spaces, or the various marginal distribution among feature spaces [7]. (3) Unsupervised transfer learning. The situation is similar to inductive transfer learning, but the goal duties are unsupervised learning tasks. This contain no labelled data either for source or for target domain [7]. (4) Instance transfer: For target duties, the Reutilization of source domain knowledge is usually an ideal scenario. In maximum cases, the supply area records cannot be reused straightforwardly as it is. Rather, there are positive occurrences from the supply area that may be reused alongside goal records to enhance results. A good example of this is a modification to Adaboost [8]. (5) Feature-representation transfer: This method focuses on minimizing domain divergence and reduction of error rates by spotting good feature representations which can be in turn used from the source to target domains. Depending upon the accessibility of labeled data, feature-representation-based transfers may apply supervised or unsupervised methods [8]. (6) Parameter Transfer: models for associated duties share prior distribution of hyperparameters or some parameters. Differing multitask learning, where both the origin and goal responsibilities used to learn simultaneously, for transfer learning, weightage to the loss of the target domain is added to improve overall performance is preferred [8]. (7) Relational Knowledge Transfer: This attempts to handle the data which is dependent and not identically distributed. In different words, statistics, in which every statistics factor has a relation with different statistics points; for instance, social community statistics makes use of relational-knowledge-transfer techniques.

Cross Domain Recommender System using transfer learning is capable of handling data sparsity problem, providing a rich, personalized and diverse recommendations.

- a. In CDRS (with side information) [7], it is supposed that a part of secondary information such as social information, user-generated information, or product attributes entities is present. Collective matrix factorization (CMF) works on eventualities where a user-object score matrix and an object-characteristic (attribute) matrix for the identical organization of objects are available, thereby factorizing those matrices through parameters of object sharing because the objects are identical. Tag-informed collaborative filtering (TagiCoFi) [9][10][11] is an approach where a score of a user-object matrix and user-tag matrix for similar arrangements of customers are used. User similarities restored from tags that are shared are utilized for helping the ranking matrix's factorization. Tag cross-domain CF (TagCDCF) extends TagiCoFi to two area eventualities each having facts from those matrixes. By concurrently integrating intra-area and inter-area correlations to matrix factorization, TagCDCF tweaks recommenders' overall functioning with inside the goal area.

- b. CDRS with non-overlapping entities manage two domain with non-overlapping entities transfer at group level. Clustering of users and items into groups are done and sharing of knowledge is through group-level rating patterns [7].
- c. CDRS with fully or partially overlapping entities[12] [13] [14] [7] suppose that entities among two domains source and goal overlap and are connected with constraints. This factorizes two matrices of user and item domain with the help of sharing a part of parameters after factorizing. Collective Transfer Factorization (TCF) has evolved for the application of implicit facts present within the source domain to predict explicit comments i.e. H. of rankings in the goal domain. Cross-domain triadic factorization (CDTF) is used to person -item domain tensor to merge implicit and explicit feedback of each. Due to full overlapping of users, user factor matrix is similar by connecting all existing domains. Cluster-based matrix factorization (CBMF) attempts to push the CDTF onto partially-overlapping entities. Because the correspondence between entities is partially available, a number of approaches have been invented that combine users or items in two areas. Unknown user/item linkings are recognized in applying latent space matching. All these are shallow learning based. The next knowledge transfer between two is achieved by mapping both item and user features in target domain with combined features from both. To differentiate user/item embedding features into different areas, GAN is tried with an extra objective function.[15] [16] [17] A general CDRS framework along with Generative Adversarial Network is proposed to concern on all the three scenarios above [7].

Apart from these, two famous strategies for DL with transfer learning are [8]: 1. Off-the-shelf Pre-trained Models as Feature Extractors: Crux is to use pre-trained model with weighted layers to squeeze out features but weights of the model's layers are not updated during training with new data for the new task. 2. Fine Tuning Off-the-shelf Pre-trained Models: Here top layers of network are cut off and replaced with supervised objective of target task, and the network is fine-tuned with backprop using labels for target domain till validation loss begins to increase. Bottom n layers are freezed when those are not updated during backprop and target task labels are scarce as well as overfitting needs to be avoided or fine-tuned when those are updated during backprop and target labels are in abundance [8].

Here are mentioned some types of Deep transfer learning [8]

1. Domain Adaption: Domain adaption is commonly related to layout where the marginal probabilities among the initial and goal domains varies. There is a built-in drift (or shift) in both source and target domains data distributions that needs refinements to transfer the learning.
2. Domain Confusion: Different sets of features are captured by variety of DL network. This information can be used to recognize domain-invariant features and in turn ameliorate their transferability across domains.
3. Multitask learning: In this model, a variety of tasks are learned at same time without any dissimilitude between the source and targets. Unlike in Transfer learning where learner in the beginning doesn't have idea about target task, here, learner acquires information about multiple tasks instantaneously.
4. One-shot learning: Here inference is made for targeted output dependent on just one or little amount of training examples.
5. Zero-shot learning: Clever judgments are made at time of training phase to discover and use additional information to know unseen data.

[40] Used DL along with transfer learning and dimension reduction to extract features from images, thereby applying these features for user-based collaborative filtering for movie recommendations. [41] Proposed value-based neighborhood model using cosine similarity and Pearson coefficient along with KNN to predict customer loyalty addressing sparsity. [42] introduced a method with zero-shot learning and autoencoders to learn language naturally to learn item-item representations. [43] summarizes a feedback loop for understanding user ratings using their discoveries and blind spots, thereby using bias and fairness for next iteration. [44] uses cross convolution filters consisting of variable embedding, cross convolution filters and rating prediction module to effectively capture nonlinear user-item interactions. [45] Proposes a deep learning model with deep autoencoders to unveil nonlinear, nontrivial, and hidden user-item relations for multicriteria purposes. [46] Employs autoencoders to extract trust relation features and tag information from user-user and user-tag matrices. In turn, these extracted features are deployed for the calculation of similarity values among users for the prediction of unseen items. [47] Proposed a method of embeddings for item and user representations for learning non-linear latent factors. The method used decreasing learning rate along with increasing weight decay by varying the values across epochs for improving efficiency. [48] Compared Collaborative Denoising Auto Encoders (CDAE), Deep Auto Encoders for Collaborative Filtering using Content Information (DAE-CI) and Deep Auto Encoders for Collaborative Filtering (DAE-CF), showing DAE-CF with best results. [49] Represented a model combining Hidden Markov Model with ANN. [50] used deep learning with collaborative filtering. They extracted aspects by word embedding and POS tagging, thereafter generation of ratings through LDA and lexicon based methods.

3.3 Active Learning

The task of active learning is to select the data precisely so that ML algorithms can perform better. The labelling process is always time consuming especially for online systems, active learning can successfully be applied here. Some of the active learning methods are query-by-committee, uncertainty sampling, anticipate model change, variance reduction, expected

reduction in error, and density-weighted methods [7]. Its techniques include rating impact analysis and bootstrapping, decision trees and matrix factorization.

[51] Used Gibbs sampling with Bayesian Nonnegative Matrix Factorization for movie recommendations. [52] Proposed an approach for recommendation where initially users were clustered to different groups, then a representative from each was chosen. Using one-hot representation of selected users, the likeliness of users to rate the item and its variance was calculated. Using these ratings, further estimation was done. [53] Proposed a framework called POLAR++ which used Bayesian neural networks to grab the uncertainties in user preferences for the selection of articles for feedback. For task of article recommendation, attention based CNN was used for similarity between user preference and recommended articles.

3.4 Reinforcement Learning:

Reinforcement Learning (RL) is a form of ML technique that allows an agent to study in an participative surroundings via way of means of trial and blunders the usage of remarks from its personal movements and experiences [18]. This makes use of remunerates and penalties as alerts for negative and positive behavior. In desire to build up an optimal solution, the agent has to explore new states along with maximization its universal reward simultaneously. This is known as Exploration vs Exploitation trade-off. To stabilize both, the great standard approach may also contain quick time period sacrifices. Therefore, the agent is ought to acquire sufficient data to make the best possible verdict for future.

- Markov Decision Process [18]: An MDP is made up of a set of possible actions $A(s)$ in each state, a set of finite environment states S and a real valued reward function $R(s)$ with a transition model denoted by $P(s', s | a)$. In most of cases, real world environments don't have any advance knowledge of environment dynamics.
- Q-learning [18] It is a model-free approach which is commonly applied for creating a self-playing PacMan agent. It better believes in concept of overhauling Q values indicating value of executing action a in state s .

$$Q(s_t, a_t) \leftarrow (1-\alpha) \cdot Q(s_t, a_t) + \alpha (r_t + \gamma \cdot \max_a Q(s, a))$$

$Q(s_t, a_t)$ is the old value, α is learning rate, r_t is reward, γ is discount factor and $\max_a Q(s, a)$ is the estimated value of optimal feature. The term $(r_t + \gamma \cdot \max_a Q(s, a))$ represent learned value.

- SARSA [18] is an on-policy technique which grasps the value based on its existing action a extracted from its existing policy.
- Deep Q-Networks (DQNs) [18] uses Neural Networks to predict Q-values. But only can handle discrete, low-dimensional action spaces.
- Deep Deterministic Policy Gradient (DDPG) [18] is a off-policy, model-free, actor-critic algorithm that efficiently solves this problem by comprehending policies in high dimensional with continuous action spaces. This model has proven to be state of art on music recommendations. Here the action of the DDPG learner contains a particular song selected from a huge pool. Following this action, by representing every tune with the help of a set of continuous features expanding the action space from discrete to continuous, it became possible and easy to scale up the number of candidate songs it can accommodate, while maintaining its accuracy and diversity[19].
- A multi-armed bandit is a complex slot device in which in place of 1, there are numerous levers present, which a gambler can select, with every lever giving a unique return or reward. The probability distribution for the return associated with each lever is different and is completely unknown to the gambler [20]. Solutions to this problem includes action value function, regret, greedy approach, epsilon greedy approach, softmax exploration, decayed epsilon greedy, upper confidence bound and regret comparison [20].

[54] Proposed a multi-agent reinforcement learning framework, Temporary Interest Aware Recommendation (TIARec) which can grasp and differentiate between normal preferences and seasonal ones. [55] Partitioned a graph to three sub-graphs user graph, item graph and user-item mapping graph. User and item graphs are pre-trained for user-item interaction graph with the help of the proposed model Attentive Multi-relational Graph Neural Network (RAM-GNN) learning relations through attention layer. To sample top-k similar neighbors from the graph, Reinforced Neighbor Sampler (RNS) was employed to discover the optimal filtering threshold. For recommendation, an aggregation-based GNN model is initiated with user-item embeddings. [56] Used DQN based technique to recommend item list. [57] Proposed deep Q learning with double exploration-based framework- DEN-DQL to get offline data trained for news recommendations. [58] uses RL based dynamic algorithm for user feedback. To maximize click-through rate(CTR) and cumulative estimated reward(CMR), Q learning was used. And finally, to provide top-k online context-aware recommendations, frameworks named Cross Feature Bandits(CFB) and Sequential Feature Bandits(SFB) were proposed. [59] Introduces deep reinforcement learning based deep hierarchical category-based recommender system (DHCRS) with two deep Q- networks, one for selection of item-category relationship and another for selection of item to recommend thus addressing problem space with large action. [60] Introduces reinforcement learning based recommender systems in mobile edge computing to improve throughput. [61] Utilizes Markov decision processes (MDP) with RNN for cold start and warm start scenarios.

3.5 Soft computation techniques: Fuzzy system and evolutionary algorithms:

Fuzzy Logic: In content based systems, fuzzy logic is applied for profiling the customers and matching of relevant items. In memory based, it is utilized for profile uncertainties in customer tastes. Besides, fuzzy sets can express precariousness in item features, lacking item descriptions, subjective user feedback on those items and for item portrayals [7]. These systems are capable of measuring user interest consistency and it's imprecision from given vast data [7].

Evolutionary Algorithms: The use of evolutionary algorithms can be seen in problem of multi objective optimization, considering multiple performance like accuracy, diversity, etc, to combine outputs of multiple recommender system. These systems are also applied to create user-item profiles to find core users and for handling ratings by combining a multi objective optimization problem to a single objective problem [7]. Ratings also include Pareto optimal recommendations or tasking multiple criteria as constraints. These algorithms also play a role in recommender’s security.

[62] Made use of soft rough sets along with context-aware recommender systems for the task of video recommendations. [63] Proposed rough C-means type methods for recommendations of e-commerce sites as well as video streaming services. [64] Attempted to solve the issue of missing ratings by three-way classification using game-theoretic N-soft sets.

3.6 NLP:

NLP can be used in recommender systems for topic modelling, in ratings, item description from text, identifying user generated tags, for user reviews, summary generation and extracting many more relevant features. Relevant techniques in NLP for above tasks include vector models (tf-idf, BM25), LDA, embeddings (Glove, Word2vec, FastText), semantic text similarity, summarization, encoders (BERT) [8] and many more [21]. Recommendations with features like item experience and sentiment of users to provide pertinent items according to user query have been evolved [7]. The neural embedding algorithm in order to deduce product similarity correlations, and multi-level item organization for application on personalized ranking, has been associated with a CF framework [7].

[65] Used CNN, VGG-16 architecture along with NLP techniques for product recommendations. [66] Makes use of Neural Network Language Models for semantic analysis of documents. [67] Developed a recommender system to recommend papers and articles that are related to particular topic of interest. [68] Employed Named Entity Recognition based extractor along with rule based and grammar based extractors to obtain the logical relations. Based on these data, similarity scores were calculated and then considered for recommendation.

IV. Summary:

Paper	Dataset	Methodology	Task
[39]	Yelp, Amazon, Beer	DeepCoNN	User-item recommendation
[40]	MovieLens	Transfer learning with collaborative filtering	movie recommendation
[41]	Amazon movies review	Value based neighborhood model	Customer loyalty
[42]	(sample created)	Zero shot learning with auto-encoders	Learn item-item representations
[43]	MovieLens	Feedback loop setting	Study behavior of iterative recommender system
[45]	Yahoo movies, TripAdvisor	Auto encoder based model	Learn non-trivial and non-linear user item relations

[47]	MovieLens, FilmTrust, Book-Crossing	Embeddings	Address gap in collaborative filtering
[48]	(created from MOOC)	DAE-CF	E-learning recommendations
[49]	MovieLens	Hidden Markov model with ANN	Pattern recognition for recommendations
[50]	Amazon , Yelp	Weighted aspect-based opinion mining using MCNN model.	Product recommendations
[51]	MovieLens	Bayesian Nonnegative Matrix Factorization	movie recommendation
[52]	movielens-Imdb	Active learning with clustering	To address cold start
[53]	Dataset from AMiner, Patent Full-Text Databases of the United States Patent, Related-Article Recommendation Dataset (RARD)	Bayesian neural network with attention based CNN	Article recommendation.
[56]	---	DQN based technique	Product recommendation
[57]	---	DEN-DQL	News recommendation
[58]	News article data	RL based dynamic algorithm	News recommendation
[59]	---	deep reinforcement learning based deep hierarchical category-based recommender system (DHCRS)	Action- space issue
[60]	---	reinforcement learning based	Mobile edge computing

[61]	---	Markov decision processes (MDP) with RNN	Address Cold start problem
[62]	LDOS-CoMoDa	Context-Aware Recommender System	Context relevancy in video
[63]	NEEDS-SCAN/PANEL, d MovieLens	Rough C-means based methods	A recommendation in e-commerce.
[64]	LDOS-CoMoDa, InCarMusic	N-soft sets based method	Context aware recommendations
[65]	Amazon Apparel database	NLP with CNN based methods	Product recommendation
[66]	Patent documents (sample)	Neural Network Language Models	Document / article recommendations
[67]	---	Weighted Average and text similarity	Document / article recommendations
[70]	MNIST, Fashion-MNIST	Bandit algorithms	Learning in non-stationary environments
[71]	MovieLens-100K	Auto-encoders based methods	Movie recommendations
[72]	Microsoft Academic Graph	Metric for exposure bias with text embeddings	Research Papers Recommendations

V. Observations and Research Gap:

The primary issue faced by the recommender system is requirement of bulky data. A good system needs data from a catalog or from other form for which then it must capture and analyze user behavioral events. Thus it can become a chicken and egg problem [35]. Another issues include changing trends and user preferences or sometimes includes a large number of variables. Next Cold Start problem crop up when new user or product enters the system. Another problems includes synonymy (n item is represented with more than two different names or entries having similar intents), Shilling attacks (if a baleful user or competitor sets foot in the system and starts giving fictitious ratings on some items to astray information of item popularity), overspecialization (Meagre content analysis leading to CB recommendation not to suggest novel items.), grey sheep (in run-of-mill CF systems, opinions of a user do not match with any group), sparsity, scalability, latency problem (when fresh items are added to dataset over and over again), unavailability of benchmark dataset and context awareness (users' short and long term history with geographic locations.) [36].

Issue of Data Sparsity exists when collaborative filtering based methods are employed. This problem to some extent can be solved if preferences relations are used instead of absolute ratings [69]. Time awareness regarding popularity span [69], or seasonal products are also important. Besides, online data are a vast pool including structured as well as unstructured data. Personalization of search engine for relevancy is also an important aspect. Sometimes decisions along with recommendations come to roles where the system is like Resource Allocation Problem. [69]

VI. Future Directions and Conclusion:

Traditional recommender system supposes that user preferences are static over time, thus their history documentation are equally weighted. But as these changes over time based on individual experiences personal preferences, or popularity-driven influences, this behavior of big data is known as concept drift. To resolve this issue, instanced decay and time window was introduced [7], which determines the weights of data occurrences alongside the timeline and compares to that old data which weighs less. Apart from penalizing the old data, some techniques were introduced that operated on dynamic matrix factorization, for which time is observed as one more dimension of the data. Still, these methods cannot regulate the increase in bias resulting due to the directions of change. Another topic includes long-tail, where items are unpopular, are rarely noticed by the users. With the development of these systems, the development of privacy conserving and secure systems are in demand. While some systems are too complex to understand or lack clarity in explanation. This challenging limitation can be solved using integrating visualization to recommender systems.

Current advancements in recommender system aims at assisting decision making support system that includes vast information involving items metadata, user profiles, user reviews, images and social networks. This paper reviews different available techniques in recommender system, issues faced in it. Apart from these, it also identifies the gap in current existing systems and paves path for future directions.

References:

- [1] What is content-based Filtering?, Towards Data Science [online] Available At <https://www.edpresso.io/edpresso/what-is-content-based-filtering>
- [2] Collaborative Filtering, Science Direct [online] Available At: <https://www.sciencedirect.com/topics/computer-science/collaborative-filtering>
- [3] Introduction to collaborative Filtering, Towards Data Science, [online] Available At: <https://towardsdatascience.com/intro-to-recommender-system-collaborative-filtering-64a238194a26>
- [4] A Simple Introduction to Collaborative Filtering, built-in beta, [online] Available At: <https://builtin.com/data-science/collaborative-filtering-recommender-system>
- [5] Classifying Different Types of Recommender Systems, bluepi, [online] Available At: <https://www.bluepiit.com/blog/classifying-recommender-systems/#:~:text=There%20are%20majorly%20six%20types,system%20and%20Hybrid%20recommender%20system.>
- [6] A Gentle Introduction to Transfer Learning for Deep Learning, Machine Learning Mastery , [online] Available At: <https://machinelearningmastery.com/transfer-learning-for-deep-learning/#:~:text=Transfer%20learning%20is%20a%20machine,model%20on%20a%20second%20ask.&text=Common%20examples%20of%20transfer%20learning,your%20own%20predictive%20modeling%20problems.>
- [7] Qian Zhang , Jie Lu, Yaochu Jin , “Artificial intelligence in recommender systems”, 1 November 2020, Complex & Intelligent Systems (2021) 7:439–457
- [8] A comprehensive hands on Guide to Transfer Learning with Real World Applications in Deep Learning, towards data science, [online] Available At: <https://towardsdatascience.com/a-comprehensive-hands-on-guide-to-transfer-learning-with-real-world-applications-in-deep-learning-212bf3b2f27a>

- [9] Qian Zhang; Peng Hao; Jie Lu; Guangquan Zhang , “Cross-domain Recommendation with Semantic Correlation in Tagging Systems”, July, 2019, IEEE
- [10] Zhen Y, Li WJ, Yeung DY TagiCoFi: Tag informed collaborative filtering. In: RecSys’09—Proceedings of the 3rd ACM Conference on Recommender Systems, 2009, pp 69–76
- [11] Hao P, Zhang G, Martinez L, Lu J Regularizing knowledge transfer in recommendation with tag-inferred correlation. IEEE Trans Cybern
- [12] Pan W, Yang Q Transfer learning in heterogeneous collaborative filtering domains. Artif Intell 197:39–55
- [13] Hu L, Cao J, Xu G, Cao L, Gu Z, Zhu C Personalized recommendation via cross-domain triadic factorization. In: Proceedings of the 22nd International Conference on World Wide Web, 2013, pp 595–606
- [14] Mirbakhsh N, Ling CX Improving top-n recommendation for cold-start users via cross-domain information. ACM Trans Knowl Discov Data 9(4):33
- [15] Hu G, Zhang Y, Yang Q (2018) Conet: collaborative cross networks for cross-domain recommendation. In: Proceedings of the 27th ACM International Conference on Information and Knowledge Management, 2018, pp 667–676
- [16] Wang C, Niepert M, Li H (2019) Recsys-dan: discriminative adversarial networks for cross-domain recommender systems. IEEE Trans Neural Networks Learn Syst
- [17] Yuan F, Yao L, Benatallah B (2019) DARec: Deep domain adaptation for cross-domain recommendation via transferring rating patterns. In: Proceedings of the Twenty-Eighth International Joint Conference on Artificial Intelligence, 2019
- [18] Reinforcement Learning 101, towards data science, [online] Available At: <https://towardsdatascience.com/reinforcement-learning-101-e24b50e1d292>
- [19] Recommender System with Reinforcement Learning, towards data science, [online] Available At: <https://towardsdatascience.com/recommendation-system-with-reinforcement-learning-3362cb4422c8>
- [20] Reinforcement Learning Guide: Solving the Multi-Armed Bandit Problem from Scratch in Python, Analytics Vidhya, [online] Available At: <https://www.analyticsvidhya.com/blog/2018/09/reinforcement-multi-armed-bandit-scratch-python/>
- [21] Aanal Raval, Agrima Lohia, “A SURVEY ON TECHNIQUES, METHODS AND APPLICATIONS OF TEXT ANALYSIS”, IJCRT2105349, May 2021.
- [22] Salakhutdinov R, Mnih A, Hinton G Restricted Boltzmann machines for collaborative filtering. In: Proceedings of the 24th International Conference on Machine Learning.
- [23] Truyen TT, Phung DQ, Venkatesh S Ordinal Boltzmann machines for collaborative filtering. In: Proceedings of the 25th Conference on Uncertainty in Artificial Intelligence, 2009, pp548–556
- [24] Zhang S, Yao L (2017) Deep learning based recommender system: a survey and new perspectives. ACM J Comput Cult Herit Artic1(35):1–35
- [25] Cheng et al. HT (2016) Wide and deep learning for recommender systems. arXiv Prepr. pp 1–4
- [26] He X, Liao L, Zhang H, Nie L, Hu X, Chua TS (2017) Neural collaborative filtering. In: Proceedings of the 26th International Conference on World Wide Web, 2017, pp 173–182
- [27] perceptron
- X (2017) DeepFM: a factorization-machine based neural network for CTR prediction. In: International Joint Conference on Artificial Intelligence, 2017, pp1725–1731
- [28] Sedhain S, Menon AK, Sanner S, Xie L AutoRec: autoencoders meet collaborative filtering. In: Proceedings of the 24th International Conference on World Wide Web, 2015, pp 111–112

- [29] Zhang S, Yao L, Xu X (2017) AutoSVD++: an efficient hybrid collaborative filtering model via contractive auto-encoders. In: Proceedings of the 40th International ACM SIGIR Conference on Research and Development in Information Retrieval, 2017, pp 2–5
- [30] Ying H et al. (2018) Sequential recommender system based on hierarchical attention network. In: International Joint Conference on Artificial Intelligence, 2018.
- [31] He X, He Z, Du X, Chua TS (2018) Adversarial personalized ranking for recommendation. In: The 41st International ACM SIGIR Conference on Research & Development in Information Retrieval, 2018, pp 355–364
- [32] Yang D, Guo Z, Wang Z, Jiang J, Xiao Y, Wang W (2018) A knowledge-enhanced deep recommendation framework incorporating GAN-based models. In: 2018 IEEE International Conference on Data Mining, 2018, pp 1368–1373
- [33] Tang J, Du X, He X, Yuan F, Tian Q, Chua T-S (2019) Adversarial training towards robust multimedia recommender system. *IEEE Trans Knowl Data Eng* 32(5):855–867
- [34] Deep Learning Based Recommender System, Sciforce, [online] Available At: <https://medium.com/sciforce/deep-learning-based-recommender-systems-b61a5ddd5456>
- [35] 5 Problems with Recommender System, readwrite, [online] Available At: https://readwrite.com/2009/01/28/5_problems_of_recommender_systems/?_cf_chl_jschl_tk_=pmd_UZyz8bgl.30UoIXhsuXwJqLz2DxSjHu4LMvOrVSuf0-1629638145-0-gqNtZGzNAICjcnBszQh9
- [36] Shah Khusro, Zafar Ali, Irfan Ullah, “Recommender Systems: Issues, Challenges, and Research Opportunities”, Springer
- [37] Silvia Milano, Mariarosaria Taddeo & Luciano Floridi, “Recommender systems and their ethical challenges”, Springer, 2020.
- [38] Baichuan Liu et al, “A survey of recommendation systems based on deep learning”, *Journal of Physics: Conference Series*, ISPECE 2020.
- [39] Lei Z, Vahid N, Philip S. Joint Deep Modeling of Users and Items Using Reviews for Recommendation [C]. Proceedings of the Tenth ACM International Conference on Web Search and Data Mining, 2017. Cambridge United Kingdom. 425-434.
- [40] Aymen Ben Hassen and Sonia Ben Ticha, “Transfer Learning to Extract Features for Personalized User Modeling”, In Proceedings of the 16th International Conference on Web Information Systems and Technologies (WEBIST 2020)
- [41] Abhishek, Srivastava Pradip, Kumar Bala Bipul Kumar, “TRANSFER LEARNING FOR RESOLVING SPARSITY PROBLEM IN RECOMMENDER SYSTEMS: HUMAN VALUES APPROACH”, *JISTEM J. Inf. Syst. Technol. Manag.* 14 (3), Dec 2017
- [42] Tao Wu, Ellie Ka-In Chio, Heng-Tze Cheng, Yu Du, Steffen Rendle, Dima Kuzmin, Ritesh Agarwal, Li Zhang, John Anderson, Sarvjeet Singh, Tushar Chandra, Ed H. Chi, Wen Li, Ankit Kumar, Xiang Ma, Alex Soares, Nitin Jindal, Pei Cao, “Zero-Shot Heterogeneous Transfer Learning from Recommender Systems to Cold-Start Search Retrieval”, arXiv:2008.02930v2 [cs.LG] 19 Aug 2020
- [43] Khenissi S, Boujelbene M, Nasraoui O. “Theoretical Modeling of the Iterative Properties of User Discovery in a Collaborative Filtering Recommender System.” 2020.
- [44] Seungyeon Lee, Dohyun Kim, “Deep learning based recommender system using cross convolutional filters”, Elsevier, Volume 592, May 2022
- [45] Qusai Shambour, “A deep learning based algorithm for multi-criteria recommender systems”, Elsevier, Volume 211, 9 January 2021.

- [46] SajadAhmadiana, MiladAhmadianb, MahdiJalili, “A deep learning based trust- and tag-aware recommender system”, Elsevier, Neurocomputing, November 2021
- [47] KiranR, PradeepKumar, BharatBhasker, “DNNRec: A novel deep learning based hybrid recommender system”, Elsevier, Expert Systems with Applications, Volume 144, 15 April 2020.
- [48] EvertonGomede, Rodolfo Mirandade Barros, Leonardo de SouzaMendes, “Deep auto encoders to adaptive E-learning recommender system”, Elsevier, Computers and Education: Artificial Intelligence, Volume 2, 2021
- [49] ChoukriDjellali, Mehdiadda, “A New Hybrid Deep Learning Model based-Recommender System using Artificial Neural Network and Hidden Markov Model”, Elsevier, Procedia Computer Science, Volume 175, 2020
- [50] AminuDa’u, NaomieSalim, IdrisRabiu, AkramOsman, “Weighted aspect-based opinion mining using deep learning for recommender system”, Elsevier, Expert Systems with Applications, Volume 140, February 2020
- [51] Gonul Ayci, Abdullatif Köksal, M. Melih Mutlu, Burak Suyunu, A. Taylan Cemgil, “Active Learning with Bayesian Nonnegative Matrix Factorization for Recommender Systems”, 2019 27th Signal Processing and Communications Applications Conference (SIU), IEEE.
- [52] Junxin Zhou, Raja Chiky, “Improving the Attribute-Based Active Learning by Clustering the New Items”, IEEE, 2019 IEEE World Congress on Services (SERVICES)
- [53] Zhengxiao Du, Jie Tang, Yuhui Ding, “POLAR++: Active One-Shot Personalized Article Recommendation”, IEEE Transactions on Knowledge and Data Engineering (Volume: 33, Issue: 6, June 1 2021)
- [54] Ziwen Du, Ning Yang, Zhonghua Yu, Philip Yu, “Learning from Atypical Behavior: Temporary Interest Aware Recommendation Based on Reinforcement Learning”, IEEE Transactions on Knowledge and Data Engineering, 2022
- [55] Xiaohan Li, Zhiwei Liu, Stephen Guo, Zheng Liu, Hao Peng, Philip S. Yu, Kannan Achan, “Pre-training Recommender Systems via Reinforced Attentive Multi-relational Graph Neural Network”, IEEE International Conference on Big Data (Big Data), 2021
- [56] Haicheng Chen, “A DQN-based Recommender System for Item-list Recommendation”, IEEE International Conference on Big Data (Big Data), 2021
- [57] Zhanghan Song, Dian Zhang, Xiaochuan Shi, Wei Li, Chao Ma, Libing Wu, “DEN-DQL: Quick Convergent Deep Q-Learning with Double Exploration Networks for News Recommendation”, IEEE, International Joint Conference on Neural Networks (IJCNN), 2021
- [58] Anubha Kabra, Anu Agarwal, “Personalized and Dynamic top-k Recommendation System using Context Aware Deep Reinforcement Learning”, IEEE 45th Annual Computers, Software, and Applications Conference (COMPSAC), 2021
- [59] Mingsheng Fu, Anubha Agrawal, Athirai A. Irissappane, Jie Zhang, Liwei Huang, Hong Qu, “Deep Reinforcement Learning Framework for Category-Based Item Recommendation”, IEEE Transactions on Cybernetics, 2021
- [60] Elnaz Rabieinejad, Shahriar Mohammadi, Mahdi Yadegari, “Provision of a Recommender Model for Blockchain-Based IoT with Deep Reinforcement Learning”, IEEE, 5th International Conference on Internet of Things and Applications (IoT), 2021
- [61] LiweiHuang, MingshengFu, FanLiaHongQu, YangjunLiu, WenyuChen, “A deep reinforcement learning based long-term recommender system”, Elsevier, Knowledge-Based Systems, Volume 213, 2021

- [62] Syed Manzar Abbas, Khubaib Amjad Alam, “Exploiting Relevant Context with Soft-Rough Sets in Context-Aware Video Recommender Systems”, IEEE International Conference on Fuzzy Systems (FUZZ-IEEE), 2019
- [63] Seiki Ubukata, Shu Takahashi, Akira Notsu, Katsuhiko Honda, “Basic Consideration of Collaborative Filtering Based on Rough C-Means Clustering”, IEEE, Joint 11th International Conference on Soft Computing and Intelligent Systems and 21st International Symposium on Advanced Intelligent Systems (SCIS-ISIS), 2020
- [64] Syed Manzar Abbas, Khubaib Amjad Alam, Kwang-Man Ko, “A Three-way Classification with Game-theoretic N-Soft Sets for Handling Missing Ratings in Context-aware Recommender Systems”, IEEE International Conference on Fuzzy Systems (FUZZ-IEEE), 2020
- [65] AkhileshKumarSharma, BhavnaBajpai, RachitAdhvaryu, SutharDhruvi Pankajkumar, PrajapatiParthkumar Gordhanbhai, AtulKumar, “An Efficient Approach of Product Recommendation System using NLP Technique”, ScienceDirect, Materials Today Proceedings, 2021
- [66] AmyTrappey, Charles V.Trappey, AlexHsieh, “An intelligent patent recommender adopting machine learning approach for natural language processing: A case study for smart machinery technology mining”, Elsevier, Technological Forecasting and Social Change, 2021
- [67] Julia A. Sterling, Matthew M. Montemore, “Combining Citation Network Information and Text Similarity for Research Article Recommender Systems”, IEEE Access (Volume: 10), 2021
- [68] Tong Zhang, Mingyi Liu, Chao Ma, Zhiying Tu, Zhongjie Wang, “A Text Mining based Method for Policy Recommendation”, IEEE International Conference on Services Computing (SCC), 2021
- [69] Madhusree Kuanr, Puspanjali Mohapatra, “Recent Challenges in Recommender Systems: A Survey”, Springer, Progress in Advanced Computing and Intelligent Engineering, 2021
- [70] Giuseppe Di Benedetto, Vito Bellini, Giovanni Zappella, “A Linear Bandit for Seasonal Environments”, Amazon Science, Proceedings of the 37 th International Conference on Machine Learning, Vienna, Austria, 2020
- [71] Jinjin Zhao, Lei Wang, Dong Xiang, Brett Johanson, “Collaborative Deep Denoising Autoencoder Framework for Recommendations”, Amazon Science, SIGIR’19, July, Paris, France, 2019.
- [72] Shantanu Gupta, Hao Wang, Zachary Lipton, Yuyang Wang, “Correcting Exposure Bias for Link Recommendation”, Amazon Science, Proceedings of the 38 th International Conference on Machine Learning, PMLR 139, 2021