Cross Domain Item Based Book Recommendation System

By

Saad Khattak

A thesis submitted to the faculty of Electrical Engineering Department, Military College of Signals, National University of Sciences and Technology, Islamabad, Pakistan, in partial fulfillment of the requirements for the degree of MS in Telecommunication (Electrical) Engineering

May 2022

ABSTRACT

Recommendation systems (RCS) are particularly extremely resourceful systems which can anticipate the type for content the user would like to stream, rate, upload, download or subscribe to. It is based on feedback process, where it tracks the type of content a user is rating, streaming, downloading, uploading and subscribing to and based on that, Recommendation system further recommends the user the type of content they would further like to stream, rate, download, upload or subscribe to. Basically Recommendation system keeps up with the appetite of the user by satisfying their needs online. In the modern world all the giant companies for example like Facebook, YouTube, Netflix, Spotify, Instagram, Amazon, Ebay, Ali Baba (companies from Entertainment Industry, Telecommunication industry and E-commerce) are heavily relying on Recommendation systems for their businesses. The Aforementioned Recommendation systems are single domain Recommendation system, meaning the Source domain and the Target domain is the same. For example if a user is listening to a song on YouTube of a particular singer and likes it, the user will be further recommended more videos from that singer. So here our source and target domains are the same, which is YouTube videos. When we talk about Cross domain, here Our Source and Target are two completely different domains. We takes users from Source domain and recommend them things from target domain. For Cross Domain to work, there has to be some sort of link between them.

In this thesis, a Cross domain item based book recommendation system is proposed. The proposed technique is based on initially building a user item rating matrix for movies, then using KNN algorithm to make movie recommendations, then taking recommended movie genres and computing their semantic similarity with the books genres using wpath method. Then those books genres are shortlisted which have semantic similarity score of more than 0.5 with the movie genres. Lastly Multi label Binarizer approach is used to break a book into its genres and make sequences of the books. For final recommendations three things are taken into account, 1) No of times book occurred in a sequence 2) Total rating count of the book 3) Average ratings of the book. This Cross Domain Item based Book Recommendation system is capable of providing better recommendations but also establishes strong relation between source and target domains.

The output results and their comparison with other different techniquesis provided which shows the overall improved results of this approach.

DEDICATION

This thesis is dedicated to

MY Loving FAMILY Specially my Sisters, my FRIENDS who were there for me during my hard times AND lastly my TEACHERS during my stay here at MCS, most notably Lt.Col Hasnat for motivating me and inspiring me to become a Data Scientist.

ACKNOWLEDGEMENTS

I am grateful to God Almighty who has bestowed me with the strength and the passion to accomplish this thesis and I am thankful to Him for His mercy and benevolence. Without his consent I could not have indulg ed myself in this task.

TABLE OF CONTENTS

LIST OF FIGURES

LIST OF TABLES

ACRONYMS

Recommender Systems	RS
Cold Start	CS
Collaborative Filtering	CF
Singular Value Decomposition	SVD
Latent Dirichlet Allocation	LDA
Principle Component Analysis	PCA
Markov Decision Processes	MDP
Probabilistic Matrix Factorization	PMF
Content Based Filtering	CBF
Mean Absolute Error	MAE
Root Mean Square Error	RMSE
Cross Domain Recommendation system	CDRS
Contend Based Filtering	CBF

Chapter 1

INTRODUCTION

The vast amount of information on various items for example gadgets, Movies, Books, Newspaper articles, songs, academic papers etc available online has provided users with a wide range of options in the last decade or so. Most online websites today are severely overburdened with information, necessitating more time and effort on the part of users to locate their desired online information or product. Recommendation systems are being developed to assist users in locating the correct required information in short amount of time from a huge chunk of information available online on the websites. These Recommendation systems simply remove the unnecessary content and provide people the only information that is relevant to them based on their consumption pattern.[1].

Recommendation Engines are a form of intelligent systems that are being used in a variety of Entertainment, Telecommunication, E-commerce industry as well as social media to propose things to consumers such as Movies to watch, Books to read, which clothing to buy, which news article to read and a variety of other items. Giant corporations throughout the world, such as LinkedIn, Hulu, Disney+, Twitter, Instagram, Facebook, YouTube, Amazon, Netflix, Spottily etc are heavily relying on their Recommendation Engines to track users behavior pattern and recommend them different things based on their consumption pattern. The profitability and user's happiness of aforementioned enterprises and more are highly dependent on the effectiveness of their Recommendation Engine [1], [2].

1.1 Problem Statement and Objectives

A lot of Research has been carried out in the single domain recommendations (mentioned above) but not much work has done so far in the field of Cross domain recommendation

systems (especially in Cross domain Book recommendations). In my thesis, I have built a Cross domain Item based Book Recommendation system which takes into account Collaborative filtering technique called Item based Collaborative filter using KNN (K-Nearest Neighbor) to make movie recommendations, followed by computing semantic similarity between recommended movie genres with all the book genres using state of the art technique called Wpath. Only those book genres are shortlisted which have semantic similarity score of 0.5 and above. Multi label Binarizer approach is used to break a book into its genres and make sequences of the book. The Final Recommendation is based on the 1) No of times a book occurred in a sequence 2) Total rating count of the book 3) Average rating of the book. The proposed system not only establishes a strong semantic similarity relations with source and target domain but also provides really low Root mean square error which is an essential evaluation matric for recommendation engine. The proposed system also outputs really high F1 score, Recall and Precision. The KNN algorithm that is used contain the following input parameter i.e Brute force, Cosine similarity and values of N set to 10 (to get top 10 nearest neighbors).

Objectives of thesis work are:

- To build a robust and effective Cross domain recommendation engine for books.
- To Build strong semantic similarity relation with source and target domain.
- To provide low RMSE and High F1, Recall and Precision.

1.2 Contributions

The contributions of my work are summarized as follows,

- To propose an item based Cross domain recommendation
- Build an effective Recommendation system using KNN algorithm at the source domain with low RMSE and high F1 score, Recall and precision.
- Computation of semantic similarity between recommended movie genres and all book genres using Wpath approach to establish strong relation between source and target

domain.

Cross Domain Item based Book Recommendation system:

Cross domain Recommendation engines heavily rely on the link between Source domain and Target domains. In the real environment when a user starts to assign random tags to different items both in source and target domain, this is where, Cross Domain Recommendation Engines fails to perform. So 1 on 1 link or 1 to many links doesn't work in real work environment. So to make Cross Domain Recommendation engines work (in case of Book recommendation systems), there has to be strong semantic similarity between movies and book genres. Also, there is no clear defined approach to compute the quality of the Book recommendations at the output. All Cross Domain Recommendation systems heavily rely on the quality of the link between Source and Target domains.

The proposed Cross Domain Item based Book Recommendation Engine approach not only solves the data sparity issues at the source domain [26], but it also develops a strong relation between source and target domains which can work in real world environment as well. The aforementioned approach also defines a new way of computing the quality of the Book recommendations at the Target domain as well.

The proposed technique is based on initially building a user item rating matrix for movies, then using KNN algorithm to make movie recommendations, then taking recommended movie genres and computing their semantic similarity with the books genres using wpath method. Then those books genres are shortlisted which have semantic similarity score of more than 0.5 with the movie genres. Lastly Multi label Binarizer approach is used to break a book into its genres and make sequences of the books. For final recommendations three things are taken into account, 1) No of times book occurred in a sequence 2) Total rating count of the book 3) Average ratings of the book. This Cross Domain Item based Book Recommendation system is capable of providing better recommendations but also establishes strong relation between source and target domains

Collaborative filtering technique called Item based Collaborative filter using KNN (K-Nearest

Neighbor) is used to make movie recommendations, followed by computing semantic similarity between recommended movie genres with all the book genres using state of the art technique called Wpath. Only those book genres are shortlisted which have semantic similarity score of 0.5 and above. Multi label Binarizer approach is used to break a book into its genres and make sequences of the book. The Final Recommendation is based on the 1) No of times a book occurred in a sequence 2) Total rating count of the book 3) Average rating of the book. The proposed system not only establishes a strong semantic similarity relations with source and target domain but also provides really low Root mean square error which is an essential evaluation matric for recommendation engine. The proposed system also outputs really high F1 score, Recall and Precision. The KNN algorithm that is used contain the following input parameter i.e Brute force, Cosine similarity and value of N set to 10 (to get top 10 nearest neighbors).

Applications: This proposed research improves upon the current research and propose approaches but also improves the relation between source and target domain which can work in real world environment. It also defines a new way to compute the quality of the recommendations at the Target domain. The proposed method can be applied in Telecommunication industry, Entertainment (movies, songs, books, pictures etc), as well as in E-commerce industry.

1.3 Thesis Outline

This thesis is divided into five chapters:

- Chapter 1: The first chapter covers the introduction, problem statements and objectives of this thesis. It covers the work done in this research.
- Chapter 2: The background study and literature review, as well as a brief summary of current procedures and quantitative measurements employed in this thesis report, are presented in the second chapter.

Chapter 3: An improved hybrid recommender system to produce accurate and

effective recommendations is proposed and experimental evaluation of proposed techniquealong its comparison with existing techniques are provided.

- Chapter 4: This chapter contains the works proposed in this research i.e Cross Domain Item based Book Recommendation system

- Chapter 5: In this Chapter, I conclude my work by presenting results and give prospect to the way forward

Chapter 2

LITERATURE REVIEW

2.1 Single Domain Recommendation System

In the 21st Century, the quantity of vital information obtainable online is hastily mounting second by second because of which the volume and complexity of presented data on internet. The online world presents a huge range of item choices (for example songs, current news, books to read, Movies to watch, home appliances, items to buy, educational research papers etc) to its daily clients/users. The assortment of appropriate item/product from huge quantity of items is an extremely difficult. Currently, majority of online services are extremely burdened with surplus of information that requires large amount of time and galvanization from clients/users to shortlist their attracted items/products. In order to counter this issue smart systems have been developed which takes into account various information filtering techniques that filter out the superfluous information from huge quantity of information presented on the internet or online that protects the client energy and time by locating appropriate information or items/products. The smart systems are also proficient enough to generate the personal recommendations to the clients based on their consumption pattern and as a result, the clients stay happy and remains loyal to your brand [3], [7].

Recommendation Engines are specific smart systems which forecast the rating of a product or an item a user would prefer to give, shown in Figure 2.1. Recommendation Engines are constantly being used by stores that are operating online, Ecommerce, Social media platforms such as Facebook, Instagram, Twitter, Youtube, Snapchat, Netflix, Amazon, Hulu etc to recommend different things to users. These Recommendation engines helps these companies make a lot of money annually and make their users stay loyal to their

brand. All the giant companies in the world right now like (Facebook, Instagram, Twitter, Youtube, Netflix, Amazon, Hulu, Disney, HBO max etc) are using recommendation engines to track and satisfy there users by recommending them the right items or products [2], [8], [9], [10]. Recommendation systems can be broken down as follows (1) collaborative filtering, (2) Content based filtering (3) Hybrid filtering. The structure of Recommendation engines is given in Figure 2.2. Collaborative filtering is widely used approach in which we take into account the items and the ratings given to them by the users.

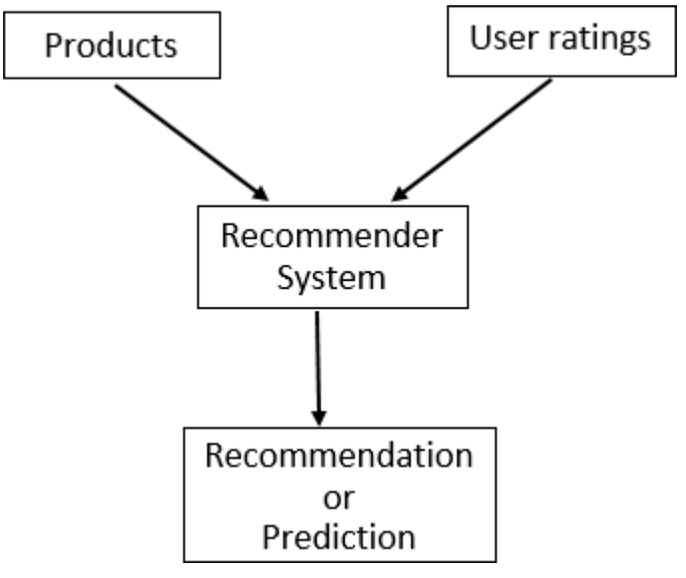

Figure 2.1: Recommender Systems.

Collaborative filtering (CF) is based on the link between user item rating matrixes. It takes into account the criticism for example ratings on scale of 1 to 5 provided by the users on various products in order to make prediction on remaining products. Where Collaborative filtering (CF) fails is, when it tries to make to make recommendations for new productsor clients. This problem is known as “cold start problem”. [10],[11]

In order to counter the cold start issues, there comes the content based filtering (CBF). CBF

takes into account product features in order to recommend various other products related to what the client might like (taking into account) their preceding actions or explicit feedback. Prior to making recommendations, CBF is heavily dependent upon products metadata and entails detail product description and clients structured account.[9], [12].

CBF and CF both approaches have pros and cons, aptness of both methods hangs on the situation in which it will be employed. Hybrid based Recommendation system is best of both world. It takes into account Both Content based Filtering and Collaborative filtering technique.[14], [15], [16].

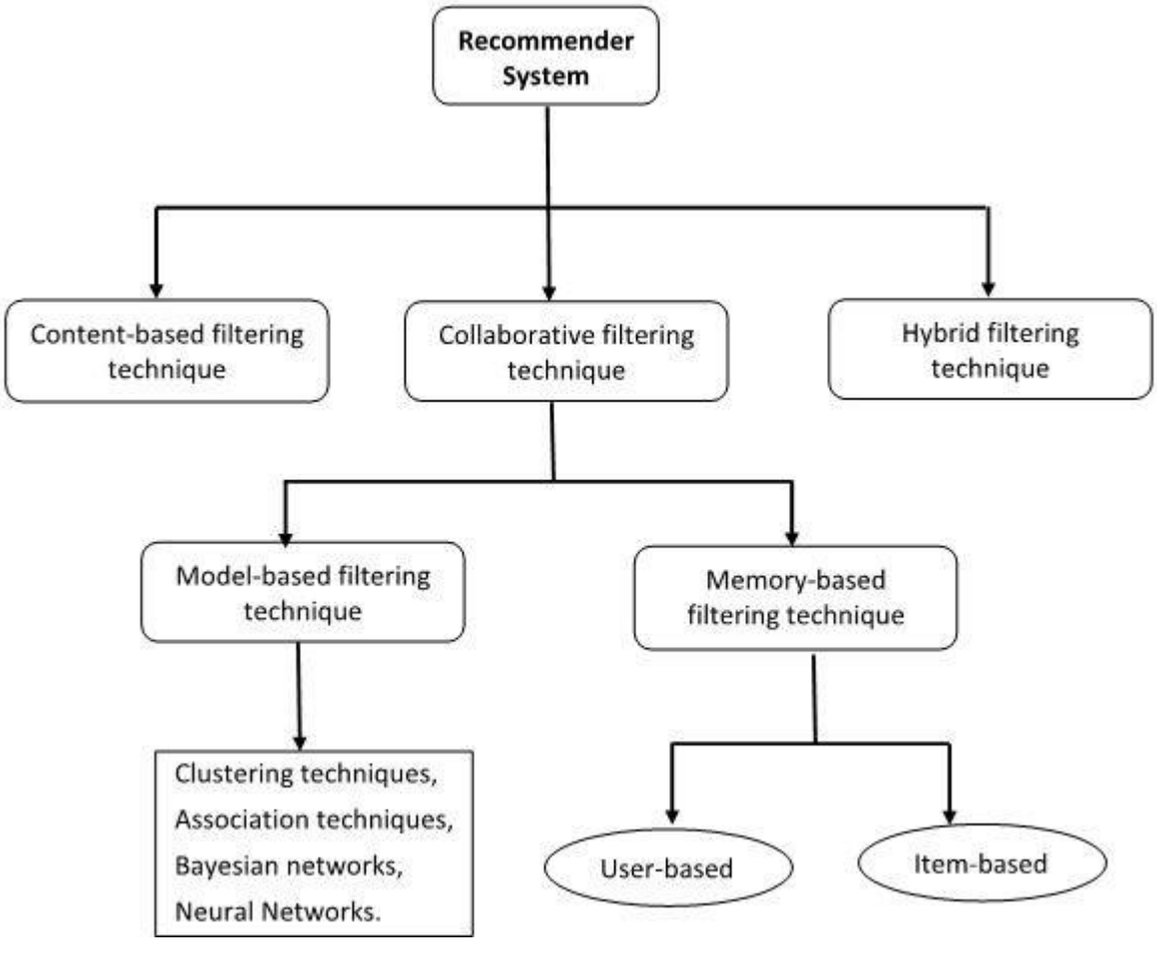

Figure 2.2: Recommendation Techniques [14].

2.1.1 Collaborative Filtering Technique

CF method is the most famous technique used in case of making recommendations. Large quantity of data is gathered from various sources for example client's consumption pattern and past behavior. After gathering the aforementioned data, recommendations are made to

the user based on his or her choices [7]. Let's suppose, there are two guys named Saad and Ali who both likes Action and Horror genre movies. But Ali also like Romantic movies, then Saad will also like Romantic movies since Saad and Ali have same interest in movies. CF splits into two branches 1) Model based filtering 2) Memory based filtering [1], [14].

Model Based Filtering:

Model based filtering takes into account of clients input on various products, data is collected from dataset to train the model by using various Machine learning (Data mining). Model based filtering approach takes into account many algorithms such as KNN (K nearest neighbor), Matrix Factorization algorithms for examples SVD,PMF,NMF and lastly Deep learning algorithms for example Neural Network, Hidden Markov models (MDP), Latent dirichlet allocation (LDA). Dimensionality Reduction methods are also utilized in Model based approaches for examples the PCA approach and the SVD matrix factorization approach. Low dimensional representation of item user rating matrix will be taken into account with item based or user based approaches to predict the ratings of missing values in item user rating matrix [18], [19].

Advantages of Model Based Filtering:

- It counters Data Sparsity issues as well as Scalability issues in excellent way.
- Given very good recommendations.
- Provides very quick recommendations with good precion.
- Counters Over fitting problems.

Disadvantages of Model Based Filtering:

- Requires a lot of resources to deploy.
- Understanding of the data is lost when Techniques such PCA or SVD are used.
- Trade-off among prediction and scalability.

- Eminence of predicted values is totally depended upon the model that you build

Memory Based Filtering:

Memory based approach takes into account the whole of the Database to compute similarity between items and users. Some similarity measures are as follows 1) Cosine similarity 2) Pearson correlation 3) Jacquard coefficient 4) Adjusted Cosine similarity. The methods are used among users or item to compute user preferences on certain products and these preferences are used to predict ratings for a given client. The aforementioned approaches takes into account the data such as ratings, clicks, votes etc to build correlations between users or items. Memory based filtering contains [20], [21]. Memory based filtering approach splits into two types 1) User based filtering 2) Item based filtering. In User based filtering method, the algorithm tries to find similar users based on the content they love or like and recommend them that content. In short, similar users are recommended similar content. In item based filtering, the algorithm takes into account users past history and tries to find the item that a user would like to buy or like or subscribe. [22], [23].

Advantages of Memory Based Filtering:

- Deployment in real environment is easy.
- Provides scalability with co-rated products.
- Items are recommended based on their similarity.

Disadvantages of Memory Based Filtering:

- Suffers from Data Sparisty problems.
- Requires data from the users like rating or voting.
- Suffers from Cold start issues.

In [25] Author presented a Collaborative Item based filtering approach in order to counter data sparsity issues and data scalability problems by computing the link between products and making recommendations on similar items to clients .Various matrix factorization Approaches have been shared based on Collaborative filtering techniques. In Matrix factorization we split our main matrix into two smaller matrixes (user item rating matrix). Then the algorithm tries to find hidden underlying features for each user and recommend them items accordingly. In [26] the author presents a PMF model method which takes into account the client that gave rating to similar films will have similar likings. This approach counters Data sparsity issues. Similarly A Non-Probalistic model is used in collaborative filtering approach which groups the rating matrix to lessen rating dimensionality these dimensions represents the users taste [27].

2.1.2 Content-Based Filtering Technique

Content based Filtering approach is taken into account where Collaborative filtering falls short. Which is, it is used to counter Data sparsity issues. It studies the product attributes in depth to produce data features so to compute client's profile. Content based Filtering approach is mostly famous because of it is easy to use. Content based Filtering approach solely relies on the contextual information while making recommendations. CBF has a lot of advatages as well as cons. CBF counters cold start issues as it takes into account the similarity between the user and product profile Content based Filtering approach is handy in forecasting predictions for new users or products. Content based Filtering approach also has cons. Extracting features of users/ items can be a challenging task. CBF depend on the items'/ users' metadata and requires structured data while making recommendations. CBF has a limitation that it cannot recommend items to user out of user profile content, also known as overspecialization problem [28], [29]. Figure 2.3 gives a block diagram of CBF.

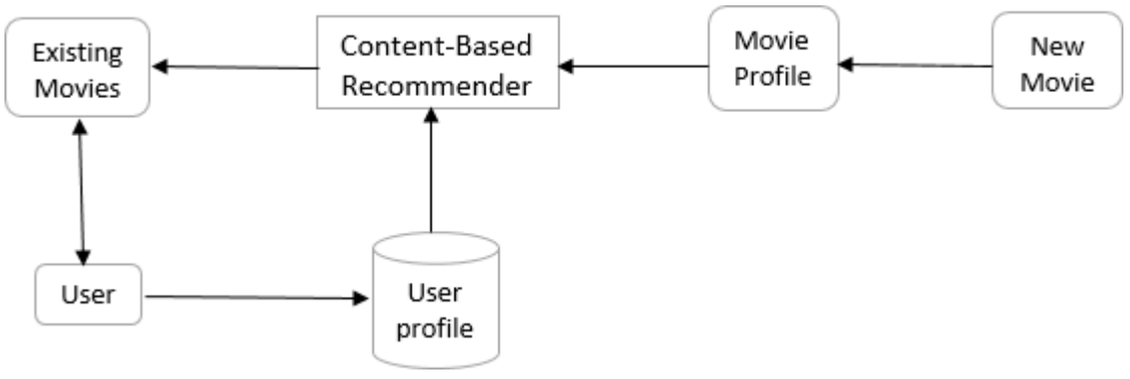

Figure 2.3: Content-Based Recommender Systems.

Content-based filtering approach takes into account various algorithms to study user's profiles. They include Bayesian classifier, ANN (Artificial neural network), etc. The aforementioned algorithms are extremely efficient compared to normal similarity measures as these algorithms learn a model by tracking hidden consumption patterns from training data and then output recommendations. [30], [31].

Advantages of Content-Based Filtering:

- Counters data sparsity issues.
- Presents recommendations for items which suffers from cold start issues.
- Better Prediction quality.
- Clients are recommended items based on their likeness.

Disadvantages of Content-Based Filtering:

- Suffers from Restricted content issues.
- This approach requires a large database to store huge data of users.
- Recommendations for new comer users is difficulty as no data of them is in the database

2.1.3 Hybrid Filtering Technique

Hybrid filtering approach is a combination of Collaborative fileting approach and Content based filtering approach by taking into account the content features and user feedback for outputting recommendations. Hybrid filtering approach counters cold start issues and data sparsity problems (which Collaborative fileting approach suffers from) by taking into account the content information and deletes the overspecialization. Figure 2.4 represents Hybrid filtering block diagram. Hybrid filtering is split into 3 parts based on integration technique of Collaborative fileting approach and Content based filtering approach that are linear, sequential and mixed. In case of linear approach, Collaborative fileting approach and Content based filtering approach are forged together for forecasting predictions. In case of Sequential method content based filtering is utilized for finding out appropriate clients [32]. A total of 7 various hybrid filtering approaches have been presented in [33], such as

Weighted: This approach gives outputs which are numerically forged to give one recommendation.

Switching: In this approach various Recommendation approaches are swapped depending upon the scenario.

Mixed: In this approach various standings generated by different methods are forged together as one recommendation.

Feature augmentation: In this approach, the output of one recommendation engine acts as input to another method.

Feature combination: In this approach various Features are mined from multiple sources are grouped together and then given as a one recommendation model as input.

Cascade: Recommendation of a single recommendation engine is made better and

optimized by a 2nd recommendation engine.

Meta−level - A machine learning model, learned through a single recommendation engine, is then given as input to another recommendation engine.

.

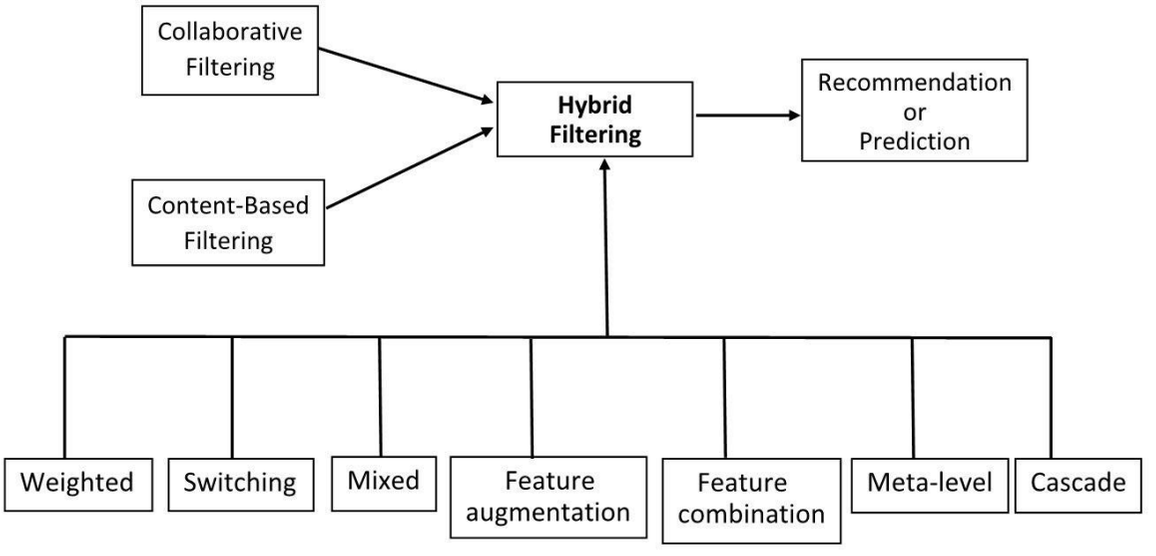

Figure 2.4: Hybrid Recommender Systems.

Advantages of Hybrid Filtering:

- Counters Data sparsity issues in excellent way.
- Counters cold start issues related to items and users
- Counters users and product scalability problems.
- Provides enchanted performance and counters all the restrictions related to both Collaborative fileting approach and Content based filtering approach.

Disadvantages of Hybrid Filtering:

- Increased complexity because of CF and CBF approach.
- Deployment is expensive.
- Requires a lot of data which is not easily available.

2.2 Cross Domain Recommendation Systems

Generally Cross Domain Recommendation systems can be grouped into 3 groups: (1) Content based technique (2) Embedding based technique (3) Rating based technique. Content based approach primarily deals with Cross domain recommendation system issues with content significance, tries to establish the link various fields by classifying comparable content data, for example product details, client reviews on product and tags given by users to items [35] In comparison, the embedding-based approach [36], primarily takes into account the Cross domain recommendation engine problems with client-level relevance and product-level relevance. The above mentioned approach involves 1st training different Collaborative filtering models (like SVD), 2) Maximum-margin matrix factorization, 3) Probabilistic matrix factorization, 4) Bayesian personalized ranking, 5) Neural collaborative filtering and lastly 6) deep matrix factorization to get client/product information. Then these embedding's are transferred through a common or similar client/product across domains. The rating pattern approach shifts an independent knowledge for example pattern of the rating, across fields. When comparing the content based technique and embedded based approach, the pattern of the rating technique takes into account machine learning approaches, for example multitask learning [38], transfer learning [39], clustering [40], and the NN (Neural Networks) [41], to transport knowledge through domains.

Aforementioned Cross domain Recommendation system approaches are defined for single-target technique which can only import the data from a wealthier filed to help a less rich domain. Nevertheless, each of the individual domains can be comparatively richer in definite types of data (for example ratings given by users, reviews left by users, user profiles, product details, and social tags given by the users). Such information can improve recommendations in all the domains rather than a target domain only if it can be imported. In recent years, Dual-target Cross domain recommendation engines [42] and multi-target Cross domain recommendation engines [43] have been discussed to improve the recommendations in both dual and multiple fields.

2.2.1 Single Target Cross Domain Recommendation

Single-target cross domain recommendation system is an orthodox recommendation engine setup in Cross Domain recommendation system area and a lot of the current Cross Domain recommendation system approaches focus on this technique.

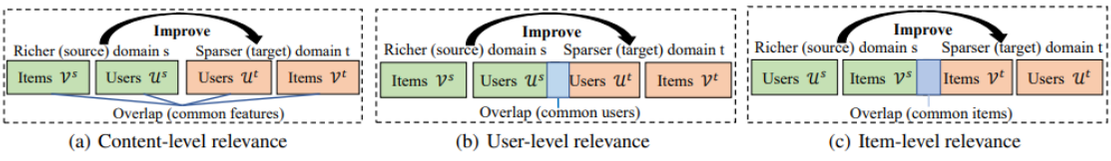

Figure 2.5: Single Target CDR

Single Target CDR can be defined as "Given a source field S (comprising a client set U(s) and an product set V(s)) with wealthier data for example explicit feedback (like client ratings and clients comments), implicit feedback (like clients purchasing and surfing histories), and lastly the side data (for example clients online profiles and product specifics) and the target field T (comprising a client set U(t) and product set V(t)) with sparser data, Single target cross domain recommendation system is to increase the recommendation accuracy in target field T by leveraging the auxiliary information in source field S".

Single Target Cross domain recommendation system can be further split into 3 sub-groups mentioned above which are 1) Content base approach 2) Embedding based technique 3) Rating pattern based technique.

Figure 2.5 (a), shows us the Content base approach. In this method, in order to increase the recommendation accuracy in the target domain, we initially build content-based relations, after that, we take into account alike client/products based on their mutual features, and lastly, we transfer additional features between alike clients/products across domains.

In Figures 2.5(b) and 2.5(c), in order to increase the recommendation accuracy in the target domain, we initially produce accurate client/product rating patterns, and then transfer the embeddings of mutual client/product or rating patterns of mutual users through domains.

2.2.2 Dual Target Cross Domain Recommendation

Dual-target cross domain recommendation system is a brand new recommendation system technique in cross domain recommendation area. It is defined as "Provided two fields Source and Target, comprising of clients sets U(1) , U(2) and product sets V(1) , V(2) respectively, dual-target cross domain recommendation system increases the recommendation accuracy in both source and target fields concurrently by leveraging their data"

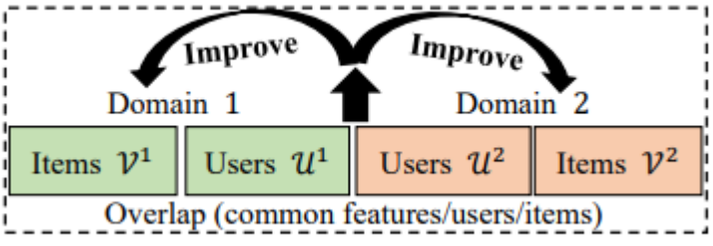

Figure 2.6: Dual Target CDR

Just like, single target cross domain recommendation system, the dual target cross domain recommendation system can be sub grouped into 3 groups are 1) Content base approach 2) Embedding based technique 3) Rating pattern based technique. Dual target cross domain recommendation system can take into account (content level approach), mutual clients (user level relevance), and lastly mutual products (product level relevance), In order to establish a strong relation the source and target domains. Mutual knowledge is transferred between domains. While Single Target Cross Domain Recommendation increases accuracy in only one domain, Dual target cross domain recommendation system increases accuracy in both source and target fields.

2.2.3 Multi Target Cross Domain Recommendation

The Multi target cross domain recommendation system aims to increase the recommendation output in numerous fields concurrently. The fundamental impression of Multi_target cross domain recommendation system is to import additional information from additional fields in order to achieve an additional enhancement of recommendations. Multi target cross domain recommendation system is defined as, "Provided the multiple domains from 1 – (N), comprising clients sets from U(1) to U(n) and product sets from V(1) to V(n) Multi target cross domain recommendation system increases recommendation accuracy in every field concurrently by importing their data."

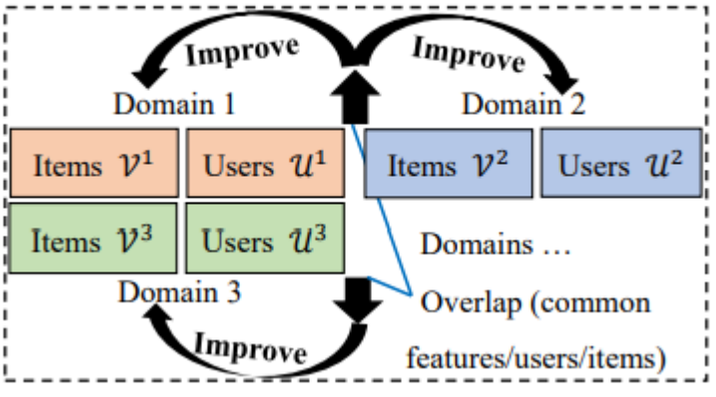

Figure 2.7: Multi Target CDR

Multi target cross domain recommendation system objective is to accomplish even larger aim, for example giving a complete whole Answer/Pipeline/Framework for data sparsity problems. In theory, if the Multi target cross domain recommendation system can discovery sufficient connected domains and take into account the auxiliary information from these connected domains then the data sparsity issues in recommendation engines can be greatly eased and even resolved.

2.2.4 Sequential Cross Domain Recommendations

Sequential Cross Domain recommendation system have garnered much lime light as it can recommend products to clients by modeling the sequential dependencies over the client-Product interactions [44]. Obviously, Cross Domain recommendation system suffers from the problem of sequentially modeling of clients and products, the same way as orthodox recommendation engines do. Preceding research on sequential recommender systems primarily focuses on educating the upper order, long term, and deafening client-Product communications in sequence. This has become additionally tough for sequential Cross Domain recommendation system since it not only requires to model sequential client-Product communications, but also share data across domains [45]. Hence, Sequential Cross Domain recommendation system have become the 2nd favorable research prospect.

2.2.5 Privacy Preserving Cross Domain Recommendations

Currently most techniques in Cross Domain recommendation system take up that information through fields are accessible in simple pain document text, which disregards the data remoteness issue in training. Seemingly, a lot of recommendation engines have been constructed using clients delicate data, for example, check in data, client's profile and browsing history. In Cross Domain recommendation system, such information is typically detained by different domains, for example E-bay and Amazon_. In few cases, this information across domains cannot shared with others openly as they encompass delicate data. Therefore, it is crucial to construct Cross Domain recommendation systems in order to defend information privacy [46]. A new research on privacy preserving Cross Domain recommendation system shows that it is only equipped to handle simple social matrix factorization models [47], and a lot of research has to be done in this field so far.

2.3 Quantitative Measures for Recommendation Systems

Various evaluation methods are utilized to compute the performance of recommendation Engines. The performance of recommendation engine heavily relies on the evaluation methods. The criteria of evaluation method is totally dependent upon the type of filtering approach and recommender applications. The evaluation methods for recommendation engines are as follows

2.3.1 Prediction Based Metrics:

Root Mean Square Error The Root Mean Square Error is the most commonly used metric to evaluate the difference between actual rating given to an item vs the predicted rating of them item given by recommendation system. Low RMSE score means the recommendation engine is performing well there is minute difference between actual and predicted rating. The RMSE formula is given as follows:

$$RMSE = \sqrt{\frac{1}{N}\sum_{u,x}(R_{ux} - \hat{R}_{ux})^2} \tag{2.1}$$

Where;

- u for users and x for CS items.
- $\hat{R}_{ux}$ is ratings predicted by recommendation engine.
- R_{ux} is ratings given by the user.
- N represents total number of predicted ratings.

Mean Absolute Error The Mean Absolute Error (MAE) takes into account the average of the absolute difference between predicted ratings by RCS and the actual ratings:

$$MAE = \frac{1}{N}\sum_{u,x}|R_{ux} - \hat{R}_{ux}| \tag{2.2}$$

Where;

- u for users and x for CS items.

- $\hat{R}_{ux}$ represents predicted ratings.
- R_{ux} represents known ratings.
- N represents total number of predicted ratings.

2.3.2 Classification Based Metrics:

Precision Precision is basically explains, Out of all the positive predicted, what percentage is truly positive.

$$Precision = \frac{Correctly\ recommended\ \ items}{Total\ \ recommended\ \ items} \tag{2.3}$$

$$Precision = TP / TP + FP \tag{2.4}$$

Here, TP = True Positive and FP is False Positive

Precision is also called positive predictive value.

Equations 2.3 and 2.4 both represent the same thing. TP basically is telling us the count of correctly recommended items. And TP + FP is basically represents Total recommended items.

Recall Out of the total positive, what percentage are predicted positive. It is the same as TPR (true positive rate).

$$Recall = \frac{Correctly\ recommended\ items}{Total\ useful\ recommended\ \ items} \tag{2.5}$$

$$Recall = TP / TP + FN \quad (2.6)$$

Here TP is True Positive and FN is False Negative

Recall is also known as sensitivity.

Equations 2.4 and 2.5 also both represent the same thing. TP is basically telling us the count of correctly recommended items. And TP + FN basically represents Total useful recommended items.

F1-Score F1 score is the harmonic mean of recall and precision. It takes both Precision and Recall into account. Therefore, it performs well on an imbalanced dataset.

$$F1 - measure = \frac{2PR}{P + R} \quad (2.7)$$

Here, P stands for Precision score and R stands for Recall.

Summary:

In Chapter 2, a detailed study of Single domain Recommendation systems and Cross Domain Recommendations systems has been shared. Concerns in prevailing Single domain and Cross domain recommendations has been offered along with prevailing exploration is also shared. Evaluation meterics which are used to judge the recommendation engines are also discussed. In chapter 4, a Cross domain Item based Book recommendation system has been discussed. A lot of research has been done in single domain in last 14 years. Cross domain CDRS is relative a new field and it provides better recommendations compared to single domain. It overcome challenges faced in single domain like data sparsity issues and recommendations for new clients. Cross Domain Recommendation systems opens a new world of research in the world of Recommendation engines.

Chapter 3

Cross Domain Item based Book Recommendation System

For a Decade or so after the Netflix 1 million dollar prize competition, a huge majority of research has been done in Single domain Recommendation systems. By single domain, it means, our Source and Target domains are the same. In Source domain, we have the users and their data of consumption for a particular domain for example like Movies, books, songs, e-commerce, News articles, shopping etc and we recommend them various things from the same domain. Hence source and target domains are the same. Cross domain recommendation systems are a new types of Recommendation systems. In Cross Domain recommendation systems we have two domains. One domain is called source domain and second domain is called Target domain. In source domain we have users along with their data and they are recommended things from the Target domain, which is completely entirely different domain. For Cross Domain Recommendation systems to work, there has to be a link between Source and Target domain. In this chapter, I have present in detail about the Pipeline that I have built for Cross Domain Item based Book Recommendation system. Figure 3.0 gives a basic understanding of Cross domain recommendation system along with the link.

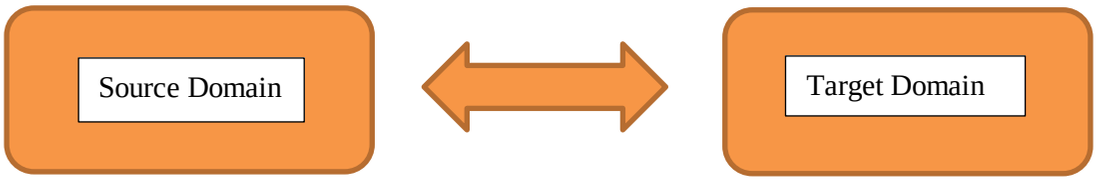

Figure 3.0: Cross Domain Block diagram

In this chapter I propose a model for Building a Cross domain Item based Book recommendation systems. The proposed model improves upon the current research in this field as well as provides extremely good F1 score, Recall, Precision at the source domain. The proposed model also builds a strong semantic similarity between Source and Target domain through a new method called Wpath. And lastly, in this proposed model, I introduce a new technique at the Target domain called Multi Label Binaizer approach which breaks a book into its genres and make sequences of the books. This helps in providing final Recommendations to the user.

3.1 Proposed Model/Pipeline Framework

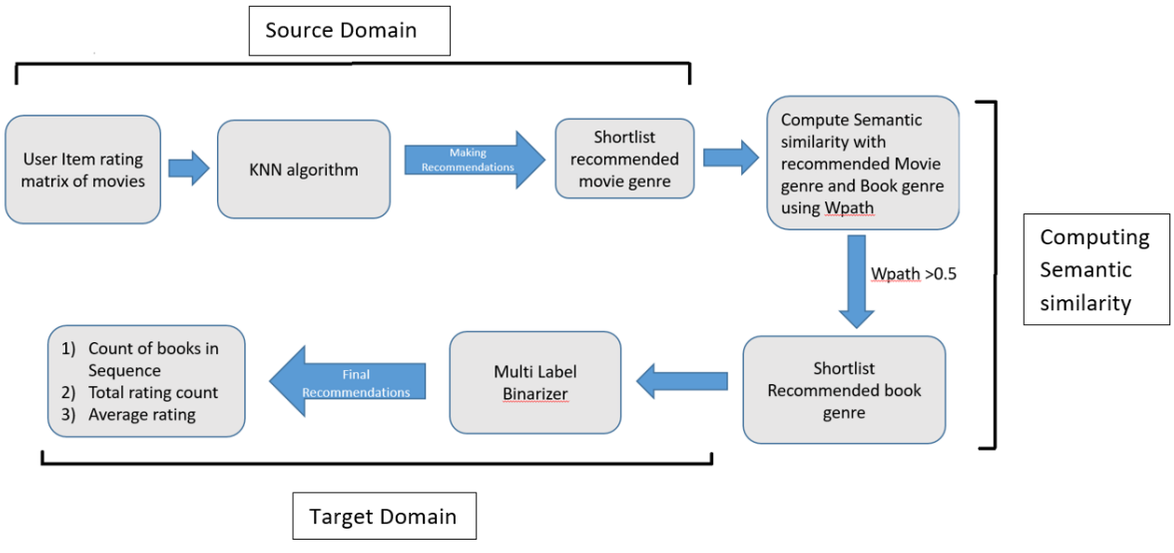

Figure 3.1: Proposed Model flow chart.

3.2 Source Domain

In the source domain part, in the first part, we build an item based movie recommendation system using KNN algorithm. The dataset we are using is Movelens 100k dataset. This dataset is consists of 100,000 ratings, ranging from 1 to 5 votes, from a total of 943 users on 1682 movies. The first step in building the recommendation engine at the source part, I

build a user ite`m rating matrix. Basic understanding of how this matrix looks something like given in the figure 3.2.

	Item 1	Item 2	Item 3	Item 4	Item 5
User 1	0	3	0	3	4
User 2	4	0	0	2	0
User 3	0	0	3	0	0
User 4	3	0	4	0	3
User 5	4	3	0	4	4

Figure 3.2: User item rating matrix

In item based collaborative filtering approach, once user item rating matrix is build, we take into account the users past behavior history on various items and try to compute what item the user would like to buy, rate or subscribe. In our case, it would be what movie a user would like to watch. For this, we initially compute the similarity between items based on the ratings given to them by the users. This would show us which items/movies are similar to which items/movies. There are numerous methods of computing similarity between items based on the ratings given to them by the users for example like 1) Euclidean distance 2) Adjusted Cosine Similarity 3) Cosine Similarity etc. The method which I used is called Cosine similarity. Its equation is given as follows

$$Similarity(\vec{A}, \vec{B}) = \frac{\vec{A} \cdot \vec{B}}{||\vec{A}|| * ||\vec{B}||} \quad (3.1)$$

Here in the formula, Cosine similarity is the dot product between two different items A and B based on the ratings given to them by the client. Once Cosine Similarity is computed to find movies that similar to each other based on the rating given to them by the users, we now want recommend those movies to the users which are not seen by users. For rating predictions for unseen movies, we use the following formula:

$$rating(U, I_i) = \frac{\sum_j rating(U,I_j) * s_{ij}}{\sum_j s_{ij}} \quad (3.2)$$

Here, the rating prediction of the unseen movie for a given user is computed.

- U here is the particular user for whom the rating is being predicted for a particular item/movie labeled as I
- I here is the unseen movie for which the rating is being predicted for a given user U.
- S here stands for Cosine similarity score.

To implement the following item based collaborative filtering approach at source domain**,** I take into account the KNN algorithm. KNN algorithm is the best go to algorithm as it is extremely good baseline for recommendation engines. It takes into account the database in which the data points are separated into several clusters to make inference for new samples.

The KNN algorithm does not make any conventions on the fundamental data distribution but it heavily relies on the similarity between the items. When KNN algorithm outputs an implication regarding a film, it calculates the space among the target film and every other film in its database and lastly it then ranks its distances and returns the top K nearest neighbor film as the most similar film recommendations.

In the proposed model, when KNN algorithm is being used, it requires some input parameters, the ones which I used in my model are

- Algorithm = Brute Force approach
- Evaluation metric to be Cosine Similarity
- And K nearest neighbors to be 10

In proposed model, I used Cosine similarity metric to find the similarity between movies. I used value of K to be 10, to get 10 closest recommended movies. And lastly I used brute force approach to find all possible every possible recommended movie. After the recommendations are made for a particular user, the genres are shortlisted from those recommended movies. The genres are basically recommended movie genres.

The genres that the movies contained in the 100k movielens dataset are as follows

- 'Mystery'
- 'Western'
- 'Adventure'
- 'Action'
- 'Crime'
- 'Documentary'
- 'Thriller'
- 'Animation'
- 'Comedy'
- '(no genres listed)'
- 'Sci-Fi'
- 'Horror'

- 'Romance'
- 'Film-Noir'
- 'IMAX'
- 'Drama'
- 'War'
- 'Children'
- 'Musical'
- 'Fantasy'

There are a total of 20 Genres of movies in the 100k movielens dataset

3.3 Link between Source and Target Domain

In Cross domain recommendation systems, there are two domains as mentioned above. In Source domain we have users and their consumption pattern and in Target domain, we recommend users different things. Target domain is completely entirely different domain. For Cross Domain Recommendation systems to work, we need to have a strong Link between them. Or in our case, a strong semantic similarity relation between source and target domain. In [33] the authors tried to establish 1 on 1, pair wise link between Source and Target domain, but that approach fails in real world as soon as users start to assign random tags to items. In [34] the authors tried to counter the aforementioned issue, by having multiple domains at the source so that Target domain can have some link with the source. But this approach also fails when deal with problems which have only one domain at the source. This yields poor link between source and target domain. There are numerous ways to computing semantic similarity between two different nodes. The Conventional way of computing semantic similarity is through KG (Knowledge graph) approach. But Knowledge graphs suffers from “Uniform distance” problem. According to Uniform distance problem, two nodes with same path length, their semantic similarity is same.

In order to counter this issue, in the proposed model I am using a new method of computing semantic Similarity called Wapth similarity method [35]. Wpath similarity method has advantage over the Knowledge graph approach. One Biggest advantage of Wpath over Knowledge graph is the elimination of Uniform distance. Wpath approach combines IC (Informativeness of Concept) in measuring the semantic similarity between concepts and path lengths. Its formula is as follows:

$$WpathSimilarity = \frac{1}{1 + length(C_i, C_j) * K^{(IC)(C_{lcs})}} \tag{3.3}$$

Here, C(j) and C(i) represents source and target domains. IC here represents the shared knowledge or concept between the two nodes. The nodes in our case, are the movies and books genres.

The Books dataset is built by combining three datasets. Those datasets are

1) Amazon books dataset
2) Goodreads books dataset
3) Booksummaries.txt

The books datasets contained the book title along with it author, as well as the total rating counts of the book, average rating of the books and lastly the genres belonging to the books. The Genres contained in the books dataset are as follows:

- ‘Humor'
- 'Mystery'

- 'Adventure'
- 'Action'
- 'Crime'
- 'Thriller'
- 'Science Fiction'
- 'Fiction'
- 'Suspense'
- 'Comedy'
- 'Historical'
- 'Classics'
- 'Comic'
- 'History'
- 'Novel'
- 'Horror'
- 'Romance'
- 'Drama'
- 'War'
- 'Children'
- Fantasy

A brief look of the Wpath similarity computation between Movies and Book genres looks something like this:

	Movie_genre	Book_genre	wpath
0	Mystery	Mystery	1.000000
1	Mystery	Action	0.655052
2	Mystery	Thriller	0.655052
3	Mystery	Fiction	0.729835
4	Mystery	Novel	0.642979
5	Mystery	Romance	0.740157
6	Mystery	Fantasy	0.642979
7	Adventure	Adventure	1.000000
8	Adventure	War	0.502001
9	Action	Mystery	0.655052
10	Action	Action	1.000000

Figure 3.3: Sample of Wpath Similarity between Movies and Books genre.

In the proposed model, while computing the Wpath semantic similarity, I only shortlisted those books genre which have Wpath score of above 0.5 so that I only get those shortlisted books genres which have really strong semantic similarity with the recommended movie genres.

Wpath similarity outputs a score between 0 and 1.

3.4 Target Domain

In the proposed model, after computing the Wpath semantic similarity between recommended movies genres and all of the books genre, only those books genres are shortlisted which have the Wpath semantic similarity score of 0.5 and above as I want the best books to be recommended at the Target domain. After this, in the proposed model, I use a new approach called “Multi Label Binarizer”. This approach helps me with two things.

1) Breaks a book into its genre

2) Make sequences of the books

What Multi Label Binarizer approach do is that, it encodes the book genre into 1's and 0's.

A brief look of how this looks, is given as follows:

	title	authors	average_rating	ratings_count		Action	Adventure	Children	Classics	Comedy	...	History	Horror	Humor	Mystery	Novel	Ro
215	The Stand	Stephen King, Bernie Wrightson	4.34	438832	0	0	0	0	1	0	...	0	0	0	1	1	
216	It	Stephen King	4.18	292592	0	0	0	0	1	0	...	0	0	0	1	1	
217	The Gunslinger	Stephen King	3.99	332494	0	0	0	0	1	0	...	0	0	0	1	1	
218	Carrie	Stephen King	3.93	356814	0	0	0	0	1	0	...	0	0	0	1	1	
219	Misery	Stephen King	4.11	334647	0	0	0	0	1	0	...	0	0	0	1	1	
...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	
307	Inferno	Dan Brown	3.80	287533	0	0	1	0	0	0	...	0	0	0	1	1	
308	The Girl on the Train	Paula Hawkins	4.10	79446	0	0	1	0	0	0	...	0	0	0	1	1	
309	The Girl Who Kicked the Hornet's Nest (Millenn...	Stieg Larsson	4.70	77747	0	0	0	0	0	0	...	0	0	0	1	1	
310	The Girl Who Played with Fire (Millennium)	Stieg Larsson	4.70	76251	0	0	0	0	0	0	...	0	0	0	1	1	
311	The Girl with the Dragon Tattoo (Millennium Se...	Stieg Larsson	4.40	505519	0	0	0	0	0	0	...	0	0	0	1	1	

Figure 3.4: Multi Label Binarizer Implementation

Once the genres of the books are encoded into 1's and 0's, in the next step, One by one I equate recommended books genres to 1 and extract those books specifically belonging to that particular genre and built their Dataframe. Once this step is done for all of the recommended book genres, I join all of the books dataframes and this help generate sequences of the books. The books which

occur most number of times in the sequences, are the books which have the most number of recommended genres.

The top 100 books are shortlisted in the final recommendations but top 10 books which occur the most number of times in the sequence are presented as the final recommendations.

To make the Books recommendations even more exciting, we can further take into account, the Total rating counts of the books as well as the average ratings of those books. After this we can present the top 10 books as Final Recommendations.

Chapter 4

EXPERIMENTAL RESULTS AND ANALYSIS

In proposed Pipeline, the results are compared in three difference domains i.e

a. Source Domain

b. Link between Source domain

c. Target domain

The results and Analysis are compared with the existing methods and the proposed pipeline provides better results compared to the existing approaches.

4.1 Source Domain Results and Comparison

In the source domain, the results are compared with the existing KNN approaches and other methods. For comparison three movies were shortlisted and their RMSE, Recall score, F1 score and Precision is computed. The result comparisons are as follows:

4.1.1 Toy Story

Evaluation Metrics	**Proposed Method**	**Existing Approach (CD-SPM)**
RMSE	**1.005**	**1.3**
Recall	**0.966**	**0.933**
F1 Score	**0.964**	**0.933**
Precision	**0.962**	**0.933**

4.1.2 Sabrina

Evaluation Metrics	Proposed Method	Existing Approach (CD-SPM)
RMSE	**0.994**	**1.8**
Recall	**0.964**	**0.933**
F1 Score	**0.971**	**0.95**
Precision	**0.978**	**0.968**

4.1.3 Twelve Monkeys

Evaluation Metrics	Proposed Method	Existing Approach (CD-SPM)
RMSE	**0.974**	**0.99**
Recall	**0.965**	**0.933**
F1 Score	**0.967**	**0.95**
Precision	**0.970**	**0.968**

When we compare the results of Propose approach with existing CD-SPM (Cross-Domain Sequential pattern mining) approach, in terms of RMSE, the proposed approach performs better. RMSE tells us the difference between the actual predictions to that particular movie vs rating predicted by the model for that particular movie. Higher the RMSE, the bad prediction is given by the model.

In terms of Recall, F1 score and Precision, all of these scores are better in the proposed approach compared to the existing CD-SPM approach. Precision basically tells us that, of all the positively forecasted (TP + FP) ratings, what percentage is truly_positive (TP). Its value is from 0 to 1. The closer the value is near to 1, the better. Recall basically tells us that, out of all the correctly predicting ratings (TP + FN), what is the percentage of correctly positively predicted (TP) ratings.

Recall value is also between 0 and 1. The closer the value is to 1, the better Recall it is.

F1 score is the harmonic mean of recall and precision. Its value is also between 0 and 1. Closer the value of F1 score is to 1, the better it is.

The comparison here at the source domain shows us that the Propose approach gives better RMSE, F1 score, Recall and Precision compared to the existing CD-SPM approach.

4.2 Link between Source and Target Domain Results and Comparison

In order to understand the relation of one genre with another, different approaches have been used. The following table provides a very good comparison between different Methods which compute the semantic similarity. We can clearly see that Wpath similarity provides a better understanding between Books and Movies genres.

Genre	**Path**	**Li**	**Jcn**	**Wpath**
Romance-Mystery	0.33	0.112	1	1
Romance-Fantasy	0.25	0.547	0.119	0.642
Comedy-Mystery	0.09	0.112	0.0591	0.152
Romance-Fiction	0.33	0.669	0.194	0.729
Fantasy-Thriller	0.2	0.448	0.089	0.574
Romance-Thriller	0.25	0.548	0.086	0.655

Table 4.1.4 : Comparison between different Semantic similarity measures

The reason why Wpath semantic similarity techniques explains the relation compared to the orthodox knowledge-based techniques is that, Knowledge-based techniques suffers from Uniform distance issue, where two nodes have the similar semantic similarity score if their path-length is the same. Wpath technique combines the path length with the shared concept or the knowledge or Informativness of concept between the two genres. Hence it provides better understanding of the relationship between movie genre with book genre.

4.3 Target Domain Results and Comparison

In the Target domain, after the Wpath Similarity is computed, only those books genres are shortlisted which have the Wpath score of above 0.5 for better recommendations. For results comparison, a random user with **user ID "15"** was picked up. At the Source Domain, the recommended genres for User ID 15 were as follows:

- Drama
- Adventure
- Sci-Fi
- Children
- Horror
- Action
- War
- Crime
- Comedy
- Fantasy
- Thriller

The Recommend movies genres Wpath Similarity was computed with all of book genres and the following book genres were shortlisted after filtering only those book genre which had Wpath similarity score of above 0.5. They are as follows:

- Fantasy
- Fiction
- Children
- War
- Drama

- Horror
- Comedy
- Thriller
- Crime
- Action
- Adventure
- Suspense
- Novel
- Romance

At Target domain, Multi Label Binarizer approach is applied on the Books dataset to break a down into its genres and secondly to make sequences of the books. Only those books are shortlisted which have the aforementioned genres in them. The Final recommendations looks like this:

title	authors	average_rating	ratings_count	genre	counts
The Lost Symbol	Dan Brown	4.20	369428	Fiction|Mystery|Thriller|Novel|Adventure|Crime...	6
The Girl on the Train	Paula Hawkins	4.10	79446	Fiction|Mystery|Thriller|Novel|Adventure|Crime...	6
The Da Vinci Code	Dan Brown	3.79	1447148	Fiction|Mystery|Thriller|Novel|Adventure|Crime...	6
Deception Point	Dan Brown	3.67	455610	Fiction|Mystery|Thriller|Novel|Adventure|Crime...	6
Digital Fortress	Dan Brown	3.60	423019	Fiction|Mystery|Thriller|Novel|Adventure|Crime...	6
Inferno	Dan Brown	3.80	287533	Fiction|Mystery|Thriller|Novel|Adventure|Crime...	6
From a Buick 8	Stephen King	3.42	47320	Fiction|Fantasy|Classics|Mystery|Science Ficti...	5
Lisey's Story	Stephen King	3.65	50721	Fiction|Fantasy|Classics|Mystery|Science Ficti...	5
The Bachman Books	Richard Bachman, Stephen King	4.10	52824	Fiction|Fantasy|Classics|Mystery|Science Ficti...	5
Nightmares & Dreamscapes	Stephen King	3.90	54401	Fiction|Fantasy|Classics|Mystery|Science Ficti...	5

Figure 4.2: Top 10 Book Recommendations for user 15.

The Top 10 Books are listed based on the number of **"Counts"**. The **"Counts"** column here

characterizes the total amount of times a book has occurred in the sequence. Or in other words, “Counts” characterizes the number of recommended genres in the book.

To make the recommendations at the Target domain a bit more generalized, we can take into account, (1)“Rating_count”, which is total number ratings given to the book or Number of users who have read the book and (2) Average ratings of the books given to them by the users. The Final generalized Recommendations now look something like this:

title	authors	average_rating	ratings_count	genre	counts
Complete Harry Potter Boxed Set	J.K. Rowling	4.74	190050	Fiction|Fantasy|Classics|Adventure|Children	4
The Essential Calvin and Hobbes: A Calvin and ...	Bill Watterson	4.65	93001	Fiction|Classics|Comic|Humor|Comedy	3
Calvin and Hobbes	Bill Watterson	4.61	117788	Science Fiction|Fantasy|Adventure|Novel|Children	4
Harry Potter and the Deathly Hallows	J.K. Rowling, Mary GrandPrÇ¸	4.61	1746574	Fiction|Fantasy|Classics|Adventure|Children	4
The Hobbit and The Lord of the Rings	J.R.R. Tolkien	4.59	90907	Fiction|Fantasy|Classics|Novel|Adventure	4
Fullmetal Alchemist Vol. 4	Hiromu Arakawa/Akira Watanabe	4.55	210752	Fiction|Fantasy|Adventure	3
Harry Potter and the Half-Blood Prince	J.K. Rowling, Mary GrandPrÇ¸	4.54	1678823	Fiction|Fantasy|Classics|Adventure|Children	4
Harry Potter and the Goblet of Fire	J.K. Rowling, Mary GrandPrÇ¸	4.53	1753043	Fiction|Fantasy|Classics|Adventure|Children	4
Harry Potter and the Prisoner of Azkaban	J.K. Rowling, Mary GrandPrÇ¸, Rufus Beck	4.53	1832823	Fiction|Fantasy|Classics|Adventure|Children	4
The Return of the King	J.R.R. Tolkien	4.51	463959	Fiction|Fantasy|Classics|Novel|Adventure	4
Fullmetal Alchemist Vol. 1	Hiromu Arakawa/Akira Watanabe	4.50	111091	Fiction|Fantasy|Adventure	3
The Complete Sherlock Holmes	Arthur Conan Doyle	4.50	109754	Fiction|Classics|Mystery|Thriller|Adventure|Crime	4
The Hobbit	Chuck Dixon, J.R.R. Tolkien, David Wenzel, Sea...	4.48	155338	Fiction|Fantasy|Classics|Novel|Adventure	4
Harry Potter and the Sorcerer's Stone	J.K. Rowling/Mary GrandPrÇ¸	4.47	1407778	Fiction|Fantasy|Classics|Adventure|Children	4
Harry Potter and the Philosopher's Stone	J.K. Rowling	4.47	3852657	Fiction|Fantasy|Classics|Adventure|Children	4
The Lord of the Rings	J.R.R. Tolkien	4.47	389054	Fiction|Fantasy|Classics|Novel|Adventure	4
Harry Potter and the Order of the Phoenix	J.K. Rowling, Mary GrandPrÇ¸	4.46	1735368	Fiction|Fantasy|Classics|Adventure|Children	4
Harry Potter and the Philosopher's Stone	J.K. Rowling, Mary GrandPrÇ¸	4.44	4602479	Fiction|Fantasy|Classics|Novel|Adventure	4
The Green Mile	Stephen King	4.42	173950	Fiction|Fantasy|Classics|Mystery|Science Ficti...	5
The Two Towers	J.R.R. Tolkien	4.42	480446	Fiction|Fantasy|Classics|Novel|Adventure	4

Figure 4.3: Top 20 Generalized Book Recommendations for User 15.

When we compared the proposed method approach Target domain recommendations with the existing approach (CD-SPM) by using their dataset, it looks like this:

```
recommended_books
```

```
ay(['The Time Machine', 'Angels and Demons', 'World War Z',
     'Jungle Tales of Tarzan', 'Dracula'], dtype=object)
```

Figure 4.4: Existing approach CD-SPM Target domain recommendation for user 15

When we compare the results at the Target domain of proposed approach with existing approach, we can clearly see that the existing approach is much more detailed and well explained. It provides much better understanding of the Books being recommended to the user. It takes into account the number of books occurring in a sequence but also it takes into account the average ratings of the book along with total rating count.

Chapter 5

CONCLUSION AND THE FUTURE WORK

A Cross Domain Item based Book Recommendation system was proposed in this thesis. The Proposed approach provided improved results over the existing CD-SPM approach.

Recommendations systems have been around more than a decade and a lot of research has been done in Single domain recommendation systems. But Cross Domain is relatively a new field and not much work has been done in this field so far. Cross Domain Recommendation systems opens new opportunities for advancement and development in the world of E-commerce, Social media, online businesses, Entrainment industry etc. It opens new doors and research ideas in the domain of Machine learning, Deep Learning and AI.

In the way forward, the proposed approach can be further worked upon as well. In this approach the semantic similarity between Movies and Books Genres was computed with the help Wpath similarity. To further work upon this, we can establish the link between Movies and Books by taking into account the summary of the Movies and Books. We can also take into account that A lot of movies are based upon the books by the same name. So in the way forward, this approach can further make Cross Domain Item based Book recommendations even better. It can also be sold as a complete product to different companies as well.

BIBLIOGRAPHY

[1] G. Adomavicius and A. Tuzhilin, "Toward the next generation of recommender systems: A survey of the state-of-the-art and possible extensions,"*IEEE Trans. Knowl. Data Eng.*, vol. 17, no. 6, pp. 734-749, 2005.

[2] P.G. Campos, F. Dez and I. Cantador, "Time-aware recommender systems: a comprehensive survey and analysis of existing evaluation protocols," *User Modeling and User-Adapted Interaction*, vol. 24, no. 1-2, pp. 67-119, 2005.

[3] F. Ricci, L. Rokach, and B. Shapira, "Introduction to recommender systems handbook", In *Recommender systems handbook*, New York, NY, USA: Springer, pp. 1-35, 2011.

[4] Y. Koren and R. Bell, "Advances in collaborative filtering," In *Recommender Systems Handbook*, New York, NY, USA: Springer, pp. 145-186, 2011.

[5] Y. Koren, R. Bell, and C. Volinsky, "Matrix factorization techniques for recommender systems," *Computer*, vol. 42, no. 8, 2009.

[6] B. Sarwar, G. Karypis, J. Konstan, and J. Riedl, "Item-based collaborative filtering recommendation algorithms", *In Proceedings of the 10th international conference on World Wide Web*, pp. 285-295, 2001.

[7] X. Su and T. M. Khoshgoftaar, "A survey of collaborative filtering techniques," *Advances in artificial intelligence*., vol. 2009, pp. 4, 2009.

[8] J. Wang, A. P. de Vries, and M. J. T. Reinders, "Unifying user-based and item-based collaborative filtering approaches by similarity fusion," in *Proc. 29th Annu. Int. ACM SIGIR Conf. Res. Develop. Inf. Retr.*, pp. 501-508, 2006.

[9] M. Balabanovi, and Y. Shoham, "Fab: content-based, collaborative recommendation", *Communications of the ACM*, vol. 40, no. 3, pp.66-72, 1997.

[10] J. Bennett, C. Elkan, B. Liu, P. Smyth, and D. Tikk, "Kdd cup and workshop 2007", *ACM SIGKDD Explorations Newsletter*, vol. 9, no. 2, pp. 51-52, 2007.

[11] D. Agarwal, and B. C. Chen, "Regression-based latent factor models", In *Proceedings of the 15th ACM SIGKDD international conference on Knowledge discovery and data mining*, pp. 19-28, 2009.

[12] D. Zhang, C. H. Hsu, M. Chen, Q. Chen, N. Xiong, and J. Lloret, "Cold-start recommendation using bi-clustering and fusion for large-scale social recommender systems", *IEEE Transactions on Emerging Topics in Computing*, vol. 2, no. 2, pp. 239-250, 2014.

[13] J. Das, P. Mukherjee, S. Majumder, and P. Gupta, "Clustering-based recommender system using principles of voting theory". In *Contemporary Computing and Informatics (IC3I), 2014 International Conference on*, pp. 230-235, 2014.

[14] F. O. Isinkaye, Y. O. Folajimi, and B. A. Ojokoh "Recommendation systems: Principles, methods and evaluation", *Egyptian Informatics Journal*, vol. 16, no. 3, pp. 261-273, 2015.

[15] M. Balabanovi, and Y. Shoham, "Fab: content-based, collaborative recommendation", *Communications of the ACM*, vol. 40, no. 3, pp. 66-72, 1997.

[16] D. Joaquin, I. Naohiro, and U. Tomoki, "Content-based collaborative information filtering: Actively learning to classify and recommend documents", *Cooperative Information Agents II Learning, Mobility and Electronic Commerce for Information Discovery on the Internet*, pp. 206-215, 1998.

[17] P. Lops, M. De Gemmis and G. Semeraro, "Content-based recommender systems: State of the art and trends", In *Recommender systems handbook*, pp. 73-105, 2011.

[18] X.Su, and T.M. Khoshgoftaar, "A survey of collaborative filtering techniques", *Advances in artificial intelligence*, vol. 2009, pp. 4, 2009.

[19] T. Hofmann, "Latent semantic models for collaborative filtering", *ACM Transactions on Information Systems*, vol. 22, no. 1, pp. 89-115, 2004.

[20] D. Billsus and M. Pazzani, "Learning collaborative information filters", In*Proceedings of the 15th International Conference on Machine Learning (ICML 98)*, 1998.

[21] B. M. Sarwar, G. Karypis, J. A. Konstan, and J. Riedl, "Itembased collaborative filtering recommendation algorithms", In*Proceedings of the 10th International Conference on World Wide Web (WWW 01)*, pp. 285-295, 2001.

[22] B. M. Sarwar, G. Karypis, J. A. Konstan and J. Riedl, "Recommender systems for large-scale E-commerce: scalable neighborhood formation using clustering", In *Proceedings of the 5th International Conference on Computer and Information Technology (ICCIT 02)*, 2002.

[23] M. Deshpande and G. Karypis, "Item-based top-N recommendation algorithms", *ACM Transactions on Information Systems*, vol. 22, no. 1, pp. 143-177, 2004.

[24] A. Popescul, D. M. Pennock, and S. Lawrence, "Probabilistic models for unified collaborative and content-based recommendation in sparse-data environments", In *Proceedings of the Seventeenth conference on Uncertainty in artificial intelligence*, pp. 437-444, 2001.

[25] Y. Zhou, D. Wilkinson, R. Schreiber, and R. Pan, "Large-scale parallel collaborative filtering for the netflix prize", *Lecture Notes in Computer Science*, vol. 5034, pp. 337-348, 2008.

[26] A. Mnih, and R. R. Salakhutdinov, "Probabilistic matrix factorization", In *Advances in neural information processing systems*, pp. 1257-1264, 2008.

[27] H. Shan, and A. Banerjee, "Generalized probabilistic matrix factorizations for collaborative filtering", In *Data Mining (ICDM), 2010 IEEE 10th International Conference on*, pp. 1025-1030, 2010.

[28] R. Salakhutdinov, and A. Mnih, "Bayesian probabilistic matrix factorization using Markov chain Monte Carlo", In *Proceedings of the 25th international conference on Machine learning*, pp. 880-887, 2008.

[29] Y. Shi, M. Larson, and A. Hanjalic, "Collaborative filtering beyond the user-item matrix: A survey of the state of the art and future challenges", *ACM Computing Surveys (CSUR)*, vol. 47, no. 1, pp. 3, 2014.

[30] T. Yoneya, H. Mamitsuka, "Pure: a pubmed article recommendation system based on content-based filtering", *Genome informatics. International Conference on Genome Informatics*, vol. 18, pp. 267-276, 2007.

[31] D. Billsus and M. J. Pazzani, "User modeling for adaptive news access", *User modeling and user-adapted interaction*, vol. 10, no. 2-3, pp. 147-180, 2000.

[32] R. J. Mooney and L. Roy, "Content-based book recommending using learning for text categorization", In *Proceedings of the fifth ACM conference on Digital libraries*, pp. 195-204, 2000.

[33] Zhang, Q., Wu, D., Lu, J., Liu, F., Zhang, G., 2017. A cross-domain recommender system with consistent information transfer. Decis. Support Syst. 104, 49–63

[34] Hao, P., Zhang, G., Lu, J., 2016. In: Enhancing cross domain recommendation with domain dependent tags In Fuzzy Systems (FUZZ-IEEE), 2016 IEEE International Conference on. IEEE, pp. 1266–1273S. T. Park, and W. Chu, 2009, "Pairwise preference regression for cold-start recom- mendation", In *Proceedings of the third ACM conference on Recommender systems*,pp. 21-28, 2009.

[35] Agarwal et al., 2011] Deepak Agarwal, Bee-Chung Chen, and Bo Long. Localized factor models for multi-context recommendation. In SIGKDD, pages 609–617, 2011.

[36] Fernandez-Tob ´ ´ıas et al., 2012] Ignacio Fernandez-Tob ´ ´ıas, Ivan Cantador, Marius Kaminskas, and Francesco Ricci. ´ Cross-domain recommender systems: A survey of the state of the art. In CERI, page 24, 2012.

[37] Fu et al., 2019] Wenjing Fu, Zhaohui Peng, Senzhang Wang, Yang Xu, and Jin Li. Deeply fusing reviews and contents for cold start users in cross-domain recommendation systems. In AAAI, volume 33, pages 94–101, 2019.

[38] Gao et al., 2019] Chen Gao, Xiangning Chen, Fuli Feng, Kai Zhao, Xiangnan He, Yong Li, and Depeng Jin. Crossdomain recommendation without sharing user-relevant data. In WWW, pages 491–502, 2019.

[39] Hu et al., 2019] Guangneng Hu, Yu Zhang, and Qiang Yang. Transfer meets hybrid: A synthetic approach for cross-domain collaborative filtering with text. In WWW, pages 2822–2829, 2019.

[40] Kumar et al., 2014] Anil Kumar, Nitesh Kumar, Muzammil Hussain, Santanu Chaudhury, and Sumeet Agarwal. Semantic clustering-based cross-domain recommendation. In CIDM, pages 137–141. IEEE, 2014.

[41] Li and Tuzhilin, 2020] Pan Li and Alexander Tuzhilin. Ddtcdr: Deep dual transfer cross domain recommendation. In WSDM, pages 331–339, 2020.

[42] Liu et al., 2020a] Jian Liu, Pengpeng Zhao, Fuzhen Zhuang, Yanchi Liu, Victor S Sheng, Jiajie Xu, Xiaofang Zhou, and Hui Xiong. Exploiting aesthetic preference in deep cross networks for cross-domain recommendation. In TheWebConf, pages 2768–2774, 2020

[43] [Loni et al., 2014] Babak Loni, Yue Shi, Martha Larson, and Alan Hanjalic. Cross-domain collaborative filtering with factorization machines. In European conference on information retrieval, pages 656–661. Springer, 2014.

[44] [Lu et al., 2018] Yichao Lu, Ruihai Dong, and Barry Smyth. Why i like it: multi-task learning for recommendation and explanation. In RecSys, pages 4–12, 2018.

[45] Singh and Gordon, 2008] Ajit P Singh and Geoffrey J Gordon. Relational learning via collective matrix factorization. In SIGKDD, pages 650–658, 2008.

[46] Tan et al., 2014] Shulong Tan, Jiajun Bu, Xuzhen Qin, Chun Chen, and Deng Cai. Cross domain recommendation based on multi-type media fusion. Neurocomputing, 127:124–134, 2014.

[47] [Wang and Lv, 2020] Jiaqi Wang and Jing Lv. Tag-informed collaborative topic modeling for cross domain recommendations. Knowledge-Based Systems, 203:106–119, 2020

[48] [Wang et al., 2016] Beidou Wang, Martin Ester, Yikang Liao, Jiajun Bu, Yu Zhu, Ziyu Guan, and Deng Cai. The million domain challenge: Broadcast email prioritization by cross-domain recommendation. In SIGKDD, pages 1895–1904, 2016.

[49] Wang et al., 2019a] Shoujin Wang, Liang Hu, Yan Wang, Longbing Cao, Quan Z Sheng, and Mehmet Orgun. Sequential recommender systems: challenges, progress and prospects. In IJCAI, pages 6332–6338, 2019

[50] [Xue et al., 2017] Hong-Jian Xue, Xinyu Dai, Jianbing Zhang, Shujian Huang, and Jiajun Chen. Deep matrix factorization models for recommender systems. In IJCAI, pages 3203–3209, 2017.