A Comparison of Traditional Approach and Deep Learning Approach to E-learning Recommender Systems

MSc Research Project
Data Analytics

Naga Venkata Satyasainath Pulaparthi
x17101778

School of Computing
National College of Ireland

Supervisors:
Dr. Dympna O Sullivian
Dr. Pramod Pathak
Dr. Paul Stynes

National College of Ireland
Project Submission Sheet – 2017/2018
School of Computing

Student Name:	Naga Venkata Satyasainath Pulaparthi
Student ID:	x17101778
Programme:	Data Analytics
Year:	2018
Module:	MSc Research Project
Lecturer:	Dr. Dympna O Sullivian, Dr. Pramod Pathak, Dr. Paul Stynes
Submission Due Date:	13/08/2018
Project Title:	A Comparison of Traditional Approach and Deep Learning Approach to E-learning Recommender Systems
Word Count:	7100

I hereby certify that the information contained in this (my submission) is information pertaining to research I conducted for this project. All information other than my own contribution will be fully referenced and listed in the relevant bibliography section at the rear of the project.

ALL internet material must be referenced in the bibliography section. Students are encouraged to use the Harvard Referencing Standard supplied by the Library. To use other author's written or electronic work is illegal (plagiarism) and may result in disciplinary action. Students may be required to undergo a viva (oral examination) if there is suspicion about the validity of their submitted work.

Signature:	
Date:	10th August 2018

PLEASE READ THE FOLLOWING INSTRUCTIONS:

1. Please attach a completed copy of this sheet to each project (including multiple copies).
2. **You must ensure that you retain a HARD COPY of ALL projects**, both for your own reference and in case a project is lost or mislaid. It is not sufficient to keep a copy on computer. Please do not bind projects or place in covers unless specifically requested.
3. Assignments that are submitted to the Programme Coordinator office must be placed into the assignment box located outside the office.

Office Use Only	
Signature:	
Date:	
Penalty Applied (if applicable):	

A Comparison of Traditional Approach and Deep Learning Approach to E-learning Recommender Systems

Naga Venkata Satyasainath Pulaparthi
x17101778
MSc Research Project in Data Analytics

10th August 2018

Abstract

Nowadays, with rapid growth in learning material and available internet access have made it easy to gain knowledge from online. But, the biggest challenge in e-learning education is how to teach learners effectively. Recommender systems are the most common information filtering systems in several domains to suggest relevant items to users. In the context of an e-learning platform, recommender system is an agent that suggests learners with learning courses based on their interests and previous behavior. The solution to effective teaching problem is preference elicitation that suggests learners based on their desired characteristics. The main objective of this research is to build an accurate recommender system model using a unique deep learning approach, Restricted Boltzmann Machines(RBM) and then compare with successful Matrix Factorization techniques. Our experiment proved that RBM outperformed Singular Value Decomposition(SVD) by 1.6% and Probabilistic Matrix Factorization(PMF) by 5.7%. These results will help data scientists to apply deep learning approach to e-learning recommender platforms.

Index terms— Deep learning, collaborative filtering, E-learning recommender systems, Restricted Boltzmann Machines,Matrix Factorization

Contents

1 Introduction

The massive growth in the number of available learning courses in e-learning sites created a challenge of information overload on learners which consumes time to find relevant material of interest. Comparing with the traditional way of teaching and classroom style, e-learning became a revolutionary way in education by attracting many learners to gain knowledge easily (Oliveira; 2016). With the increase in technology and usage of internet, these platforms became more convenient for many learners to educate themselves effectively. In a survey, it has provided that several learners are preferring e-learning than traditional teaching style and gaining benefits from it on the various types of courses(Sikka et al.; 2012). Most of the e-learning platforms are generalized sites which suggest common trend courses to all learners. However, these non-personalized platforms face a challenge with the model it follows in which generalized learning courses are provided to all learners(Sikka et al.; 2012). It is fact that every learner could have own interests in learning course and may also have a different level of expertise (Martnez et al.; 2014). So, a personalized recommender system is must needed to attract learners and also provide useful learning material or courses.

1.1 Background and Research Motivation

Recommender systems are well known for its common usage in domains like e-commerce (Amazon) (Linden et al.; 2003), music, travel, and movie(Netflix)(Zheng; 2016). 80 percent of Netflix movies watched came from recommender systems[1], 60 percent of Youtube video clicks came from Recommender systems(Zhou et al.; 2010). These recommender systems are widely used in e-commerce but not explored into depth in the e-learning domain. Usage of these recommender systems is not new but the era has started over two decades using non-personalized systems with suggestions to learners based on general trend items. As stated, these non-personalized recommender systems suggest items to all users irrespective of demographic content or any other characteristics consideration. The main role of any recommender system is to analyze data and predict the preference of user given by data intelligently. Technically, recommender systems can be implemented using collaborative filtering, content-based and hybrid techniques. But the main motive of any technique is to predict accurate preference of learners to suggest relevant courses(Bobadilla et al.; 2013).

From the above-mentioned techniques, the collaborative filtering technique is the most widely used technique in companies which can be broadly divided into memory based (uses similarity) and model-based(machine learning) techniques(Bobadilla et al.; 2013). Memory-based approaches find similar learners based on cosine similarity or Pearson correlation but the main disadvantage of this approach is its non-scalability and reduces performance on sparse data. As most of the recommender systems data is sparse data, dimensionality reduction techniques of machine learning can deal with it and model-based techniques like singular value decomposition(SVD), matrix factorization or neural nets of collaborative filtering are effective. In-detail description of these techniques is provided in section 2.2.

A lot of research work has been done on collaborative filtering using matrix factorization

[1]https://www.wired.co.uk/article/how-do-netflixs-algorithms-work-machine-learning-helps-to-predict-what-viewers-will-like

techniques which is the most effective technique among them(Bokde et al.; 2015). But, e-learning platform recommender systems are different from other domain systems as learners may have variant features like interests, background, style, and level of expertise which can be hidden features that effect on learner preference(Oliveira; 2016). In order to use these hidden features incorporated between learner and course, a deep trained model is required but very less research work done on how to use deep learning approach for e-learning collaborative filtering systems.
On the other hand, deep learning has shown tremendous results in image processing, speech recognition, and natural language processing(Zheng; 2016). Recently, many companies are switching to deep learning approach because of its significant performance improvement than traditional methods. Deep learning proved that it works better than traditional methods on large datasets to provide better results and performance (Zhang et al.; 2018).

1.1.1 Research Question

It is important to provide accurate and relevant courses to learners while reducing selection overload in e-recommender systems. Preference elicitation is a process of discovering accurate preferences using a model that find hidden patterns and avoid redundancy[2]. How accurate courses are recommended to learners is the biggest challenge in any e-learning recommender system. This research work investigates the accuracy and results of Singular Value Decomposition(SVD) and Probabilistic Matrix Factorization(PMF) of the traditional approach and compares with Restricted Boltzmann machines(RBM) of deep learning in e-learning recommender systems.

1.1.2 Business UseCase

This research work will help the data scientists whose work is on e-learning recommender systems. In a business perspective, an e-learning website can be integrated with an outperformed model of this research work to get more accurate preferences of learners. This will intelligently predict the preferred course of learners for better suggestions. This model can be used as the backbone in finding relevant courses to learners which will reduce the overload of selection.

1.1.3 Research Project Objectives

The most important thing in collaborative filtering branch is that almost all datasets are sparse dataset which creates problems in predicting the accurate predictions. Either it is 5 rating scale system like 1-5 movie rating or binary scale like course completed or not, error reduction in prediction is the primary task of any machine learning model in process of accurate predictions. Matrix factorization techniques like SVD 3.4.1 and PMF 3.4.1 use stochastic gradient descent function to reduce the error of model trained whereas unique deep learning approach Restricted Boltzmann Machines(RBM) use a gradient function called contrastive divergence. The main objective of this research work is to investigate the deep learning approach model in comparing it with the current successful models in existence. The research work is divided into objectives at stages as shown below.

[2]https://en.wikipedia.org/wiki/Preference-elicitation

Objective 1: Build SVD and PMF traditional models and compare them using evaluation metrics RMSE and MAE.
Objective 2: Build RBM model and compare it with the outperformed model of objective 1 using the same evaluate metrics.
Objective 3: Evaluate three model built beyond accuracy using precision and recall metrics.
Objective 4: Error reduction analysis of contrastive divergence(CD) on outperformed model using RMSE and MAE.

1.1.4 Research Contributions

This project is significant because the results will enable to provide insightful information to data science community regarding preference elicitation accuracy and performance of which Restricted Boltzmann machines model outperformed Singular Value Decomposition and Probabilistic Matrix Factorization models. This report also provides a glimpse of various techniques applied to e-learning recommender systems.

1.1.5 Report Structure

The rest of the technical report is structured as follows. Chapter 2 presents an investigation of the related work on recommender systems using different approaches. Chapter 3 will describe the methodology used. Chapter 4 contains implementation and chapter 5 with evaluation. Finally, Chapter 6 of the report concludes research work with a conclusion and future work.

2 Related Work

2.1 Recommender Systems Approaches Overview

Research work on recommender systems has been going along with the web in parallel (Bobadilla et al.; 2013). These systems were started with non-personalized content in websites, incorporated with social information and will be used widely in the Internet of things in the future (Bobadilla et al.; 2013). In the context of an e-learning platform, these systems act as an agent that reduces the course selection overload on the learner.
Technically, recommender systems are classified according to the technique used for domain requirement. They are generally classified as collaborative filtering, content-based and hybrid techniques. Several techniques can be implemented to build recommender systems but the most effective technique is collaborative filtering techniques (Sielis et al.; 2012). This research paper focuses on collaborative filtering techniques which compare the traditional methods and deep learning method of collaborative filtering.
In this division, section 2.2 briefly describes various approaches on recommender systems. section 2.3 discuss related work on collaborative filtering traditional approaches to recommender systems and section 2.4 provides information on related work using deep learning approach.

2.2 Approaches to Recommender Systems

Techniques of Recommender systems:

- Collaborative Filtering
- Content Based Systems
- Hybrid Systems

As mentioned above, recommender systems techniques are classified under collaborative filtering(CF), content-based(CB) and hybrid systems. Content-based techniques of recommender systems use previous actions of the learner and description of that course interested by the learner for suggestions (Pazzani and Billsus; 2010). These systems suggest a learner when a new course shows up that match the characteristics desired by the learner (Oliveira; 2016). It is a very effective technique when all features of learner are considered but the major limitation is learners are not suggested courses which are liked by other learners who have similar characteristics (Pazzani and Billsus; 2010).
Hybrid systems are the combination of collaborative filtering(CF) and content based(CB) technique in which researchers try to smoothen the disadvantage of CF and CB (Sharma and Mann; 2013). These two mentioned techniques require explicit data and content feature data but our research work is focused on implicit data. The major concern of this research is to find the latent similarities between learners so that a certain number of courses can be suggested among learners without using explicit data. Another disadvantage of hybrid is dimensionality reduction in process of finding latent features among learners (Sharma and Mann; 2013). Recommender systems is a broad research area and the main objective of this research work is to focus on collaborative filtering technique to find hidden features and suggest learners accurately and effectively. Section 2.3 discusses the related work in collaborative filtering branch of recommender systems. Figure 1 shows the overview of approaches to recommender systems.

2.3 Collaborative Filtering Techniques

Categories of Collaborative Filtering:

- Memory Based approach
- Model-Based approach

Collaborative filtering technique is the most popular and commonly used technique to build recommender systems Zheng (2016). The main advantage of collaborative filtering is that it is not required to analyze the content of course when trying to find "like-minded" learners(Miyahara and Pazzani; 2000). It follows the observation that users would prefer an item based on other users who have a similar taste. For instance, if user X and user Y likes item I1 and user Y prefers I2, then user X most likely to prefer I2 because of similar taste (Zheng; 2016). In this context, user is a learner and course is an item. Collaborative filtering technique is sub-classified or categorized into memory based2.3.1 and model-based2.3.2 techniques which are discussed in further sections.

2.3.1 Memory Based-Collaborative Filtering

In this technique, course preferences are computed and suggested using similarity values. Preference value given by learners like rating is used to compute the weights and similarity between learners(Aditya et al.; 2016). The main advantage of this technique is easy implementation because of only similarity calculation. Neighborhood approach is the representative of this category which uses only a set of close users to compute similarity. Linden et al. (2003) did research

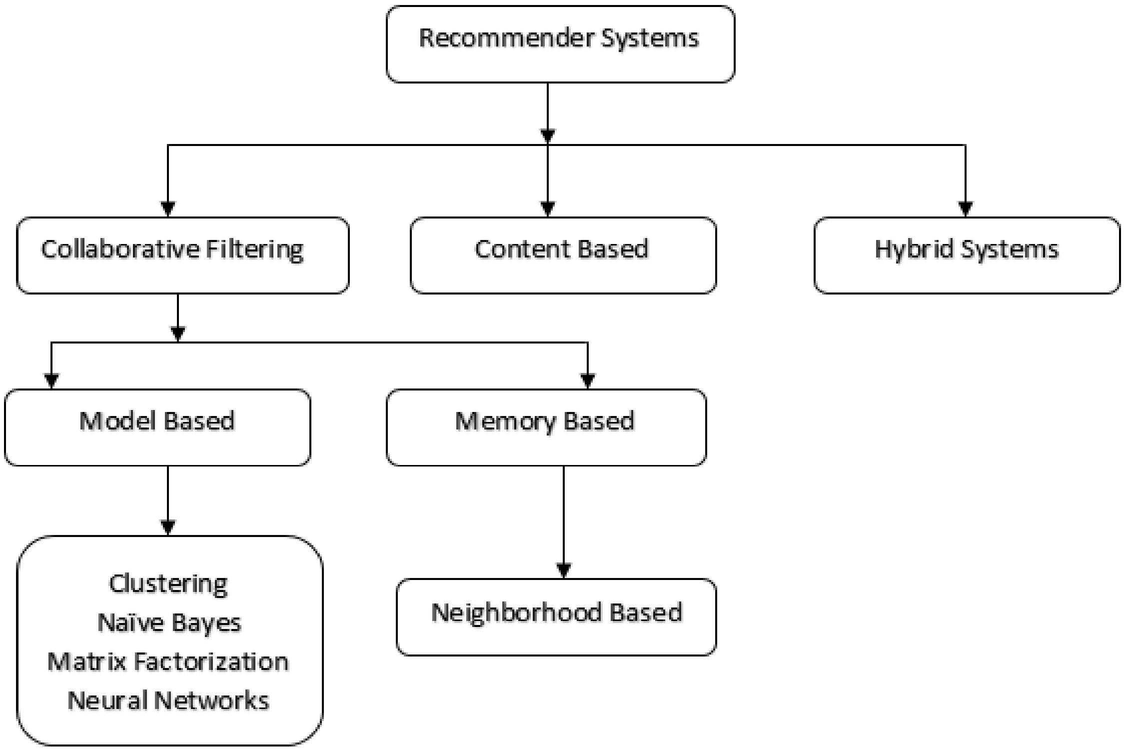

Figure 1: Techniques for Recommender Systems

on neighborhood approach which involves recommendations in two steps: similarity calculation and prediction of preferences. Generally, similarities are calculated using Pearson coefficient considering N nearest neighbors and use that similarity in predicting the preferences (Linden et al.; 2003). But, the main disadvantage of this model approach is inaccurate and unreliable predictions when data is largely sparse. But, most of the recommender system datasets are sparse dataset and to overcome this limitation, several researchers choose model-based techniques of collaborative filtering which are discussed in section 2.3.2.

2.3.2 Model Based-Collaborative Filtering

This category of collaborative filtering is an interesting area that attracted several researchers because of its implementation using machine learning models. There are several approaches to predict and recommend courses to learners using the model-based technique but applying a suitable model for a Recommender system is a challenge because of various models available and also domain requirements. These machine learning models provide intelligent results with its capability and overcome the limitations of memory-based models(Amatriain et al.; 2011). The initial approach using machine learning model on recommender systems started with (Miyahara and Pazzani; 2000) using Naive Bayes (NB) model on a binary matrix and outperformed memory based technique.
Clustering is another known and simple model of unsupervised machine learning which can be used as an intermediate step in collaborative filtering. In this technique, Recommender systems can categorize the data into related groups and the probability can be calculated using its group information (Ungar and Foster; 2002). All these simple models use the existing features to find the similarities among learners but there can be some hidden features which can't be interpreted by humans easily. these features can be anything related to characteristics like background, style, level of expertise, interest, etc,.

To find above mentioned like latent features among learners or users in collaborative filtering, (Koren et al.; 2009) have come up with Matrix Factorization(MF) technique. In the context of E-learning domain, these hidden factors can be the interest, qualification, education background, or pattern style of study which may affect the preference of course. Matrix factorization technique factorizes the original matrix into two matrices which are latent factors of learners and courses. For an instance, a matrix R in which n learners in rows and v courses in columns and values as their preference values. Matrix D is factorized in W and T which are latent factors of learners and courses respectively as shown in Figure2 [3]. Matrix factorization techniques can be Singular Value Decomposition(SVD) and Probabilistic Matrix Factorization(PMF) in collaborative filtering.

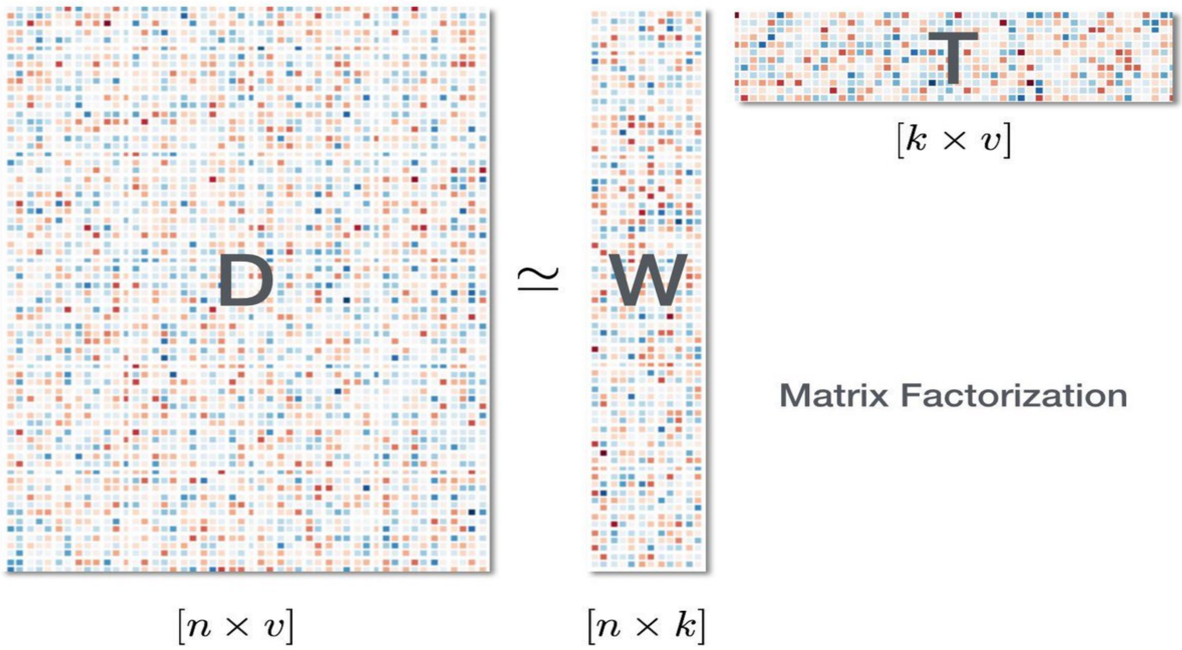

Figure 2: Matrix Factorization

$$D = W * T$$

All these matrix factorization techniques are famous for finding hidden patterns and also considering the importance of each factor to the learner and course. Singular Value Decomposition(SVD) technique factorize the original matrix into two or more matrices so that the original matrix is obtained by multiplying them as shown in Figure 2. Probabilistic Matrix Factorization(PMF) is also the same as SVD as a base algorithm but it finds the probability of learner picking a random course that belongs to a random latent feature. In this case, D is the original matrix, W and T are factorized as Probabilistic matrices (Mnih and Salakhutdinov; 2008). Matrix factorization provided solutions to several limitations which other models are not capable of solving and these techniques became the most successful and effective technique of collaborative filtering (Zheng; 2016). Section 2.4 discusses the success towards deep learning for collaborative filtering.

[3]http://www.mattmoocar.me/recsys/

2.4 Deep Learning

Deep learning is a part of machine learning which uses deep hidden layers while training a model. Models mentioned in the above section are shallow architecture networks which are a dominant and good performance in 90's (Zheng; 2016). Recent research works suggest that with the increase in a large amount of data, deep architecture is needed to train a better model. Deep learning has shown massive results in image processing, text analytics and audio recognition (Graves et al.; 2013). Deep learning plays a prominent role in the recommender system and youtube is the best example of deep learning usage in recommender systems. (Covington et al.; 2016) provided usage of deep learning in youtube recommender systems and its prominent role. Several researchers are interested in deep learning usage on recommender systems as very less research work has been done.Section 2.4.1 discuss deep learning for collaborative filtering.

2.4.1 Towards Deep Learning-Collaborative Filtering

With the evolution of deep learning, researchers have started their work to solve problems and train better model for recommender systems using deep learning. several attempts have made to achieve but for a successful implementation, a complete observed data is required. The challenge of missing values in the sparse matrix to work on neural networks was first tackled by (Salakhutdinov et al.; 2007). In (Salakhutdinov et al.; 2007) research work, the researcher has proved that Boltzmann machines works well on a large sparse dataset and used MovieRulz dataset as a benchmark. Although still there is an issue of the cold-start problem, the initial step brought success towards a deep learning approach. Cold-start problem is an issue in recommender systems which is incapable of recommendations to new users because of no prior information or actions(Pandey and Rajpoot; 2016). This limitation was later addressed by(Strub et al.; 2016) using Auto-encoders which uses explicit and implicit data.
Most of the approaches are implemented to replace matrix factorization techniques, which is a representative of the model-based technique of collaborative filtering. Although many researchers are working on deep learning approaches, most of them are 1-5 rating scale dataset which is commonly considered in e-commerce, movies sites(Zhang et al.; 2007). In the context of an e-learning platform, some recommender systems should be capable of considering implicit data to suggest learners based on their behavior without any explicit data like rating. This makes systems not to rely on explicit data to recommend courses. In our research work, we consider binary data based on learner completed the course or not and find similarities between learners to suggest more courses.
This paper investigates and suggests a deep learning approach to e-learning recommender systems which will support to recommend accurate courses to learners accurately.

3 Methodology

This research paper is based on knowledge and data discovery(KDD) for recommender systems and this section contains information about the methodology used.

3.1 Business Understanding

E-learning platform requires an intelligent agent to suggest relevant courses to learners in process of selection overload. For this, an accurate machine learning model for recommender systems is needed which will intelligently recommend these relevant courses to learners. Generally, recommender systems consider explicit data provided by users like rating in 1-5 scale but sometimes recommender system should consider implicit data by observing the behavior of learner. This research work does not require any explicit data and model is trained based on previous actions of learners.

3.2 Data Selection and Understanding

In process of contribution to e-learning domain, data chosen for this experiment is from an On-line source which was provided by McKinsey company as part of the recommender systems design challenges[4]. This dataset consists of approximately 1 million observations with nearly 69532 learners and 5348 courses. Below table shows the attribute information of the dataset. As already mentioned, this experiment does not require any explicit data and all latent hidden features are observed in the training model process.

3.3 Data Preprocessing and Transformation

As part of implementation work, data provided in tabular form is converted into matrix form which is suitable to build recommender systems. Now, the matrix form of data looks like all learners in rows and all courses in columns and the values of this matrix is in binary form whether course completed by the particular learner(1) or not(0). This matrix table is just a representation of transformed matrix 2. A further step of the experiment is explained in section 3.4.

Attribute	DataType
LearnerID	numeric
CourseSequences	int
CourseID	numeric
Completed	Categorical

Table 1: Glimpse of Data Attributes

$$
\begin{array}{c}
\begin{array}{ccccc} course1 & course2 & course3 & course4 & course5 \end{array} \\
\begin{array}{c} learner1 \\ learner2 \\ learner3 \\ lerner4 \\ learner5 \end{array}
\begin{bmatrix}
1 & 1 & 0 & 1 & 1 \\
0 & 1 & 1 & 0 & 0 \\
0 & 0 & 1 & 0 & 0 \\
0 & 1 & 0 & 1 & 1 \\
1 & 0 & 0 & 0 & 1
\end{bmatrix}
\end{array}
$$

Table 2: Transformed Matrix Data

3.4 Data Mining

As the transformed matrix now contains 69532 rows and 5348 columns, it is necessary to reduce the feature space for ease of computation. So, Matrix Factorization techniques are required to reduce the dimensions of a matrix for better computational benefits. Dataset is divided into train and test datasets so that test dataset can be used to evaluate the performance of models trained on train dataset. We train the machine learning model on train dataset and use the test dataset to predict the preferences using the trained model. we then calculate the loss function which is the difference between actual values and predicted values of the test dataset. This tells us how effectively our trained model will work on unseen data in future. In this experiment, we built three recommender system models, Singular Value Decomposition(SVD)3.4.1, Probabilistic Matrix Factorization(PMF)3.4.1 and Restricted Boltzmann machines(RBM)3.4.2.

[4]https://datahack.analyticsvidhya.com/contest/mckinsey-analytics-online-hackathon-recommendation/

3.4.1 Matrix Factorization

The most effective and important technique of collaborative filtering is matrix factorization because of its work on latent hidden features and underlying interactions between learners and courses. The basic idea of matrix factorization technique is to factorize the original matrix into at least two matrices and the same original matrix can be obtained by multiplying these matrices. We built SVD 3.4.1 and PMF 3.4.1 models of matrix factorization technique as a traditional approach for this experiment.

Singular Value Decomposition(SVD)

Considering a matrix $n * v$ D with n learners and v courses is factorized into W and T as shown in Figure 2. In a binary rating recommender systems, values are Boolean whether the user likes the item or not as shown in Table 2. The main motive of SVD is to discover K underlying features and also to find matrices W and T such that multiplication of W and T provides original matrix D.

$$D = W * T$$

Probabilistic Matrix Factorization(PMF)

PMF is also similar to SVD in finding latent features but uses the probability that learner would prefer the course or not. The basic idea of PMF is to predict the probability that learner picks a latent random feature and a random course belongs to that feature(Mnih and Salakhutdinov; 2008). For instance, learner prefers a random latent feature like Data science and picks a random data mining course that belongs to data science feature and PMF predicts the probability that this learner prefers it.

$$P\left(i/u\right) = \sum_{Z} P\left(i/z\right) P\left(z/u\right)$$

where,$P\left(i/u\right)$ is the probability of learner i picking course u,$P\left(z/u\right)$ is learner picks a random latent feature, $P\left(i/z\right)$ is that course belongs to that latent feature.

To make the model more accurate, the difference between predicted preference value and real preference value should be minimum and stochastic gradient descent method is used to minimize the error value 3.4.1. The difference between actual value and the predicted value is usually referred to as an error or loss function.

Stochastic Gradient Descent(SGD)

Stochastic gradient descent is a method to find the optimal values in order to reduce the error calculated during prediction [5]. SGD is the loss function used in above mentioned SVD and PMF and finds the optimal values based on the gradient at current values and moves towards the minimum error approaching(Li et al.; 2014).

We also trained a deep learning approach in this experiment and section 3.4.2 explains about this approach.

3.4.2 Restricted Boltzmann Machines(RBM)

There is a lot of research work available on recommender systems using matrix factorization. But very less research work done on these systems using deep learning. It can be considered as an extension work of matrix factorization in collaborative filtering. Restricted Boltzmann machines are undirected neural networks that learn probability distribution from its input. The main reason for the implementation of this network is its unique process of learning. Unlike other neural networks, Restricted Boltzmann machines have no connections between hidden neurons and also between input neurons. This network consists of only two layers with visible

[5]https://en.wikipedia.org/wiki/Stochastic-gradient-descent

layer and hidden layer. RBM is comprised of multiple visible neurons $v = \{v0, v1, v2, v3\}$ and hidden neurons $h = \{h0, h1, h2\}$ with no connection links of same layer as shown in Figure 3. These RBMs are more powerful enough to observe hidden features which increase the capacity of model[6]. These hidden features are background, style, interest, level of expertise of the learner, etc,. in our e-learning context. All courses are inputs of input visible layer for each neuron. Unlike other approaches, it uses Contrastive Divergence(CD) approach to minimize the predicted error.

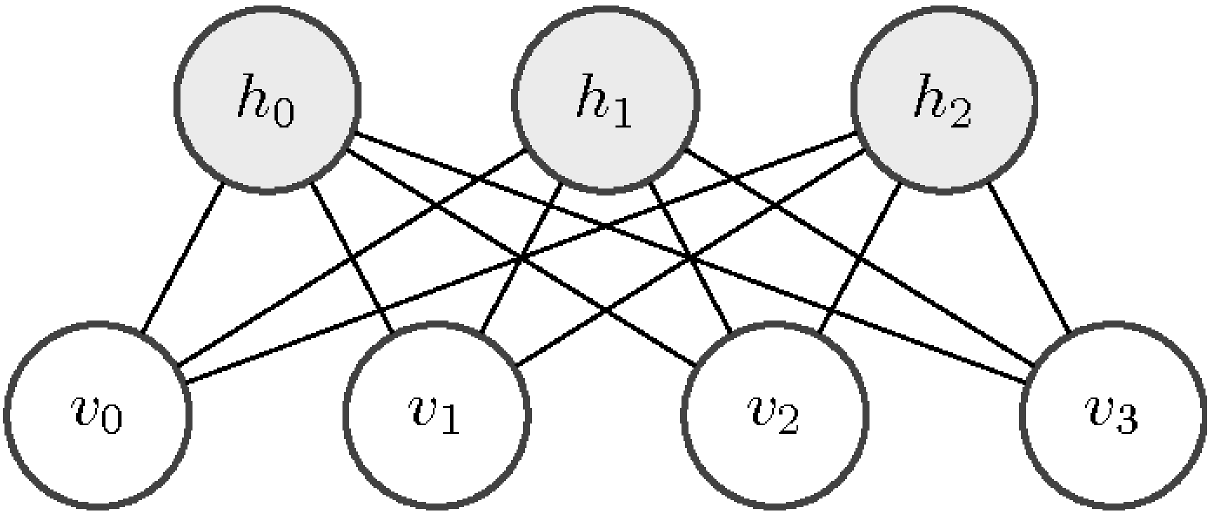

Figure 3: Restricted Boltzmann Machine(RBM)

Contrastive Divergence(CD)
CD is a gradient function used to train probabilistic undirected graphical model like RBM6 . It minimizes the predicted error using Gibb's sampling technique Fischer and Igel (2012). For this experiment, we have a large number of observations and features, direct sampling of the stochastic process is difficult and the main advantage of Gibbs sampling is it helps in the sequence of observations as input to train the model(Cowell and Ghahramani; 2005).
Generally, the recommender system's models are evaluated using some important metrics to conclude how well a model is being trained. Evaluation metrics considered in this experiment are discussed in section 3.5.

3.5 Evaluation Methods

It is important to evaluate the performance of models built for recommender system using unused test dataset so that it will give an approximate future performance on unseen data. The evaluation of recommender system is an important aspect for choosing better learning algorithm(Cremonesi et al.; 2008). Predicted values are compared with the actual values of the test dataset by calculating the error value to check the quality of the model.
The quality of a recommender system can be evaluated with different types of techniques or evaluation methods. Statistical accuracy metrics is the suitable metric to evaluate the recommender system by calculating error value using predicted value and actual value. Root Mean Square Error(RMSE) and Mean Absolute Error(MAE) are usually used as the metrics to evaluate the accuracy of a recommender system(Isinkaye et al.; 2015). Lesser the value, model is more accurate.
Mean Absolute Error(MAE) is a usual metric used to evaluate recommender system because of

[6]http://deeplearning.net/tutorial/rbm.html

its easy implementation and direct interpretation. MAE can be computed as follow,

$$MAE = 1/n \sum_{i=1}^{n} |p_i - a_i|$$

Also, Root Mean Square Error(RMSE) computes the error based on the square of mean absolute value. according to this metric, lower RMSE value, better the accuracy of recommendation. It can be computed as follows,

$$RMSE = \sqrt{1/n \sum_{u,i} (p_i - a_i)^2}$$

where p_i is predicted preference value and a_i is actual preference value.
There are other metrics like Precision and Recall which are mostly used in an evaluation of Top-N Recommendations(Cremonesi et al.; 2008). These metrics are often used in e-commerce sites like Amazon listing top-N recommendations in their user interface. Precision and Recall is also an important metric for binary classification whether the user will like it or not(Cremonesi et al.; 2008). These metrics are also most suitable to evaluate our model's performance. Briefly defining, Precision is the proportion of recommended courses that are relevant to learners. The recall is the proportion of relevant courses that are found in recommended courses.

$$\boxed{\begin{aligned} \text{Recall} &= \frac{TP}{TP + FN} \\ \text{Precision} &= \frac{TP}{TP + FP} \end{aligned}} \tag{1}$$

where TP is an interesting course recommended, TN is an uninterested course not recommended, FN is an interesting course not recommended and FP is an uninterested course recommended to a learner(Cremonesi et al.; 2008). According to Herlocker et al. (2004), usually the number of courses completed by learners is much smaller than the courses available and the number of relevant courses in the validation dataset is much smaller than the whole dataset. So, these precision and recall measures are only to compare the algorithms but not to be interpreted as absolute measures.

4 Implementation

This chapter discusses the implementation of three models for the recommender system and technical specifications considered for implementation. Section 4.1 provides the information related to the technical configuration which includes tools used and packages. Section 4.2 outlines the workflow used to build these models.

4.1 Technical Configuration

Table 3 provides information about technical tools and packages used to build models.

4.2 Models Built

This workflow section describes the approaches considered to build three models of e-learning recommender system. Python programming language is the common language to implement all

Approach	SVD	PMF	RBM
Programming Language	Python 3.6	Python 3.6	Python 3.6
Preprocessing Packages	Pandas	Pandas	Pandas, Pytorch
Model Packages	Surprise	Surprise	Pytorch
Machine RAM	16 GB	16 GB	16 GB
Processor	Intel I7	Intel I7	Intel I7

Table 3: Technical Specifications Table

the models in this experiment but different packages used as mentioned in Table 3.
As part of our experiment, we built three models in which two belongs to traditional approach technique, singular value decomposition(SVD) and probabilistic matrix factorization(PMF) and one deep learning approach which is restricted Boltzmann machines(RBM).

Table 4 provides information about different models to recommender system.

Traditional Approach	Deep Learning
Singular Value Decomposition	Restricted Boltzmann Machines
Probabilistic Matrix Factorization	

Table 4: Models Built

Firstly, the experiment started with the traditional approaches singular value decomposition(SVD) and probabilistic matrix factorization(PMF) techniques. Data has been imported and transformed it to bring into suitable matrix form 2 so that it can be used to build recommender system models. Dataset was split into train and validation datasets so that model trained on train dataset can be evaluated using the validation dataset later 3.4.
SVD model trained using parameters as 100 hidden features and 6 epochs. Hidden features are the latent features which are indirectly relating between learners and courses.
Probabilistic matrix factorization(PMF) technique with the same set of SVD parameters was considered to build the model so that results are more valid to evaluate and compare. It is important to note that Stochastic Gradient Descent function(SGD)3.4.1 is used to reduce the error in these two models.

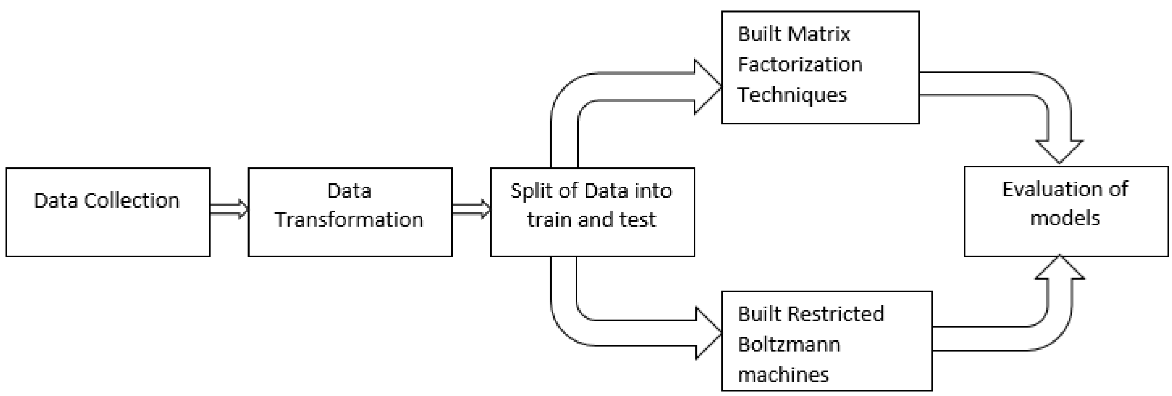

Figure 4: workflow of models built

Restricted Boltzmann Machines(RBM) is a unique approach to deep learning which uses Contrastive Divergence(CD)3.4.2 function instead of Stochastic gradient descent to reduce error.

Same like the previous model, the original transformed matrix was split into train dataset and validation dataset for further performance evaluation 3.4. The number of hidden features was set to 100 that means hidden layer contains 100 neurons and visible layer contains the number of neurons same that of courses. To speed up the training process of the model, we subdivided the training dataset into batches in training process where each batch contains 20000 learners out of 69532 learners. the model was trained in 6 epochs and weights of network updated at every epoch. Here, an epoch is a complete pass of a dataset into the model which is a complete update of a model with weights. Above diagram 4.2 is the workflow of models built.

5 Evaluation

This chapter discusses the evaluation of models built as part of research work. In this experiment, we compared Restricted Boltzmann Machines(RBM) with Probabilistic Matrix Factorization(PMF) and Singular Value Decomposition(SVD) models using general predictive evaluation metrics like Mean Absolute Error(MAE), Root Mean Square Error(RMSE) and also decision-based metrics, Precision and Recall 3.5.

5.1 Objective 1: SVD vs PMF

In our experimental approach, we first built and compared the results of two traditional approaches SVD and PMF. We investigated the performance of models based on usual evaluation metrics MAE and RMSE. The lesser the value, the greater the accuracy.

Model	Mean Absolute Error	Root Mean Squared Error
Singular Value Decomposition	0.0181536	0.0392444
Probabilistic Matrix Factorization	0.5748119	0.686356

Table 5: SVD vs PMF

In this experiment of comparison between two traditional approaches, SVD outperformed PMF with large variation providing a better accurate model. This shows that SVD technique is far better than PMF technique in the context of e-learning recommender systems which has a large number of learners and courses.

5.2 Objective 2: SVD vs RBM

In the second stage of research, we built deep learning model RBM and compared it with the results of an outperformed model of objective 1, SVD. We used the same usual metrics to compare the results and found as shown in table 6.

Model	Mean Absolute Error	Root Mean Squared Error
Singular Value Decomposition	0.0181536	0.0392444
Restricted Boltzmann machines	0.0014686	0.0294584

Table 6: RBM vs SVD

We noticed that the deep learning approach RBM outperformed traditional successful method SVD by 1.6% on huge data trained. We evaluated these three models using accuracy metrics but accuracy is not everything. It is also important to evaluate machine learning models beyond accuracy. Precision and recall are the decision-based metrics which will allow evaluating models beyond accuracy.

5.3 Objective 3: Beyond accuracy-Precision and Recall

As mentioned above, models built are evaluated using Precision and Recall to check beyond accuracy. These metrics are to be considered only to compare the algorithm but not to interpret as an absolute measure(Herlocker et al.; 2004). Precision and recall are classic binary classification evaluation metrics which are translated to help researchers in the evaluation of recommender systems. In Recommender systems context, generally, top N recommendations to a user are considered for valid evaluation of precision and recall rather than all items of users. These

Model	Precision Value	Recall Value
Singular Value Decomposition	0.6816	0.5322
Probabilistic Matrix Factorization	0.2648	0.2924
Restricted Boltzmann machines	0.8515	0.6901

Table 7: Precision and Recall

precision and Recall values are average values of all the learners that are considered for top 5 courses. In practical, Precision result values of SVD, PMF and RBM show that 68%,26%,85% courses recommended are relevant to learners respectively. whereas, 53%,29% and 69% courses that are relevant to learners have appeared in recommendations.

5.4 Objective 4: Effect of Contrastive Divergence(CD)

Taking the research work forward with the results of the above experiments, we were interested in the change of error in every epoch of the RBM model that outperformed the other two models. We evaluated the RBM model at 6 epochs and observed how Contrastive Divergence(CD) function reduces the error at different epochs using RMSE and MAE. Figures 5 and 6 are findings of evaluation metrics at 6 different epochs.

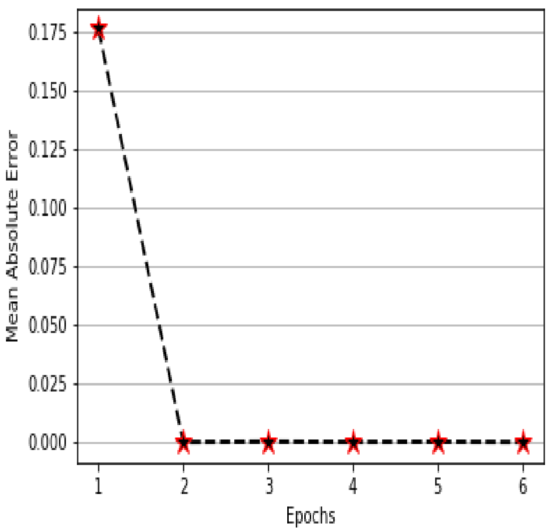

Figure 5: MAE with Epochs

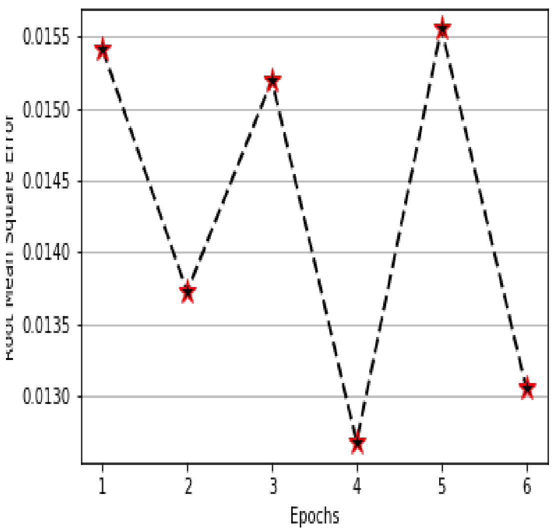

Figure 6: RMSE with Epochs

This shows that the MAE metric reduces the maximum error after 1st epoch whereas RMSE maintained a mean error value throughout the Epoch process.

5.5 Discussion

In this section, we provide a detailed discussion of the experiment results. We sub-divided this section into significance, generalization and research scope.

5.5.1 Significance/Validity

The main important question that follows many medium and small-sized companies regarding recommender systems is that which algorithm is best suitable. The quality of a recommender system model lies in the validation using correct metrics. RMSE and MAE are the most common metrics generally used to evaluate and precision and recall metrics to evaluate top k recommendations. These measures make the model significant and valid results in evaluation of recommender systems.
From all stages of our experimental process, we noticed that RBM outperformed SVD by 1.6% and PMF by 5.7%. These results are significant because of the accurate predictions measured on models using RMSE and MAE. Our expectation has come true that deep learning approach RBM reduced the prediction error rate better than traditional approaches SVD and RBM. This makes valid that Contrastive Divergence(CD)3.4.2 function of RBM reduces error rate better than Stochastic Gradient Descent(SGD)3.4.1 on e-learning data. Although RBM slightly outperformed SVD in this experiment, it makes a significant difference on a large dataset with a large number of users. Below Table 8 shows the comparison of all the three models built as part of this research work.

Model	Mean Absolute Error	Root Mean Squared Error
Restricted Boltzmann Machines	0.0014686	0.0294584
Singular Value Decomposition	0.0181536	0.0392444
Probabilistic Matrix Factorization	0.5758119	0.686656

Table 8: Models Comparison

5.5.2 Generalization

The models built for this research work are completely based on collaborative filtering technique and used e-learning dataset as a benchmark to compare between models. In general, these models are not specific to a domain and also can be applied to other domains. In fact, Recommender systems are part of the success of Amazon and Netflix. Personalized recommender systems for any user or customer plays an important role in the business world and finding hidden patterns using filtering techniques makes accurate predictions in personalized systems. Other domains like a book, Online advertisement, music, news, e-commerce, travel recommendations, etc. also use personalized recommender systems to attract customers. The research work of this paper can be used in general to other domains mentioned above. Online advertisement business can use this RBM model which will use the data like whether the user is clicked the advertisement(1) or not(0) and suggest relevant items. Book recommendations are more similar to our E-learning course recommendations which will suggest relevant books to readers.
An e-commerce site can use this model which has implicit data like whether the customer purchased the product or not. Music and news recommender system can also use the model with the implicit data of listening and reading behavior respectively. This makes more generalization

to other domains and attracts researchers towards the success of deep learning in recommender systems.

5.5.3 Scope

As concerned with the research scope, these models can be used to build recommender systems on a large scale of data. In this research work, we used 1 million observations which have 69532 users and 5348 courses and converted into matrix form where users are in rows and courses are in columns. In general, other domains too have a large number of users and items and the model of this paper can be used to build recommender system as it outperformed traditional models on huge data. Generally, with the availability of computational support, deep learning approaches perform better than the traditional approach on a large dataset. This experiment was built using restricted Boltzmann machines and it can be scaled to architectural advancement like a stack of restricted Boltzmann machines(RBMs). The stack of RBMs is also called Deep Belief Networks(DBM) which can scale to deep advanced architecture for deep learning.

Interestingly, this model can be used in distributed machine learning methodology too which is an approach where the vast amount data have grown can be stored at different nodes of a cluster and deploy individual machine learning models in nodes to predict more accurately. This successful model can be deployed in different nodes of a cluster if the huge data is stored at different places. The best example is the Amazon e-commerce site which has business worldwide and the data is stored in different countries. So, it is technically and computationally difficult to bring all the data to one place and build a model for better performance. In this scenario, models can be deployed in its base location of data and the predictions can ensemble for better accuracy results.

6 Conclusion and Future Work

These days, most of the people prefer to educate themselves by e-learning platform rather than the traditional way of teaching. With the increase in e-learners and availability of courses based on learners requirements, e-learning recommender systems which intelligently suggest courses to learners face challenge how to suggest them relevantly and effectively. Despite the traditional way of teaching and non-personalized content, every learner may have distinct characteristics like own interests, level of expertise, background, study style in learning courses, etc. These features are may not available in data for all the learners and these features are also called latent hidden features. But, these hidden features definitely effect in collaborative filtering techniques while finding similarities among learners or courses. To find these hidden features among learners in a collaborative filtering approach, a deep learning architecture is required. An accurate e-learning recommender system model trained with deep learning architecture help in prediction and suggestion of relevant preferred courses to learners effectively.

In this report, we produced a unique deep learning approach Restricted Boltzmann machines(RBM) to e-learning recommender systems and compared it with other most widely used traditional techniques Singular Value Decomposition(SVD) and Probabilistic Matrix Factorization(PMF). We also demonstrated through our experiment that this technique can be successfully applied to a large dataset with a large number of learners and courses in e-learning platforms. The results of this paper will enable the data science community with insightful information regarding e-learning recommender systems.

Finally, concluding our experiment, proposed deep learning approach Restricted Boltzmann Machines outperformed traditional successful approaches SVD by 1.6% and PMF by 5.7% on the huge data provided and trained at several epochs.

6.1 Future Work

In the contribution to E-learning domain and extension to this research, researchers may work on building a hybrid model using Restricted Boltzmann Machines and considering the content of courses like course description. This model considered existing learner and courses of the system to recommend but researcher may work on suggesting new courses and also to new learners but not only with existing courses of the system. This will solve the problem of cold-start which is generally defined as the incapability of the recommender system to recommend a user or item due to lack of insufficient information. A researcher may also use some important features of courses like the course language, domain, course background, etc if available in data.
The most interesting future work can be integrating natural language processing with the recommender system considering the explicit feedback given by learner. The feedback content given by learner may have information regarding their sentiment whether they like the course or not and also any negative and positive opinions. This may improve the recommendation accuracy to learners based on their feedback content and suggest based on their feedback. A researcher may also use the existing models and compare the outperformed model of this experiment among different domains to check the performance and accuracy.

7 Acknowledgment

I thank Dr. Paul Stynes, Dr. Pramod Pathak and Dr. Dympna O Sullivian for their superior guidance of this research work and also thank McKinsey for making E-learning platform dataset available which is used as a benchmark to evaluate my research work. I also thank my parents for their support towards my career and it would not be possible without their guidance. I am grateful towards everyone with whose encouragement I successfully accomplished my masters on time.

References

Aditya, P. H., Budi, I. and Munajat, Q. (2016). A comparative analysis of memory-based and model-based collaborative filtering on the implementation of recommender system for E-commerce in Indonesia: A case study PT X, *2016 International Conference on Advanced Computer Science and Information Systems (ICACSIS)*, pp. 303–308.

Amatriain, X., Jaimes*, A., Oliver, N. and Pujol, J. M. (2011). Data Mining Methods for Recommender Systems, *in* F. Ricci, L. Rokach, B. Shapira and P. B. Kantor (eds), *Recommender Systems Handbook*, Springer US, Boston, MA, pp. 39–71.

Bobadilla, J., Ortega, F., Hernando, A. and Gutirrez, A. (2013). Recommender systems survey, *Knowledge-Based Systems* **46**: 109–132.
URL: *http://linkinghub.elsevier.com/retrieve/pii/S0950705113001044*

Bokde, D., Girase, S. and Mukhopadhyay, D. (2015). Matrix Factorization Model in Collaborative Filtering Algorithms: A Survey, *Procedia Computer Science* **49**: 136–146.
URL: *http://www.sciencedirect.com/science/article/pii/S1877050915007462*

Covington, P., Adams, J. and Sargin, E. (2016). Deep Neural Networks for YouTube Recommendations, ACM Press, pp. 191–198.
URL: *http://dl.acm.org/citation.cfm?doid=2959100.2959190*

Cowell, R. and Ghahramani, Z. (2005). Proceedings of the Tenth International Workshop on Articial Intelligence and Statistics, p. 461.

Cremonesi, P., Turrin, R., Lentini, E. and Matteucci, M. (2008). An Evaluation Methodology for Collaborative Recommender Systems, IEEE, pp. 224–231.
URL: *http://ieeexplore.ieee.org/document/4688072/*

Fischer, A. and Igel, C. (2012). An Introduction to Restricted Boltzmann Machines, *in* D. Hutchison, T. Kanade, J. Kittler, J. M. Kleinberg, F. Mattern, J. C. Mitchell, M. Naor, O. Nierstrasz, C. Pandu Rangan, B. Steffen, M. Sudan, D. Terzopoulos, D. Tygar, M. Y. Vardi, G. Weikum, L. Alvarez, M. Mejail, L. Gomez and J. Jacobo (eds), *Progress in Pattern Recognition, Image Analysis, Computer Vision, and Applications*, Vol. 7441, Springer Berlin Heidelberg, Berlin, Heidelberg, pp. 14–36.

Graves, A., Mohamed, A. r. and Hinton, G. (2013). Speech recognition with deep recurrent neural networks, *2013 IEEE International Conference on Acoustics, Speech and Signal Processing*, pp. 6645–6649.

Herlocker, J. L., Konstan, J. A., Terveen, L. G. and Riedl, J. T. (2004). Evaluating Collaborative Filtering Recommender Systems, *ACM Trans. Inf. Syst.* **22**(1): 5–53.
URL: *http://doi.acm.org/10.1145/963770.963772*

Isinkaye, F., Folajimi, Y. and Ojokoh, B. (2015). Recommendation systems: Principles, methods and evaluation, *Egyptian Informatics Journal* **16**(3): 261–273.
URL: *http://linkinghub.elsevier.com/retrieve/pii/S1110866515000341*

Koren, Y., Bell, R. and Volinsky, C. (2009). Matrix Factorization Techniques for Recommender Systems, *Computer* **42**(8): 30–37.
URL: *http://ieeexplore.ieee.org/document/5197422/*

Li, F., Wu, B., Xu, L., Shi, C. and Shi, J. (2014). A Fast Distributed Stochastic Gradient Descent Algorithm for Matrix Factorization, p. 11.

Linden, G., Smith, B. and York, J. (2003). Amazon.com recommendations: item-to-item collaborative filtering, *IEEE Internet Computing* **7**(1): 76–80.
URL: *http://ieeexplore.ieee.org/document/1167344/*

Martnez, O. S., Crespo, R. G. and Franco, E. T. (2014). Using Recommendation System for E-learning Environments at degree level, **1**: 4.

Miyahara, K. and Pazzani, M. J. (2000). Collaborative Filtering with the Simple Bayesian Classifier, *PRICAI 2000 Topics in Artificial Intelligence*, Lecture Notes in Computer Science, Springer, Berlin, Heidelberg, pp. 679–689.

Mnih, A. and Salakhutdinov, R. R. (2008). Probabilistic Matrix Factorization, p. 8.

Oliveira, R. (2016). Recommender System for an E-learning platform, p. 68.

Pandey, A. K. and Rajpoot, D. S. (2016). Resolving Cold Start problem in recommendation system using demographic approach, IEEE, pp. 213–218.
URL: *http://ieeexplore.ieee.org/document/7980578/*

Pazzani, M. J. and Billsus, D. (2010). Content-based Recommendation Systems, p. 18.

Salakhutdinov, R., Mnih, A. and Hinton, G. (2007). Restricted Boltzmann machines for collaborative filtering, ACM Press, pp. 791–798.
URL: *http://portal.acm.org/citation.cfm?doid=1273496.1273596*

Sharma, M. and Mann, S. (2013). A Survey of Recommender Systems: Approaches and Limitations, *International Journal of Innovations in Engineering and Technology* p. 9.

Sielis, G. A., Tzanavari, A. and Papadopoulos, G. A. (2012). Recommender Systems Review: Types, Techniques and Applications, p. 12.

Sikka, R., Dhankhar, A. and Rana, C. (2012). A Survey Paper on E-Learning Recommender System, *International Journal of Computer Applications* **47**(9): 27–30.
URL: *http://research.ijcaonline.org/volume47/number9/pxc3880024.pdf*

Strub, F., Gaudel, R. and Mary, J. (2016). Hybrid Recommender System based on Autoencoders, ACM Press, pp. 11–16.
URL: *http://dl.acm.org/citation.cfm?doid=2988450.2988456*

Ungar, L. H. and Foster, D. P. (2002). Clustering Methods for Collaborative Filtering, p. 16.

Zhang, Q., Yang, L. T., Chen, Z. and Li, P. (2018). A survey on deep learning for big data, *Information Fusion* **42**: 146–157.
URL: *http://www.sciencedirect.com/science/article/pii/S1566253517305328*

Zhang, Z., Li, T., Ding, C. and Zhang, X. (2007). Binary Matrix Factorization with Applications, IEEE, pp. 391–400.
URL: *http://ieeexplore.ieee.org/document/4470263/*

Zheng, L. (2016). A Survey and Critique of Deep Learning on Recommender Systems, p. 31.

Zhou, R., Khemmarat, S. and Gao, L. (2010). The impact of YouTube recommendation system on video views, ACM Press, p. 404.
URL: *http://portal.acm.org/citation.cfm?doid=1879141.1879193*