# THE USE OF MACHINE LEARNING ALGORITHMS IN RECOMMENDER SYSTEMS: A SYSTEMATIC REVIEW

**Prof. Parinita. J . Chate**
Assistant Professor, Computer Engineering Department, Bharati Vidyapeeth's College of Engineering Lavale, Pune, India.

**Abstract:** Recommendation system is one of the most popular applications of Artificial Intelligence which attracts many researchers all over the globe. The advent of the Internet era has brought wide implementation of recommendation system in our everyday lives.

We are dealing with; systematic review of the literature that analyzes the use of recommender system done using a machine learning algorithm often has problems and open questions that must be evaluated, so software engineers know where to focus research efforts.

Mainly focused on filtering algorithms based on the neighborhood of users or objects, and based on content, the description of these algorithms includes: similarities, disadvantages and advantages, measures for evaluating the algorithm, and calculation of the sample value of the evaluation prediction. The design part of the work begins with the description of the used databases from the Movie Lens portal.

*Index Terms* - recommender system, machine learning, systematic review

## I. INTRODUCTION

A recommender system is one type of application used in machine learning process, which can predict large scale data as per the needs, the system gets outstand results from recognition of user use "system work on platform or engine of information filtering system that seeks to predict the "rating"

Recommender systems are utilized in a variety of areas including movies, music, news, books, research articles, search queries, social tags, and products in general. There are also recommender systems for experts, collaborators, jokes, restaurants, garments, financial services, life insurance, romantic partners (online dating), and Twitter pages

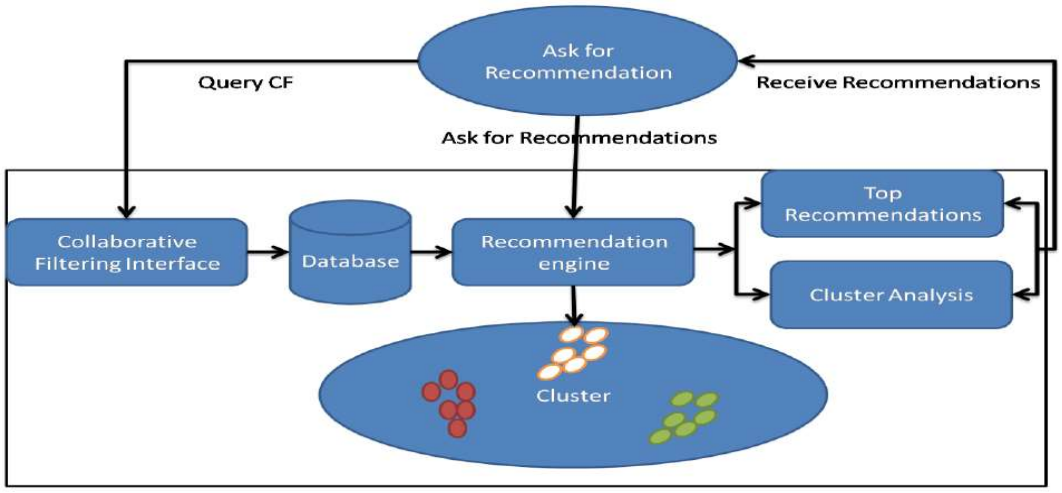



**Fig 1: Architecture diagram for collaborative filtering interface with database cluster analysis.**

A large amount of data flows through the internet and it gives away a lot of information regarding the user searching activity. The information extracted from the pattern of previously searched data can be molded into the prediction of relevant data for the user; the implementation of the system can be performed by various techniques.

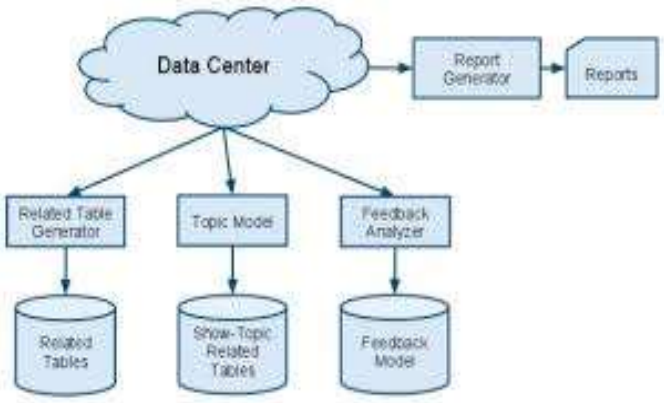



Figure 2 : Data center for prediction.

## II. **Statistical techniques for prediction**

Predictive analytics encompasses a variety of statistical techniques from data mining, predictive modeling, and machine learning that analyze current and historical facts to make predictions about future or otherwise unknown events.

The prediction of analytics is based on different learning methods that use the understanding of the past, by uncovering relationships and patterns within large volumes of data to make predictions about future events. It can, for instance, be applied to make predictions about customers' behavior, or even when a factory floor machine is likely to break. Today, predictive analytics is mostly used within marketing for customer acquisition, campaign management, budgeting and forecasting models, cross selling etc.

Analytics is the discovery and communication of meaningful patterns in data (corporate, product, channel, and customer). It's not the data but the signals buried in and inferred from data. There are four distinct types of analytics:

By analyzing the past and the present, the user can extend descriptive analytics with predictive analytics, which will provide him with possible answers to the questions:

What happened, where and when – Descriptive Analytics

Why did it happen – Diagnostic or Prescriptive Analytics

What is likely to happen – Predictive Analytics

Guided actions and steps – Machine Learning, AI and Cognitive Learning

Conversational AI – **Chatbots**

All of these need data. Data is the new raw material. Cloud is the new pipeline. Machine Learning is the new refinery. **Digital** use cases is the new experience frontier

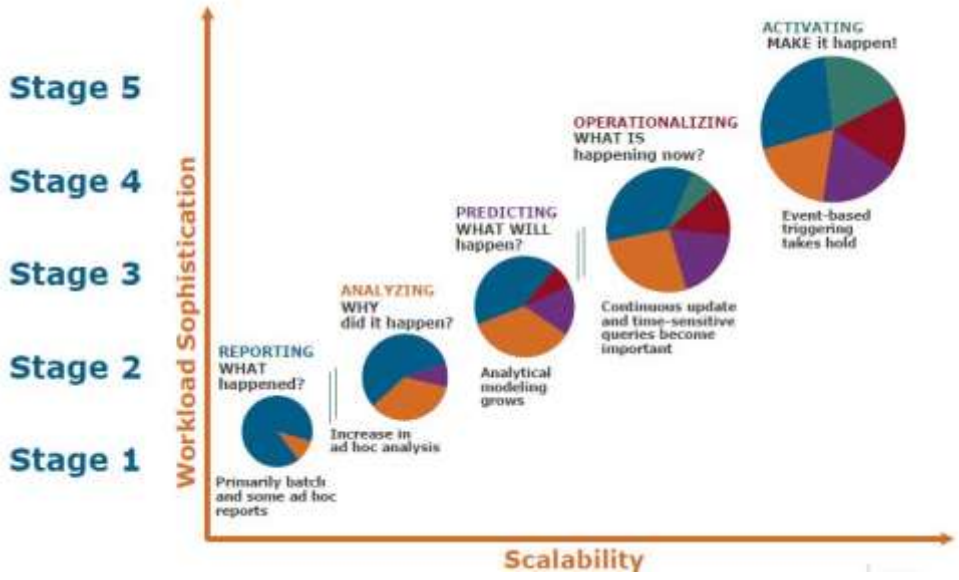



Figure 3: Stage in a prediction.

In the paper we are trying to highlight algorithms deal with dataset in order to ask for recommendation the movies, music, news, books, research articles, search queries, social tags, and products in general and calculate the precision along with tackling the cold-start problem.

The cold start problem concerns the personalized recommendations for users with no or few past history (new users). Providing recommendations to users with small past history becomes a difficult problem for CF models because their learning and predictive ability is limited. Multiple researches have been conducted in this direction using hybrid models. These models use auxiliary information (multimodal information, side information, etc.) to overcome the cold start problem.

## III. Machine learning

Over the past two decades Machine Learning has become one of the mainstays of information technology and with that, a rather central, albeit usually hidden, part of our life. With the ever increasing amounts of data becoming available there is good reason to believe that smart data analysis will become even more pervasive as a necessary ingredient for technological progress.

Machine learning is an application of artificial intelligence (AI) that provides systems the ability to automatically learn and improve from experience without being explicitly programmed. **Machine learning focuses on the development of computer programs** that can access data and use it learn for themselves.

A Taste of Machine Learning

Machine learning can appear in many guises. We now discuss a number of applications, the types of data they deal with, and finally, we formalize the problems in a somewhat more stylized fashion. The latter is key if we want to avoid reinventing the wheel for every new application. Instead, much of the art of machine learning is to reduce a range of fairly disparate problems to a set of fairly narrow prototypes. Much of the science of machine learning is then to solve those problems and provide good guarantees for the solutions, **the primary aim is to allow the computers learn automatically** without human intervention or assistance and adjust actions accordingly.

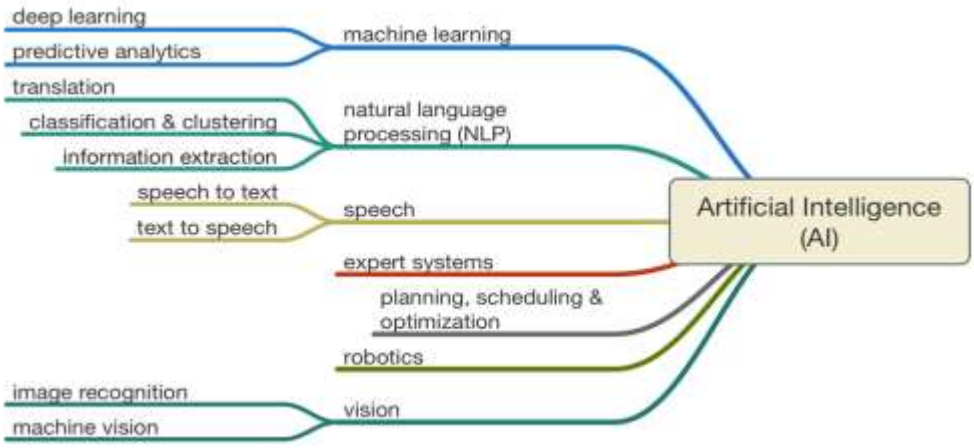



Figure 4: Face Technology is one of the many applications of AI

## IV. **Systematic Review**

Developing recommender system's, the engineers must decide specific recommender algorithm of all those available. The number of algorithm variations and combinations in the literature makes this choice a challenging task.

Trying to develop tools to make RS development easier is a moving target, as new studies must be done to observe new open problems and trends, and further enrich the knowledge base. For these reasons the authors decided to do a systematic review of the ML field to analyze the development of RS containing ML algorithms and to see if there could be a more organized approach to software engineering of RSs. This review has two main goals:

- To identify which ML algorithms are used in RSs
- We review the relevant articles in the field of scholar recommendations.
- We explore contextual information influential in scholar recommendations.
- We examine recommending approaches.
- Contextual information are categorised in three groups.
- The most recommending approaches are collaborative filtering, content based, knowledge based and hybrid

To answer first question are done read each publication returned by a search queryand list the proposed algorithms. The approach to the second question is different. First,
the authors limit the scope of SE areas to the five stages of the waterfall model for
software lifecycle: requirements, design, implementation, verification, and maintenance.

EC1: Publications must have been peer-reviewed and published in a conference, journal, press, etc. Publications that were not peer-reviewed or formally published are excluded.
EC2: Publications must describe the recommender system and the machine learning algorithm sufficiently well that it is possible to identify the algorithm. Publications that do not clearly state the machine learning algorithm being used are excluded.
EC3: Publications must be unique. If a publication is repeated, other copies of that publication are excluded.

The search query was used on the popular academic search engine Scopus 10, Some publication entries were excluded based on the exclusion criteria previously explained, and they are summarized on Table

**Table 1 - Publications that were included and excluded from the systematic review**

| Total retrieved | | **10** |
|---|---|---|
| **Reason** | **Publication** | **Total** |
| Conference / Proceedings entries | | **2** |
| Not describing a case study or implementation [46] 1 | **5** | **3** |

**Systematic Review Results**

The results and conclusions are presented in the following, Processing ML algorithm names means that separated algorithms into categories based on [10]. Some algorithms had obvious classifications because they were small variations of well-established algorithms (e.g. incremental matrix factorization is a variant of the matrix factorization algorithm, and collaborative filtering, content based, knowledge based and hybrid formation taken in to consideration)

their investigation of different techniques inrecommender systems also showed that a hybrid approach produced the best recommendations. Besides that, grouping different users using *clustering* techniques in such systems, turned out to increase the accuracy and effectiveness of the system that they proposed.

## V. Algorithms and Techniques

### 5.1. K-Nearest Neighbor

The simplest algorithm computes cosine or correlation similarity of rows (users) or columns (items) and recommends items that k—nearest neighbors enjoyed

K-nearest neighbor (k-NN) algorithms are unsupervised algorithms, and are very popular in recommender systems, due to their simplicity and efficiency (Ricci, et al., 2015). These algorithms are applied in both content-based and collaborative-filtering approaches to compute and find the most similar entities of a given entity in the system. Let $T = \{e_1, e_2, \dots, e_n\}$ be a set of observations with $e_i$ being an entity consisting of a set of features $\{f_1, f_2, \dots, f_n\}$ defined as numeric values. The k-NN algorithm will take a new entity $\hat{e}$ and

compute the *k* most similar entities of $\hat{e}$ in *T* using a similarity function. The results are then ranked, and the selected set of *k* most similar users are called the *k*-nearest neighbors.

Common similarity functions used in k-NN algorithms for recommender systems are the *Euclidean distance*, the *Cosine similarity*, and the *Jaccard similarity* (Qamar & Gaussier, 2012). Another popular similarity method used in CF-based recommender algorithms is the *Pearson correlation* (Hahsler, 2011).

The distance *d* using the Euclidean distance between a vector *e*) and *e* is calculated as:

$$d_{ED}(e_i, \hat{e}) = \sqrt{(f_1 - \hat{f}_1)^2 + (f_2 - \hat{f}_2)^2 + \cdots + (f_n - \hat{f}_n)^2} \qquad \textbf{Eq. 1}$$

When comparing the similarities between two entities, it can be more meaningful to use the *Normalized Euclidean distance*:

$$d_{NED}(e_i, \hat{e}) = \frac{d_{ED}(e_i, \hat{e})}{|e_i|} \qquad \textbf{Eq. 2}$$

According to some researchers, the cosine similarity should be preferred over Euclidean
distance when dealing with non-textual data (Qamar & Gaussier, 2012). It is calculated by applying the following formula:

$$sim_{cosine}(e_i, \hat{e}) = \frac{\sum_j f_j \hat{f}_j}{\sum_j f_j^2 \times \sum_j \hat{f}_j^2} \qquad \textbf{Eq. 3}$$

where $sim_{cosine}(e_i, \hat{e}) \in [0,1]$ and $j$ being the index of the $jth$ feature of an entity.

When comparing vectors consisting solely of binary data, it is only interesting to consider the intersection of features between entities, then the Jaccard similarity is a favorable measure (Hahsler, 2011). It is calculated as:

$$sim_{jaccard}(e_i, \hat{e}) = \frac{|e_i \cap \hat{e}|}{|e_i \cup \hat{e}|} \qquad \textbf{Eq. 4}$$

The Jaccard similarity represents the intersection of the entities $e$) and $e$ in relation to their union.

### 5.2. Cold start and content based recommendation

Sometimes interactions are missing. Cold start products or cold start users do not have enough interactions for reliable measurement of their interaction similarity so collaborative filtering methods fail to generate recommendations.

Cold start problem can be reduced when attribute similarity is taken into account. You can encode attributes into binary vector and feed it to recommender.

### 5.3. Collaborative Filtering

Collaborative filtering methods are based on collecting and analyzing a large amount of information on users' behaviors, activities or preferences and predicting what users will like based on their similarity to other users. A key advantage of the collaborative filtering approach is that it does not rely on machine analyzable content and therefore it is capable of accurately recommending complex items such as movies without requiring an "understanding" of the item itself. Many algorithms have been used in measuring user similarity or item similarity in recommender systems

**Algorithm 2**. Collaborative Filtering

**Input:** users $X$, movies $m$, rating $r$, Number of movies to be recommended$(\mu)$.

**Output**: Recommended movies $R$.

1. for all users do
2. Select seen movies $s$, unseen movies $s'$
3. Find similarity $(sim_i)$ w.r.t $s$, where $i = 1$ to $n$.
4. Select highest $sim_i$ user
5. Select $m' \in s$ of user obtained in step 4 and $s'$ of $i^{th}$ user.
6. Calculate weight $W(m_e')$ where $e \in m'$
7. Return top $\mu$ weight recommendations.
8. end for

In this algorithm, the notations used have the following meaning : $sim_i$ represents common movies between user $i$ and other users.

$weight(m_e') = rating\ of\ particular\ movie_e\ /\ max\ rating.$

An example of the prediction procedure in a user-based approach, using 5 users and 6 items,is illustrated in Figure 5. Given the 5×6 users-item preference matrix below, a k-NN algorithm is applied to find the most similar users (grayed out rows) to the active user (colored row) in the matrix.

**Table 2: User-based collaborative filtering example for predicting a user's preferences for unknown items**

| | $i_1$ | $i_2$ | $i_3$ | $i_4$ | $i_5$ | $i_6$ | |
|---|---|---|---|---|---|---|---|
| $u_1$ | 0 | 0 | 0 | 1 | 0 | 3 | |
| $u_2$ | 3 | 3 | 2 | 1 | 0 | 0 | 1th NN |
| $u_a$ | 4 | 3 | **1.5** | 2 | **1** | **0** | Active User |
| $u_4$ | 0 | 0 | 3 | 0 | 4 | 2 | |
| $u_5$ | 4 | 0 | 1 | 0 | 2 | 0 | 2th NN |

### 5.4. Hybrid Filtering

Hybrid approaches can be implemented in several ways: by making content-based and collaborative-based predictions separately and then combining them; by adding content-based capabilities to a collaborative-based approach (and vice versa); or by unifying the approaches into one model (see[21] for a complete review of recommender systems).

Suppose, the user appreciates mostly movies in $g \subset G$ genres, and the collaborating users also give high ratings to the $g \subset G$ genres, then $g$ will be taken as the metric to recommend movies to the user.

**Algorithm 3** .Hybrid Filtering

**Input:** users *X*, movies *m*, rating *r*, movie genre *mg* , Number of movies to be recommended($\mu$).

**Output**: Recommended movies *R*.

1. for all users do

2. Select seen movies *s*, unseen movies *s'*, association of each genre *agj* w.r.t *s'*, where *i* is 1 to n and *j* is 1 to *m*.

3. Calculate *scorej* .

4. Select highest three *scorej*

5. Select *m''* Є *s* of the *ith* user according to highest three *scorej*

6. Find similarity (*simj*) w.r.t m"

7. Select highest *simj* user.

8. Select *m'* according to its highest three *scorej* Є *s* of user obtained in step 7 and *s'* of the *ith* user under consideration.

9. Calculate weight *W(me')* where *e* Є *m'*

10. Return top $\mu$ weight recommendations.

11. end for

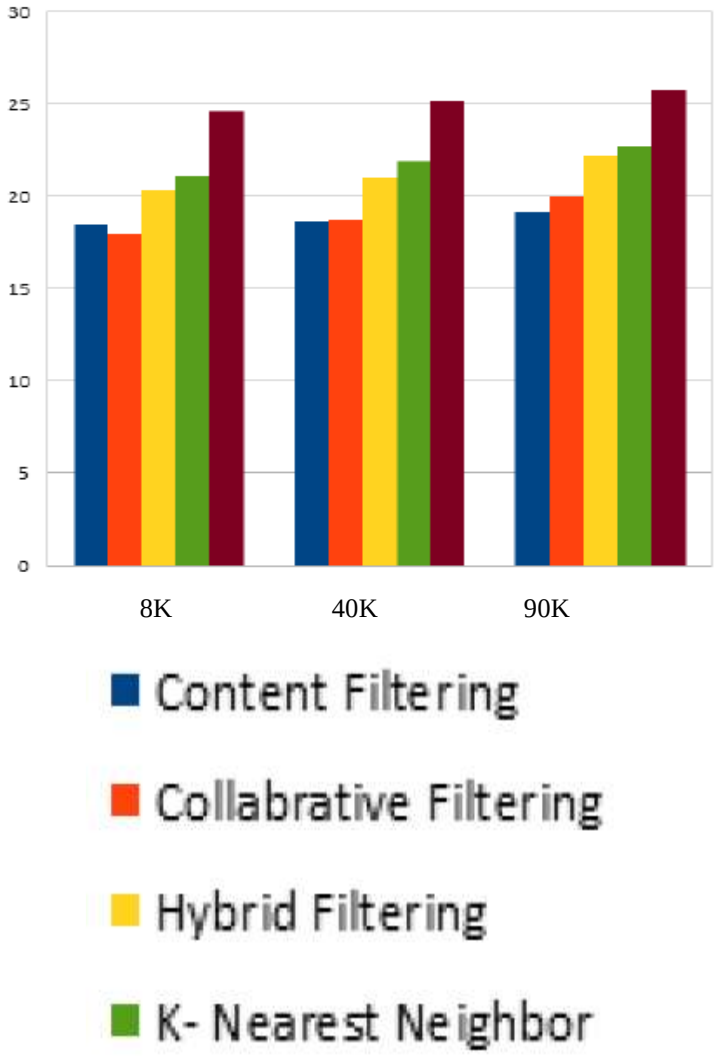
12. end

## VI. EXPERIMENTAL RESULT

The results from the conducted evaluations of the implemented systems and their variants, to compare their accuracy we have used the MovieLens dataset of 8K, 40K and 90 K. The dataset varies in sparsity.

Precision = ($\Sigma$tc / $\Sigma$T ) * 100

For all the experiments, we are taking value of $\mu$ = 4 and value of $k$ = 10.




Bar chart 1: Precision Comparison

## VII. **Methodology**

the research methodology used in this study. First, the chosen research approach is presented. Then follows a description of how the feature selection, algorithm selection, and model evaluation were conducted. Finally, the data set used for testing and evaluation is described as shown in the flow chart.

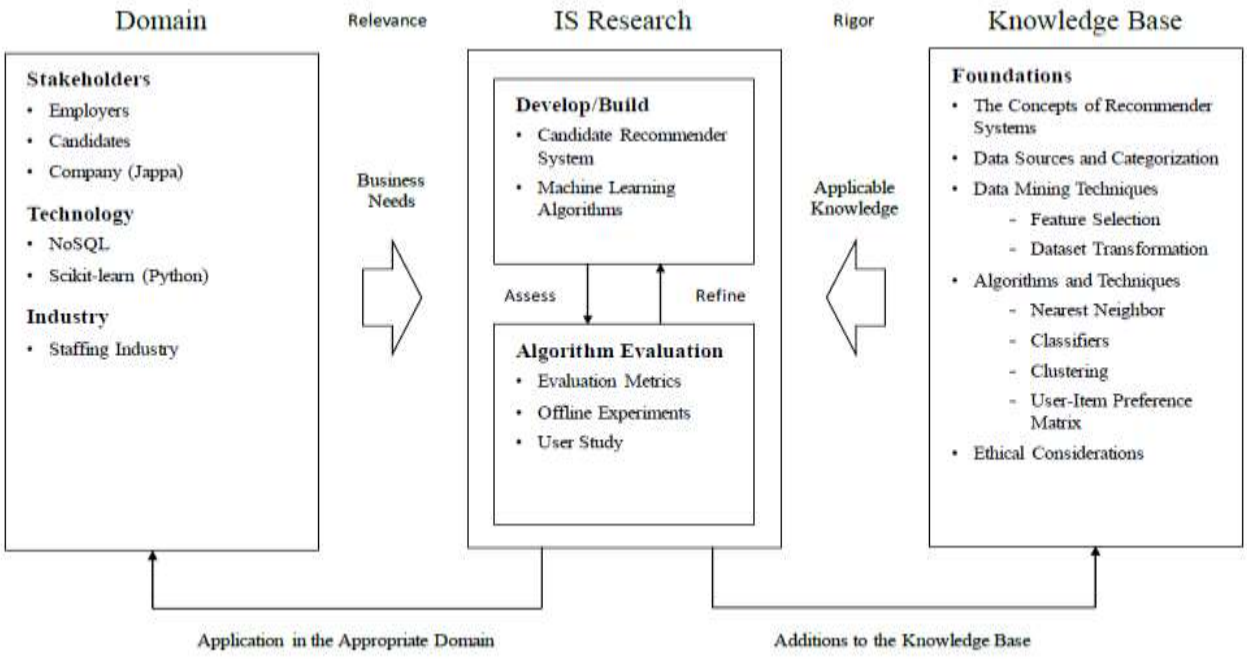



Figure 5: Overview of research approach based on a framework for design research studies by Hevner et. al(2004, p.80).

## VIII. Implementation

All algorithms in this study were implemented using Scikit-learn, a collection of machine learning libraries for Python. Scikit-learn is built on NumPy, SciPy and matplotlib, popular Python libraries for scientific and mathematical computing and visualization. Scikit-learn contains tools for classification, regression, clustering, data pre-processing, dimensionalityreduction and model evaluation.

## IX. CONCLUSION

Currently, recommender systems (RS) are widely used in e-commerce, social networks, and several other domains. One progressive step in RS history is the adoption of machine learning (ML) algorithms, which allow computers to learn based on user information and to personalize recommendations further.

The literature lacks a classification system for algorithms showing the environment in which they are most suitable. Therefore, choosing an ML algorithm to be used in RSs is a difficult task.

The systematic review collected 26 publications, after filtering out some based on exclusion criteria. All publications were read and the conclusions are as follows. In RS development, the ML algorithm.

All the algorithms described in this paper are compared with respect to their precision rates. This comprehensive analysis depicts the strength and the weakness of each one of them in different versions of the MovieLens dataset. The experiments performed are the witness of the sparsity handling by these algorithms. Our experiments have shown promising results

## X. FUTURE WORK

This study serves as a basis for investigating RS development. In the future, more studies on the use of Bayesian algorithms in RSs can be done to observe the implications of their use, performance, and utility.

In the real time sophisticated recommendation systems there is a need of high accuracy.

## XI. Acknowledgements

## XII.

The authors would like to thank Bharati Vidyapeeth's College of Engineering, Lavale, Pune for supporting and guiding me. Also I am thankful to Principal Sir, Staff who encourage me all the time and willingness to provide assistance whenever it was needed.

**References**


1 Adomavicius, G., & Tuzhilin, A. (2011). Context-aware recommender systems. In *Recommender systems handbook* (pp. 217-253). Springer US.

2 Katore L.S., and Umale J.S. Comparative Study of Recommendation Algorithms and Systems using WEKA, International Journal of Computer Applications, Volume 110 – No. 3, pp14-17, 2015.

3 Anand Nautiyal University of Hyderabad and Mahendra Prasad Indian Institute of Technology (ISM) Dhanbad,Machine Learning Algorithms for Recommender System - a comparative analysis, International Journal of Computer Applications Technology and Research Volume 6–Issue 2, 97-100, 2017, ISSN:-2319–8656

4 Evaluation of Machine Learning Algorithms in Recommender Systems, Candidate Recommender Systems in the Staffing Industry, Master's Thesis in Software Engineering, Adam Myrén, Piotr Skupniewicz Neto

5 Arenas-García, J., Meng, A., Petersen, K. B., Lehn-Schioler, T., Hansen, L. K., & Larsen, J. (2007, August). Unveiling music structure via plsa similarity fusion. In *Machine Learning for Signal Processing, 2007 IEEE Workshop on* (pp. 419-424). IEEE.

6 **Ivens Portugal,** David R. Cheriton School of Computer Science University of Waterloo Waterloo, ON, Canada iportugal@uwaterloo.ca,The Use of Machine Learning Algorithms in Recommender Systems:A Systematic Review

7 Vinyals, O. and Le, Q. V. A neural conversational model, *Proceedings of the 31st International Conference in Machine Learning*, Vol.37, arXiv:1506.05869v3, 2015.

8 Campbell, M., Hoane Jr. A. J., and Hsu, F.-H. Deep blue, *Artificial Intelligence* 134(1–2): 57-83, 2002.