

Hybrid Recommender for Research Papers and Articles

Alhassan Jamilu Ibrahim, Peter Zira, Nuraini Abdulganiyyi

Faculty of Science, Federal University Kashere, Gombe, Nigeria

Email address:

jamilualhassan@yahoo.com (A. J. Ibrahim), zirrapeter64@gmail.com (P. Zira), agnuraini@gmail.com (N. Abdulganiyyi)

To cite this article:

Alhassan Jamilu Ibrahim, Peter Zira, Nuraini Abdulganiyyi. Hybrid Recommender for Research Papers and Articles. *International Journal of Intelligent Information Systems*. Vol. 10, No. 2, 2021, pp. 9-15. doi: 10.11648/j.ijis.20211002.11

Received: September 18, 2019; **Accepted:** March 22, 2021; **Published:** May 14, 2021

Abstract: In digital libraries and other e-commerce sites, recommender system is the solution that supports the users in information search and decision making. Some of these recommender systems will make predictions by matching the content of an item against the user profile otherwise known as Content-Based recommendation approach. Other recommenders will provide recommendation based on ratings of items from current user and other users and then use it to recommend similar items the current user has not seen, this is known as Collaborative-Based recommender approach. There exist several other approaches that are used in recommending articles and other items to users of different search engines. Over the years several researchers have tried combining these approaches in an attempt to design more efficient recommendations in search engines. This research proposed and designed a prototype of a Hybrid recommender called Zira, which is a model that combines both the Collaborative filtering, Content-based filtering, attribute-based approach to look at contextual information as well as an item-based approach that will solve the issues associated with cold-start problems all working concurrently to complement one another. The proposed system supports multi-criteria ratings, provide more flexible and less intrusive types of recommendations to ensure the improvement in recommendations of e-learning materials to users of digital libraries.

Keywords: Recommender System, Content-Based Approach, Collaborative Filtering Technique, Hybrid Recommender, Digital Library

1. Introduction

According to researchers, the roots of recommender systems can be traced back to the extensive work in information retrieval, approximation theory, cognitive science, forecasting theories, consumer choice modeling in marketing, as well as evidences and links to management sciences [16, 7, 2, 27, 13]. In the mid 1990's, when recommendation problems that unambiguously rely on ratings structure started attracting researchers with diverse approaches, recommender systems surfaced as an independent research area. As they gain more attraction, the issues of recommendations is reduced to that of estimating ratings for items which the user has not seen [1].

Over the years, various researchers used diverse approaches and techniques to spawn algorithms and systems that will recommend different categories of items to users. In recent years other researchers worked towards improving some existing models to overcome certain limitations, among them are the authors in Agosti et. al. and Anderson whom

used content-based approach to research paper recommenders [3, 6]. Other researchers like used past ratings of active users otherwise collaborative-based approach to recommend e-learning materials [8]. Researchers clearly outlined the drawbacks associated with using either of the two most popular approach to recommendation which are Content-based filtering (CBF) and Collaborative-based filtering (CF) techniques [5]. Over the past years many researchers attempted to overcome the drawbacks associated with using single recommender by implementing hybrid ones. Various researchers used wide range of approach, from using simple techniques to complex algorithms that will combine more than one existing ones [9]. Successfully many researchers did manage to improve recommendation by overcoming some of the limitations. Among them is BellKor solution which combines predictions from over hundred baseline recommender algorithms and going ahead to win the Netflix prize [20]. Improving recommendation accuracy had always been the driving force for merging various predictors often used in settings of machine learning [4].

Melville et. al., proposed a *feature-augmentation* algorithm

denoted “Content boosted collaborative filtering” (CBCF), an approach which even Burke’s survey affirms as one of the best recommendation algorithms [25]. The technique basically made from creating ‘augmented user profiles’ by adding ‘pseudo-ratings’ to the initial user profiles prior to generating recommendations [11, 36, 38]. For given items’ features or characteristics, Pseudo-ratings are generated using a content-based Naïve Bayes classifier which as well can be interpreted as the ratings that users would give to items that were never been rated [24]. Various researchers showed how rating prediction is computed with a variant of the user-based collaborative approach, where user-to-user similarities are computed as the Pearson Correlation Coefficient between the original user profiles and the augmented user profiles [12]. According to Fan *et al.*, the weights assigned to pseudo-ratings depend on the number of rated and co-rated items for each user [15]. As detailed by Hu *et al.*, in terms of Mean Absolute Error (MAE), their algorithm performed better than the content-based Naïve Bayes and the user-based collaborative algorithms [18]. Researchers like Igbe *et al.*, also proved that their algorithm is less vulnerable to problems induced by sparsity [19].

hybrid-based recommender systems continue to gain more and more attention from researchers to overcome recommendation issues like over-specialization, content description, sparsity, early-rater problem, gray-ship, subjective domain and, most sinister, cold-start problems [14, 17]. A paper recommender called Scienstein was developed in using hybrid approach in providing recommendations to first targeted users [33]. Citation analysis and sources analysis were used to increase the quality of recommendation [21]. Papers are considered relevant that are published by the same author or journal, which at times is inaccurate. Another hybrid-based paper recommender system was also released as the output of the research work by Pandya *et al.*, so as to increase the quality of recommendations [27]. The research cited the issues associated with using most of the existing recommender algorithms. The researcher also recommended the combination of many recommendation algorithms in order to overcome the disadvantages of either one of the algorithms.

Although various researchers continue to display remarkable effort towards devising hybrid algorithms for recommendations of various products, many at times failed to produce models that will combine multiple techniques to equally overcome multiple drawbacks associated with such e-learning recommendations. One of them is looking at the contextual information such as attribute of the materials as well as overlooking cold-start issues like *new item*, *new user* and *new system* [23, 29]. This research combines Collaborative-based and Content-based filtering techniques to produce a hybrid recommender, putting into consideration the attribute-based of paper materials [22, 26]. Extensions where implemented to the attribute-based model in developing the trivial hybrid algorithm thus eliminating the problem of *new user*, *new item* and other problems or limitations [34]. Several testing methodologies are also

implemented in order to evaluate the quality of the recommender algorithm in, especially, the presence of *new items* [28]. These research where generally to improve the users and items understanding, incorporate the contextual information into the recommendation process, supports multi-criteria ratings and provide more flexible and less intrusive types of recommendations [2, 10, 30, 31, 32].

2. Approach

This research develops a recommender system that implements five different approaches; Collaborative Filtering (CF), Content Based Filtering (CBF), Nearest User Rated material (NUR), Most Frequently Visited materials (MFV), and Most Similar Visited Material to the targeted learner (MSV). To mirror learner’s complete range of interests, Matrix Preference (MP) was used to consider multi-dimensional attributes of materials. The importance for specific attribute of each learner can be determined based on rating of learner’s access to educational materials. In accordance with the specific attributes, learning material profile is defined as a vector in which the values of attributes are assigned to a material. The attribute based model is defined as a multi-dimensional vector $I = A_1, A_2 \dots A_k$ where A_k is the K -th attribute’s name. Putting into account the weight for each attribute, AW_k is the weight value and $\sum_{k=1}^K AW_k = 1$. Given by the learning ratings on materials, the attributes of learning materials will transfer to learner’s profile. For each learner the proposed system will make a personal preference matrix including K -rows corresponding to the attributes of category and T -columns corresponding to T visited materials. In this matrix, S_{ijk} denotes the score of the attribute A_k by learner i in the observance of material j which is computed as follows:

$$S_{ijk} = AW_k r_{ij} \quad (1)$$

Where r_{ij} is rating of learner i for the material j visited.

2.1. Content-Based Recommendation

The similarity between learner behavior and special material, p is given by the following equation:

$$sim(L_i, M_p) = \frac{\sum_{j=1}^T w_{ij} \cdot \sum_{k=1}^K S_{ijk} \cdot m_k(L_j, M_p)}{T \cdot K} \quad (2)$$

In which w_{ij} is a weighing value for observation of material j by learner i and is normalized with I-norm. Since learner’s recent accessed material plays an important role in to the future interests. The relative importance of each observation is pre-determined as follows:

$$w_{ij} = e^{-\lambda(t(L_{ij})-T)} \quad (3)$$

Where $t(L_{ij})$ is the order of materials j in the recent observation by learner i and λ in an adjustable parameter used to describe the change rate of learner’s preference. This formula gives more weight to recent visited materials. $m_k(L_{ij}, M_p)$ is the matching function between k -th attribute

of material L_{ji} and M_p that is calculated as follows:

$$m_k(L_{ij}, M_p) = \begin{cases} 1 & \text{if } value(A_k, M_p) = value(A_k, L_{ij}) \\ 0 & \text{otherwise} \end{cases} \quad (4)$$

Therefore if the attribute of k -th of L_{ij} and M_p be same $m_k(L_{ij}, M_p)$ gets 1 otherwise 0. Materials are ranked by calculating the similarity between the preference matrix made up of weighted behavior attribute sets as materials. Highly ranked materials are then recommended to the learner.

2.2. Collaborative Filtering Recommendation

STEP ONE: For reflecting the similarity between the preferences of two learners, the similarity is calculated as follows

$$sim(L_i, L_j) = \frac{\sum_{h=1}^T \sum_{f=1}^T w_{th} w_{tf} \sum_{k=1}^K s_{ikh} s_{jkf} m_k(L_{jh}, L_{jf})}{T^2 K} \quad (5)$$

Where

$$m_k(L_{jh}, L_{jf}) = \begin{cases} 1 & \text{if } value(A_k, L_{ih}) = value(A_k, L_{jf}) \\ 0 & \text{otherwise} \end{cases} \quad (6)$$

STEP TWO: Here the proposed system determines explicit attribute-based neighborhood of learner a , $N(L_a)$ according to the calculated similarity in step 1.

STEP THREE: The prediction rating of material i by using implicit attribute based for active learner a is $P(L_a, i)$ that is gained by the rating of L_a , neighborhood, $N(L_a)$, that have rated i before. The computation formula is as follows:

$$P(L_a, i) = \bar{R}_{L_a} + \frac{\sum_{j \in N(L_a)} sim(L_a, L_j) \cdot (R_{L_j}(i) - \bar{R}_{L_j})}{\sum_{L_i \in N(L_a)} sim(L_a, L_j)} \quad (7)$$

Where \bar{R}_{L_a} and \bar{R}_{L_j} average rating of items rated by active learner L_a and L_j respectively.

2.3. Other Recommendations

Nearest User Rated Material (NUR): the weights of the current user are affected by each cold-start question. These numbers correspond to how much each material tag is weighted by the Euclidean distance for the KNN algorithm. The difference between two vectors as defined in Equation 8 where $u_{a,i}$ is the i th answer from user a and n is the number of questions.

$$sim(\bar{u}_a, \bar{u}_b) = \frac{1}{\sqrt{diff(\bar{u}_a, \bar{u}_b) + 1}} \quad (8)$$

$$diff(\bar{u}_a, \bar{u}_b) = \sum_{i=1}^n |\bar{u}_{a,i} - \bar{u}_{b,i}| \quad (9)$$

The NUR will go a long way towards solving the cold-start problems, which are *cold-start new item*, *cold-start new user*, *cold-start new system*.

Most Frequently Visited Materials (MFV): It looks into $N(L_a)$ for each neighbor, scans through the database and counts the visit frequency of materials. Therefore the score in this method is defined as:

$$Score_p = \sum_{L_i \in N(L_a)} N_{FV}(L_i, p) \quad (10)$$

Score p denotes the visit frequency of material p by neighbors of L_a (Symeonidis et. al. 2008). This method assumes that the more a material is visited, the more popular it becomes.

Most Similar Materials to the Targeted Learner (MSL): is employed to give the system the capability to make suggestion outside the scope of what the user has already shown interest in. According to sequential combination model of hybrid recommendation systems, those materials that visited by $L_i (L_i \in N(L_a))$ and are the most similar to $L_a (Sim(L_a, M_p))$ will be considered, the score is defined as:

$$Score_p = Sim(L_a, p) \quad (11)$$

Most Similar Visited to the Most Similar Learners (MSV-MSL): Since the similarity of the neighbors will vary significantly for each material visited by $L_i (L_i \in N(L_a))$, this method admeasures not just the similarity of materials to $L_i (Sim(L_i, M_p))$ but to the similarity of L_i and $L_a (Sim(L_a, L_i))$ as well. In this method the score p is defined as:

$$Score_p = Sim(L_i, p) \cdot Sim(L_a, L_i) \quad (12)$$

MVF-MSL: the researchers made combination that will make alternative replacements of scores in both the base algorithms. The method scores according to their visit frequency by learner L_i within $N(L_a) (N_{FV}(L_i, p))$ and the similarity of L_i and $L_a (Sim(L_a, L_i))$ where:

$$Score_p = N_{FV}(L_i, p) \cdot Sim(L_a, L_i) \quad (13)$$

The server stores each action from a user using field *source* that describes which learning material is looked up or which action is performed on a certain learning material.

2.4. Implementation

The recommender system (*Zira*), consists of 3 components; the Client (web), Backend (Server) and the Recommender engine which is implemented on the server.

2.4.1. Client

The client used for developing *Zira* is HTML5 embedded in a web view and packed by most web clients. The HTML5 is used along with Cordova Project so that it is easier to communicate with the servers Application Programming Interface (API) [37]. The client uses Hypertext Transfer Protocol (HTTP) request to communicate or acquire all contents visible for the end-user. It uses object oriented approach to distinguish between authorized and not authorized classes to help keep track of more complex tasks. For a particular end user session in a recommendation cycle, the server gives a list of papers or materials to the client so that if the user recommends one, it is recorded to the API and taken into consideration for later recommendations. The client also provides search for other papers and when the user clicks on any searched item it will be collected as a view of the materials.

2.4.2. Backend

The backend (or server) of the system is where all the

clients get their learning materials, recommendations and post their feedback. All information is saved on the backend, therefore it is where the client gets the information they display to the user. Using the programming language Scala with Play Frame, the server is implemented using modern web architecture style Representational State Transfer (REST) [39]. Apart from modern language and framework, it also allows the use of Java library whenever the need arises. The researchers hosted the server on Heroku Cloud Service because it allows easy deployment without much configuring on your own. The client request from the server through different URLs called endpoints.

2.4.3. Recommender

The recommender system is part of the server while client communicates with it through one of the API endpoints. Initially, the KNN algorithm calculates the distance between two users before it can have enough ratings from explicit star ratings [35]. Similarity measure is used to find most similar users. The ratings will be measured with the same similarity and is the product of their actual ratings and similarity to similar user. For collaborative based filtering, the system is developed with a relation between user and material tag so that score for the user is saved stored for the tags. Scores on each tag are made from both cold-start question and user ratings which forms the user profile.

GroupLens called LensKit, along with set of tools for such system was used to implement Collaborative filtering algorithm. This research uses only the LensKit-core and LensKit-data-structures modules to implement this section of the algorithm. The core module is required to initialize the building of collaborative filtering recommendations as it is also a dependency for the other modules. An item recommender is built from an access object to the database where it will gain access to other ratings, users and items. In order to help LensKit compute recommendations, problem specific is needed, therefore, LensKit-data-structure module was used to provide data structure to help with the configuration.

Various field *sources* are keyed-in to the algorithm to support multiple attributes of the materials. The system gathers explicit feedbacks in two ways, either (1) when the user agrees or clicks certain buttons of the recommendation list, or (2) when user reports a star rating after viewing a material. The system also has two mechanisms for gathering implicit feedbacks. (1) A timer is started anytime a user looks at a material so as to predict if the user did actually read the article or (2) Timestamp are logged for every request so that same material within same period of time are not recommended.

2.4.4. Recommender Composition

The composition of recommendation list is mainly from Collaborative filtering and Content-based filtering which constitutes 80% (40% each). The remaining percentage is filled in by the NUR algorithm in the event of cold-start problems; otherwise it will be filled by either of the best two attribute-based MSV or MSV-MSL algorithm, if no enough

knowledge for CF or CBF algorithms is available. Lower score than defined threshold can outperform CBF which will then exclude a material, while CF can be outperformed if the user has not yet rated any material and the engine has nothing to compare against. The other 20% is a set of random materials which is a necessary strategy in order to be able to compare other algorithms to random in the evaluation and help integrate new materials with no ratings. The recommendation list is not deterministic and therefore randomized so as to help unrated materials gather some ratings as well.

3. Experiments and Results

A course management system (CMS) which is a free open source for online researchers, MOODLE (Modular Object Oriented Dynamic Learning Environment) is used for this research. A real-world dataset from the usage data of course management Moodle is used for the research experiments. The learning dataset consists of 1600 lending records from 600 learners on 3700 learning materials that contain timestamp and rating information from each record. The dataset is divided to training and testing sets where, 75% was used as training set and 25% as testing set.

As used by several researchers, this research also used precision and recall as evaluation metrics for the recommender performance. These researchers focused on accuracy metrics that estimates the fraction of relevant items actually recommended (recall) or the fraction of the recommended items that are actually relevant (precision).

The standard definition of recall that is used in information retrieval settings is given by:

$$recall = \frac{|relevant \wedge retrieved|}{|retrieved|} \quad (14)$$

3.1. Impact of Neighborhood size in the CF Component

This research carried out an experiment to determine the sensitivity of the neighborhood size in the CF component with respect to F1. The numbers of neighbors are changed and the corresponding F1 metric are computed. Optimal choice of neighborhood size was set to 18 and our finding is that the number of neighborhood sizes does not affect the quality of top-N recommendations.

3.2. Impact of Number of Recommended Materials and λ in the CBF Component

Two parameters may affect the recommendation result in the CBF component. Experiment was carried out to determine the sensitivity of the parameters where number of recommendations and λ where varied and the corresponding F1 metric is computed. From the result shown in Figure 2, it can be observed that both the number of recommendations and λ affects the quality of the CBF based recommendations. Best results are obtained where numbers of recommendations are 5 or 6 and $\lambda = 0.3$.

3.3. Impact of Recommendation Generation Method

Experiment was performed to compare relative performance of all the seven approaches which included CF, CBF, NUR, MSV, MFV, MFV-MSL, MSV-MSL methods in generating recommendations. Figure 1 shows the relative performance of these approaches may change with different number of recommendations. The experiment shows that five out of these approaches have better performance.

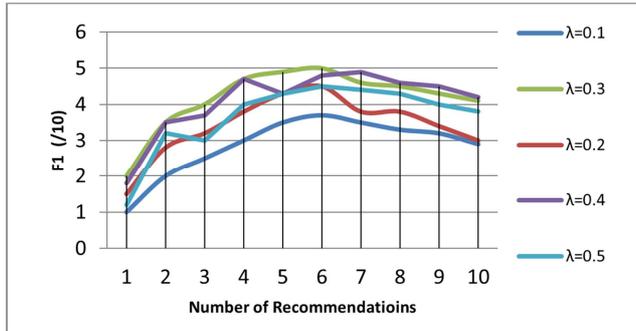


Figure 1. Impact on the number of recommendations and λ in the CBF component with respect to F1.

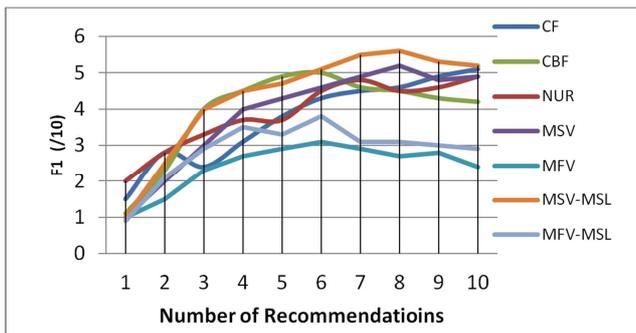


Figure 2. Comparison of various recommendation methods with respect to F1.

3.4. Quality Comparison

As soon as the optimal values of different parameters of the recommendation approaches are obtained, recommendation quality of the different attribute-based can be compared with memory-based method CF which uses similarity between items for recommendation generation. Table 1 summarizes the experimental results and it clearly shows that attribute-based approaches outperform the item-based.

Table 1. Comparison between attribute-based and item-based approaches.

Approach	Precision	Recall	F-Measure
CF	0.673	0.341	0.455
CBF	0.698	0.357	0.472
NUR	0.584	0.324	0.417
MSV	0.972	0.384	0.509
MSV-MSL	0.754	0.402	0.524

4. Conclusion

Despite effort made by several researchers to improve e-

learning recommendations, there still exist gap in recommending research materials in digital libraries. This research presents a hybrid recommender that takes into account both the item-based as well as the attribute-based approach for noble and more accurate recommendations. Although table 1 shows NUR displaying low performance but Figure 2 clearly indicates that NUR performed better in the absence of fewer learning materials. Both MFV and MFV-MSL are removed from the algorithm since they clearly showed poorer performance while MSV and MSV-MSL producing the best output. Therefore CF, CBF, MSV, MSV-MSL and NUR are combined to complement one another in order to have maximum performance at every instance. In other words, even though attribute-based approaches showed more promising recommendation results, the Nearest User Rated material (NUR) played a vital role in, especially the *new user* or *new item* situation. Evaluation metrics was used to measure the performance or effectiveness of the Ensemble recommender along with other traditional recommenders which showed promising results.

In summary, the main contributions drawn from the research includes extension of previous or traditional algorithms to work with implicit datasets as well as perform additional tasks of overcoming various limitations. Also the research contributed an extensive accuracy comparison where traditional algorithms were put to test. However, there are possibilities these performances can be improved by future researchers using different dataset, feature combinations and/or when other parameter settings are explored. The researchers also suggest that future work should be added to the algorithm to include researcher's suggestions for information sharing.

References

- [1] Adomavicius, G., & Tuzhilin, A. (2001, August). Extending recommender systems: A multidimensional approach. In Proceedings of the International Joint Conference on Artificial Intelligence (IJCAI-01), Workshop on Intelligent Techniques for Web Personalization (ITWP2001), Seattle, Washington, August (pp. 4-6).
- [2] Adomavicius, G., & Tuzhilin, A. (2005). Toward the next generation of recommender systems: A survey of the state-of-the-art and possible extensions. *IEEE Transactions on Knowledge & Data Engineering*, (6), 734-749.
- [3] Agosti, Maristella, & Tasso, Carlo. (2016). A Content-Based Approach to Social Network Analysis: A Case Study on Research Communities. Paper presented at the Digital Libraries on the Move: 11th Italian Research Conference on Digital Libraries, IRCDL 2015, Bolzano, Italy, January 29-30, 2015, Revised Selected Papers.
- [4] Alotaibi, Faisal Fahd. (2016). Perceived dimensions of quality in higher education. Wichita State University.
- [5] Anastassiou, G. A. (2010). *Fuzzy mathematics: Approximation theory* (Vol. 251, pp. xii+443). Heidelberg: Springer.

- [6] Anderson, N. L., Polanski, M., Pieper, R., Gatlin, T., Tirumalai, R. S., Conrads, T. P.,... & Lobleby, A. (2004). The human plasma proteome a nonredundant list developed by combination of four separate sources. *Molecular & Cellular Proteomics*, 3 (4), 311-326.
- [7] Anselone, P. M. (1971). *Collectively compact operator approximation theory and applications to integral equations*. Prentice Hall.
- [8] Benesty, J., Chen, J., Huang, Y., & Cohen, I. (2009). Pearson correlation coefficient. In *Noise reduction in speech processing* (pp. 1-4). Springer, Berlin, Heidelberg.
- [9] Boran, F. E., Genç, S., Kurt, M., & Akay, D. (2009). A multi-criteria intuitionistic fuzzy group decision making for supplier selection with TOPSIS method. *Expert Systems with Applications*, 36 (8), 11363-11368.
- [10] Burke, Robin. (2002). Hybrid recommender systems: Survey and experiments. *User modeling and user-adapted interaction*, 12 (4), 331-370.
- [11] Champiri, Zohreh Dehghani, Salim, Siti Salwah Binti, & Shahamiri, Seyed Reza. (2015). The Role of Context for Recommendations in Digital Libraries. *International Journal of Social Science and Humanity*, 5 (11), 948.
- [12] Chang, Na, Irvan, Mhd, & Terano, Takao. (2016). Designing a Hybrid Recommendation System for TV Content Intelligent Decision Technology Support in Practice (pp. 217-229): Springer.
- [13] Chintagunta, P., Erdem, T., Rossi, P. E., & Wedel, M. (2006). Structural modeling in marketing: Review and assessment. *Marketing Science*, 25 (6), 604-616.
- [14] de Magalhães, Cleyton Vanut Cordeiro, Souza, E, Neto, JSC, & Vilar, Guilherme. (2013). Recommender Systems: an Experience With NenNet Health-Care Social Network. Paper presented at the eTelemed: The Fifth International Conference on eHealth, Telemedicine and Social Medicine.
- [15] Fan, Z. P., Ma, J., & Zhang, Q. (2002). An approach to multiple attribute decision making based on fuzzy preference information on alternatives. *Fuzzy sets and systems*, 131 (1), 101-106.
- [16] Frakes, W. B., & Baeza-Yates, R. (Eds.). (1992). *Information retrieval: Data structures & algorithms* (Vol. 331). Englewood Cliffs, NJ: prentice Hall.
- [17] Fox, Edward A, Eaton, John L, McMillan, Gail, Kipp, Neill A, Mather, Paul, McGonigle, Tim,... DeVane, Brian. (1997). Networked digital library of theses and dissertations: An international effort unlocking university resources. *D-lib magazine*.
- [18] Hu, R., & Pu, P. (2011, October). Enhancing collaborative filtering systems with personality information. In *Proceedings of the fifth ACM conference on Recommender systems* (pp. 197-204). ACM.
- [19] Igbe, Tobore, & Ojokoh, Bolanle. (2016). Incorporating User's Preferences into Scholarly Publications Recommendation. *Intelligent Information Management*, 8 (02), 27.
- [20] Koren, Y. (1). The bellkor solution to the netflix grand prize. 2009. URL http://www.netflixprize.com/assets/GrandPrize2009_BPC_BelIKor.pdf.
- [21] Leydesdorff, L. (2008). On the normalization and visualization of author co-citation data: Salton's Cosine versus the Jaccard index. *Journal of the American Society for Information Science and Technology*, 59 (1), 77-85.
- [22] Leuski, A., & Allan, J. (2000, April). Improving interactive retrieval by combining ranked lists and clustering. In *Content-Based Multimedia Information Access-Volume 1* (pp. 665-681). LE CENTRE DE HAUTES ETUDES INTERNATIONALES D'INFORMATIQUE DOCUMENTAIRE.
- [23] Linden, Greg, Smith, Brent, & York, Jeremy. (2003). Amazon.com recommendations: Item-to-item collaborative filtering. *IEEE Internet computing*, 7 (1), 76-80.
- [24] Lops, P., De Gemmis, M., & Semeraro, G. (2011). Content-based recommender systems: State of the art and trends. In *Recommender systems handbook* (pp. 73-105). Springer, Boston, MA.
- [25] Melville, N., Kraemer, K., & Gurbaxani, V. (2004). Information technology and organizational performance: An integrative model of IT business value. *MIS quarterly*, 28 (2), 283-322.
- [26] Mooney, Raymond J, & Roy, Loriene. (2000). Content-based book recommending using learning for text categorization. Paper presented at the Proceedings of the fifth ACM conference on Digital libraries.
- [27] Pandya, S, Shah, J, Joshi, N, Ghayvat, H, Mukhopadhyay, SC, & Yap, MH. (2016). A novel hybrid based recommendation system based on clustering and association mining. Paper presented at the Sensing Technology (ICST), 2016 10th International Conference on.
- [28] Pazzani, M. J. (1999). A framework for collaborative, content-based and demographic filtering. *Artificial intelligence review*, 13 (5-6), 393-408.
- [29] Rendle, S, Tso-Sutter, K, Huijzen, W, Freudenthaler, C, Gantner, Z, Wartena, C,... Wibbels, M. (2011). Report on state of the art recommender algorithms (update). *My-Media public deliverable D*, 4.
- [30] Resnick, Paul, & Varian, Hal R. (1997). Recommender systems. *Communications of the ACM*, 40 (3), 56-58.
- [31] Schafer, J Ben, Konstan, Joseph A, & Riedl, John. (2001). E-commerce recommendation applications Applications of Data Mining to Electronic Commerce (pp. 115-153): Springer.
- [32] Severyn, A., & Moschitti, A. (2012, August). Structural relationships for large-scale learning of answer re-ranking. In *Proceedings of the 35th international ACM SIGIR conference on Research and development in information retrieval* (pp. 741-750). ACM.
- [33] Shinde, Shraddha B, & Potey, Mrs MA. (2016). Research Paper Recommender System Evaluation Using Coverage.
- [34] Sintorn, E., & Assarsson, U. (2008). Fast parallel GPU-sorting using a hybrid algorithm. *Journal of Parallel and Distributed Computing*, 68 (10), 1381-1388.
- [35] Spiegel, S. (2009). A hybrid approach to recommender systems based on matrix factorization. Department for Agent Technologies and Telecommunications, Technical University Berlin.

- [36] Su, Xiaoyuan, & Khoshgoftaar, Taghi M. (2009). A survey of collaborative filtering techniques. *Advances in artificial intelligence*, 2009, 4.
- [37] Tata, S., & Patel, J. M. (2007). Estimating the selectivity of tf-idf based cosine similarity predicates. *ACM Sigmod Record*, 36 (2), 7-12.
- [38] Vailaya, Aditya, Figueiredo, Mário AT, Jain, Anil K, & Zhang, Hong-Jiang. (2001). Image classification for content-based indexing. *IEEE transactions on image processing*, 10 (1), 117-130.
- [39] Witten, Ian H, Nevill-Manning, Craig G, & Cunningham, Sally Jo. (1996). Building a digital library for computer science research: technical issues. *Australian Computer Science Communications*, 18, 534-542.