UIJRT ISSN: 2582-6832

United International Journal for Research & Technology

Volume 04, Issue 07, 2023 | Open Access | ISSN: 2582-6832

An Enhancement in Result Retrieval Process in Knowledge-Based Recommendation That Uses Case-Based Reasoning by Utilizing Feature Weighting and Feature Normalization

Lance Gerard V. Ronquillo[1], Gabriel C. Villa[2], Richard C. Regala[3], Fernandez C. Pineda[4], and Herminiño C. Lagunzad[5]

[1,2]Student, College of Engineering - Pamantasan ng Lungsod ng Maynila

[3,4,5]Professor, College of Engineering - Pamantasan ng Lungsod ng Maynila

***Abstract*—** Since the Information Age came, a tremendous quantity of information has been made available online. The data may consist of general information, different fields of study, or even E-Commerce. This immense quantity of information often leads to a phenomenon known as information overload. The phenomenon led to the creation, development, and enhancement of different types of recommendation systems. Knowledge-Based Recommendation System (KBRS) suffers significantly in its performance since KBRS relies on user input and does not use other user preferences such as liked, visited, and trends. This study proposes an enhancement of the result retrieval process in the KBRS method that uses Case-Based Reasoning. The aim is to improve the recommendation process using Feature Weighting, Feature Normalization, and Weighted Cosine based on a study conducted by Knowledge/ Domain Experts in real-estate recommendation systems. The results demonstrate significant improvements in performance metrics such as Precision and NDCG, providing promising directions for future studies and practical implications in enhancing user satisfaction and engagement.

***Keywords*—** Knowledge-Based Filtering, Case-Based Approach, Feature Weighting, Feature Normalization.

I. INTRODUCTION

In the current Information Age, a diverse range of information covering general topics, science, news, and even shopping is readily accessible, coming at an astonishingly fast rate. Nevertheless, this ease comes with a cost, wherein a person experiences a state of being overflowed with information for a person's capacity to process. When information is hard to filter due to its enormous amount, recommendation systems are now needed. A recommendation system solves the problem of information overload by providing users with personalized service recommendations that utilize different filtering methods [1]. By considering different perspectives, the recommendation system aims to have a recommended item/ service based on the user's past behavior [2]. There are several types of recommendation systems, and these approaches are the following: Collaborative, Content-Based, Demographic-based, Utility-Based, Knowledge-Based, and Hybrid recommendation systems [3].

There are three main types of recommendation systems: Collaborative-Filtering, Content-Based, and Knowledge-Based Recommendation Systems. Collaborative Filtering evaluates products that use users' ratings from their historical data. It functions by developing a database that stores users' preferences. [4] Content-Based Recommendation System tries to give recommendations to a user through the usage of its past liked items. By analyzing its previously rated and liked items or products, the system could eventually build a model or personalized profile based on the user's preferred features [5]. Knowledge-Based Recommendation System (KBRS) does not rely on the item description, ratings, and user trends but on more profound knowledge about the items offered to the user. KBRS usually uses semantic knowledge to describe it more in detail, allowing a different way of recommending process.[6]

A KBRS must have a solid grasp of its product domain to generate practical user recommendations. The system must have the capability to be able to weigh essential features and be able to access the knowledge base where it is stored inferentially. Consequently, a KBRS necessitates knowledge engineering with concomitant challenges [7].

The study's main objective is to improve a KBRS that uses Case-Based Reasoning (CBR) in providing clear and transparent recommendations to the user, enhancing the rationality of the recommendation process. In order to attain the desired result, the researchers used Feature

Normalization, Feature Weighting, and Weighted Cosine to give a range of recommendations to the users.

II. RELATED WORKS

In this section, the researchers introduced the recommendation system, knowledge-based filtering, and case-based approach to building a recommender system. Discussed in this section are the strengths and weaknesses of the approach.

A. Recommender System

Recommendation systems aim to solve a phenomenon called information overload. It aims to give the user recommendations of items or products using user input or preference as the basis for the recommendation process [8]. The primary purpose of a recommendation system is to generate suitable recommendations according to the user's preferences and input. Real-world examples of industry-strength recommender systems operation include book and movie recommendations on Amazon and Netflix [9].

B. Knowledge-Based Recommendation System

KBRS differentiates itself from other forms of recommendation systems by employing different techniques to generate a recommendation. A KBRS generates recommendations based on domain-specific knowledge. A user will receive a recommendation based solely on his profile; the behavior of other users will not be considered at all, or if it is, it will not play a significant role in determining the recommendation. [10]. The knowledge-based filtering method uses the users' knowledge to generate a recommendation for an existing product using a knowledge-based approach. It has two main approaches: the case-based approach and the constraint-based approach, which are both similar in the recommendation process but differ in how they use the knowledge provided by the user in the system. In contrast, a constraint-based recommendation approach relies on a set of recommendation rules. In case-based recommendation approach uses the specified customer requirements to get the similarity metrics of other items. [11]. The KBRS does not depend on any reviews/critiques made by other users. Hence, its approach is more sensitive than other recommendation systems regarding changing preferences of the user [12].

C. Case-Based Approach

CBR is a type of methodology used in problem-solving. CBR works by finding a previous case that is similar and already solved, and the previous case will be used to solve the new problem [13]. Early CBR systems rely on concrete experiences rather than having the possibility to have problem-solving knowledge in the form of codified rules and robust domain models. Early CBR systems can be distinguished from more traditional solving techniques. Researchers and companies used early CBR systems in classification and problem-solving tasks [14].

D. Feature Normalization

Normalization involves categorizing the attributes of data stored inside the model to strengthen the connection between different entity kinds. The flexibility of the data can be increased through normalization [15]. The Standard Scaler (SS) approach normalizes each feature by eliminating its mean and scaling its variance to one. Because the normalized value is determined solely by the mean and variance, it has advantages such as being linear, reversible, rapid, and highly scalable. On the other hand, Standard Scaler has certain drawbacks, including high sensitivity to outliers and a preference for regularly distributed data [16].

E. Feature Weighing

In most cases, it is common knowledge that not all traits are equally indicative of the underlying pattern, particularly in problems that arise in real-world scenarios. The primary feature of feature weighting is to alter or give weights to the elements so that they contribute to the Machine Learning algorithm metric in proportion to their projected level of importance [17].

F. Calculating Similarities

Calculating similarities between two texts is fundamental to various text-mining applications. For instance, if we had a foolproof mechanism for determining the degree to which two pieces of text are comparable, we might construct the perfect information retrieval system. The cosine, which determines the angle between two vectors, is the most often used metric currently available [18].

III. METHODOLOGY

A. Research Design / Experimental Setup

The study proposes modifying the existing Knowledge-based Filtering method, which employs CBR. This modification is achieved by incorporating feature normalization and weighted cosine similarity in the information retrieval process. The experimental setup involves implementing the modified method on a real estate property dataset. The study employs the Python programming language and relevant libraries for implementing these modifications.

B. Data Collection

The dataset for this study, comprising information about real estate properties, was sourced from Kaggle.com. Specifically, it includes properties listed for sale on the website Lamudi. The dataset comprises 9000 property entries, each detailing the location, price, number of bedrooms and bathrooms, floor and land area, and latitude and longitude coordinates [19].

C. Data Preprocessing

Before retrieving recommendations, the dataset underwent data cleaning procedures to ensure data quality and consistency. Listed is the pre-processing done with the dataset.

- Duplicate entries were identified and removed from the dataset to eliminate redundancy.
- Missing values in the dataset were handled by giving default values of zero or an empty string].

D. Algorithms, Models, Techniques

The modified Knowledge-based Filtering method incorporates feature normalization and weighted cosine similarity. Feature normalization involves scaling the numerical features using the Standard Scaler. In contrast, weighted cosine similarity applies attribute weighting based on knowledge engineering. A team of researchers found that users that find real estate on different websites selected the following queries: location, price, and housing unit properties [20].

The formula for feature normalization is as follows:

$$z = \frac{x - \mu}{\sigma}$$

Where:

μ = mean

σ = standard deviation

The formula for Weighted Cosine Similarity is as follows:

$$\frac{\sum_{i=1}^{n} w_i x_i y_i}{\sqrt{\sum_{i=1}^{n} w_i {x_i}^2}\sqrt{\sum_{i=1}^{n} w_i {y_i}^2}}$$

Where:

w = weights

x = user input

y = property/item

The algorithm for the modified Knowledge-based filtering method is:

1.) User Input Acquisition: Obtain the user input, which consists of preferences or criteria provided by the user.
2.) Preprocessing Techniques:
    a. Handle Null Values: Exclude attributes with null values from further computations to ensure data completeness.
    b. Categorical Encoding: Apply OneHotEncoder to transform categorical values into binary representations, allowing for numerical computations.
    c. Feature Normalization: Normalize the numerical values of the user input using StandardScaler, ensuring that each attribute contributes equally during similarity computation.
    d. Feature Weighting: Apply feature weights to different attributes of the user input, emphasizing the significance of specific attributes in the recommendation process.
3.) Similarity Metric Computation: Calculate the similarity metric, typically weighted cosine similarity, between the preprocessed user input and the items in the dataset. This step measures the similarity between the user input and the different items.
4.) Recommendation Generation:
    a. Determine the Number of Recommendations: Select the desired number of recommendations to be generated, typically denoted as k.
    b. Retrieve Similar Cases: Identify the k most similar items based on the computed similarity metric.
5.) Evaluation Metric Computation: Assess the performance of the recommendation system by computing relevant evaluation metrics, such as precision, recall, or normalized discounted cumulative gain (NDCG). These metrics measure the accuracy and relevance of the recommendations provided by the system.
6.) Case Base Update: Save the recommended solution and the evaluation metric results, to the case base. This step allows for the storing and retrieving of past recommendations and their associated performance metrics.

To test the performance of the modified system, it ran through 100 system-generated user inputs. The system-

Volume 04, Issue 07, 2023 | Open Access | ISSN: 2582-6832

generated user inputs consisted of different user preferences. Each of the user input was then given recommendations based on the preferences, and the evaluation metrics were computed based on the results.

E. Evaluation Metrics

The system's performance is evaluated using Precision@k, Recall@k, and Normalized Discounted Cumulative Gain (NDCG)@k metrics. Each metric is critical in assessing the recommendation system's performance.

Precision@k is significant as it evaluates the relevance of the top k recommendations. The metric for Precision@k is as follows:

$$\frac{\#\ of\ recommended\ items\ @k\ that\ are\ relevant}{\#\ of\ recommended\ items\ @\ k}$$

A higher score indicates a higher ratio of relevant suggestions within the top k recommendations.

Recall@k assesses the fraction of relevant recommendations within the top k recommendations out of the total relevant recommendations. The metric for Recall@k is as follows:

$$\frac{\#\ of\ recommended\ items\ @k\ that\ are\ relevant}{total\ \#\ of\ relevant\ items}$$

A higher score signifies the system's efficiency in suggesting a broader range of relevant items.

NDCG@k is important for determining the quality of the ranking of recommendations. It considers both the relevance of recommendations and their rank order. The metric for NDCG@k is as follows:

$$nDCG_k = \frac{DCG_k}{IDCG_k}$$

Where:
DCG = Discounted Cumulative Gain, the sum of the relevance scores of th recommended items, discounted at each rank.
IDCG = Ideal Discounted Cumulative Gain, the highest possible DCG@k that can be obtained for a given set of relevance scores.

A higher score indicates a more accurate ranking of items according to their relevance.

The calculated Precision@k, Recall@k, and NDCG@k values are used to compare the performance of the proposed method to the baseline methods. These comparisons demonstrate the proposed method's efficacy in terms of precision, recall, and ranking quality.

IV. RESULTS AND DISCUSSION

A. Performance Results and Comparison to Baseline Method

The proposed modification significantly improved the experimental evaluation, outperforming the baseline. The data for both the baseline and the modified methods across different k-values are summarized in Table 1.

Table I. *Performance Results of the Modified Method*

	Baseline			**Modified**		
	Prec.	**Recall**	**NDCG**	**Prec.**	**Recall**	**NDCG**
K=1	82.00%	4.39%	82.00%	100%	4.46%	100%
K=3	68.00%	6.70%	83.92%	83.17%	6.69%	99.68%
K=5	62.60%	8.09%	83.36%	76.75%	8.16%	98.46%
K=10	56.60%	10.35%	82.94%	66.27%	10.12%	97.75%

For k = 1, the Average Precision@k for the baseline method was 0.82, significantly improved to 1.0 in the modified method. The Average Recall@k slightly increased from 0.0439 in the baseline to 0.0446 in the modified method. The Average NDCG@k, a crucial measure of the ranking quality, improved from 0.82 to 1.0, signifying a more accurate ranking of properties.

For k = 3, the modified method improved the Average Precision@k from 0.68 in the baseline to 0.83. The Average Recall@k remained comparable, with a slight decrease from 0.0670 in the baseline to 0.0669 in the modified method. The Average NDCG@k score displayed a remarkable uplift from 0.8392 to 0.9968.

Similar improvements were observed for k = 5 and k = 10. For k = 5, the modified method improved the Average Precision@k from 0.626 to 0.769, the Average NDCG@k from 0.8336 to 0.9846 while maintaining a similar Average Recall@k value. For k = 10, the modified method improved the Average Precision@k from 0.566 to 0.663 and the Average NDCG@k from 0.8294 to 0.9775 while keeping a similar Average Recall@k.

B. Discussion of Findings

The findings of this study provide solid empirical support for the proposed modification to the Knowledge-based Filtering method. As evidenced by the results (Table 1), the modified method consistently outperformed the baseline method across varying k-values, which indicates its effectiveness and adaptability.

The enhanced method increases in Average Precision@k, especially at lower k-values, which signifies an enhancement in the system's capability to present more relevant recommendations to users at the top of the list, which is particularly important in real-world applications where users may only consider a few top recommendations. While the Recall@k values remained comparable, the significant improvement in the NDCG@k scores across all k-values denotes a substantial upgrade in the ranking quality. The modifications thus have maintained the breadth of the system's ability to retrieve relevant items and significantly improved its precision and ranking accuracy.

These results provide strong empirical evidence of the potential benefits of incorporating feature normalization and weighted cosine similarity in improving Knowledge-based Filtering methods that employ CBR.

C. Implications

The implications of these findings extend to various aspects of the field of Recommender Systems, particularly Knowledge-based Filtering methods that utilize CBR.

The improvements seen in our proposed modifications, specifically in Average Precision@k and NDCG@k scores, mark a significant direction for future studies in this field. The enhancements in precision and ranking accuracy due to the implementation of feature normalization and weighted cosine similarity highlight the potential for increasing the overall quality of recommendations.

The increase in Average Precision@k suggests that the enhanced system's proficiency in providing highly relevant recommendations within the first few suggestions has been improved substantially. This attribute is vital in user-centric applications, where delivering relevant content is vital.

In addition, the increase in the NDCG@k score indicates that the items recommended are not just relevant but are also ordered effectively according to their relevance. This improvement in the quality of the ranking of recommendations can significantly enhance user satisfaction and engagement.

While the Recall@k remained relatively unchanged, it is noteworthy that the modifications maintained a good recall level while enhancing precision and ranking quality. This result highlights a balanced approach to improving the performance of recommendation systems without sacrificing the breadth of relevant items retrieved.

In summary, this study confirms the proposed modifications' effectiveness and paves the way for further advancements in recommender systems. The practical implications of these enhancements could reach far and wide, promoting greater user satisfaction and engagement in various digital platforms.

V. CONCLUSION AND FUTURE WORK

In this study, the researchers implemented enhancing the Knowledge-Based Filtering Method, which uses CBR in its result retrieval process. Based on the domain expertise of different authors regarding real estate, this expertise was used as a basis for knowledge engineering used in Feature Weighting. Feature Normalization is also used as an additional pre-processing algorithm in this enhancement. The experimental dataset is based on datasets that Kaggle.com validates.

In our proposed methodology, the researchers utilized a different pre-processing algorithm and similarity metrics to get the desired improvements since Knowledge-Based Filtering generally does not use user preferences other than the user inputs. This may give an incorrect recommendation process; thus, the proposed methodology relies on how the knowledge engineering within the filtering method is implemented.

Since KBRS relies on user inputs, similarity metrics used in the recommending process are vital in giving suitable recommendations for users to navigate. Using weighted cosine and Feature normalization helped give better results regarding Precision and NDCG performance metrics. An increase in Precision and NDCG suggests high relevance for the first few recommended items simultaneously ordered while Recall remained unchanged. This suggests that the algorithms approached the improvement needed while retaining the relevant items retrieved during the recommendation process.

As the world's economy changes during this recovery time after the pandemic, the researchers expect more problems to be found in different recommendation systems. As this study only focused on the results retrieval of KBRS that uses CBR, many steps need to be improved regarding the case-based approach, especially on retaining and revising. A much bigger and more enhanced knowledge-engineering may improve the results of the recommendation process depending on the ontologies or tags used in the system.

ACKNOWLEDGMENT

The researchers wish to give God appreciation for giving them mental, physical, and emotional stability throughout this study. To Mr. Richard Regala, for accepting the proposition as the study's adviser. To the researcher's families for their assistance throughout this period. To the friends who give the researcher motivation. To Pamantasan ng Lungsod ng Maynila - College of Engineering's faculty and staff. And The Department of Science and Technology for funding the study and giving the researchers a chance to this opportunit.

REFERENCES

[1] Isinkaye, F. O., Folajimi, Y. O., & Ojokoh, B. A. (2015). Recommendation systems: Principles, methods and evaluation. Egyptian Informatics Journal, 16(3), 261–273. doi:10.1016/j.eij.2015.06.005

[2] Das, Debashis & Sahoo, Laxman & Datta, Sujoy. (2017). A Survey on Recommendation System. International Journal of Computer Applications. 160. 6-10. 10.5120/ijca2017913081.

[3] Recommendation systems | Engati. (n.d.). Engati. Retrieved January 13, 2023, from https://www.engati.com/glossary/recommendation-systems

[4] Fayyaz, Z., Ebrahimian, M., Nawara, D., Ibrahim, A., & Kashef, R. (2020). Recommendation Systems: Algorithms, Challenges, Metrics, and Business Opportunities. Applied Sciences, 10(21), 7748. doi:10.3390/app10217748

[5] Lops, P., de Gemmis, M., & Semeraro, G. (2010). Content-based Recommender Systems: State of the Art and Trends. Recommender Systems Handbook, 73–105. doi:10.1007/978-0-387-85820-3_3

[6] Felfernig, A., Jeran, M., Ninaus, G., Reinfrank, F., Reiterer, S., & Stettinger, M. (2013). Basic Approaches in Recommendation Systems. Recommendation Systems in Software Engineering, 15–37. doi:10.1007/978-3-642-45135-5_2

[7] Tran, Thomas. (2007). Combining Collaborative Filtering and Knowledge-Based Approaches for Better Recommendation Systems.

[8] Fayyaz, Z., Ebrahimian, M., Nawara, D., Ibrahim, A., & Kashef, R. (2020). Recommendation Systems: Algorithms, Challenges, Metrics, and Business Opportunities. Applied Sciences, 10(21), 7748. doi:10.3390/app10217748

[9] Melville, P., & Sindhwani, V. (n.d.). Recommender Systems 1 Definition.

[10] Bouraga, S., Jureta, I., Faulkner, S., & Herssens, C. (2014). Knowledge-Based Recommendation Systems. International Journal of Intelligent Information Technologies, 10(2), 1–19. doi:10.4018/ijiit.2014040101

[11] Jannach, D., Zanker, M., Felfernig, A., & Friedrich, G. (n.d.). Knowledge-based recommendation. Recommender Systems, 81–123. doi:10.1017/cbo9780511763113.006

[12] Shishehchi, Saman & Banihashem, Seyed & Mat Zin, Nor Azan & Mohd Noah, Shahrul Azman. (2012). Ontological Approach in Knowledge Based Recommender System to Develop the Quality of E-learning System. Australian Journal of Basic and Applied Sciences. 6. 115-123.

[13] Lorenzi, F., & Ricci, F. (n.d.). Case-Based Recommender Systems: a Unifying View.

[14] Barry Smyth , Case-Based Recommendation. P. Brusilovsky, A. Kobsa, and W. Nejdl (Eds.): The Adaptive Web, LNCS 4321, pp. 342-376, 200

[15] Raju, V. N. G., Lakshmi, K. P., Jain, V. M., Kalidindi, A., & Padma, V. (2020). Study the Influence of Normalization/Transformation process on the Accuracy of Supervised Classification. 2020 Third International Conference on Smart Systems and Inventive Technology (ICSSIT). doi:10.1109/icssit48917.2020.9214160

[16] Ferreira, P., C. Le, D., & Zincir-Heywood, N. (2019). Exploring Feature Normalization and Temporal Information for Machine Learning Based

Insider Threat Detection. 2019 15th International Conference on Network and Service Management (CNSM). doi:10.23919/cnsm46954.2019.9012708

[17] Niño-Adan, I., Manjarres, D., Landa-Torres, I., & Portillo, E. (2021). Feature weighting methods: A review. Expert Systems with Applications, 184, 115424. doi:10.1016/j.eswa.2021.115424

[18] Li, B., & Han, L. (2013). Distance Weighted Cosine Similarity Measure for Text Classification. Lecture Notes in Computer Science, 611–618. doi:10.1007/978-3-642-41278-3_74

[19] Vinluan, P. (2022, August). Real Estate Price in the Philippines, Version 1. Retrieved form May 22, 2023 from https://www.kaggle.com/datasets/stanner0012/real-estate-price-in-the-philip

[20] Yuan, X., Lee, J.-H., Kim, S.-J., & Kim, Y.-H. (2013). Toward a user-oriented recommendation system for real estate websites. Information Systems, 38(2), 231–243. doi:10.1016/j.is.2012.08.004