Enhancing Self-Directed Learning with Personalized Online Resource Recommendations

Wijayasena W.D.N.D.T
Faculty of Computing
Sri Lanka Institute of Information Technology
Colombo, Sri Lanka
dinethwija1806@gmail.com

Senaweera T.I.S
Faculty of Computing
Sri Lanka Institute of Information Technology
Colombo, Sri Lanka
ishmasenaweera@gmail.com

Weerasekara N.N
Faculty of Computing
Sri Lanka Institute of Information Technology
Colombo, Sri Lanka
nisansith@gmail.com

Prasanna S. Haddela
Faculty of Computing
Sri Lanka Institute of Information Technology
Colombo, Sri Lanka
prasanna.s@sliit.lk

***Abstract*— Online learning has become a popular way for students to study. Among all of these approaches, online reading has assumed a significant place. While performing online reading, students may encounter obstacles such as difficulty understanding the reading content or the need for additional clarification. In these cases, learners will look for extra study materials on the internet. However, it would be impossible for them to locate the most appropriate materials for both the learner and the query that user given to the system. To overcome this problem, this strategy encourages self-directed learning by providing students with tailored learning opportunities based on the difficulty level of the user's study material that they refer to and employing modern ranking systems**.

Keywords*— *Online learning, Self-studying, self-directed learning, Learning opportunities.

I. INTRODUCTION

University students face enormous pressure and stress when it comes to their academics today. Many students find themselves struggling to keep up with a wide range of subjects to cover and a huge number of assignments and exams. Some students do not have enough time to refer to all lectures, some students may not know how to study efficiently in order to achieve good grades. Many students try to do self-learning to stay on the correct track. But unfortunately, despite all the hard work they may still get lower grades for subjects than they anticipated. There can be many reasons for this, but through our research, we found that one of the main reasons may be that the student did not fully understand their study material.

In such case, students try to find extra study resources to get an understanding about the subject or module. However, when seeking online resources learners frequently struggle to find appropriate materials to help them learn specific ideas that they wish to cover. The resources that they found might not match their expected level and not be in the standard of their study material. This problem is particularly prominent when students try to manually select a few words to form a query that can be answered by a recommendation system when they do not have a proper understanding about the reading paragraph. Most search engines transform the queries and candidate resources into vectors or bags of words, thereby neglecting the semantic topics underlying the content. Additionally, most existing information retrieval and recommendation systems rank resources solely based on their relevance to the user request, overlooking the suitability of the resources for the user's comprehension level.

The major goal of this project component is to provide an online extra resource recommending system that suggests appropriate and engaging resources based on the content and reading level of the user . This approach promotes self-directed learning by giving students individualized learning opportunities based on their level of provided paper difficulty and using modern ranking methods. The aim of this project is to develop an online resource recommendation system that uses advanced technologies to recommend the most relevant and engaging resources to enhance the students' learning experience.

The system utilizes a paragraph that in the student's study material as input, The system processes this input through a query processing section that conducts preprocessing, topic generation, and topic compression. It uses the Latent Dirichlet Allocation (LDA) topic modeling algorithm for topic generation and the cosine similarity method for topic compression. A topic model is based on the premise that when a document is about a specific topic, certain words should occur more frequently. Documents are topic mixtures, with each subject being a probability distribution over words. The generated topics are then used to retrieve relevant resources, considering the top k words in each topic. The system retrieves URLs of resources and ranks these resources based on their relevancy and difficulty level. The ranking system leverages the relevance ranking method and the Flesch Reading Ease method for assessing the difficulty of documents, while videos are evaluated using semantic analysis. These ranking scores are combined to provide the most relevant resources for the user. This research component's final product is a fully completed web application that recommends extra study resources for students.

A. Initial Survey

Initially we conducted a survey to find the how students are involved with online recommendation systems when they are doing self-learning. A survey was conducted among undergraduate students. The survey involved the distribution of a questionnaire, which required the students' participation in providing answers. The objective of the survey was to assess the level of students' acquaintance with online recommendation systems (Fig.1) and to identify any problems they encounter when utilizing existing online resource recommendation systems (Fig.2).

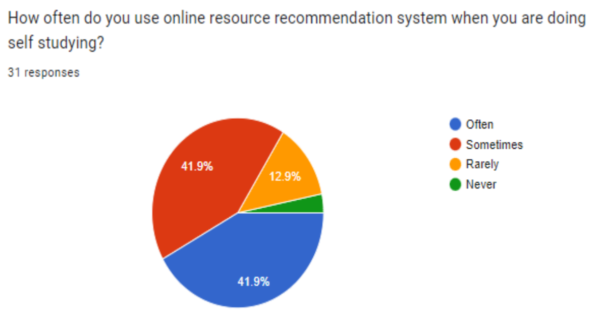

Fig. 1. User responses on usage of resource recommendation.

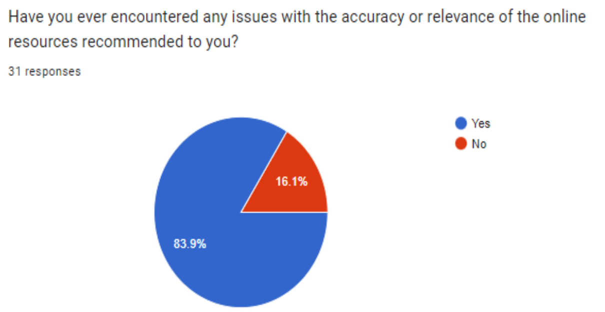

Fig. 2. User responses on the accuracy of resources suggested by existing recommendation systems.

We sought further testimonials by evaluating survey results to augment our research on the identified difficulties. These testimonies were critical in verifying the issues raised in our study. We conducted an exhaustive literature review in addition to the testimonials to acquire useful information from current scholarly sources. The combination of testimonies and the literature study provided extensive insights into the difficulties students have while using existing online resource recommendation systems.

II. Literature Review

Online resource recommendation systems have drawn a lot of attention from researchers in recent years who are investigating various technologies to propose the best resources to users. The methodologies and tools that researchers used to create online resource recommendation systems are described in the section that follows.

In research done by Xin Wei et al in 2021 investigate the creation of a personalized extra learning resource recommendation system that enhances students' learning results by utilizing educational psychology theory and artificial intelligence technologies [1]. It discusses the difficulties in delivering relevant learning resources in online learning and suggests an algorithm for recommending learning resources based on LinUCB. The algorithm modifies the proportion of exploration and exploitation during recommendation using a unique exploration coefficient.

In research done by Hongtao Sun et al in 2022 proposed a learner-model-based collaborative filtering recommending approach for online study resources, which takes into account the traits and actions of learners when making recommendations in order to improve the accuracy of the content that is suggested [2]. In order to build learner models and to calculate the user similarity this paper proposed methods such as machine learning algorithms and learner modeling algorithms. The learner's evaluation of resources is revealed by how they use online learning resources, and the algorithm used to generate suggestions is based on how similar the learners are to one another and well rated regarded the resources are.

Another research done by Danyang Shen in 2020 proposed a neural network-based recommendation model that uses concept maps to represent knowledge points, learning behavior based on learning style scales, and Bloom's taxonomy to rank educational materials [3]. Then the score is calculated using the multilayer perceptron network based on the created embedding vectors. Studies carried out for this study's experiments showed that this recommendation model produced excellent results.

In research done by Raghad Obeidat, Rehab Duwairi and Ahmad Al-Aiad in 2019 also done research reading this topic [4]. In here, depending on commonalities in their prior course experience, a collaborative recommendation system for online courses has been developed. To identify trends among the courses, this system uses data mining techniques. Then, based on the students' behavior, clustering algorithms for the datasets use to put similar students into the same cluster.

Another research done by Gina George and Anisha M. Lal in 2019 proposed an ontology-based resource recommender system [5]. Web Ontology Language (OWL) can be use to represent ontology in a model. An ontology can serve as a database's equivalent of a XML schema. There, feature extraction, pattern classification utilizing ontology mapping, and ontology merging are used as methodologies to uncover and enhance the discovery of sets of related learning materials.

In research done by Bhaskar Mondal et al in 2020 suggest a machine learning strategy for recommending relevant online courses depending on student performance history [6]. A new student is initially categorized using the k-means clustering technique by the framework based on their prior performance. The cluster will use collaborative filtering to suggest appropriate courses.

Another research done by Jun Xiao, Minjuan Wang, cBingqian Jiang and Junli Li in 2017 developed a personalized resource recommending system for online learning that makes use of association rules, content screening, and cooperative filtering to make suggestions for relevant learning resources to students taking online courses [7]. The authors highlight the value of personalized learning, which has grown more significant with the quick growth of online and mobile technology.

In research done by Honggang Wang & Weina Fu in 2020 developed a dynamic collaborative filtering algorithm-based strategy for recommending unique learning resources [8]. The authors suggest using dynamic k-nearest-neighbor and the Slope One method to optimize collaborative filtering algorithms in order to address the issues of sparse data and low scalability. They also examine the network's learning resource data sparsity in considering the outcomes of neighbor selection.

In research done by Ronghua Shi, Lei Mao, Chao Hu and Sixiang Lire in 2018 provides a novel method for recommending educational resources that is based on the learner's current knowledge structure [9]. The suggested algorithm seeks to offer students individualized recommendations that are in sync with their interests while also considering their current knowledge base. The proposed method was tested in a controlled experiment by the authors, and the findings indicate encouraging gains in both the correlation score and learning process score.

Another research done by Roshan Bhanuse and Sandip Mal in 2021 focuses on a thorough analysis of deep learning-based e-learning recommendation systems (RS) and the problems with their accuracy, scalability, cold-start, and data scarcity [10]. The authors have noted that as the use of online learning materials has increased, it has become harder for students to sift through vast amounts of data for specific knowledge. The authors emphasize the significance of learning analytics (LA) and educational data mining (EDM) in the development of e-learning RS and contend that deep learning algorithms have the potential to improve RS's efficacy in individualized instruction.

III. METHODOLOGY

A. Overall System Architecture

The external resource recommendation system aims to assist learners in enhancing their understanding of study materials by leveraging online resources. Figure 3 depicts the overall system architecture. The system utilizes the learner's personal study material as input, wherein the learner selects a paragraph from their reading material and uploads it into the system. The 'Process Query' section is responsible for processing the uploaded query passage and generating topics from it. Then, 'Retrieve Resource' section will retrieve resources that are relevant to processed query. At last , 'Resource Ranking' section ranks the best resources and provides to the students.

Fig. 3. Overall process of the system

The system allows learners to select specific paragraphs that require further clarification, thereby addressing their individual learning needs. This personalized approach ensures that learners receive tailored recommendations that directly align with their areas of difficulty and interest.

B. Procesiing the Query

Initially the user is able to select the level of resources that he/she wants (Easy, Medium and Hard) and the type of resource (Document or Video). Selected paragraph is executed in a search engine as a query. There for we have to process the query before using it in a search engine to capture the most relevant details from the given paragraph. In the query processing part, it preprocesses the paragraph by stemming and removing noisy and stop words. Figure 4 shows the architecture of the query process section.

When the user input query passage to the system, it will receive by query processing section which do preprocessing, topic generating and topic compression. The topic generating section will use topic modeling algorithms to identify patterns and topics in large collections of documents. Latent Dirichlet Allocation (LDA) is use as the topic modeling algorithm for this research. When creating several topics from the given passage some topics can be associated with similar concepts. To get rid of such duplication we use topic compression module. This will remove duplicate topics after taking the word distribution of every topic into account. To compare and identify similarity between topics, we can use correlation or similarity methods. In similarity methods, widely used two methods are cosine similarity method and Jaccard similarity method. For this research we use cosine similarity method because it's useful when examining text similarity in situations where duplication is considered.

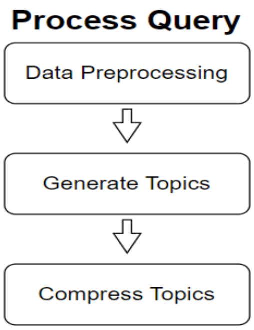

Fig. 4. Process of the query processing section

C. Resource Retrieving

Topics that are generated from the previous part will be used here to retrieve relevant resources. This considers the top k words to create query with top keywords in each topic, this new query will be run through the search engine and retrieved resources. The system retrieves the URLs from the Google search results using the Google API. This gives URLs for both documents and videos.

D. Resource Ranking

Main part for the recommendation system is to provide most relevant and accurate resources to the end user. In this system users have to select which level of contents that they want (Easy, Medium or Hard) and also they will able to select which type of resources that they want (Videos or Documents). This involves considering several factors to choose the materials that are the most valuable and pertinent. This approach ranks resources according to relevancy and measures and ranks reading difficulty of the input passage. First this approach does relevance ranking for the retrieved resources. Figure 5 shows the architecture of the query process section.

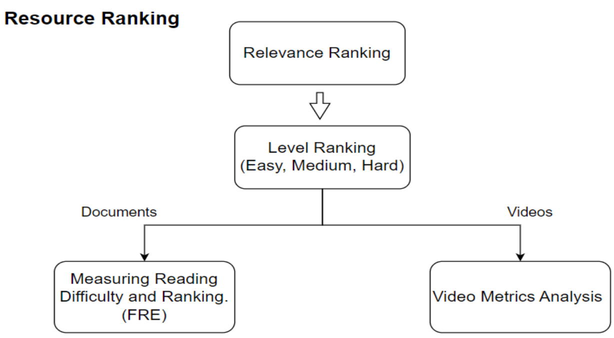

Fig. 5. Process of Resource Ranking Section

Traditional search engines use term similarity instead of topic similarity when matching documents to a query. Then the retrieved resources should be rank again according to their topic similarity not only to the topic query but also to the whole query passage. For this purpose, each set of candidate resources along with the original query passage is treated as a content bucket. For each bucket, we generate a set of topics as the semantic features with the same topic generation method discussed earlier. We use the topic representations generated for the documents and the query in each content bucket to re-rank the documents of the bucket with respect to the query. Any similarity or distance function could be utilized here. We use the cosine similarity. We can then select the top k ($k < K$) documents to show for each query topic discovered from the query passage. The final output can be biased to consider the importance of each topic in the query passage.

After ranking the resources according to relevancy, documents and videos will rank again separately in two separate methods. Document resources will be categorized according to the user interest. To find the level of the documents (Easy, Medium, and hard) we use Flesch Reading Ease (FRE) method.

Flesch reading ease test use to evaluate the readability of a text. FRE model rates the text based on two factors. This will give a score for the reading difficulty of the given resources. The range of score used is (1-100) where 1- 50 is the range of hard level, 51-60 medium level, and 60-100 easy level. Finally for each level resources will be ranking according to the relevance score that calculated previously to sort out the most relevant documents first.

When it's comes to the videos, video resources will rank again considering the metrics of the videos that can be extracted from the YouTube Data API such as like count, comment count and view count. This section provides scores between 0 and 1. Then the relevance ranking score and the metrics ranking score will merge using a formula and provide the most relevant resources to the user.

Finally, retrieved and well ranked resources that relevant to the user given content will be displayed to the user in an interface.

IV. Results and Discussion

Below are test cases used to check the quality and the performance of the work.

Test Case 1: Topic Generation Accuracy

- Objective: To ensure the LDA topic modeling algorithm correctly identifies the main topics from a given paragraph.
- Input: A known paragraph with defined topics.
- Steps:
  - Upload the paragraph to the system.
  - Let the system generate topics using the LDA algorithm.
- Expected Outcome: The topics generated should align with the predefined topics of the paragraph.

Test Case 2: Topic Compression Efficiency

- Objective: To validate that the topic compression module effectively removes duplicate topics.
- Input: A paragraph known to produce overlapping or duplicate topics.
- Steps:
  - Upload the paragraph.
  - Generate topics and pass them through the topic compression module.
- Expected Outcome: The final list of topics should have no duplicates or very similar topics.

Test Case 3: Resource Retrieval Relevancy

- Objective: To ensure that the system retrieves resources that are relevant to the topics provided.
- Input: A set of predefined topics.
- Steps:
  - Input the topics to the system.
  - Let the system retrieve resources based on these topics.
- Expected Outcome: A majority of the retrieved resources (e.g., 80%) should be directly relevant to the input topics.

Test Case 4: Reading Level Categorization

- Objective: To ensure documents are categorized correctly based on their reading level using the Flesch Reading Ease method.
- Input: A document with a known Flesch Reading Ease score.
- Steps:
  - Upload the document.
  - Let the system calculate the reading level.
- Expected Outcome: The system should categorize the document into the correct reading level (Easy, Medium, or Hard).

Test Case 5: Video Retrieval from YouTube API

- Objective: To test the effectiveness of the system in retrieving relevant videos from YouTube based on generated topics.
- Input: A specific topic known to have related content on YouTube.
- Steps:
  - Input the topic to the system.
  - Let the system retrieve videos from YouTube.
- Expected Outcome: The retrieved videos should be relevant to the input topic.

Test Case 6: Resources Retrieving Speed

- Objective: To validate the resources retrieving speed of the web application.
- Input: Paragraphs with different lengths.
- Steps:
  - Access the application via the web.
  - Let the system process the input paragraph.
- Expected Outcome: The resources retrieval should be completed within a reasonable timeframe.

TABLE 1. TEST CASES

Test Case Number	*Test Case Name*	*Outcome Summary*	*Performance Rate*
1	Topic Generation Accuracy	Identified main topics correctly in 9 out of 10 paragraphs.	90%
2	Topic Compression Efficiency	Effectively removed duplicates in 8 out of 10 paragraphs.	80%
3	Resource Retrieval Relevancy	Achieved 85% relevancy in retrieved resources for 20 topics.	85%
4	Reading Level Categorization	Correctly categorized 47 out of 50 documents.	94%
5	Video Retrieval from YouTube API	Retrieved relevant videos for 12 out of 15 topics.	80%
6	Resources Retrieving Speed	Average retrieval time was 3.5 seconds for varied paragraphs.	-

The system achieved a high accuracy rate of 90% in topic generation. This demonstrates the robust nature of the LDA algorithm when applied to diverse paragraphs, making it a reliable choice for topic modeling in the context of the system.

The system exhibited a commendable 80% efficiency in removing overlapping topics. This suggests that while the existing compression module is effective, there's potential for further optimization.

The system displayed 85% relevancy in resource retrieval. This indicates the system's algorithms are adept at scouring and categorizing web-based content but may need occasional updates to maintain relevancy due to the dynamic nature of online resources.

With a 94% accuracy rate, the system's ability to classify documents based on the Flesch Reading Ease score was found to be highly precise, making it a dependable feature for users seeking resources of a specific reading level. The system fetched relevant videos at an 80% success rate, showcasing its effectiveness. However, this also points to the potential for enhancing the precision of video retrieval, ensuring users get the most pertinent video resources.

The system's average resource retrieval speed was measured at 3.5 seconds. This rapid response time is crucial in providing real-time feedback to users, but there may be scope to optimize it further, especially during peak times to ensure consistent performance.

V. Conclusion

Based on the methodology we've explored, it's clear we've taken a comprehensive, data-centric approach to enhance online resource recommendations. By harnessing the power of data science and machine learning, we hope to revolutionize self-study. Our methodology isn't just about algorithms and code; it's about understanding the student and customizing their learning journey. As we consider the importance of catering to individual student needs, this methodology sets a strong foundation. With the digital transformation in education, this kind of research isn't just innovative, it's essential. As we move forward, we anticipate our approach becoming a cornerstone in shaping personalized, enriched learning experiences for students everywhere.

References

[1] Wei, X., Sun, S., Wu, D., & Zhou, L. (2021). Personalized Online Learning Resource Recommendation Based on Artificial Intelligence and Educational Psychology. Front Psychol, 12.

[2] Sun, H., Huang, L., Chen, Z., & Zheng, Q. (2022). Research on collaborative filtering recommendation method of online learning resources based on learner model. In 2022 IEEE 5th International Conference on Information Systems and Computer Aided Education, ICISCAE 2022 (pp. 202–206). Institute of Electrical and Electronics Engineers Inc.

[3] Shen, D. (2020). Recommendation of Online Educational Resources Based on Neural Network. In Proceedings - 2020 International Conference on Artificial Intelligence and Computer Engineering, ICAICE 2020 (pp. 32–36). Institute of Electrical and Electronics Engineers Inc.

[4] Obeidat, R., Duwairi, R., & Al-Aiad, A. (2019). A Collaborative Recommendation System for Online Courses Recommendations. In Proceedings - 2019 International Conference on Deep Learning and Machine Learning in Emerging Applications, Deep-ML 2019 (pp. 49–54). Institute of Electrical and Electronics Engineers Inc..

[5] George, G., & Lal, A. M. (2019). Review of ontology-based recommender systems in e-learning. Comput Educ, 142.

[6] Mondal, B., Patra, O., Mishra, S., & Patra, P. (2020). A course recommendation system based on grades.

[7] Xiao, J., Wang, M., Jiang, B., & Li, J. (2018). A personalized recommendation system with combinational algorithm for online learning. J Ambient Intell Humaniz Comput, 9(3), 667–677.

[8] Wang, H., & Fu, W. (2022). Personalized Learning Resource Recommendation Method Based on Dynamic Collaborative Filtering. Mobile Networks and Applications, 26(1), 473–487.

[9] Shi, R., Mao, L., Hu, C., & Li, S. (2018). A Recommendation Method of Educational Resources Based on Knowledge Structure. IEEE.

[10] Bhanuse, R., & Mal, S. (2021). A Systematic Review: Deep Learning based E-Learning Recommendation System. In Proceedings - International Conference on Artificial Intelligence and Smart Systems, ICAIS 2021 (pp. 190–197). Institute of Electrical and Electronics Engineers Inc.

[11] Aditya, A., Arpit, J., Tulasi, S. S. C., Bharath NVS, A., & Mohan, B. (2019). Performance Evaluation of Topic Modeling Algorithms for Text Classification.

[12] L. Zahrotun (2016). Comparison Jaccard similarity, Cosine Similarity and Combined Both of the Data Clustering With Shared Nearest Neighbor Method. Computer Engineering and Applications, 5(1).