Optical image recognition strategy for keyword extraction and page ranking for slide recommendation system

S.M Laique Abbas
Fakultät für Informatik
Otto von Guericke Universität
Magdeburg, Germany
syed.abbas@st.ovgu.de

Visakh Padmanabhan
Fakultät für Informatik
Otto von Guericke Universität
Magdeburg, Germany
visakh.padmanabhan@st.ovgu.de

Taruna Tiwari
Fakultät für Informatik
Otto von Guericke Universität
Magdeburg, Germany
taruna.tiwari@st.ovgu.de

Tathagata Ghosh
Fakultät für Informatik
Otto von Guericke Universität
Magdeburg, Germany
tathagata.ghosh@st.ovgu.de

***Abstract*—Understanding the learning environment, the feedback it provides, and the task requirements are necessary for an organized learning habit. Yet, as seen in our SQLValidator's activity logs, students primarily rely on trial and error to find the right solution. Few students ultimately understand the SQL language's rules; most pupils instead turn to their peers for assistance. Yet, we could lessen the time cost of poor involvement if we received instructional feedback in the form of a tip. For this reason, we added a recommendation subsystem to our SQLValidator that delivers automatic instructive feedback during online exercise sessions. We demonstrate how cosine similarity may be utilized to provide effective recommendations using a mapping between SQL tasks, course slides, and the appropriate cosine similarity**

***Index Terms*—Recommendation Systems, Optical Character Recognition (OCR), German language, Natural Language Processing, Tf-idf, cosine similarity**

I. Introduction

——————Needs to be changed The paper aims to improve the slide recommendation system from the German language course by implementing an optical character recognition (OCR) strategy for keyword extraction and page ranking. The paper discusses the importance of personalized learning and the challenges of identifying relevant content for individual learners in large-scale educational settings.

Optical character recognition (OCR) is a technology that converts scanned images or printed text into machine-readable text. It involves the use of computer algorithms to analyze an image and extract text and has numerous applications in various fields, including education. OCR technology is used in education to help digitize paper-based documents and make them accessible online. It is also used to convert printed materials into electronic text that can be edited, searched, and analyzed more easily.

There are several OCR software available in the market, including open-source options such as Tesseract, GOCR, and OCRopus, and proprietary software such as Adobe Acrobat, ABBYY FineReader, and Readiris. OCR technology has been integrated into many educational platforms, such as Learning Management Systems (LMS) and Electronic Document Management Systems (EDMS), to help make document retrieval and storage processes more efficient.

In addition to OCR technology, recommendation systems are also widely used in education to help students personalize their learning experiences. A recommendation system is a type of artificial intelligence that analyzes data to make user suggestions. In education, these systems can analyze data such as student performance, interests, and learning style to suggest appropriate resources, such as articles, videos, or courses, that match their preferences and needs.

In conclusion, OCR technology and recommendation systems are powerful tools in education that can help to streamline document processing and personalize learning experiences. Further research can focus on exploring new approaches and techniques to improve the accuracy and effectiveness of these technologies in the education domain.

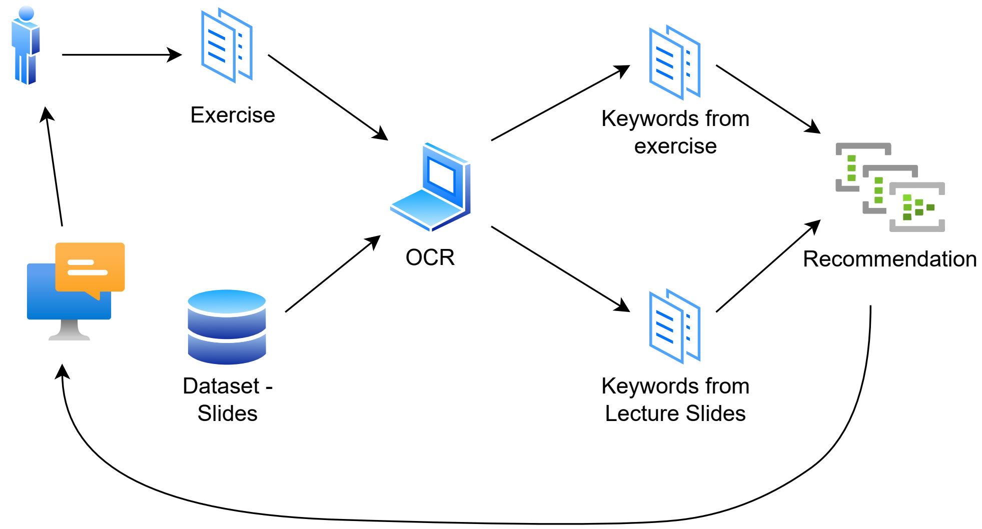

Fig. 1: Overview of system

II. Related Work

https://www.overleaf.com/project/63fe3eb4a18cebc57dcb665a Optical Character Recognition (OCR) technology has seen widespread usage in recent times among various sectors to convert printed or handwritten documents into digital documents for ease of usage, storage, and availability. OCR technology has found its implementation in different sectors like education, healthcare, finance, legal, and many more. While this led to large-scale development in OCR technology, most of the studies have been focusing either on video text retrieval or on the conversion of printed or handwritten text into digital format for improving the student learning experience. With the. This paper focuses on the development of slide recommendation systems for students based on OCR technology [1].

Based on its input, the optical character recognition system can be categorized as either handwritten or printed character recognition. Because handwritten characters are not consistent and vary depending on the person, handwritten character recognition is challenging. On the contrary, the uniformity of dimensions and their structures make the printed character recognition system that is utilized in this paper favorable [1].

Various phases of optical character recognition are as follows:

- **Pre-processing** : To enhance the image, several pre-processing techniques are used after the images are collected. As a result, the images are more useful for further actions. Here, skew removal, thinning, and noise removal techniques are applied.
- Segmentation : Here the characters are separated to make it more readable.
- **Feature Extraction** : Features from the segmented images are extracted and these features aid in character recognition.
- **Classification** : The segmented characters are further classified to several classes and categories. Various types of classification techniques are used such as structural classification that classifies characters based on the local features. Several statistical classification methods that classify based on probabilities such as Bayesian classification. Nearest neighbourhood is one the most popularly used classification methods that is used in image recognition to classify several data points.
- **Post processing** : This includes the phases that are carried out to improve the efficiency of OCR in terms of context, or working with different languages [1].

The paper Optical Character Recognition Techniques: A survey describes about the steps involved in the optical character recognition. This includes segmentation, feature extraction, classification and context analysis. It concludes that n grams introduced in OCR lead to better performance but decrease the speed of OCR. There is a trade off between speed and performance in order to optimise the option character recognition system [2].

[3]

The optical character recognition system is widely used in many applications such as healthcare, receipt text extraction, handwriting recognition and banking etc [4]. These daily applications make it more relevant in the current time. The major challenges of OCR are multilingual support environment, blurred and degraded images [4]. The quality of OCR output mainly depends on the input provided to it. In our case the input is lecture slide and exercise which are of more uniform in nature. But have some noise that is removed in pre processing. According to ”A Detailed Analysis of Optical Character Recognition Technology” pre processing plays important role in optimising the OCR output [4] [5]. The major phases of optical character recognition are segmentation, feature extraction, normalisation and classification [4] [1].

The automated indexing comprises timestamped automatic acquisition, processing, and keyword extraction from lecture slides and lecture video, according to ”Automated Hypermedia Authoring for Personalized Learning [5].” After being optimised, the captured images are sent through an optical character recognition system. It is necessary to decrease the OCR output by eliminating connectives, prepositions, and common language because the OCR process takes a lot of text from lecture presentation images. The remainder of the scanned text is processed as a set of keywords that serve as the foundation for the lecture index. The study also offers some initial assessments and measures of the index extraction technique’ effectiveness. According to the experiments, processing lengthy visual presentations takes much longer than processing text-only slides. [5].

To raise student performance, an effective teaching and learning approach is necessary. It is crucial to create a framework that may link a specific exercise task to the pertinent slides. This is accomplished using a system made up of SQL keywords and a string of characters taken from lecture slides that contain those relevant keywords [6].

Firstly the Lecture slides are converted to string and pre processed followed by term frequency and inverse term frequency calculation for each slide based on SQL keywords. Similarly exercise slides are pre processed and term frequency and inverse term frequency is calculated for the SQL keywords detected from the task. Cosine similarity is calculated based on two word vectors [6] [7]. If the angle between the vectors is low then the there is more similarity. So the relevant exercise tasks and slides are mapped for similarity and the recommendations for that task is provided when the user engages with it [6].

Tesseract is a more adaptable optical character recognition engine since it can accept a range of inputs, including PDFs and scanned documents, according to ”An Introduction of the Tesseract OCR Engine” [8]. Because it employs cutting-edge image processing methods, the Tesseract OCR engine provides great accuracy. The fact that the tesseracr OCR engine supports hundreds of different languages is one of its main features. Because of this, it may be used for both the German language and many other languages [8].

The paper ”Open source optical character recognition for historical research” gives the performance of customized OCR against open-source commercial OCR tools like tesseract OCRopus in the area of digitizing historical documents. The authors discuss the challenges faced by conventional OCR tools in the area of historical research since those tools are optimized for large-scale commercially relevant documents [9]. They suggest that an OCR which is customizable will be much more beneficial rather than the conventional ones mentioned for extracting good-quality of information from historical data. The various steps in the OCR engine were discussed that includes preprocessing, layout analysis, processing, and post-processing [9].

The OCR tools that are being used for the test are Tesseract 3.0 and OCRopus toolkit. Most commercial engines including them expect the input to their engine in highly optimized binary form, but OCRopus provides preprocessing tools which are customizable. Hence OCRopus was considered and was used in the OWP workflow. In the first case study with historical British newspapers, researchers were given the option to manually interpret the steps within the OCR cycle. At the same time, a trade-off was maintained with automation as well to balance the human workload. The other case study discusses enriching the OWP workflow with extra semantic information to obtain context-rich outputs. The final conclusion says that the best approach for an OCR engine in the case of historical data is to integrate the robustness of commercial engines and their capacity to read a broader spectrum of characters with the customizable pre-processing, layout analysis would give the most optimum results [9].

III. System Schematic

Several tools and techniques have been used to implement the workflow shown in schematic diagram. These include python, python libraries such as python image library, Nltk package. We used visual studio code and pycharm for development environment. PyQt is used for user interface has been used in order to provide optimized and convenient user interface to students as well as administrator.

The system schematic is divided into three major modules as Optical character recognition module, Feature extraction module and Recommender module. This in turn also has different administrator accessibility and users accessibility. The working of these modules is given below.

A. Administrator accessibility

The individual who holds the most rights is the administrator, who can manage OCR operations, input exercises, and upload lecture slides. The keyword dictionary belongs to the administrator. It includes of the numerous SQL terms that appear in German-language exercise questions and lecture presentations. In order to offer precise keywords for feature extraction, several German words are additionally translated into English SQL keywords [10]. The administrator may choose the lectures that correspond with a given exercise in order to offer recommendations based on those particular

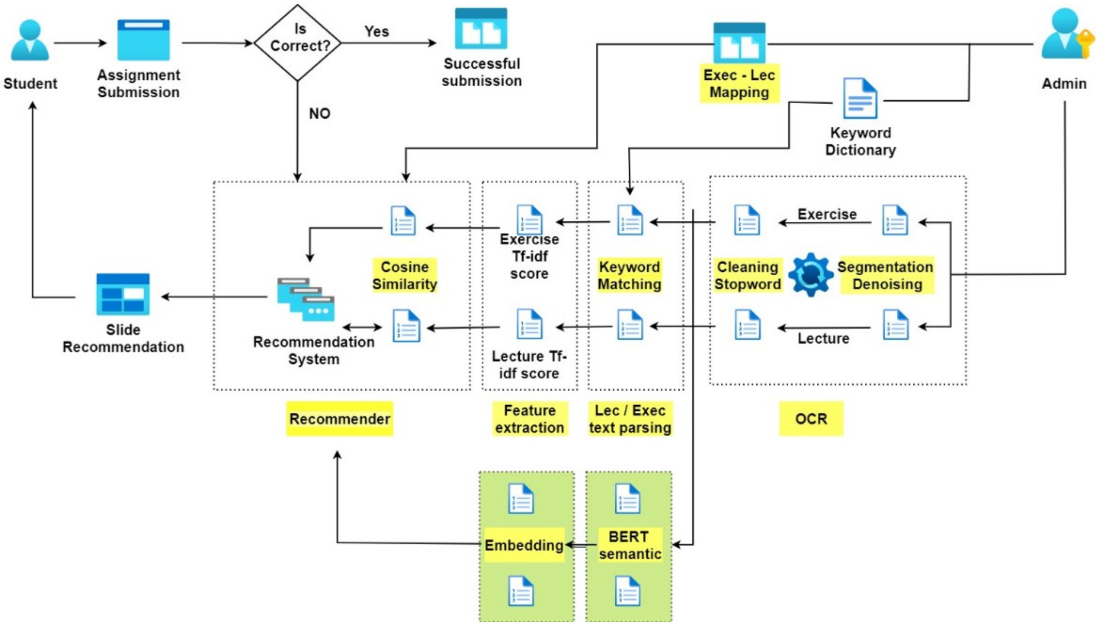

Fig. 2: Schematic representation of implementation

lectures. Additionally, the administrator has the option of selecting between English or German as the recommended language.

The administrator uses the user interface to feed the keyword mappings for each SQL exercise [10]. The administrator can then decide which lecture slides should be taken into account for that specific exercise and offer recommendations. The administrator can then initiate the OCR, and for each question containing SQL keywords the page numbers of lecture slides is generated as a recommendation.

B. User accessibility

Only recommendations are permitted under the user's rights. If the user submits the exercise solution correctly to the SQL validator, the submission is successful. If the submission is erroneous, a recommendation is made to the user, who is then given the lecture slides that should be taken into account for that specific task. The user can then review the suggested lecture slide presentation and submit the appropriate solution.

C. Optical character recognition

1) Exercise slides: Administrator feeds OCR with the exercise and lecture slide PDFs. There are multiple tasks on each exercise sheet, and those tasks and their sub-parts are scattered out over several PDF pages. Each task is separated and routed via the Pre-processing pipeline using OCR with Page Segmentation Mode -6. Stop-words are eliminated using the nltk package during the pre-processing stage by going over each question one at a time. When stop words are eliminated, we continue scan for the keywords and determine whether they appear as a value in any of the dictionary's keys. Every time a word that is mapped to that keyword appears, the count for that keyword is updated. The next step is to create a data frame with an index serving as the exercise number-question number and a column containing the keywords repeated in accordance with the count.

2) Lecture slides: Additionally, the OCR pre-processing pipeline is used to segment the pages of the lecture slides using Page Segmentation Mode -6. The logo provided in each slide is here hidden using the Python image library because it is perceived to be noise by the OCR engine. In order to use it afterwards, headers and page numbers are also separated. The nltk package is used to remove stop-words during the pre-processing phase. The specified keyword dictionary is then used to calculate the number of keywords on each page. Finally, a data frame with the page numbers of the lecture slides and the keywords found on each page is generated [6].

D. Feature extraction and representation

The term frequency and inverse term frequencies are calculated for the each keywords that has been detected in the exercise as well as lecture slides.

E. Recommendation

The cosine similarity is computed 0.2 threshold

IV. Implementation

Lecture Slides- The lecture slides are kept in a folder and the admin can trigger the OCR process by giving the path with the option Lecture from the drop-down. We convert all the slides to images and apply some preprocessing steps before saving them. Image Preprocessing: Since our lecture slides are already clean enough for identifying the text, only cropping the image to remove the logo, and unwanted information is required. Hence with the python provided PIL package, we mask certain areas of all the lecture slides so that only the useful parts including the title and content of the slides are being present in the image.

Extracting text: Once the images are saved, each image is read one by one by using the PIL and by tesseract. Since slides would contain bullet points, it should be handled accordingly, hence the page segmentation mode(psm) of 6 is initialized inside the pytesseract function. The language option available inside them also allows us to extract proper text from the slides,hence we chose deu and eng for German and English respectively. This ensures that the OCR engine classifies the text accurately and avoids ambiguity for German characters such as umlauts with English characters such as i,j, etc.

Text Preprocessing: From the saved images, the text is extracted and converted to lowercase. Then the stopwords are removed from the extracted text using the NLTK package. There are separate packages for German and English. A difficulty that we faced in this phase was, some sql keywords such as between, where, not in,etc are sql keywords are removed since they are present in the NLTK stopwords. This is handled by manually excluding them from looking at the results.

Text Processing: Once the cleaned text and preprocessed text are obtained, they are stored in the form of a data frame with the PDF page number as the key and keywords column. The keywords column contains the words according to their count present in each slide. This will be saved in the form of a csv file for further computations.

Maintaining dictionary- A dictionary of keywords that could be sql based is stored separately for German and English. Words such as 'full outer join', 'full-outer-join' is mapped to a single word 'fullouterjoin' to avoid the possibility of treating them as separate words. If it is treated as separate words, then the slides containing the 'join' keyword could also be recommended. This same the for other keywords such as 'group by','order by',etc.

Exercise Slides: Tht exercise slides contains similar steps as of the the lecture slides. Here we can skip the image preprocessing part since they are already good for extracting text. Extracting text : In this phase, we primarily focus on segmenting the text properly since a task could have sub-tasks and it should be considered along with the main part to get all the information regarding that question. Hence the primary focus is given to the parsing and the same pipeline as that lecture slide is used.

Once the text is cleaned, and processed and the data frame is obtained containing the keyword present in a particular task, the data frame is saved in the form of csv for further processing.

MORE DETAILS REGARDING EX-LEC filtering and similarity computaiton needs to be written

A. *Method selection*

B. *Method description*

V. Evaluation and Result

confusion matrix Performance matrix - based on hyper-parameter of slides based on aggregate function and join detection

VI. Proposed Improvement Strategy

The limitations and challenges faced opens a few opportunities for improvement. These improvement strategies are discussed in detail.

A. *Team formation algorithm*

Implemented team formation algorithm is based on the statistical mean of the available academic score. Having outliers in the data can lead to the formulation of teams with a wide difference in their average academic scores. The algorithm can be enhanced to be tolerant of outliers by using a median-based approach instead of a mean.

B. *Correlation and threshold values*

VII. Summary and Future work

Improved parsing BERT - Word embedings based on present word - to provide context

References

[1] N. Islam, Z. Islam, and N. Noor, “A survey on optical character recognition system,” *arXiv preprint arXiv:1710.05703*, 2017.

[2] S. Singh, “Optical character recognition techniques: a survey,” *Journal of emerging Trends in Computing and information Sciences*, vol. 4, no. 6, 2013.

[3] K. Mandavia, P. Badelia, S. Ghosh, and A. Chaudhuri, “Optical character recognition systems for different languages with soft computing,” *Spring. Inter. Publish.*, vol. 352, pp. 9–41, 2017.

[4] K. Hamad and K. Mehmet, “A detailed analysis of optical character recognition technology,” *International Journal of Applied Mathematics Electronics and Computers*, no. Special Issue-1, pp. 244–249, 2016.

[5] M. Liška12, V. Rusňák, and E. Hladká1, “Automated hypermedia authoring for individualized learning,” 2007.

[6] V. Obionwu, V. Toulouse, D. Broneske, and G. Saake, “Slide-recommendation system: A strategy for integrating instructional feedback into online exercise sessions,” in *Proceedings of the 11th International Conference on Data Science, Technology and Applications*, 2022, pp. 542–549.

[7] A. R. Lahitani, A. E. Permanasari, and N. A. Setiawan, “Cosine similarity to determine similarity measure: Study case in online essay assessment,” in *2016 4th International Conference on Cyber and IT Service Management*. IEEE, 2016, pp. 1–6.

[8] S. Mori, C. Y. Suen, and K. Yamamoto, “Historical review of ocr research and development,” *Proceedings of the IEEE*, vol. 80, no. 7, pp. 1029–1058, 1992.

[9] T. Blanke, M. Bryant, and M. Hedges, “Open source optical character recognition for historical research,” *Journal of Documentation*, vol. 68, no. 5, pp. 659–683, 2012.

[10] G. Nagy, T. A. Nartker, and S. V. Rice, “Optical character recognition: An illustrated guide to the frontier,” in *Document recognition and retrieval VII*, vol. 3967. SPIE, 1999, pp. 58–69.