Journal on Artificial Intelligence

Tech Science Press

DOI: 10.32604/jai.2024.050706

ARTICLE

Enhancing Exam Preparation through Topic Modelling and Key Topic Identification

Rudraneel Dutta* **and Shreya Mohanty**

School of Computer Engineering, Kalinga Institute of Industrial Technology, Bhubaneswar, 751024, India

*Corresponding Author: Rudraneel Dutta. Email: its.rudraneel@gmail.com

Received: 14 February 2024 Accepted: 17 June 2024 Published: 19 July 2024

ABSTRACT

Traditionally, exam preparation involves manually analyzing past question papers to identify and prioritize key topics. This research proposes a data-driven solution to automate this process using techniques like Document Layout Segmentation, Optical Character Recognition (OCR), and Latent Dirichlet Allocation (LDA) for topic modelling. This study aims to develop a system that utilizes machine learning and topic modelling to identify and rank key topics from historical exam papers, aiding students in efficient exam preparation. The research addresses the difficulty in exam preparation due to the manual and labour-intensive process of analyzing past exam papers to identify and prioritize key topics. This approach is designed to streamline and optimize exam preparation, making it easier for students to focus on the most relevant topics, thereby using their efforts more effectively. The process involves three stages: (i) *Document Layout Segmentation and Data Preparation*, using deep learning techniques to separate text from non-textual content in past exam papers, (ii) *Text Extraction and Processing* using OCR to convert images into machine-readable text, and (iii) *Topic Modeling with LDA* to identify key topics covered in the exams. The research demonstrates the effectiveness of the proposed method in identifying and prioritizing key topics from exam papers. The LDA model successfully extracts relevant themes, aiding students in focusing their study efforts. The research presents a promising approach for optimizing exam preparation. By leveraging machine learning and topic modelling, the system offers a data-driven and efficient solution for students to prioritize their study efforts. Future work includes expanding the dataset size to further enhance model accuracy. Additionally, integration with educational platforms holds potential for personalized recommendations and adaptive learning experiences.

KEYWORDS

Topic modelling; document layout segmentation; optical character recognition; latent dirichlet allocation

1 Introduction

Walking into a library of a million books and prioritizing and planning for an exam can be difficult and time-consuming. Students often need help identifying the most important topics to focus on, leading to inefficiency and wasted time.

The research addresses this problem in identifying key topics through the manual and labor-intensive process of analyzing past exam papers individually. This approach is designed to streamline

and optimize exam preparation, making it easier for students to focus on the most relevant topics, thereby using their efforts more effectively.

At its core, our research revolves around topic modelling, a statistical technique that may be used to identify clusters of similar words within a body of text. It allows us to extract all the underlying themes and recurring subjects of interest from within a document. This allows us a novel method for identifying, ranking and prioritizing critical topics in annual standardized tests. Using deep learning-based document segmentation methods enables effective study planning with a vast scope for future expansion into other fields. Our design follows a four-step sequence for document layout segmentation, data acquisition, theme analysis, and topic ranking.

(i) *Document layout segmentation*: Document layout segmentation is facilitated through a machine-learning model built around the Detectron2 library for contextual segmentation of convoluted pages within input documents.
(ii) *Data acquisition*: Optical Character Recognition (OCR) extracts text content from the document image segments.
(iii) *Theme analysis*: Latent Dirichlet Allocation (LDA) is a popular topic modelling algorithm used to find themes within a collection of documents.
(iv) *Topic ranking*: The identified topics are ranked based on their importance in the particular field.

This approach offers several advantages over traditional study planning methods. First, it is data-driven, based on the actual content of the previous sets rather than subjective assumptions, which can often be inaccurate. Second, it is efficient, as it can identify and prioritize the most important topics in seconds, which would otherwise require hours of manual analysis. Lastly, it is also personalized, as it can be tailored to the individual student's needs and strengths.

This topic modelling approach helps students save significant time and effort analyzing historical question papers while retaining the advantage of tedious analysis. Additionally, it aids confidence with the added mathematical certainty of prioritizing the most critical topics for study.

While this project aims to improve academic efficiency, it can be further expanded into other institutions, such as journalism, business, and law firms, in processing large amounts of context-based data.

This research aims to introduce a new method for analyzing documents and identifying key topics. This method aims to be faster and more accurate than traditional study planning methods, especially when dealing with large sets of documents.

2 Literature Review

The research in this project is based on the core topics of document layout segmentation, OCR, and, most importantly, topic modelling. Topic modelling is a statistical approach to identifying abstract topics in a document group. It has been used in various applications, including text classification, document summarization, and recommendation systems [1]. This thesis provides a foundation and draws upon key works and tools in this domain.

2.1 Layout-Parser

The LayoutParser toolkit is a Python library that uses deep learning to analyze and process document images. It includes tools for layout detection, OCR, and other document image analysis tasks. It also includes a collection of pre-trained neural network models [2].

LayoutParser formulates layout analysis as an object detection problem. It accepts a document as input in the form of an image and generates a set of rectangular boxes representing their corresponding content regions [3]. LayoutParser is built on top of the Detectron2 platform.

2.2 Faster R-CNN

Region proposal algorithms are used to hypothesize the locations of objects in an image [4]. Faster R-CNN (Region-based Convolutional Neural Network) is a model introduced for object detection with an improved architecture and performance and reduced training and detection time compared to its predecessor, Fast R-CNN. Unlike its predecessor, it is designed as a single model consisting of two modules. Because of the two-module architecture, Faster R-CNN belongs to the group of two-stage detectors, in opposition to one-stage detectors such as YOLO (You Only Look Once) [5] or SSD (Single Shot Detector) [6].

2.3 The Bitter Aloe Project

The Bitter Aloe Project, examining testimonies from apartheid victims in South Africa, develops customized machine learning models to extract data from Truth and Reconciliation Commission (TRC) records and exemplifies work in this domain [7]. The key technical components of the project include a deep learning model that utilizes contextual features to identify over 100 types of entities in TRC testimonies. Word Embeddings through a custom-trained model represent words as vectors in a high-dimensional space, enabling semantic analysis, such as finding similar testimonies and exploring word relationships [8]. Finally, text classification models categorize testimonies (e.g., victim, perpetrator, eyewitness) based on training data, automating classification for new testimonies to create new research tools that can help scholars and journalists learn more about the apartheid era and its legacy.

2.4 Automatic Detection of Sections and Paragraphs in Legal Documents

Vlachos et al. propose a deep learning approach to detect paragraphs and sections in legal documents automatically, a task that can otherwise be very labour-intensive [9]. This scientific thesis delves into applying Deep Learning methods to analyze legal documents. It specifically concentrates on the initial task of identifying the paragraphs and titles within legal documents. Two computer vision models, RetinaNet and YOLOv5, enable this. A subsequent post-processing step categorizes these text blocks as either paragraphs or titles.

The research offers comprehensive insights into the data, methodologies, and outcomes of various Deep Learning models, alongside a discussion of their strengths and weaknesses. This holds significance as it illustrates the capability of these techniques to automate legal processes and generate substantial time and cost savings for legal professionals.

The study of the work above has contributed to our research by providing valuable insights and methodological foundations. It highlights the importance of deep learning techniques in topic identification. This paper aims to create an analytical tool to improve the efficiency of study planning. Our work paves the way towards better integration of technology in academia and also has the potential to benefit various sectors beyond education.

3 Methodology

The research methodology comprises three key stages: data preparation and document layout segmentation, text extraction and processing using OCR, and finally, topic modelling and analysis through LDA.

3.1 Data Preparation and Document Layout Segmentation

This stage involves preparing the question papers for text extraction and topic modelling by:

(a) Annotating a sample dataset of 40 diverse question papers using the Label Studio platform. The primary objective of this annotation process is to classify specific sections of the question papers as either "question" or "not". Any content not directly related to questions, such as instructions or headers, was labelled as "not". The annotation process is a structured procedure, which includes selecting documents, viewing its pages, and drawing bounding boxes around the distinct sections. This is crucial in training and evaluating our topic modelling approach.

(b) Data augmentation to enhance the robustness of our model, we perform data augmentation on the annotated dataset. That means creating various versions of the annotated question papers, including rotations, translations, and brightness adjustments [10]. Data augmentation aids the model in generalizing to different document layouts and formats.

(c) Layout Training using the Detectron2 model to identify two distinct parts. First, we must prepare the dataset, and then, as discussed below, we can train the model on our custom dataset.

  (i) *Preparing the dataset*: After rigorous annotation, review, and data augmentation, we obtained a well-annotated dataset for training and evaluating our topic modelling approach. Then, the labelled dataset was exported from Common Objects in Context (COCO) format and saved in a JavaScript Object Notation (JSON) file. COCO format is commonly used in object detection tasks and is an acceptable format for training using the Detectron2 library. An exported dataset consists of a JSON file and all images with at least one labelled region [3,11]. The dataset is split into two groups for training and testing. For our task, the split was approximately 85%–15% ratio for the respective sets.

  (ii) *Training the model*: The hypothesis is that referencing a pre-trained model available within the LayoutParser Model Zoo will allow us to detect questions within an example question paper with sufficient reliability [12]. All experiments to prove this hypothesis were conducted on a local environment of Python 3.9 with a GTX 1660Ti GPU. Faster R-CNN R 50 FPN 3x trained on the PubLayNet dataset was used to fine-tune the table detector from the Model Zoo. Training the model over 10,000 iterations took about 6 hours, and evaluation was performed once every 20 iterations during the training.

Using the trained and fine-tuned model on a batch of questions from previous years yields a masked version of each page of the input documents. This process results in a processed document that is more suitable for context-based text extraction depending on the identified text body (Fig. 1).

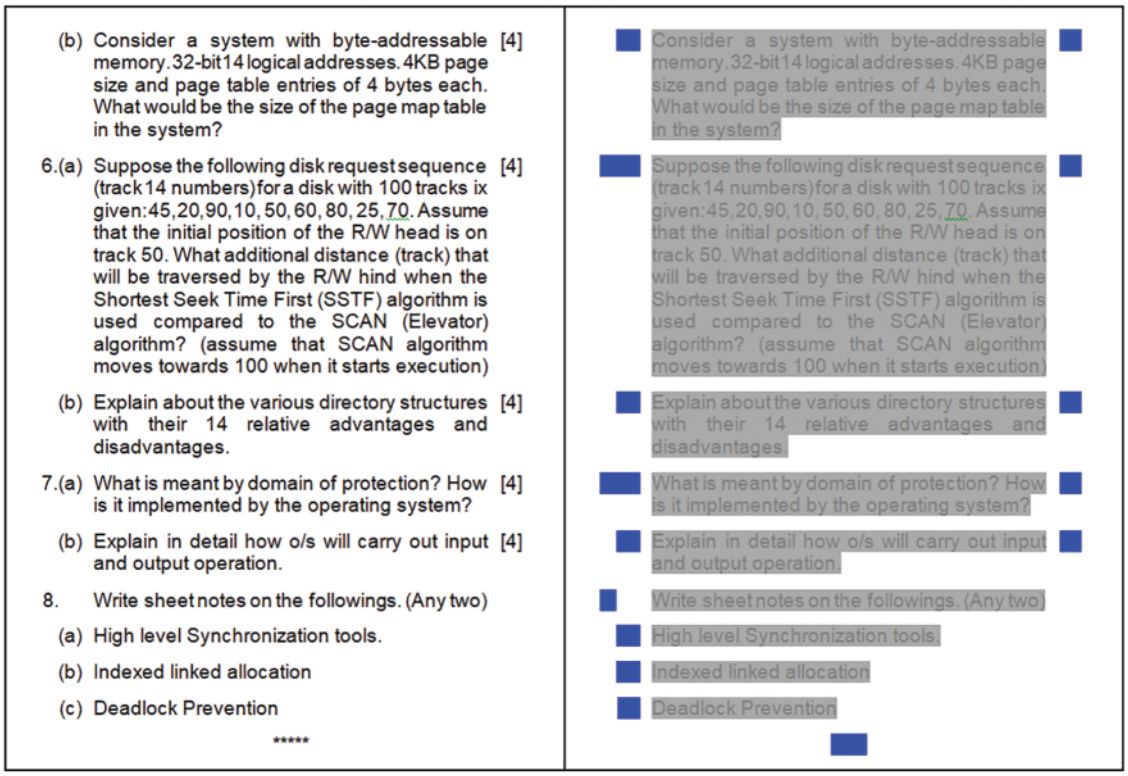

Figure 1: Sample document before and after segmentation

3.2 Text Extraction and Processing using OCR

Running the trained model on our example documents results in a list of bounding box coordinates mapped to the input documents based on the detected classes ("Question" and "Not"). We then iterate over the list of coordinates to extract the image within every bounding box tagged labelled "Question". An OCR agent configured with the Tesseract engine is used through the LayoutParser library. For every iteration, the OCR Agent processes the content within each bounding box image and appends it to a JSON file for further analysis [13].

3.3 Topic Modelling and Analysis through LDA

This stage uses Latent Dirichlet Allocation (LDA) to uncover hidden topics within the extracted question text.

3.3.1 Data Loading and Preprocessing

This involves preparing the extracted text for LDA modelling. The data is loaded as a JSON file containing all the individual questions in their respective entries, as extracted from the input documents. Following this, we employ a series of techniques to further process the data and underlying themes using LDA.

(a) *Lemmatization* is a technique used on text to reduce words to their base or dictionary form, called lemmas. That means transforming words to their root form to ensure that different grammatical forms of a word are treated as a single word. For example, lemmatization would convert words like "running" and "ran" to their base form, "run" [14]. Lemmatization is

common in natural language processing tasks such as text analysis, topic modelling, and text mining to improve the accuracy and interpretability of the results.

(b) We utilize the spaCy library to analyze and lemmatize the preprocessed text. It can be performed selectively, such as in our case, where only nouns, adjectives, verbs, and adverbs are considered for lemmatization [15]. Overall, lemmatization is applied to the extracted text to ensure a consistent representation of words for topic modelling to improve its interpretability.

(c) *After lemmatization, tokenization* breaks text into individual parts, like words or punctuation marks. This process facilitates counting, measuring word frequencies, and uncovering patterns within the data. By segmenting the text into discrete units, tokenization enhances the granularity of analysis, enabling a more detailed exploration of the document's content. This allows this structured data to be readily processed and analyzed [16].

3.3.2 LDA Model Training

The heart of the LDA process lies in training the LDA model. This probabilistic model uncovers latent topics within a collection of documents by analyzing word patterns and co-occurrence. It identifies underlying themes by examining how words are distributed across documents and topics. The model iteratively refines its understanding of these topics, making it a powerful tool for exploratory data analysis.

Blei et al. introduced key terms in their work: words, documents, and corpus [17]. Now, a topic is a distribution of words. The LDA model treats each document in the corpus as a mixture of several topics.

In this research, we aim to estimate two sets of distributions by studying the corpus: (i) the distribution of words in each topic and (ii) the distribution of topics over a corpus. We can demonstrate the LDA model as a three-level hierarchical Bayesian model with an example.

Let us assume two topics in our case: Computer Memory and Pipelining. Based on the corpus, our LDA model (Fig. 2) generates a topic distribution over each document, such as 62.5% under the topic of Pipelining and 37.5% under the topic of "Computer Memory" (Fig. 3).

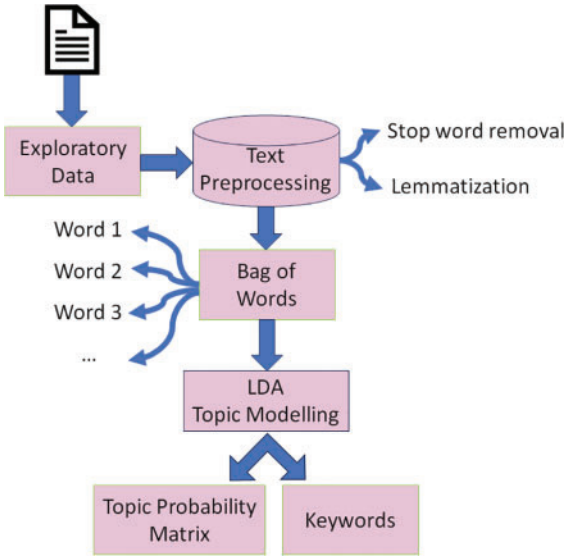

Figure 2: LDA modelling pipeline

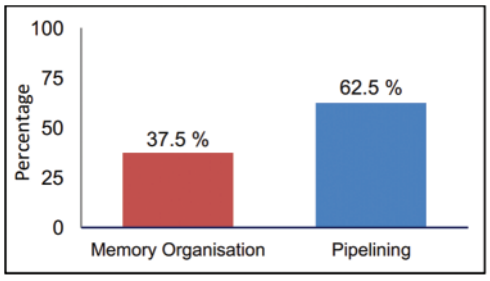

Figure 3: Distribution of topics

Each topic is a distribution of all words from an example vocabulary set {"data_dependence", "pipeline", "parallel", "access_time", "addressing", "hit_ratio", "byte-addressable", "hazard"}. Depending on the topic, we see some words with higher probabilities. For example, naturally, the word "pipeline" would show up with a higher probability in the topic "Pipelining" than in the topic "Memory Organization".

Here, we establish two latent variables. A latent variable is not directly observable in the data but can unveil concealed patterns within the data, which are valuable in constructing a probabilistic model. The initial latent variable, denoted as θ, represents the distribution of topics across each document, with 62.5% being "Pipelining" and 37.5% being "Memory Organization". The second latent variable, Z (where Z can take on values from 1 to n), signifies the topic associated with each word.

For each document 'd', there is a topic distribution 'θ_d'. For each word 'i' in the document 'd', 'w_di' is generated based on the topic distribution 'θ_d', the topic 'Z' for this word 'di', and word distribution over the topic 'Z_di'. Suppose the first word is "pipeline" in a document 'd'; it is generated by first specifying a distribution of topics: 62.5% "Pipelining" and 37.5% "Memory Organization" ('θ_d'). Then, we sample the topic "Pipelining" ('Z_d1') for the first word. In the topic "Pipelining", we sample the word "pipeline" ('w_d1') from the topic distribution.

Mathematically, the process can be represented by the Bayesian formula in Eq. (1) below:

$$P(w, z, \theta) = \prod_{d=1}^{D} P(\theta_d) \prod_{n=1}^{N_d} P\left(Z_{dn}|\theta_d\right) P(w_{dn}|z_{dn}) \tag{1}$$

By specifying probability distributions for $P(\theta_d)$, $P\left(Z_{dn}|\theta_d\right)$, and $P(w_{dn}|z_{dn})$, we can compute the joint distribution $P(w, z, \theta)$, which represents the likelihood of encountering the corpus at hand.

The use of the Dirichlet distribution in LDA is attributed to the primary behaviour of 'θ_d', which pertains to a Dirichlet Distribution.

(a) *LDA Topic Modelling with Gensim*

The topic modelling process revolves around building a Gensim dictionary and creating a corpus from the input data. A corpus is a bag-of-words representation of our text data. It captures the frequency of each word in the text, mapping words to their respective counts within each document. This corpus is used as input for the LDA model.

A Gensim dictionary plays a pivotal role in LDA topic modelling. Once the text data has been tokenized, a Gensim dictionary is created. This dictionary acts as a bridge between the raw text and the numerical world of LDA modelling. Each unique word (token) in the tokenized text is assigned a unique numerical ID. This process is crucial because LDA works with numerical data, and the dictionary facilitates this transformation and establishes a reference between words and their corresponding numerical representations, ensuring consistency in the modelling process.

(b) *Corpus Creation*

Concurrently with dictionary creation, a Gensim corpus is generated. The corpus represents text data that resembles a "bag of words." It captures the frequency of each word in the text, mapping each word to its respective counts within each document. This process results in a structured numerical representation of text data suitable for LDA modelling. The corpus summarizes the text by counting how often each word appears, a fundamental aspect of LDA [18]. Gensim creates a unique ID for each word in the document, mapping word_id and

word_frequency. For example, (10, 2) below indicates word_id 10 occurs twice in the document and so on in the following excerpt:

(0, 2), (1, 1), (2, 1), (3, 2), (4, 4), (5, 1), (6, 1),
(7, 2), (8, 1), (9, 1), (10, 2), (11, 2), (12, 1), (13, 1)

3.4 LDA Topic Visualization

After training the LDA model, the code employs the PyLDAvis library to create interactive visualizations of the discovered topics. These visualizations offer graphical representations of the topics, their word distributions, and their interrelationships (Fig. 4).

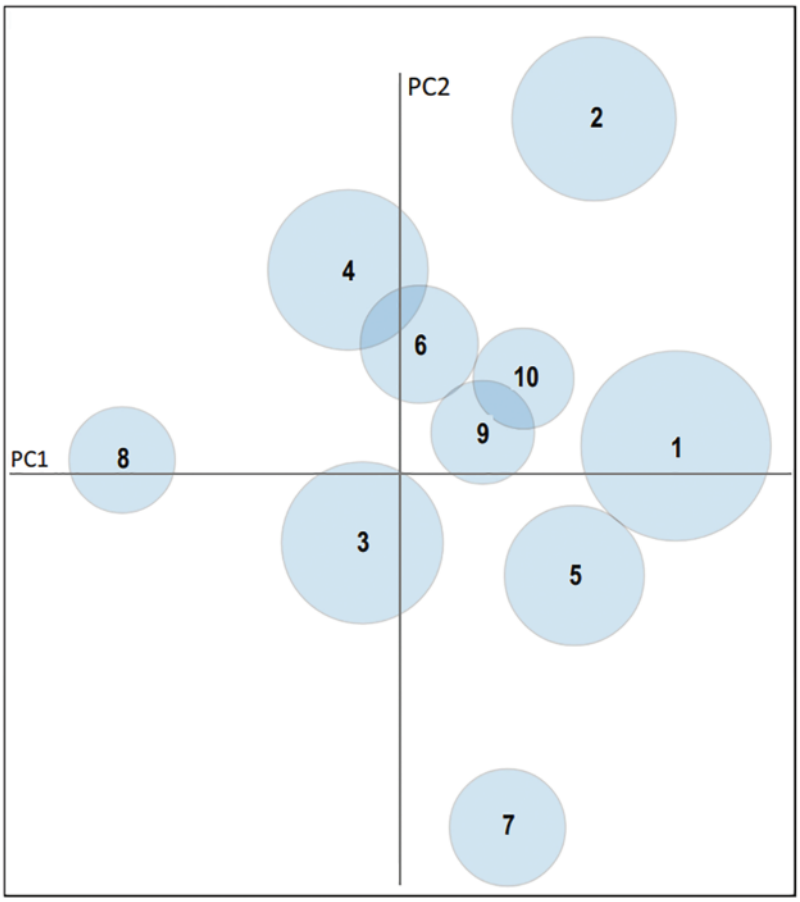

Figure 4: Topic visualisation

Researchers can interact with these visualizations, gaining valuable insights into the discovered topics and their semantic connections. This visualization step is indispensable for exploring and understanding the results of the LDA modelling, making it a critical component of the research process.

3.5 Significance of LDA

This research is based on the process of LDA. Other well-known methods also exist for clustering text content, such as Term Frequency-Inverse Document Frequency (TFIDF) in combination with K-Means clustering. Term Frequency is the ratio of the number of occurrences of a word in a document to the number of all words in a document, as shown in Eq. (2):

$$tf\ (t,d) = \frac{n_t}{\sum_k n_k} \tag{2}$$

Inverse Document Frequency is a log of the ratio of the number of all documents/strings in the corpus to the number of documents with a particular term [19] as seen in Eq. (3).

$$idf\,(t,D) = \log \frac{|D|}{|\,\{d_i \in D \mid t \in d_i\}|} \tag{3}$$

Thus, TFIDF in Eq. (4) is the product of TF and IDF.

$$tf - idf\ (t,d,D) = tf\ (t,d) \cdot idf\,(t,D) \tag{4}$$

K-Means is one of the simplest clustering algorithms for detecting common patterns in unorganized data points. It classifies all data points into K clusters by identifying close data points based on their similarities. K-Means was explored to obtain clusters of different questions from the input. Despite this, it became evident early in our research that adopting this process would not achieve our objective.

In contrast, LDA is a form of unsupervised learning for discovering latent (hidden) themes within data. The discovered themes may be referred to as topics. It is a versatile algorithm widely used in data analysis, especially in natural language processing and text analysis. LDA assigns a distribution of topics over data rather than a single topic or grouping—this is the key difference between LDA and strict clustering algorithms such as K-Means.

LDA produces a probability distribution of groupings. In document analysis (Fig. 5), clustering algorithms produce one grouping per document, while LDA produces a probability distribution of groupings (topics) per document [20].

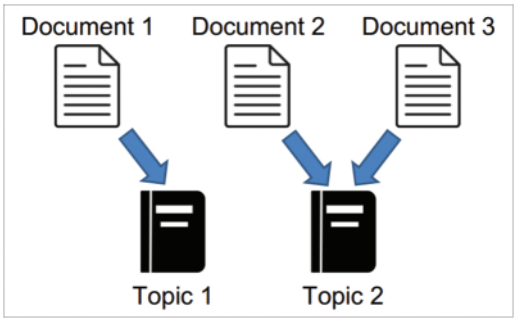

Figure 5: K-Means clustering

Thus, utilizing LDA is advantageous in analyzing previous years' question papers for educational research or exam preparation. LDA's probabilistic approach enables the generation of a probability distribution of topics (themes) per document, as opposed to traditional clustering algorithms that present a single theme per document. This distinction is critical when identifying important recurring

topics in question papers. LDA's probabilistic nature allows for a more nuanced assessment of topic relevance by assigning probabilities to topics within each document, assisting in identifying prevalent and crucial subject matter.

Additionally, methods such as K-Means clustering require specifying the number of clusters (K) in advance, which cannot be determined due to the unpredictability of our task, as opposed to LDA, which offers a distinct advantage in scenarios where the optimal number of clusters is uncertain [21]. LDA accommodates the inherent overlap and intersection of topics within question papers, providing insights into various concepts and where they frequently appear (Fig. 6).

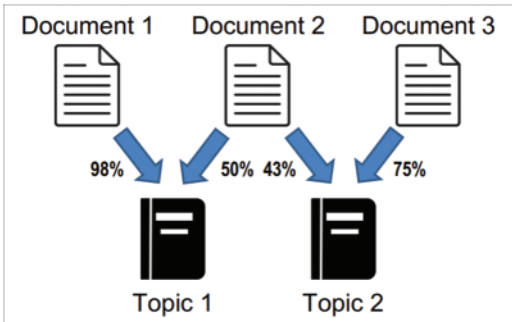

Figure 6: Latent dirichlet allocation

This probabilistic framework aids in prioritizing study efforts by focusing on topics with higher probabilities of occurrence, ultimately facilitating more efficient and comprehensive topic analysis for exam preparation or educational research.

4 Results

During training, inference and subsequent evaluation are performed on unseen images from the validation dataset. After training, a final evaluation is conducted on the testing dataset. Training of a model outputs several files. The most important is the configuration file, which includes the changed hyperparameters and final weights, and the metrics file, which contains the training progress. As shown in Fig. 7, the final trained model yields a very reliable solution to classify text on the input document based on context.

While the model demonstrates promising classification accuracy, a more rigorous analysis of the results is essential to comprehensively understand its capabilities and limitations. Here, we propose several avenues for further exploration:

Quantitative Analysis: A deeper examination of the metrics file, particularly metrics like precision, recall, and F1-score for various document categories, can reveal the model's performance on specific text types within the documents.

Error Analysis: Investigating misclassified documents can provide valuable insights. Are there specific document types or text styles that pose challenges for the model? Understanding these errors can guide targeted model improvements.

Visualization Techniques: Beyond the distribution of document word counts (Fig. 7), techniques such as confusion matrices or saliency maps can visualize the model's decision-making process. This visualization can aid in identifying potential biases or areas for further optimization.

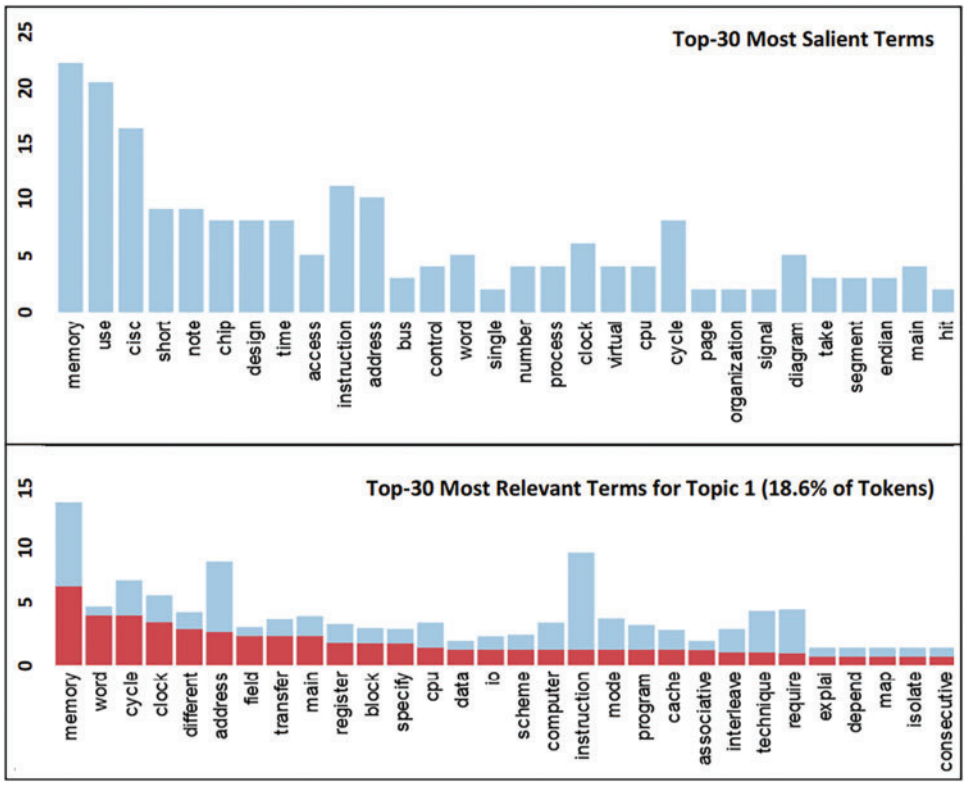

Figure 7: Distribution of document word counts

Finally, on performing LDA, we are presented with several unique underlying themes that give a good idea of the dominant themes throughout the given set of documents, as observed in the excerpt below:

Topic 1:	cisc, instruction, diagram, processing, classification, flynns, parallel, multiply, restore, restoring
Topic 2:	endian, bus_organization, control_signal, cpu, single_bus, big, instruction, advantage, execution, multi_bus, little_endian
Topic 3:	time, memory_access, access_time, average, hit, address, main_memory, ratio, cache_memory, virtual_memory, translation, block, system
Topic 4:	processor, pipeline, interrupt, perform, word, main_memory, technique, booth, multiplication, require, instruction, associative, ghz, speed

The identified topics align well with core concepts in computer architecture, including CPU architecture, memory systems, and pipelining. This reinforces the potential of LDA for summarizing key themes from a document corpus, even with limited prior knowledge of the specific content.

To enrich the analysis, we can consider the following:

- *Topic Coherence*: Evaluate the coherence of the identified topics using metrics like perplexity or coherence scores. This will assess how well the words within each topic form a meaningful group.
- *Topic Distribution*: Explore how the prevalence of these topics varies across different documents. Are there specific documents that heavily emphasize certain topics? This analysis can reveal the range and depth of concepts covered within the document set.
- *Comparison with External Knowledge*: Compare the topics with established taxonomies or knowledge graphs in computer architecture. This comparison can validate the model's findings and uncover new relationships between concepts.

By incorporating these techniques, we can better understand the model's performance and the thematic landscape of the analyzed documents. This deeper understanding will inform future refinements and applications.

5 Discussion

This discussion emphasizes the importance of parameter selection and model optimization for maximizing the system's effectiveness. By exploring the intricacies of these processes, we gain valuable insights for future research and practical applications, ultimately empowering students with a more efficient and targeted study experience. The experiments demonstrate the potential of this system to significantly benefit students in effective study planning. The research has two main sections: Document Layout Segmentation and Key Topic Identification through LDA (Fig. 8).

Optimizing the System for Peak Performance: The accuracy of topic identification relies heavily on the accuracy of layout segmentation. The layout segmentation results were achieved by fine-tuning the model several times, which was significantly affected by the model-training configuration [22]. Experiments proved that a lower learning rate value (0.0025) yielded better results than higher values. Increasing the number of iterations from 2,000 to 10,000 for model training also improved the result. Batch size in deep learning refers to the number of training samples used per iteration. Selecting a larger batch size may improve accuracy but yield poor generalization at excessive values [23]. After thorough experimentation, the parameters were set to maximize the hardware potential and achieve the best possible performance.

Understanding LDA Parameters for Effective Topic Modeling: After document layout segmentation, the next step is performing LDA topic modelling on the extracted text. The parameters of an LDA model significantly impact the quality of the results in topic modelling. Several critical parameters are specified in the LDA model configuration, such as the number of topics, passes, alpha value, and the "R" parameter for visualization, which control the topics discovered and the effectiveness of the model [24].

Nuances of Parameter Selection: The number of topics determines the number of distinct topics the model seeks to extract from the corpus. At higher values, this can result in topics that are excessively fine-grained and difficult to interpret, and at lower values can lead to broad topics that lack specificity. In our case, a value of 10 generated satisfying results corresponding to our example curriculum.

Balancing Computational Cost and Topic Quality: The 'number of passes' parameter specifies the number of passes through the entire corpus during training. Increasing this value can allow the model to refine its topics, potentially resulting in more accurate topics at the cost of increased training time. A value of 10, as in our case, means that the algorithm will go through the entire corpus 10 times during the training process.

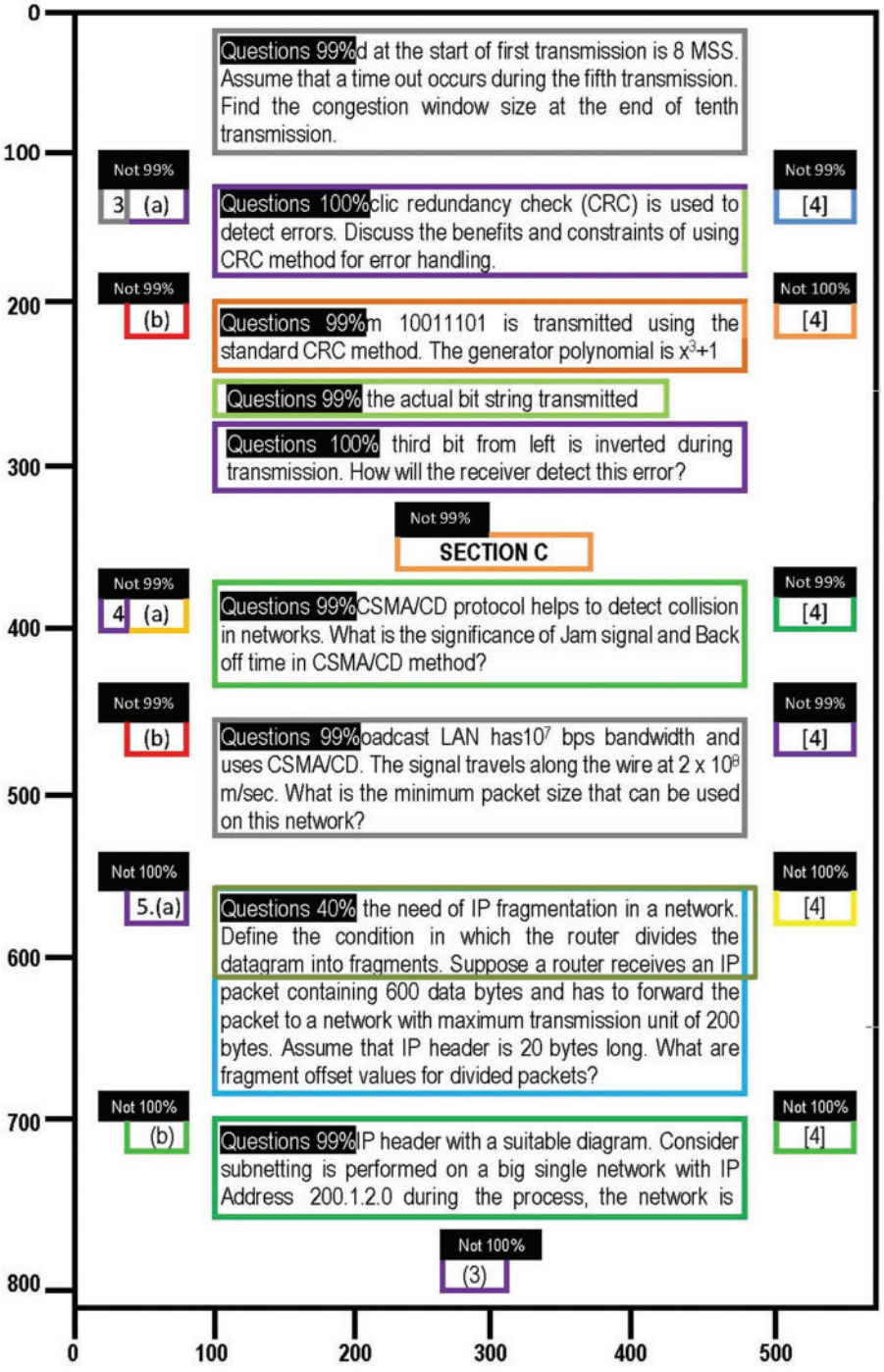

Figure 8: Detected layout

Tailoring Visualization for Clear Topic Representation: Finally, the "R" parameter for LDA visualization specifies the number of words to be shown for each topic in the visualization. A lower "R" value results in a more precise set of words per topic, presenting a more focused representation of

the themes in the question papers [25]. Setting the 'R' value to 30 indicates that the visualization will display up to 30 terms associated with each topic, as in our case.

Therefore, in this research, the authors propose an efficient pipeline to develop a system that can streamline and optimize exam preparation, making it easier for students to focus on the most relevant topics, thereby using their efforts more effectively.

6 Future Development

As with any machine learning model, with an even larger dataset, it is possible to enhance the accuracy of the results further. With advances in hardware, it will be possible to train the layout segmentation model over even larger datasets with more intensive parameters.

The method explored in this research paper can be integrated into modern educational platforms with the potential to adapt and cater to the needs and preferences of individual students. It may be developed further to categorically recommend topics based on the type of question, such as subjective or objective, from a systemized database that can search and organize large practice sets.

7 Conclusion

This research presents a novel approach to optimizing exam preparation. It leverages document layout segmentation, Optical Character Recognition (OCR), and Latent Dirichlet Allocation (LDA) for topic modelling. This data-driven approach automates identifying and prioritizing crucial topics within historical exam papers, addressing the longstanding challenge of effectively organizing study efforts.

The proposed system offers distinct advantages:

- *Data-Driven Prioritization*: Topic identification relies on the actual content of past exams, ensuring superior relevance compared to subjective approaches.
- *Enhanced Efficiency*: The system identifies critical topics in seconds, significantly reducing time spent on manual analysis.
- *Personalized Learning Potential*: Future integration with educational platforms holds promise for personalized recommendations and adaptive learning experiences.

Overall, this research empowers students to focus their studies more effectively by automating topic identification and prioritization. The success in extracting pertinent themes from past exams paves the way for broader applications in various domains requiring analysis of large, context-dependent data sets. This research is a foundation for further exploration, including expanding the dataset size for improved model accuracy and integrating the system with educational platforms for personalized learning experiences.

Acknowledgement: None.

Funding Statement: The authors received no specific funding for this study.

Author Contributions: The authors confirm contribution to the paper as follows: Study conception and design: Rudraneel Dutta and Shreya Mohanty; Data Collection: Shreya Mohanty; Analysis and interpretation of results: Rudraneel Dutta and Shreya Mohanty; Draft manuscript preparation: Rudraneel Dutta. All authors reviewed the results and approved the final version of the manuscript.

Availability of Data and Materials: Data and materials will be provided upon request.

Conflicts of Interest: The authors declare that they have no conflicts of interest to report regarding the present study.

References

[1] P. Kherwa and P. Bansal, "Topic modelling: A comprehensive review," *EAI Endorsed Transact. Scalable Inform. Syst.*, vol. 7, no. 24, pp. 1–16, 2020.

[2] Z. Shen, R. Zhang, M. Dell, B. C. G. Lee, J. Carlson and W. Li, "LayoutParser: A unified toolkit for deep learning-based document image analysis," in *Document Anal. Recognit.*, Springer International Publishing, 2021, vol. 12821, pp. 131–146. doi: 10.1007/978-3-030-86549-8_9.

[3] Kovalova, "Deep learning-based table detection in documents," Bachelor's thesis, Satakunta Univ. of Applied Sciences, Finland, 2023, pp. 1–44.

[4] S. Ren, K. He, R. Girshick, and J. Sun, "Faster R-CNN: Towards real-time object detection with region proposal networks," *Adv. Neural Inform. Process. Syst.*, vol. 28, pp. 1–9, 2015.

[5] J. Redmon, S. Divvala, R. Girshick, and A. Farhadi, "You only look once: Unified, real-time object detection," in *2016 IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Las Vegas, NV, USA, 2016, pp. 779–788.

[6] B. Leibe, J. Matas, N. Sebe, and M. Welling, "SSD: Single shot multibox detector," in *Proc. Comput. Vis.–ECCV 2016: 14th Eur. Conf.*, Amsterdam, Netherlands, Oct. 11–14, 2016, vol. 9905, no. 14, pp. 21–37.

[7] W. Mattingly, "Introduction to topic modelling and text classification," 2021. Accessed: Feb. 12, 2023. [Online]. Available: https://topic-modeling.pythonhumanities.com/intro.html

[8] R. Das, M. Zaheer, and C. Dyer, "Gaussian LDA for topic models with word embeddings," in *Proc. 53rd Annual Meet. Assoc. Comput. Linguist. 7th Int. Joint Conf. Natural Lang. Process.*, Beijing, China, 2015, vol. 1, pp. 795–804.

[9] C. Vlachos, I. Androutsopoulos, M. Papageorgiou, and D. D'Anna, "Automatic detection of sections and paragraphs in legal documents," Master thesis, Dept. of Informatics, School of Information Sciences and Technology, Univ. of Economics and Business, Athens, Greece, 2022.

[10] B. Srinivasa-Desikan, *Natural Language Processing and Computational Linguistics: A Practical Guide to Text Analysis with Python, Gensim, Spacy, and Keras*. UK: Packt Publishing Ltd, 2018.

[11] T. Wang *et al.*, "Large batch optimization for object detection: Training coco in 12 minutes," in *Proc. Comput. Vis.–ECCV 2020: 16th Eur. Conf.*, Glasgow, UK, Springer International Publishing, Aug. 23–28, 2020, pp. 481–496.

[12] M. Ataullha, M. H. Rabby, M. Rahman, and T. B. Azam, "Bengali document layout analysis with Detectron2," arXiv:2308.13769, pp. 1–4, 2023.

[13] S. Naik, R. Dinesh, and S. Prabhanjan, "Segmentation of unstructured newspaper documents," *Int. J. Adv. Eng. Res. Sci.*, vol. 4, no. 5, pp. 79–83, 2017. doi: 10.22161/ijaers.4.5.13.

[14] V. Balakrishnan and E. Lloyd-Yemoh, "Stemming and lemmatization: A comparison of retrieval performances," *Lect. Notes Softw. Eng.*, vol. 2, no. 3, pp. 262–267, 2014. doi: 10.7763/LNSE.2014.V2.134.

[15] F. Martin and M. Johnson, "More efficient topic modelling through a noun only approach," in *Proc. Australasian Lang. Tech. Assoc. Workshop*, Parramatta, Australia, 2015, pp. 111–115.

[16] M. M. Haider, M. A. Hossin, H. R. Mahi, and H. Arif, "Automatic text summarization using gensim Word2Vec and K-Means clustering algorithm," in *2020 IEEE Region 10 Symp. (TENSYMP)*, Dhaka, Bangladesh, 2020, pp. 283–286.

[17] D. M. Blei, A. Y. Ng, and M. I. Jordan, "Latent dirichlet allocation," *J. Mach. Learn. Res.*, vol. 3, pp. 993–1022, 2003.

[18] A. Murakami, P. Thompson, S. Hunston, and D. Vajn, "What is this corpus about?: Using topic modelling to explore a specialized corpus," *Corpora*, vol. 12, no. 2, pp. 243–277, 2017. doi: 10.3366/cor.2017.0118.

[19] V. Karyukin *et al.*, "On the development of an information system for monitoring user opinion and its role for the public," *J. Big Data*, vol. 9, no. 1, pp. 110–155, 2022. doi: 10.1186/s40537-022-00660-w.

[20] X. Wei and W. B. Croft, "LDA-based document models for ad-hoc retrieval," in *Proc. 29th Annual Int. ACM SIGIR Conf. Res. Develop. Inform. Retriev. (SIGIR'06)*, New York, NY, USA, Association for Computing Machinery, 2006, pp. 178–185.

[21] G. Goel, "Advancement in K-mean clustering algorithm for distributed data," *Int. Res. J. Eng. Technol.*, vol. 3, no. 8, pp. 695–698, 2016.

[22] J. C. Campbell, A. Hindle, and E. Stroulia, "Latent dirichlet allocation: Extracting topics from software engineering data," in *Art Sci. Analysing Softw. Data*, Elsevier Inc., 2015, pp. 139–159. doi: 10.1016/B978-0-12-411519-4.00006-9.

[23] S. Zhou, P. Kan, Q. Huang, and J. Silbernagel, "A guided latent dirichlet allocation approach to investigate real-time latent topics of Twitter data during Hurricane Laura," *J. Inf. Sci.*, vol. 49, no. 2, pp. 465–479, 2023. doi: 10.1177/01655515211007724.

[24] M. Lee, W. Wang, and H. Yu, "Exploring supervised and unsupervised methods to detect topics in biomedical text," *BMC Bioinformatics*, vol. 7, no. 1, pp. 1–11, 2006. doi: 10.1186/1471-2105-7-140.

[25] T. K. Aslanyan and F. Frasincar, "Utilizing textual reviews in latent factor models for recommender systems," in *Proc. 36th Annual ACM Symp. Appl. Comput.*, Republic of Korea, 2021, pp. 1931–1940.