A Compress-Expand Framework for Automatic Lesson Plan Generation

Shuyan Huang[1], Ying Zheng[2], Xiaoli Zeng[2], Zitao Liu[2*]

[1] TAL Education Group, Beijing, China
[2] Guangdong Institute of Smart Education, Jinan University, Guangzhou, China
huangshuyan@tal.com, zhengying@stu2022.jnu.edu.cn, zengxiaoli@stu.jnu.edu.cn, liuzitao@jnu.edu.cn

Abstract

Creating a well-structured lesson plan is essential for improving classroom efficiency, yet it is often a labor-intensive process. Recently, many studies have leveraged large language models (LLMs) to generate lesson plans automatically. However, existing methods heavily rely on LLMs that are pretrained on large-scale universal corpora, which often lack critical educational theory and textbook-specific information. This can lead to inconsistencies and misalignments with textbook content. To address these challenges, we propose CE-LessonPlan, a novel compress-expand framework to generate lesson plans by effectively combining external lesson plan references and textbook information. The framework consists of two key components: a compressor, which synthesizes multiple retrieved references into a cohesive document, and an expander, which integrates textbook-specific information with the parametric knowledge of LLMs to produce another enriched lesson plan. The outputs of the compressor and expander are then seamlessly integrated to create a comprehensive golden context, further enhancing the lesson plan generation process with LLMs. We conduct extensive experiments to demonstrate that CE-LessonPlan outperforms existing methods for generating lesson plans.

Code — https://github.com/ai4ed/CE-LessonPlan

Introduction

A lesson plan is a structured instructional blueprint designed by teachers to guide their in-class teaching processes. It includes comprehensive components such as lesson objectives, materials and detailed lesson procedures to encapsulate what the teacher should teach and how it should be taught (Azubike 2021; Sugianto 2020), as illustrated in Figure 1 (a). A well-designed lesson plan is particularly crucial for teachers and it not only ensures structured and coherent instructional content delivery but also significantly improves the overall efficacy of the teaching process (Liu et al. 2020; Iqbal, Siddiqie, and Mazid 2021).

Creating lesson plans is a time-consuming task that presents numerous challenges for novice teachers. With the remarkable performance in language understanding and text generation of large language models (LLMs) (Achiam et al. 2023; Touvron et al. 2023), recent studies leverage LLMs to generate lesson plans automatically by carefully designing instructional prompting (Hu et al. 2024; Zheng et al. 2024). Although prompting state-of-the-art LLMs such as GPT-4o may in some cases alleviate the problem of lesson plan generation, there still exists a large gap in instructional content creation quality between teachers and LLMs, thereby hindering LLM based lesson plans from being effectively implemented in real-classroom. First, in spite of lesson plans generated by LLMs are structured similarly to those designed by experts, the content is often vague and lacks practicality for classroom teaching. As illustrated in Figure 1 (b), the blue text indicates empty and difficult-to-conduct content for the teacher generated by the GPT-4o. Second, the LLMs are typically pre-trained on a large-scale universal corpus and have limited knowledge of educational theory and teaching materials, the generated content often lacks alignments with textbook information and may extend beyond the current lesson scopes. For example, comparing the content of textbook analysis components generated by GPT-4o and an expert illustrated in Figure 1 (b). The generation of GPT-4o is relatively inane and misses essential knowledge such as the definition of useful work and extra work in the textbook content.

To address the aforementioned challenges, we develop a lesson plan generator by adapting universal LLMs into the lesson plan generation task. Motivated by the impressive performance of supervised fine-tuning (SFT) that adapts an LLM to a specific downstream task by adjusting the model's parameters to be relevant to a particular task or domain of interest (Ouyang et al. 2022), we fine-tune LLMs on a collected educational dataset consist of 139,512 lesson plans. Furthermore, we prompt relevant handwriting lesson plans for the generator to enhance its capability to produce human-like content following the retrieval-augmented generation (RAG) paradigm (Guu et al. 2020).

Nevertheless, the retrieved lesson plans are created not only by experts but also by novice teachers, leading to varying professional quality. Directly using these initial documents to prompt the lesson plan generator may include poor instructional content, potentially resulting in counterproductive outcomes. Hence, we introduce a lesson plan compressor to refine the multiple retrieved lesson plans into a new

*The corresponding author: Zitao Liu
Copyright © 2026, Association for the Advancement of Artificial Intelligence (www.aaai.org). All rights reserved.

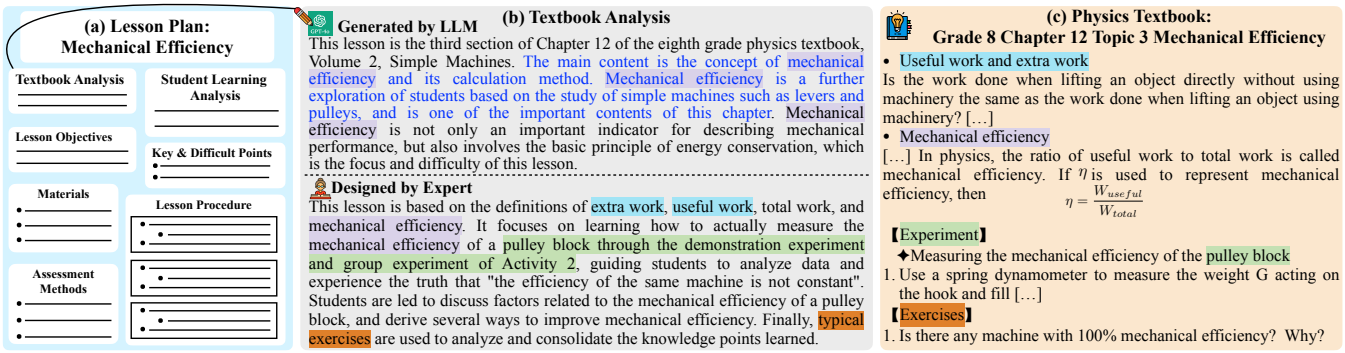

Figure 1: Illustration of lesson plans. The blue text indicates content generated by the GPT-4o that is barren and broad. The content filled with colors corresponds to the core content from the textbook.

and elegant one. Meanwhile, to prevent missing essential textbook knowledge during compression, we collect additional textbook information to enhance the instructional content. Given that textbooks are the condensed results of vast amounts of knowledge in the real educational environments, we develop a textbook expander to generate extra lesson plans with abundant expertise by incorporating the textbook with the parametric knowledge of LLMs. Consequently, we aggregate the lesson plans generated by both the compressor and expander to shape a novel compress-expand framework, *CE-LessonPlan* that compresses multiple retrieved results at the same time expanding knowledge information of the corresponding textbooks to augment the lesson plan generator. Specifically, CE-LessonPlan selects the tokens from the ensemble of the token probabilities from both the compressor and expander. This ensemble decoding simultaneously integrates the outputs of the compressor and expander to create a golden reference. The generated golden references then effectively guide the lesson plan generator to produce human-like and informative instructional content that is consistent with the textbook content, making it easy for educators to conduct in the class.

To assess the effectiveness of CE-LessonPlan, we employ ROUGE (Lin 2004) to evaluate the consistency between textbook information and the generated outputs of CE-LessonPlan. Given the inherent diversity of lesson plans, we further design a comprehensive human evaluation framework encompassing 24 dimensions, tailored to meet the characteristics of each lesson plan component. The extensive experimental results revealed that our approach generates superior lesson plans compared to existing methods in terms of automatic and human evaluation, particularly excels in setting student learning analysis, lesson objectives and key & difficult point components. The main contributions of this work are summarized as follows:

- We build up a lesson plan generator to adapt the universal LLMs into the educational domain on a constructed lesson plan dataset via the SFT and RAG techniques.
- We develop a compress-expand framework to augment the lesson plan generator by extracting and purifying the multiple retrieved lesson plans while enriching them with knowledge from external textbooks and the parametric information of LLMs.
- We propose an ensemble decoding method that simultaneously integrates the reference lesson plans from both the compressor and the expander, enabling our lesson plan generator to produce not only human-like but also knowledgeable outputs.
- We design comprehensive automatic and human evaluation metrics to evaluate generation quality to meet the characteristics of creating lesson plans in real-world educational scenarios. The experimental results demonstrate our proposed CE-LessonPlan approach outperforms other methods.

Preliminaries

Lesson Plan

A lesson plan provides teachers with a structured framework for effective curriculum instruction (Nesari and Heidari 2014; Liu et al. 2023; Li, Ding, and Liu 2020; Zheng et al. 2025). Prior studies (Zheng et al. 2024; Hu et al. 2024) generally define a lesson plan as comprising: (1) **textbook analysis (TA)** to determine instructional scope; (2) **student learning analysis (SLA)** to assess learners' requirements; (3) **lesson objectives (LO)** specifying targeted knowledge and skills; (4) **key & difficult points (KDP)** identifying critical and challenging content; (5) **materials (MAT)** listing required instructional resources; (6) **lesson procedure (LP)** outlining sequenced teaching activities aligned with objectives; and (7) **assessment methods (AM)** summarizing evaluation strategies and feedback for instructional planning.

LLMs for Lesson Plan Generation

Given the extraordinary performance of LLMs across various domains, new applications for generating lesson plans by LLMs are emerging (Zheng et al. 2024; Hu et al. 2024). For example, Lee and Zhai reduced in-service elementary school teachers' lesson preparation pressure at a South Korean university by employing ChatGPT to create science lesson plans (Lee and Zhai 2024). Zheng et al. used GPT-4 to generate each component of lesson plans step-by-step, incorporating self-critiques based on human-defined evaluation criteria to refine these plans to achieve better outputs

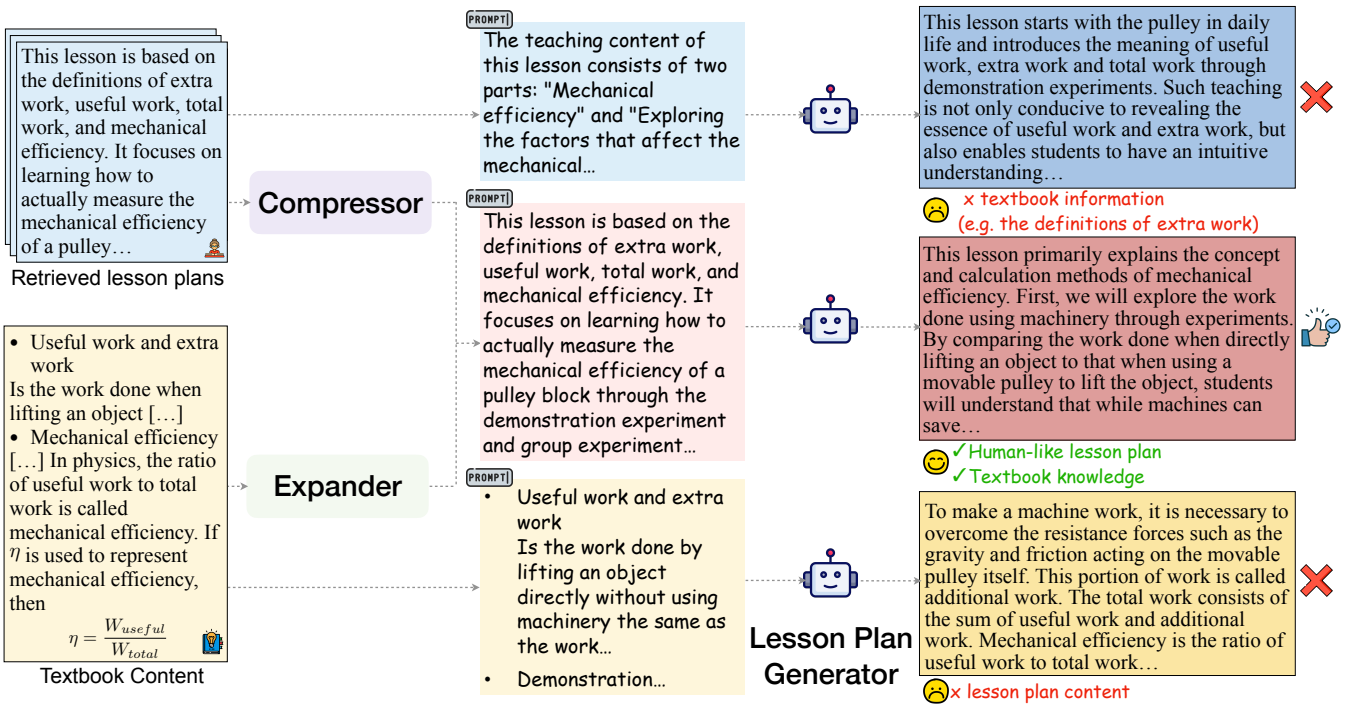

Figure 2: The overview of the proposed CE-LessonPlan.

(Zheng et al. 2024). Hu et al. explored using mathematical problem chains and corresponding prompt instructions to design instructional materials by prompting GPT-4 (Hu et al. 2024). The above approaches heavily rely on meticulously crafted instructional prompts, whereas this paper presents a lesson plan generator to fine-tune LLMs on the collected real-world lesson plans. Moreover, we design a novel compress-expand framework to implement human-like and knowledgeable outcomes by integrating the compression results of retrieval documents and expanding knowledge information of textbooks.

Context Compression

Currently, there are lots of studies aim to compress retrieved documents to filter out irrelevant information and highlight the core content for LLMs (Wang et al. 2023; Li et al. 2024; Ke et al. 2024; Yoon et al. 2024; Jiang et al. 2024; Cao et al. 2024; Xu, Shi, and Choi 2024). For instance, Jiang et al. conducted prompt compression via a small trained compression model to improve LLMs' perception of the key information by preserving only the query-relevant information (Jiang et al. 2024). Xu et al. trained a compression model to summarize the retrieved documents to relieve the burden of LLMs to identify relevant information from long contexts (Xu, Shi, and Choi 2024). Cao et al. employed a dynamic compression strategy to utilize input queries to guide the context compression process to enhance the relevance of queries and remaining contexts (Cao et al. 2024). Li et al. leveraged a single decoder-only LLM to adaptively extract query-relevant content verbatim along with the necessary context thereby restructuring key information for LLM (Li et al. 2024).

Constrained Decoding

Constrained decoding is an effective method to control the token generation process of a generative model to limit its predictions for the next token. Liu et al. designed ensemble-based and search-based decoding strategies to refine prompts for language model (Liu et al. 2024). Li et al. presented a contrastive decoding approach that optimizes a contrastive objective subject to improve the plausibility of open-ended generation (Li et al. 2023). Shi et al. implemented context-aware decoding to reduce hallucinations and unfaithful content of LLMs (Li et al. 2023). Jung et al. utilized ensemble decoding to compress externally retrieved evidence meanwhile incorporating the model's parametric knowledge during the compression process (Jung et al. 2025). Similar to (Jung et al. 2025), we employ ensemble decoding to integrate the compression results of hand-write lesson plans and expansion output of knowledge-aware lesson plans to ensure our lesson plan generator yields human-like and knowledgeable instructional content.

Methodology

The overview of our proposed CE-LessonPlan is illustrated in Figure 2. We fine-tune LLMs on a collected lesson plan dataset to construct a lesson plan generator and employ the RAG technique to provide relevant documents for the generator to improve its ability in creating human-like content. To further improve the quality of the relevant documents, we use a compressor to distill the essence of retrieved refer-

ences and eliminate unnecessary elements, thereby obtaining a condensed result of multiple lesson plans. Considering lesson plan compression solely might result in some missing knowledge information relevant to textbooks, we further design an expander to generate another informative lesson plan by incorporating textbook sources and rich parametric knowledge of LLMs. Ultimately, we propose an ensemble decoding method to synergize the compressed document and the expanded knowledgeable lesson plan as a golden context to enhance the lesson plan generator.

Lesson Plan Generator

To deeply facilitate the pedagogical theory and subject knowledge of universal LLMs in lesson plan generation, motivated by recent approaches (Radford et al. 2018, 2019), we conduct SFT on LLMs to adjust the models' parameters to the educational domain and serve as a lesson plan generator. To conduct SFT on LLMs, we develop a lesson plan dataset to allow LLMs to understand educational theories better and automatically generate more reasonable lesson plans for teachers. The dataset consists of massive and high-quality real-world and simulated lesson plans. The real-world data is crawled from a variety of third-party educational resource platforms. These lesson plans are initially stored as PDF documents. To promote better extraction of textual information, these PDFs are converted into Microsoft Word documents. Since each lesson plan is typically a long textual sequence, the complete lesson plans are split into a series of lesson plan components, such as "textbook analysis", "lesson objectives", and "teaching procedures" to improve training efficiency and avoid the limitation of the context length of LLMs. To enrich the diversity and data scale of the SFT dataset, we also create simulated lesson plans by prompting GPT-4 to design components given curriculum information and real-world component references. The prompt to generate the components is shown as follows:

You are an instructional design expert and have won multiple awards for instructional design. You are designing a lesson plan with the following curriculum information:
Curriculum: {*subject, level, grade, chapter, topic, textbook edition*}
This is an example:{*example of component content*}
Please design a new one with the given curriculum information and example.

The proposed CE-LessonPlan aims to provide golden references to the lesson plan generator to improve its generation quality based on the RAG technique. Similar to (Zhang et al. 2024), we employ retrieval-augmented fine-tuning to foster the generator's detection awareness of the given references. To this end, there are half of the training samples in the SFT dataset contain a retrieved lesson plan d corresponding to the same curriculum information to guide the lesson plan generator to generate similar outputs y during the model training stage:

$$y \sim \mathcal{G}(x, d) \tag{1}$$

where $\mathcal{G}$ represents the lesson plan generator, x represents the given curriculum information [1]. d is the given reference, which is optional.

RAG with Lesson Plan Compressor

To support a human-like generation, the CE-LessonPlan provides real-world handwriting instructional documents as references to enhance the quality of the lesson plan generator. Specifically, for each curriculum information x, there is a set of j retrieved documents $D = \{d_1, d_2, .., d_j\}$ provided to guide the generator in creating similar outputs. As described in equation (1), the lesson plan generator $\mathcal{G}$ produces a lesson plan y given the curriculum x and a reference d_j. However, the collected handwriting instructional documents are designed not only by experts but also by novice teachers, the quality of reference documents d_j varies greatly. Directly utilizing references from D could introduce suboptimal content, potentially degrading output quality. Therefore, inspired by (Jung et al. 2025), a lesson plan compressor $\mathcal{C}$ is presented to condense multiple documents of varying quality D into a cohesive and refined lesson plan $\tilde{d}$:

$$\tilde{d} \sim \mathcal{C}(x, [d_1, d_2, .., d_j]) \tag{2}$$

The objective of the compressor is to take the essence and discard the dross of multiple retrieved lesson plans to get an informative refinement. In this work, we employ an unsupervised method where the compressor is instructed to produce a curriculum-relevant refinement of D in a few-shot manner using a lesson plan compression instruction prompt I_{comp} as follows:

You are an instructional design expert, and you are integrating and compressing multiple instructional design contents of the same curriculum to generate a more refined instructional design.
Curriculum 1st: {*subject, level, grade, chapter, topic, textbook edition*}
Multiple Lesson Plans 1st: { *real-world lesson plans*}
Compression result 1st: {*Compression result*}
Curriculum 2nd: {*subject, level, grade, chapter, topic, textbook edition*}
Multiple Lesson Plans 2nd: { *real-world lesson plans*}
Compression result 2nd: {*Compression result*}
Curriculum 3rd: {*subject, level, grade, chapter, topic, textbook edition*}
Multiple Lesson Plans 3rd: { *real-world lesson plans*}
Compression result 3rd: {*Compression result*}
Curriculum 4th: {*subject, level, grade, chapter, topic, textbook edition*} *Multiple Lesson Plans 4th:* { *real-world lesson plans*}
Compression result 4th:

We formulate the lesson plan compression in an autoregressive way by:

$$P_c(\tilde{d}|I_{comp}, x, D) = \prod_{i=1}^{|\tilde{d}|} P_c(\tilde{d}_i|I_{comp}, x, D, \tilde{d}_{<i}) \tag{3}$$

[1] Typically, the curriculum information is a set of subject, level, grade, chapter, topic and textbook edition.

where P_c denotes probability distributions of compressor, $|\tilde{d}|$ is the length of the lesson plan refinement $\tilde{d}$.

Knowledge-aware with Textbooks Expander

Considering lesson plan compression only might still result in some missing knowledge information relevant to textbooks. To better align the generated lesson plan with textbook content, textbook information is also taken into account in the CE-LessonPlan framework. Since a textbook is the condensed result of a large amount of real-world knowledge sources, to generate lesson plans with detailed knowledge content, an LLM-based expander $\mathcal{E}$ is developed to enhance the knowledge in lesson plans by combining massive model parameter knowledge and textbook information. Specifically, the expander generates another lesson plan reference $\hat{d}$ given the curriculum information x and textbook content e. This process is conducted in a zero-shot manner using a textbook expansion instruction prompt, denoted as I_{tb}:

You are a {subject} expert and have won the first prize for teaching design many times. You are writing a lesson plan. Please generate the corresponding content with the provided curriculum information and textbook content.
Curriculum: {subject, level, grade, chapter, topic, textbook edition}
Textbook Content: {textbook content}

The objective of the expander is to leverage the extensive real-world knowledge embedded in LLMs to complement the information contained in textbooks, thereby generating an additional lesson plan that remains consistent with textbook content. Similar to the proposed compressor, the expander also performs generation in an auto-regressive manner as follows:

$$P_e(\hat{d}|I_{tb}, x, e) = \prod_{i=1}^{|\hat{d}|} P_e(\hat{d}_i|I_{tb}, x, e, \hat{d}_{<i}) \tag{4}$$

where P_e denotes probability distributions of expander, $|\hat{d}|$ is the length of the generated knowledge-enhanced lesson plan $\hat{d}$.

Ensemble Decoding for CE-LessonPlan

The ideal outcome of CE-LessonPlan is a consolidated reference for the lesson plan generator that not only mirrors human writing but also remains detailed and consistent with textbook information. To simultaneously consider both lesson plans generated by compression and expansion, inspired by (Liu et al. 2024; Jung et al. 2025), the CE-LessonPlan framework engages ensemble decoding that involves a multiplicative ensemble of compressor $\mathcal{C}$ and expander $\mathcal{E}$ at each decoding step. Specifically, once the compressor and expander generate their respective probability distributions for the next token, the subsequent token is selected by maximizing the weighted sum of the log probabilities of the two models, formalized as follows:

$$\begin{aligned} d_i^* = \arg\max_{d_i^* \in V} \Big(& \alpha \cdot \log P_e(\hat{d}_i \mid I_{tb}, x, e, \hat{d}_{<i}) \\ & + (1-\alpha) \cdot \log P_c(\tilde{d}_i \mid I_{comp}, x, D, \tilde{d}_{<i}) \Big) \end{aligned} \tag{5}$$

where d_i^* represents the subsequent token and α is the ensemble coefficient used to balance the two probability distributions.

Ensemble decoding enables CE-LessonPlan to seamlessly integrate retrieved lesson plans with textbook-enhanced lesson plans generated using the parametric knowledge of LLMs. Specifically, CE-LessonPlan selects the argmax token from the expander only when its probability exceeds that of the compressor. This approach ensures that CE-LessonPlan leverages the textbook information only when necessary, such as in cases where the compressor demonstrates uncertainty about the next token. In this way, the lesson plan generator produces output y^* that are human-like lesson plans meanwhile relevant to textbook content given the curriculum information x and the golden reference d^*:

$$y^* \sim \mathcal{G}(x, d^*) \tag{6}$$

Experiment

In this section, we aim to answer the following research questions through both quantitative and qualitative experimental analysis: (1) How does our lesson plan generation performance compare with the existing methods (RQ1); (2) How do the external data sources including retrieved lesson plans and textbook information impact mainstream LLMs and CE-LessonPlan (RQ2); (3) How does the ensemble coefficient affect the performance of CE-LessonPlan (RQ3); (4) How do the selective models influence the performance of CE-LessonPlan (RQ4); (5) How does the performance of CE-LessonPlan compare with the lesson plans written by teaching experts (RQ5).

Evaluation Setup To comprehensively assess the performance of CE-LessonPlan, we incorporate extensive experimental settings and evaluation methodologies. We construct a lesson plan dataset that consists of both real-world and simulated lesson plans for the model training and testing. We additionally build a textbook database extracted from public educational resources. The SFT dataset spans grades K-12 and multiple disciplines, the textbook database contains 588 textbooks across various subjects and editions, ensuring comprehensive coverage of curriculum information.

In terms of implementation, we select Qwen2-72B with a 4K context window as the foundation model to conduct SFT as our lesson plan generator. As our compressor and expander are designed to create lesson plan references, we also utilize the generator model as the foundation model for both the compressor and expander.

For evaluation, we employ both automatic evaluation and human evaluation. We use ROUGE scores to measure the relevance of generated lesson plans to textbook content. The human evaluation assigns expert annotators to assess the quality of lesson plans across 24 dimensions spanning 7

Method	Auto Evaluation		Human Evaluation								
	ROUGE-1	ROUGE-L	TA	SLA	LO	KDP	MAT	LP	AM	AVG.	Kappa
Qwen2-72B (Yang et al. 2024)	6.576	6.236	2.223	1.976	2.034	1.750	1.764	1.938	1.920	1.951	0.653
GLM-4 (GLM et al. 2024)	8.281	7.017	2.048	2.050	2.214	1.919	2.308	2.269	2.317	2.198	0.663
Claude 3.5 sonnect	6.057	5.300	2.132	1.987	2.106	1.763	2.329	2.061	2.148	2.070	0.590
GPT-4o (Achiam et al. 2023)	6.708	5.970	2.170	2.068	2.177	1.989	2.304	2.295	2.155	2.207	0.664
Self-Critique (Zheng et al. 2024)	13.317	11.631	2.378	1.767	2.299	1.965	2.605	2.366	2.068	2.255	0.645
CE-LessonPlan	14.849	13.173	2.377	2.381	2.532	2.411	1.933	2.185	2.255	2.276	0.710

Table 1: The evaluation results of lesson plan generation performance.

components. All annotators are trained through multiple iterative sessions to ensure consistent and reliable results.

Overall Performance (RQ1)

The automatic and human evaluation results of our approach and baselines are reported in Table 1. For the human evaluation, we also report the Cohen's Kappa Coefficient which reflects the consistency between the annotators (Fleiss 1971). From Table 1, we have the following observations: (1) among all the selected baselines, the Self-Critique framework performs the best in terms of ROUGE scores and human evaluation. This is because, unlike other baselines that use the initial LLM based generation results, it iteratively refines the outputs based on the strong language understanding capabilities of GPT-4 to achieve better performance; (2) The proposed CE-LessonPlan significantly outperforms both LLM baselines and existing lesson plan generation methods regarding automatic and human evaluation, showcasing the CE-LessonPlan's superiority in generating practical lesson plans rather than other methods. Please note that the human evaluation of the MAT component in CE-LessonPlan is slightly worse than other methods. We suggest after considering textbook information, the MAT content is more detailed, which may generate content that is not covered in the LP component, thereby affecting the human annotation results. Nevertheless, the average score of the human evaluation (AVG.) of the CE-Lesson Plan is the highest among all the models. Considering all dimensions comprehensively, our method has the best performance; and (3) Compared to the foundation model of our CE-LessonPlan framework, i.e., Qwen2-72B, which is without any SFT and RAG enhancement, our approach achieves better performance in all evaluation dimensions. This indicates that after the employment of SFT and RAG techniques, an universal LLM is able to adapt to the educational domain effectively, hence generating high quality lesson plans.

Analysis of External Data Sources (RQ2)

We systematically investigate the impact of retrieved lesson plans and textbook information on mainstream LLMs and CE-LessonPlan. As GPT-4o achieves the best performance among all LLMs, we select GPT-4o as the representative of mainstream LLMs. We can stimulate the few-shot learning of GPT-4o with the retrieved lesson plans (GPT-4o (3-shot)). The textbook information can provide extra knowledge data to the GPT-4o (Textbook enhanced GPT-4o). For the CE-LessonPlan, we examine the effect of the retrieved lesson plans and textbook information by removing the lesson plan compressor (Comp) and textbook expander (Exp) from the framework respectively, where "w/o" denotes the exclusion of this module from CE-LessonPlan. We mainly observe the ROUGE scores due to the high efficiency evaluation consideration. From Table 2, we can observe that: (1) both retrieved lesson plans and textbook information bring improvements to two lesson plan generation models, especially the textbook information, which shows the essential of these retrieved lesson plans and textbook information; (2) whether it is GPT-4o or our model, utilizing both data sources simultaneously showcases that it is more effective than relying on a single data source, emphasizing the indispensability of these two data sources for lesson plan generation; (3) in terms of using two data sources simultaneously, GPT-4o still performs slightly worse than our CE-LessonPlan, indicating the superiority of our compress-expand framework in generating lesson plans; and (4) since CE-LessonPlan w/o Comp achieves higher ROUGE scores than CE-LessonPlan, we further conduct additional human evaluation to compare these two methods, the average of human evaluation scores of CE-LessonPlan w/o Comp and CE-LessonPlan is 2.273 and 2.276 respectively. We suggest that CE-LessonPlan w/o Comp simply provides textbook information that does not meet the structural format requirements of a lesson plan, leading to slightly worse results. Although the generated content is highly consistent with the textbook information, it is relatively difficult to implement in real-world classroom.

Model	ROUGE-1	ROUGE-L
GPT-4o	6.708	5.970
GPT-4o (3-shot)	8.779	7.536
Textbook enhanced GPT-4o	14.016	12.775
Textbook enhanced GPT-4o (3-shot)	14.324	12.662
CE-LessonPlan w/o (Comp& Exp)	7.334	6.236
CE-LessonPlan w/o Exp	11.606	10.037
CE-LessonPlan w/o Comp	18.143	16.122
CE-LessonPlan	14.849	13.173

Table 2: The performance of different variants in GPT-4o and the proposed CE-LessonPlan. "Comp" and "Exp" denote compressor and expander respectively.

Impact of Ensemble Coefficient (RQ3)

We further explore the impacts of the ensemble coefficient α on the CE-LessonPlan performance. We mainly observe the

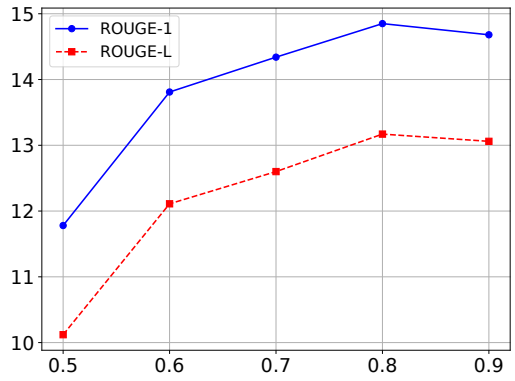

Figure 3: Impact of coefficient α on CE-LessonPlan.

ROUGE scores due to the high efficiency evaluation consideration. The results are illustrated in Figure 3. We limit the range of α to [0.5, 0.9] as the expander significantly influences the model performance more than the compressor. We observe that with the increasing values of α, CE-LessonPlan obtains better performance gradually and performs the best ROUGE scores when $\alpha = 0.8$, indicating the efficient integration of information from our lesson plan compressor and textbook expander. Once $\alpha = 0.9$, the performance of CE-LessonPlan slightly declines, which suggests that compression results and expansion results complement each other and that paying excessive attention to a sole result may encounter counterproductive consequences.

Selective Models		**Lesson Plan Generator**	
		Qwen2.5-72B	Ours
Comp & Expander	Qwen2.5-72B	13.044/11.258	11.964/10.599
	Ours	14.511/12.863	14.849/13.173

Table 3: ROUGE-1 (left) and ROUGE-L (right) of selective models for CE-LessonPlan.

Influence of Selective Models (RQ4)

There are three core modules in our CE-LessonPlan framework, including a lesson plan generator, a lesson plan compressor and a textbook expander. These three modules are served flexibly by various models. We further examine the performance of the deployment with different LLMs in each module. Please note that the model of the compressor (Comp) and expander (Exp) must be the same one to conduct ensemble decoding to integrate their timing outputs. We mainly observe the ROUGE scores due to the high efficiency evaluation consideration. We investigate the usage of universal open-source LLM, Qwen2.5-72B-Instruct (Qwen2.5-72B), and our SFT lesson plan model as the lesson plan generator, compressor and expander respectively. From table 3, we can see that compared to using Qwen2.5-72B-Instruct as a lesson plan generator or comp & expander, using our SFT model as a generator and comp & expander simultaneously achieves the highest ROUGE scores. These results emphasize the effectiveness, versatility and flexibility of our compress-expand framework.

Module	**Win** (%)	**Loss** (%)	**Tie** (%)	**Kappa**
TA	50.00%	0.00%	50.00%	1.000
SLA	60.00%	4.44%	35.56%	0.563
LO	41.27%	11.11%	47.62%	0.545
KDP	52.63%	0.00%	47.37%	0.719
MAT	33.33%	22.22%	44.44%	0.500
LP	22.62%	19.05%	58.33%	0.628
AM	0.00%	0.00%	100.00%	1.000

Table 4: CE-LessonPlan vs. Human.

Comparisons of CE-LessonPlan and Human Experts (RQ5)

To explore the human-like level of the automatically generated lesson plans, we conducted a blind test to compare lesson plans generated by the CE-LessonPlan framework with those written manually by educators collected from a third-party education platform. We randomly select 28 lesson plans and the corresponding generative results by our approach. To ensure an unbiased evaluation, we invited three annotators with at least five years of teaching experience to assess the quality of the lesson plans without informing them which ones were LLM generated and which were handcrafted. The annotators are instructed to annotate which lesson plan is better or label a "tie", by considering the proposed evaluation dimensions. As shown in Table 4, about 80% of lesson plans generated by our approach have comparable (Tie) or better performance (Win) to those created by educators. This indicates that the CE-LessonPlan framework is capable of producing high-quality lesson plans that are virtually indistinguishable from those written by experienced educators (even better). We suggest our approach potentially reduces teachers' workload in teaching preparation by providing well-organized drafts for them so that they can edit them for their individual requirements.

Conclusion

In this paper, we introduce CE-LessonPlan, a novel methodology that leverages LLMs with real-world lesson plan data and textbook information to automatically generate well-designed lesson plans. To avoid the occurrence of low-quality content in retrieved documents that might have negative impacts on model performance, we develop a compressor to refine these retrieved data. Additionally, we present an expander to augment the textbook information with the model's parametric knowledge to provide more detailed knowledge for creating instructional content. By integrating the outputs from both compression and expansion through ensemble decoding, we create high-quality references to enhance the lesson plan generation capabilities of LLMs. In this way, our CE-LessonPlan can generate drafts for teachers, thereby reducing their workload so that they only need to revise the generated content according to their preferences and classroom context with our generated results.

Acknowledgments

This work was supported in part by National Key R&D Program of China, under Grant No. 2022YFC3303600; in part by NFSC under Grant No. 62477025; in part by Key Laboratory of Smart Education of Guangdong Higher Education Institutes, Jinan University (2022LSYS003) and in part by Beijing Municipal Science and Technology Project under Grant No. Z241100001324011.

References

Achiam, J.; Adler, S.; Agarwal, S.; Ahmad, L.; Akkaya, I.; Aleman, F. L.; Almeida, D.; Altenschmidt, J.; Altman, S.; Anadkat, S.; et al. 2023. GPT-4 Technical Report. *arXiv preprint arXiv:2303.08774*.

Azubike, N. O. 2021. Lesson Plan and the Relevant Teaching Skills As Instrument for Effective Teaching Practice. *International Journal of Progressive and Alternative Education*, 7(1): 1–15.

Cao, Z.; Cao, Q.; Lu, Y.; Peng, N.; Huang, L.; Cheng, S.; and Su, J. 2024. Retaining Key Information under High Compression Ratios: Query-Guided Compressor for LLMs. In *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, 12685–12695.

Fleiss, J. L. 1971. Measuring Nominal Scale Agreement among Many Raters. *Psychological Bulletin*, 76(5): 378.

GLM, T.; Zeng, A.; Xu, B.; Wang, B.; Zhang, C.; Yin, D.; Rojas, D.; Feng, G.; Zhao, H.; Lai, H.; et al. 2024. ChatGLM: A Family of Large Language Models from GLM-130B to GLM-4 All Tools. *arXiv preprint arXiv:2406.12793*.

Guu, K.; Lee, K.; Tung, Z.; Pasupat, P.; and Chang, M. 2020. Retrieval Augmented Language Model Pre-training. In *International Conference on Machine Learning*, 3929–3938. PMLR.

Hu, B.; Zheng, L.; Zhu, J.; Ding, L.; Wang, Y.; and Gu, X. 2024. Teaching Plan Generation and Evaluation With GPT-4: Unleashing the Potential of LLM in Instructional Design. *IEEE Transactions on Learning Technologies*.

Iqbal, M. H.; Siddiqie, S. A.; and Mazid, M. A. 2021. Rethinking Theories of Lesson Plan for Effective Teaching and Learning. *Social Sciences & Humanities Open*, 4(1): 100172.

Jiang, H.; Wu, Q.; Luo, X.; Li, D.; Lin, C.-Y.; Yang, Y.; and Qiu, L. 2024. LongLLMLingua: Accelerating and Enhancing LLMs in Long Context Scenarios via Prompt Compression. In *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, 1658–1677.

Jung, D.; Liu, Q.; Huang, T.; Zhou, B.; and Chen, M. 2025. Familiarity-Aware Evidence Compression for Retrieval-Augmented Generation. *Proceedings of the 2025 Conference on Empirical Methods in Natural Language Processing*.

Ke, Z.; Kong, W.; Li, C.; Zhang, M.; Mei, Q.; and Bendersky, M. 2024. Bridging the Preference Gap between Retrievers and LLMs. In *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, 10438–10451.

Lee, G.-G.; and Zhai, X. 2024. Using ChatGPT for Science Learning: A Study on Pre-service Teachers' Lesson Planning. *IEEE Transactions on Learning Technologies*.

Li, H.; Ding, W.; and Liu, Z. 2020. Identifying At-Risk K-12 Students in Multimodal Online Environments: A Machine Learning Approach. *International Educational Data Mining Society*.

Li, X. L.; Holtzman, A.; Fried, D.; Liang, P.; Eisner, J.; Hashimoto, T. B.; Zettlemoyer, L.; and Lewis, M. 2023. Contrastive Decoding: Open-ended Text Generation as Optimization. In *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, 12286–12312.

Li, Z.; Hu, X.; Liu, A.; Zheng, K.; Huang, S.; and Xiong, H. 2024. Refiner: Restructure Retrieved Content Efficiently to Advance Question-Answering Capabilities. In *Findings of the Association for Computational Linguistics: EMNLP 2024*, 8548–8572.

Lin, C.-Y. 2004. ROUGE: A Package for Automatic Evaluation of Summaries. In *Text Summarization Branches Out*, 74–81.

Liu, Q.; Wang, F.; Xu, N.; Yan, T. L.; Meng, T.; and Chen, M. 2024. Monotonic Paraphrasing Improves Generalization of Language Model Prompting. In *Findings of the Association for Computational Linguistics: EMNLP 2024*, 9861–9877.

Liu, Z.; Liu, Q.; Guo, T.; Chen, J.; Huang, S.; Zhao, X.; Tang, J.; Luo, W.; and Weng, J. 2023. XES3G5M: A Knowledge Tracing Benchmark Dataset with Auxiliary Information. *Advances in Neural Information Processing Systems*, 36: 32958–32970.

Liu, Z.; Xu, G.; Liu, T.; Fu, W.; Qi, Y.; Ding, W.; Song, Y.; Guo, C.; Kong, C.; Yang, S.; et al. 2020. Dolphin: A Spoken Language Proficiency Assessment System for Elementary Education. In *Proceedings of The Web Conference 2020*, 2641–2647.

Nesari, A. J.; and Heidari, M. 2014. The Important Role of Lesson Plan on Educational Achievement of Iranian EFL Teachers' Attitudes. *International Journal of Foreign Language Teaching & Research*, 3(5): 25–31.

Ouyang, L.; Wu, J.; Jiang, X.; Almeida, D.; Wainwright, C.; Mishkin, P.; Zhang, C.; Agarwal, S.; Slama, K.; Ray, A.; et al. 2022. Training Language Models to Follow Instructions with Human Feedback. *Advances in Neural Information Processing Systems*, 35: 27730–27744.

Radford, A.; Narasimhan, K.; Salimans, T.; Sutskever, I.; et al. 2018. Improving Language Understanding by Generative Pre-training.

Radford, A.; Wu, J.; Child, R.; Luan, D.; Amodei, D.; Sutskever, I.; et al. 2019. Language Models are Unsupervised Multitask Learners. *OpenAI Blog*, 1(8): 9.

Sugianto, A. 2020. Applying a Lesson Plan for a Digital Classroom: Challenges and Benefits. *International Journal of English Education and Linguistics (IJoEEL)*, 2(2): 62–74.

Touvron, H.; Martin, L.; Stone, K.; Albert, P.; Almahairi, A.; Babaei, Y.; Bashlykov, N.; Batra, S.; Bhargava, P.; Bhosale, S.; et al. 2023. Llama 2: Open Foundation and Fine-Tuned Chat Models. *arXiv preprint arXiv:2307.09288*.

Wang, Z.; Araki, J.; Jiang, Z.; Parvez, M. R.; and Neubig, G. 2023. Learning to Filter Context for Retrieval-Augmented Generation. *arXiv preprint arXiv:2311.08377*.

Xu, F.; Shi, W.; and Choi, E. 2024. RECOMP: Improving retrieval-augmented LMs with context compression and selective augmentation. In *The Twelfth International Conference on Learning Representations*.

Yang, A.; Yang, B.; Hui, B.; Zheng, B.; Yu, B.; Zhou, C.; Li, C.; Li, C.; Liu, D.; Huang, F.; et al. 2024. Qwen2 Technical Report. *arXiv preprint arXiv:2407.10671*.

Yoon, C.; Lee, T.; Hwang, H.; Jeong, M.; and Kang, J. 2024. CompAct: Compressing Retrieved Documents Actively for Question Answering. In *Proceedings of the 2024 Conference on Empirical Methods in Natural Language Processing*, 21424–21439.

Zhang, T.; Patil, S. G.; Jain, N.; Shen, S.; Zaharia, M.; Stoica, I.; and Gonzalez, J. E. 2024. RAFT: Adapting Language Model to Domain Specific RAG. *arXiv preprint arXiv:2403.10131*.

Zheng, Y.; Huang, S.; Zeng, X.; Huang, Y.; Liu, Z.; and Luo, W. 2025. Knowledge-enhanced large language models for automatic lesson plan generation. *Humanities and Social Sciences Communications*, 12(1784).

Zheng, Y.; Li, X.; Huang, Y.; Liang, Q.; Guo, T.; Hou, M.; Gao, B.; Tian, M.; Liu, Z.; and Luo, W. 2024. Automatic Lesson Plan Generation via Large Language Models with Self-critique Prompting. In *International Conference on Artificial Intelligence in Education*, 163–178. Springer.