Problem-Oriented Segmentation and Retrieval: Case Study on Tutoring Conversations

Anonymous ACL submission

Abstract

Many open-ended conversations (e.g., tutoring lessons or business meetings) revolve around pre-defined reference materials, like worksheets or meeting bullets. To provide a framework for studying such conversation structure, we introduce **Problem-Oriented Segmentation & Retrieval** (POSR)[1], the task of *jointly* breaking down conversations into segments and linking each segment to the relevant reference item. As a case study, we apply POSR to education where effectively structuring lessons around problems is critical yet difficult. We present **LessonLink**, the first dataset of real-world tutoring lessons, featuring 3,500 segments, spanning 24,300 minutes of instruction and linked to 116 SAT Math problems. We define and evaluate several joint and independent approaches for POSR, including segmentation (e.g., TextTiling), retrieval (e.g., ColBERT), and large language models (LLMs) methods. Our results highlight that modeling POSR as one joint task is essential: POSR methods outperform independent segmentation and retrieval pipelines by up to +76% on joint metrics and surpass traditional segmentation methods by up to +78% on segmentation metrics. We demonstrate POSR's practical impact on downstream education applications, deriving new insights on the language and time use in real-world lesson structures.[2]

1 Introduction

Across education, business, and science, many open-ended conversations like meetings or tutoring sessions are designed to address a set of predefined topics. As a prominent example, educators often shape their lessons around worksheet problems. Structuring lessons effectively is critical but challenging, as educators must allocate the right amount of time to different problems, while addressing different student learning needs (Haynes, 2010; Henderson, 1997; Panasuk and Todd, 2005). However, many novices or educators teaching large groups of students struggle with lesson structuring and often run out of time (Stradling and Saunders, 1993; Pozas et al., 2020; Deunk et al., 2018; Takaoglu, 2017; Hejji Alanazi, 2019).

[1]Pronounced as "poser" (/ˈpozər/), a perplexing problem.

[2]You can find our code and LessonLink dataset as a zip file in our submission. If our work is accepted, the public-facing manuscript will include a GitHub link.

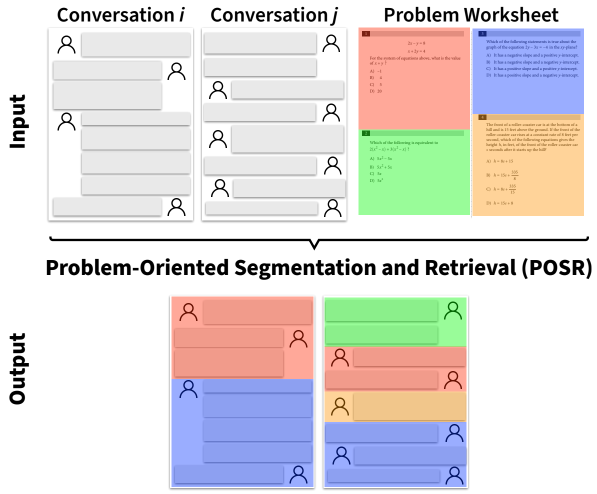


Figure 1: Problem-Oriented Segmentation and Retrieval (POSR) provides a framework for studying conversation structure around reference materials. For example, while conversations i, j discuss the same worksheet, POSR reveals that conversation i covers fewer problems than j but spends more time per problem.

Providing evidence-based insights on lesson structuring is a key step towards addressing this challenge. These insights provide educators feedback on their teaching (Fishman et al., 2003; Kraft et al., 2018; Lomos et al., 2011; Desimone, 2009), tutoring platforms on training priorities (Hilliger et al., 2020; Gottipati and Shankararaman, 2018; Hilliger et al., 2022) and curriculum developers on material design (O'Donnell, 2008; Fullan and Pomfret, 1977). Unfortunately, obtaining insights on lesson structures at scale is challenging.

The study of conversation structure around reference materials draws on concepts from two, typically distinct natural language processing (NLP) tasks: *discourse segmentation* to identify segments in the conversations and *information retrieval* (IR) to retrieve the relevant reference material for each segment. While each task has rich literature, studying them jointly reveals real-world challenges that existing works bypass. For example, discourse segmentation methods assume that conversations share the same structure (Ritter et al., 2010; Hearst and Plaunt, 1993; Chen and Yang, 2020), but education conversations have unique structures as teachers adapt their lessons to different needs. While prior IR work has studied supporting natural-language queries over conversations (Sanderson et al., 2010; Oard et al., 2004; Chelba et al., 2008), the reverse task of using open-ended conversation segments as queries for retrieving domain-specific reference materials has not received similar attention.

To address these gaps, we make several key contributions. We define the **Problem-Oriented Segmentation and Retrieval** (POSR) task for jointly segmenting conversations and linking segments to relevant reference materials, such as worksheet problems (Figure 1). Unlike segmentation or retrieval alone, the joint POSR task reflects the realistic opportunities and challenges presented by knowing the potential reference topics (from the reference materials) for conversation segments.

POSR provides a general framework for studying conversation structure around reference materials. As a case study, we apply POSR to the education setting. We contribute **LessonLink, a novel dataset of real-world tutoring lessons featuring 3,500 segments, 116 SAT (Scholastic Aptitude Test) Math problems, and over 24,300 minutes of instruction**. Our open-source dataset consists of real tutoring conversations paired with SAT math worksheets, each conversation lasting about 1.5 hr long. Each conversation is segmented and each segment is linked with one of the 116 problems. To the best of our knowledge, this is the first dataset to include real-world conversations of unique structures linked with reference materials like worksheets.

Evaluating POSR is challenging: Existing segmentation metrics do not measure time-weighed errors and existing metrics fail to reflect the subtle ways in which segmentation and retrieval errors interact. To address this, we contribute **time-aware segmentation metrics** adapted from standard line-based metrics (e.g., WindowDiff from Pevzner and Hearst (2002)) and introduce the **Segmentation and Retrieval Score (SRS)** to jointly measure segmentation and retrieval accuracy as the proportion of conversation where the retrieved item matches the ground truth.

We **define and evaluate a suite of segmentation, retrieval and POSR methods** on LessonLink, including traditional segmentation methods like TextTiling (Hearst, 1997), popular IR methods like ColBERT (Khattab and Zaharia, 2020) and long-context large language models (LLMs) like Claude and GPT-4 (Anthropic, 2024; OpenAI, 2024). Our results highlight the importance of POSR's joint approach: POSR methods outperform independent segmentation and retrieval pipelines by up to +76% on SRS metrics and traditional segmentation methods by up to +78% on segmentation metrics. However, several challenges remain. In domains with high privacy risks like education, companies are often unwilling to share data long-term due to privacy concerns. Moreover, while LLMs achieve strong POSR performance, their high API costs on long texts raise scalability concerns. Our findings motivate the need for more cost-effective, open-sourced methods that can deliver high accuracy on joint reasoning tasks like POSR.

Finally, to further highlight the utility of POSR to real-world scenarios, we describe **two novel applications of POSR** to illustrate its potential for impacting evidence-based practices in education. First, through a linguistic analysis, we discover that tutors who spend more time on problems provide richer conceptual explanations. Tutors who spend less time provide procedural explanations. Second, POSR quantifies wide variability in how long tutors spend on the same problem. These examples point to opportunities for improving language and time-management practices.

2 Related Work

Discourse segmentation is the task of partitioning conversations into segments, traditionally a preprocessing step before retrieval or summarization of conversations (Hearst and Plaunt, 1993; Callan, 1994; Wilkinson, 1994; Galley et al., 2003; Chen and Yang, 2020; Althoff et al., 2016; Salton and Buckley, 1991a,b; Salton et al., 1996; Huang et al., 2003). Different domains like customer service or meetings define segments differently, e.g. as a speech act, a topic, or a conversation stage (Liu et al., 2023; Riedl and Biemann, 2012; Prabhakaran et al., 2018); In this work, we study *problem-oriented* segments: conversation segments that discuss individual math problems. While most existing segmentation methods assume conversations exhibit predictable structure (Ritter et al., 2010; Hearst and Plaunt, 1993; Chen and Yang, 2020), education conversations are diverse and lack such predictable structure.

Math information retrieval poses special challenges (Munavalli and Miner, 2006; Sojka and Líška, 2011; Nguyen et al., 2012) because math expressions can be difficult to represent contextually (Schubotz et al., 2016; Kamali and Tompa, 2013; Zanibbi and Blostein, 2012; Aizawa and Kohlhase, 2021). Our setting combines these challenges with the additional difficulty of treating conversational segments as queries, unlike typical retrieval using well-formed keyword queries (Wang et al., 2024). Our LessonLink dataset provides a new resource of real-world education conversations segmented and linked to math problems from worksheets. This enables the study of POSR, combining discourse segmentation with retrieval of math materials.

Evaluation metrics for segmentation include P_k (Beeferman et al., 1997) and WindowDiff (Pevzner and Hearst, 2002). Both measure the segmentation accuracy based on a *line-level* sliding window (Morris and Hirst, 1991; Kozima, 1996; Reynar, 1999; Choi, 2000; Beeferman et al., 1999) but neither account for the time duration of a line, which can confound accuracy reporting for real-world applications (Grosz and Hirschberg, 1992; Nakatani et al., 1995; Passonneau and Litman, 1997; Hirschberg and Nakatani, 1998; Repp et al., 2007). We develop a time-based version of P_k and WindowDiff and propose a time-based SRS metric for assessing the holistic performance.

3 Problem-Oriented Segmentation and Retrieval (POSR)

We define the task of Problem-Oriented Segmentation and Retrieval (POSR) as jointly dividing a *conversation transcript* into segments and retrieving the *relevant topic* (e.g., problem) discussed in each segment. While segmentation and retrieval are individually challenging, POSR jointly addresses them together to improve ecological validity and expose new system design tradeoffs. We hypothesize (and show in Section §6) that systems aware of retrieval topics will segment better, and vice versa, motivating joint POSR methods.

Algorithm 1 POSR vs. non-POSR methods

Require: T, R
  if with POSR **then**
    $s_1, \ldots, s_N \leftarrow \texttt{segment}(T, R)$
  else
    $s_1, \ldots, s_N \leftarrow \texttt{segment}(T)$
  end if
  $w_1, \ldots, w_N \leftarrow \texttt{retrieve}([s_1, \ldots, s_N], R)$

3.1 Task Definition

Given a transcript $T = \langle T_1, ..., T_N \rangle$ of N lines and a corresponding reference corpus $R = \langle R_1, ..., R_W \rangle$ (e.g., a worksheet of problem entries), the POSR objective is to output an array of segment id and problem reference id for each line in the transcript, $Y = [(s_1, w_1), (s_2, w_2), \ldots, (s_N, w_N)]$:

- $s_1, \ldots, s_N$ is the segment id for each line in line. So, s_1 is the segment id for the line 1, s_2 the segment id for line 2, and so on.
- $w_1, \ldots, w_N \in \{R_1, \ldots, R_W\}$ indicate the problem reference id from the corpus.[3]

Since these transcripts originate from real-world conversations, each line T_i is associated with a start and end timestamp, $t_i^{\text{start}}, t_i^{\text{end}}$. Algorithm 1 highlights POSR methods, which take both transcript T and retrieval corpus R into account for segmentation, in contrast to independent segmentation and retrieval methods.

3.2 Metrics

To evaluate the effectiveness of POSR methods, we introduce the standard and our novel metrics for evaluating segmentation and retrieval individually and jointly. As evident in Algorithm 1, the segmentation metrics help capture how segmentation may be improved by accounting for the retrieval corpus. We additionally adapt standard metrics to also take time into account. Finally, we also account for practical considerations by reporting cost.

Existing, line-based segmentation metrics. We use two established metrics for segmentation accuracy: WindowDiff from Pevzner and Hearst (2002) and P_k metric from Beeferman et al. (1999). Both

[3] If $s_i = s_j$ then $w_i = w_j$.

use a line-based sliding window approach that measures boundary mismatches within the window. Lower values are better for both metrics. For example, WindowDiff is computed as:

$$\text{WindowDiff}(Y, Y^*) = \frac{1}{N-k}\sum_{j=1}^{N-k} \mathbb{1}(|b(s_{j:j+k}) - b(s^*_{j:j+k})| > 0),$$

where $b(\cdot)$ represents the number of boundaries within the $\cdot$ window and k is typically set to half of the average of the true segment line size. P_k is similar but penalizes false-negatives more, i.e., missed segments. For conciseness, we leave P_k's definition in Appendix §A.

New, time-based variants of segmentation metrics. Existing segmentation metrics operate at a line-level and do not account for the time duration of segments. However, in education settings, time spent per segment is crucial to understanding lesson structures (Stevens and Bavelier, 2012; Martens and Wyble, 2010; Heim and Keil, 2012; Eze and Misava, 2017). To address this, we propose Time-WindowDiff and Time-P_k, new *time-based* variants of P_k and and WindowDiff. Time-Windowdiff is calculated as:

$$\text{Time-WindowDiff}(Y, Y^*) = \frac{1}{N-k}\sum_{j=1}^{N-k} \mathbb{1}(|b(s_{t_j^{\text{start}}:t_j^{\text{end}}+\Delta_k}) - b(s^*_{t_j^{\text{start}}:t_j^{\text{end}}+\Delta_k})| > 0),$$

where Δ_k, the time duration of the sliding window, is half of the average true segment duration (similar to k). $b(s_{t_j^{\text{start}}:t_j^{\text{end}}+\Delta_k})$ refers to the number of boundaries within the window that starts at t_j^{start} and ends at $t_j^{\text{end}} + \Delta_k$. This ensures that long and short segment durations are appropriately weighted in the evaluation. For conciseness, we leave Time-P_k's definition in Appendix §A.

API cost. Closed-sourced models result in high API usage costs, especially on thousands of long conversations such as in our setting.[4] Educational organizations may be less inclined to rely on expensive methods without justified trade-offs. Thus, we report the average cost per 100 transcripts[5].

[4]Third-party models additionally raise privacy and intellectual property concerns especially in domains that deal with sensitive data, like student data and copyrighted materials.

[5]Based on OpenAI and Anthropic pricing in 05/24-06/24.

The Segmentation Retrieval Score (SRS). Evaluating POSR methods presents unique challenges because of interdependencies between segmentation and retrieval. On the one hand, segmentation may improve with access to the retrieval corpus in disambiguating segment boundaries. On the other hand, incorrect segmentation make retrieval evaluations difficult as the retrieved content cannot be easily checked with misaligned segment boundaries and ids.

We propose the Segmentation Retrieval Score (SRS), which accounts for this by evaluating the correctness of retrieved topics, conditioned on the predicted segmentation. False positive segments overly penalize an exact segment match. Therefore, SRS only requires the retrieved topic w_j, determined based on the predicted segment s_j (rf. Algorithm 1), to match the reference w_j^* for a line to be considered correct. This allows some flexibility in segment boundaries as long as the retrieved topics are accurate. SRS is defined as:

$$\alpha\text{-SRS}(Y, Y^*) = \frac{1}{\sum_j \alpha_j}\sum_{j=1}^{N} \alpha_j \mathbb{1}(w_j(s_j) == w_j^*)$$

where line-based SRS has $\alpha_j = 1$ and time-based SRS has $\alpha_j = t_j^{\text{end}} - t_j^{\text{start}}$.

4 The LessonLink Dataset

We introduce the LessonLink dataset as a concrete case study of POSR. LessonLink contains real-world tutoring lesson transcripts segmented and linked with problems in SAT math worksheets. The dataset features 3,500 segments of over 24,300 minutes of instruction, featuring 1,300 unique speakers and 116 linked problems. Table 1 summarizes the statistics of the dataset. We release the LessonLink dataset under the CC Noncommercial 4.0 license[6].

Data source. We collected the data in partnership with Schoolhouse.world, a free peer-to-peer tutoring platform that supports over ~80k students worldwide with the help of ~10k volunteer tutors. One of their main focuses is to help high school students prepare for the SAT, a standardized test used for college admissions in the United States. The platform shared de-identified transcripts with us from their March 2023 SAT Math Bootcamp, a four week-long course where tutors met with

[6]https://creativecommons.org/licenses/by-nc/4.0/

Transcripts	Total Transcripts	300
	Total Speakers	1377
	Total Segments	3576
	Mean Speakers Per Transcript	6.37
	Mean Segments Per Transcript	11.92
	Mean Problems Per Transcript	7.43
	Mean Lines Per Transcript	495.51
	Mean Duration (mins)	81.62
Worksheets	Total Worksheets	7
	Total Problems	116

Table 1: **LessonLink dataset statistics.**

students in small groups twice a week to practice SAT math problems. We randomly picked 300 transcripts. Schoolhouse received consent from parents and students to share de-identified data for research purposes. The maximum tutor-student ratio in each bootcamp is 1:10. Tutoring lessons are 80 minutes long. Schoolhouse recommends a lesson structure that starts with 30 minutes of warm-up exercises followed by the students working on the worksheet independently and then a group review. Tutors have freedom in structuring their lesson and they typically use their students' practice test results to determine what to focus on.

Transcripts. Each tutoring lesson is recorded and transcribed automatically via Zoom. Schoolhouse de-identified the transcripts using the Edu-ConvoKit library (Wang and Demszky, 2024), with tutor and student names replaced with placeholder tokens "[TUTOR]" and "[STUDENT]".

Worksheets. Each transcript is linked to an SAT problem worksheet that the tutor and students work on during the lesson. The sheets include official, publicly available math practice problems created by College Board, the organization that administers the SAT.[7] Each worksheet has about 16 problems on average. We split each worksheet into separate problem images, and use Pytesseract, an optical character recognition (OCR) tool, to extract the text content from the images (PyTesseract, 2017). OCR does not capture the visual components (e.g., graphs). We focus only on using the text data, and leave visual data for future work.

Annotation. The definition of a segment varies across domains like customer service, meetings, and tutoring sessions (Liu et al., 2023; Riedl and Biemann, 2012). Our definition builds on Schoolhouse.world's curriculum structure that dedicates time for an introduction to the session, targeted warm-up exercises, and worksheet problems. We use the following segment categories: (1) **Informal.** These segments include introductory talk or off-task discussions (Carpenter et al., 2020; Rodrigo et al., 2013). Examples include the group doing an ice-breaker game. (2) **Warm-up problem.** These segments discuss warm-up problems that are not a part of the session's main worksheet. (3) **Worksheet problem.** These segments discuss a problem from the session's main worksheet.

We recruited 3 annotators for data annotation. Segment annotations happen at the level of a transcript line, as provided by Zoom. To determine human agreement on this task, the annotators annotated the same 30 lesson transcripts for segments and linked problems. On a line-level, the inter-rater segmentation accuracy was 98.9% and retrieval accuracy was 100%. We also use Cochran's Q (Cochran, 1950) to evaluate segmentation agreement, similar to prior work (Galley et al., 2003): Cochran's test evaluates the null hypothesis that the number of subjects assigning a boundary at any position is random. The test shows that the inter-rater reliability is significant to the 0.01 level for 98% of the transcripts. Given the high inter-rater agreement, the 3 annotators annotated 300 transcripts. We create a small 1:10 train/test split on our dataset: The train set containing 30 transcripts and the test set 270 transcripts.

5 Evaluation

This section describes the methods and evaluation setup which uses LessonLink's test split. Appendix §B includes more information on our prompting setup for GPT4 and Claude LLMs.

Segmentation. We evaluate a series of common segmentation methods. We evaluate top-10 and top-20 word segmentation, i.e. we take the top-10 and 20 words found in the segment boundaries of the train set to segment the test set. We also evaluate existing approaches like TextTiling (Hearst, 1997)[8] and topic- and stage-segmentation methods from Althoff et al. (2016) and Chen and Yang (2020). Lastly, we test zero-shot prompting long-context LLMs like GPT-4-turbo (OpenAI, 2024) and the Claude variants Haiku, Sonnet, and Opus (Anthropic, 2024).[9] We omit open-source, instruct-

[7] https://satsuite.collegeboard.org/sat/practice-preparation/practice-tests

[8] We use the NLTK libary implementation of the algorithm (Bird et al., 2009)

[9] These evaluations were performed in May 2024.

tuned LLMs like Llama-2 (Touvron et al., 2023), Llama-3 (Meta, 2024), or Mixtral (Jiang et al., 2024) because their context windows are not long enough for our transcripts.

We fit the topic and stage segmentation methods on our train split, and use three pre-trained encoders from Sentence-Transformers (Reimers and Gurevych, 2019): the base-nli-stsb-mean-tokens (originally used in Chen and Yang (2020)), all-mpnet-base-v2, all-MiniLM-L12-v2. These encoders did not vary in performance. Therefore, we report results on the first encoder and Appendix D reports the rest. Stage segmentation requires the number of segments a priori; our experiments vary this to be either the rounded average or maximum number of segments in LessonLink.

Retrieval. We evaluate several methods for IR: Jaccard similarity (Jaccard, 1912), TD-IDF (Sammut and Webb, 2011), BM25 (Robertson et al., 2009), ColBERTv2 (Santhanam et al., 2021), zero-shot prompting GPT-4-turbo, Claude Haiku, Claude Sonnet, and Claude Opus. A challenge in using traditional IR methods in our setting is specifying that nothing in the worksheet is linked to a segment, e.g., for informal or warm-up segments. For instruct-tuned LLMs, we can simply specify this in the prompt. For traditional IR methods, we must set a threshold value for what is deemed relevant enough to the segment. We perform 5-fold cross validation on the training set and set the threshold to the average value that best separates on the held-out fold. We report these thresholds in Appendix §C.

POSR. We combine the best independent segmentation method with each retrieval method and report their joint performance. We also evaluate zero-shot prompted GPT-4-turbo, Claude Haiku, Claude Sonnet, Claude Opus as POSR methods that perform segmentation and retrieval jointly.

6 Results

Table 2 summarizes the joint evaluations, and Table 3 summarizes the segmentation results. **The POSR methods outperform most independent segmentation and retrieval approaches, and at lower costs.** `POSR Opus` and `POSR GPT4` achieves slightly higher Line- and Time-SRS to their independent counterparts, and much higher to other combined independent approaches, e.g., `Opus+TDIDF` on both SRS metrics. Additionally,

Segmentation Method	Retrieval Method	POSR Metrics SRS (↑) Line	Time	Cost (↓)
`Opus`	`Jaccard`	0.62 ± 0.19	0.63 ± 0.19	17.17 ± 4.82
`Opus`	`TFIDF`	0.63 ± 0.22	0.63 ± 0.22	17.17 ± 4.82
`Opus`	`BM25`	0.51 ± 0.23	0.52 ± 0.23	17.17 ± 4.82
`Opus`	`ColBERT`	0.50 ± 0.23	0.5 ± 0.23	17.17 ± 4.82
`Opus`	`GPT4`	0.87 ± 0.13	0.88 ± 0.13	54.22 ± 15.14
`Opus`	`Haiku`	0.57 ± 0.23	0.57 ± 0.23	18.10 ± 4.91
`Opus`	`Sonnet`	0.68 ± 0.20	0.69 ± 0.20	28.30 ± 6.93
`Opus`	`Opus`	0.85 ± 0.11	0.85 ± 0.11	72.80 ± 21.57
`POSR GPT4`		$\mathbf{0.88 \pm 0.12}$	$\mathbf{0.89 \pm 0.11}$	11.71 ± 2.71
`POSR Haiku`		0.60 ± 0.22	0.60 ± 0.22	$\mathbf{0.35 \pm 0.08}$
`POSR Sonnet`		0.84 ± 0.15	0.85 ± 0.15	4.23 ± 0.93
`POSR Opus`		$\mathbf{0.88 \pm 0.11}$	$\mathbf{0.89 \pm 0.11}$	21.08 ± 4.62

Table 2: **POSR evaluations.** The best average is **highlighted**.

Method	Segmentation Metrics P_k (↓) Line	Time	WindowDiff (↓) Line	Time
`Top-10`	0.58 ± 0.04	0.28 ± 0.16	1.0 ± 0.01	1.0 ± 0.0
`Top-20`	0.58 ± 0.04	0.28 ± 0.16	1.0 ± 0.0	1.0 ± 0.0
`TextTiling`	0.58 ± 0.05	0.27 ± 0.16	0.90 ± 0.11	0.94 ± 0.06
`Topic`	0.58 ± 0.04	0.27 ± 0.16	1.0 ± 0.02	1.0 ± 0.01
$\texttt{Stage}_{avg}$	0.58 ± 0.04	0.28 ± 0.16	1.0 ± 0.0	1.0 ± 0.0
$\texttt{Stage}_{max}$	0.58 ± 0.04	0.28 ± 0.16	1.0 ± 0.0	1.0 ± 0.0
`GPT4`	0.20 ± 0.10	0.25 ± 0.17	0.33 ± 0.09	0.52 ± 0.15
`Haiku`	0.29 ± 0.14	0.30 ± 0.17	0.39 ± 0.14	0.55 ± 0.16
`Sonnet`	0.24 ± 0.14	0.23 ± 0.18	0.37 ± 0.15	0.53 ± 0.17
`Opus`	0.15 ± 0.09	$\mathbf{0.11 \pm 0.10}$	0.31 ± 0.13	0.46 ± 0.17
`POSR GPT4`	0.16 ± 0.01	0.18 ± 0.17	0.32 ± 0.09	0.53 ± 0.17
`POSR Haiku`	0.24 ± 0.10	0.22 ± 0.13	0.35 ± 0.11	0.51 ± 0.17
`POSR Sonnet`	$\mathbf{0.13 \pm 0.08}$	$\mathbf{0.11 \pm 0.12}$	0.31 ± 0.09	0.49 ± 0.17
`POSR Opus`	$\mathbf{0.13 \pm 0.08}$	0.12 ± 0.13	$\mathbf{0.28 \pm 0.10}$	$\mathbf{0.44 \pm 0.17}$

Table 3: **Segmentation evaluations.** The best average is **highlighted**.

we find that POSR methods are much more cost-effective: `POSR Opus` and `POSR GPT4` cost \$11-\$21 per 100 transcripts, while the best combined independent methods, `Opus+GPT4`, cost \$54 per 100 transcripts. This demonstrates the importance of POSR of jointly modelling segmentation and retrieval for better accuracy *and* cost performance. However, there is still room for improvement such as future work on developing and improving open-sourced long-context methods.

According to Table 3, **POSR methods perform better than most independent segmentation methods by a large margin.** For example, `POSR Opus` improves upon topic and stage segmentation methods by $\sim$ 57% on P_k and WindowDiff. The poor performance of top-10 and top-20 word segmentation indicates that segmentation cannot be solved by word-level cues alone. Addition-

Method	# Segment Diff
Top-10	236.84 ± 75.98
Top-20	305.37 ± 90.04
TextTiling	42.97 ± 17.93
Topic	148.61 ± 52.044
Stage$_{avg}$	367.40 ± 115.82
Stage$_{max}$	371.27 ± 118.84
GPT4	-1.24 ± 4.51
Haiku	0.73 ± 4.90
Sonnet	2.86 ± 5.50
Opus	4.82 ± 5.86
POSR GPT4	1.09 ± 4.47
POSR Haiku	1.02 ± 4.02
POSR Sonnet	3.67 ± 3.8
POSR Opus	2.64 ± 3.64

Table 4: **Difference in number of segments.**

Category	Bigram (log odds)
Providing Examples e.g., Let's say we have the function X squared plus 5 x plus 6.	"lets_say" (2.26), "yeah_say" (1.51)
Alternative explanations e.g., There are different ways to solve this as well.	"differ_way" (1.50), "simpler_way" (1.23)
Prompting participation e.g., So, [STUDENT], how did you like start to approach this problem?	like_start (1.51), try_find (1.48), guy_know (1.48)

Table 5: **Bigram categories founded in falsely inserted boundaries by `POSR Opus`.** Incorrect segments are inserted when the tutor provides examples ("let's_say"), alternative explanations ("diff_way"), or prompts for participation ("like_start").

ally, we find that POSR methods perform better than their independent LLM segmentation counterparts. For example, `POSR Sonnet` improves upon `Sonnet` across all segmentation metrics, such as $0.24 \rightarrow 0.13$ on Line-P_k or $0.37 \rightarrow 0.31$ on Line-WindowDiff. These results reiterate the importance of treating segmentation and retrieval *jointly*.

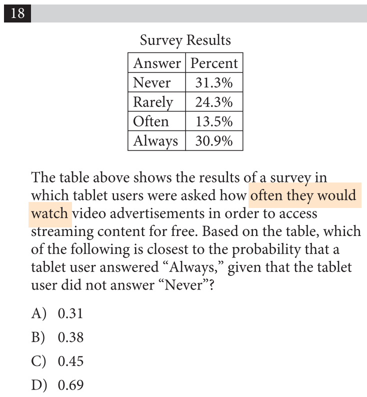
The time- and line-based metrics for segmentation and SRS are well-correlated across methods, indicating that accounting for time does not impact relative rankings. However, time-weighing is still important in accounting for errors in long segments: Time-P_k errors are lower than Line-P_k because it reduces the impact of oversegmentation whereas Time-WindowDiff amplifies errors from missing long segments.

Segmentation error analysis. To better understand sources of segmentation error, we investigate the difference in segment numbers (reported in Table 4) and we examine the bigram language in false segment insertions compared to true segment insertions with the log odds ratio, latent Dirichlet prior, measure defined in Monroe et al. (2008). Table 4 reveals that traditional methods oversegment, being sensitive to low-level topics shifts. Surprisingly, while `Haiku` has a higher segmentation error rate in Table 2, it achieves the lowest segment count difference, altogether indicating that Haiku inserts new (albeit few) segments far away from true segment boundaries. The log odds results in Table 5 indicate that incorrect segments are inserted when the tutor introduces examples (e.g., "let's say"), alternative explanations (e.g., "There are different ways to solve this"), or participation prompts (e.g., "how did you like start to approach this problem?"). This analysis signals areas for improvement in precise segmentation.

Retrieval error analysis. We conduct a qualitative analysis on retrieval errors, particularly those in the independent methods. A large error source is caused by long segments that are incorrectly segmented for reasons illustrated in the previous section. For example, long problem segments are broken up and incorrectly linked. Oversegmentation also yields shorter segment queries for retrieval, reducing the similarity to the target reference. This particularly impacts traditional methods whose similarity thresholds are set with the ground truth segments as explained in Appendix C. In Appendix E, we compare retrieval methods on *ground-truth segments* and confirm that ground truth segments significantly boosts retrieval accuracy, especially for LLM methods. Thus, we conclude that inaccurate segmentation is a critical bottleneck to mitigating downstream retrieval errors.

7 Downstream Applications

There are several applications that POSR enables for gaining insights into tutoring practices at scale. We illustrate two. One application is a language analysis to compare how tutors talk about the same problem with the long vs. short talk times (top and bottom quartile). We use the log odds ratio measure from Monroe et al. (2008) to estimate the distinctiveness of a bigram using Edu-ConvoKit (Wang and Demszky, 2024). We report the top-3 bigrams on the most popular problem from LessonLink and qualitative examples in Figure 2. The log-odds analysis reveals that in short segments, tutors tend to stick to the language from the "problem statement" and immediately explain the answer. However, in longer segments, tutors provide



Answer	Percent
Never	31.3%
Rarely	24.3%
Often	13.5%
Always	30.9%

Long segments	let_see (0.683), let_say (0.683), conditional_probability (0.602)
Example	Tutor: And then someone wants to take a look at Question 18 [...] you might deal with something called conditional probability. Right? So conditional probability means what is the probability of something occurring when something else doesn't occur. So let's say that you have 2 events A and B. The probability that a occurs assuming that B occurs which we denote like this probability of A assuming B [...] so let's say that we have some event a. and we have some event. B. So a. And then we [..]
Short segments	always_divided (2.025), often_would (1.658), would_watch (1.658)
Example	Tutor: So now 18. [..reading aloud the problem..] So let's just take 31.3. Take that off of a 100, so 68, point 7. That's going to be 30. Point 9, over 68.7, which i'm guessing is around point 4, 5, just to guess. based off of the answer choices. Yep. The answer is, See that's pretty much all there is to that problem. You just have to get rid of this.

Figure 2: **Qualitative examples & log odds.** We report the top-3 bigrams in segments talking about the left problem. We compare long segments (top quartile duration) and short segments (bottom quartile duration). Longer segments tend to provide conceptual explanations ("let's say", conditional probability). Shorter segments tend to stick more to the problem at hand.

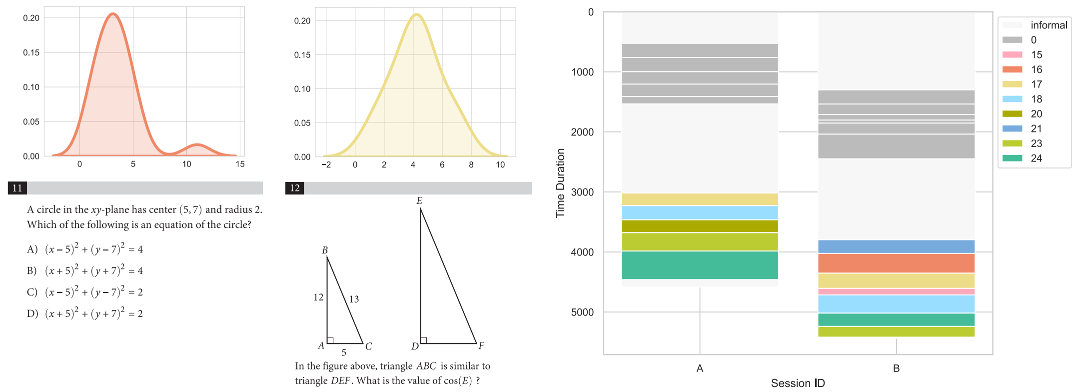


Figure 3: **Left:** Time spent (minutes) per worksheet problem. **Right:** Example of time management across two lessons.

examples to students (e.g., "let's say"), and offer conceptual explanations inferring the underlying mathematical concept (e.g., "this is a conditional probability question"). The second POSR application is the analysis of talk time distributions across different tutors and problems, such as in Figure 3: some problems have very different talk times (e.g., problem 11), while others have similar talk times (e.g., problem 12). Altogether, POSR enables these downstream applications and can tackle the large challenge of lesson structuring in education.

8 Discussion and Conclusion

We introduce the Problem-Oriented Segmentation and Retrieval (POSR), a task that jointly segments conversations and retrieves the problem discussed in each segment. We contribute the LessonLink dataset as a concrete case study of POSR in education. LessonLink is the first large-scale dataset of tutoring conversations linked with worksheets, featuring 3,500 segments, 116 linked SAT math problems and over 24,300 minutes of instruction. To evaluate the joint performance and account for time in segmentation, we introduce the Segmentation and Retrieval Score (SRS) and time-based segmentation metrics for P_k and WindowDiff. Our comprehensive evaluations highlight the importance of jointly modeling segmentation and retrieval, rather than treating them as independent tasks: POSR methods significantly outperform the independent approaches as measured against the traditional segmentation, SRS, and new time-based metrics. The LLM-based POSR methods achieve the best performance, but come at a higher cost, motivating future work on cost-effective solutions. We also demonstrate the potential of POSR by showcasing downstream applications, such as a language analysis comparing tutoring strategies. In conclusion, our work establishes POSR as an important task to study conversation structure. The LessonLink dataset and the proposed methods pave the way for further research in joint segmentation and retrieval, with broad implications for educational technology, conversational analysis, and beyond.

9 Limitations

While our work provides a useful starting point for understanding conversations (such as in education) at scale, there are limitations to our work. Addressing these limitations will be an important area for future research.

One limitation is the lack of connection to outcomes. While prior works have explored the relationship between duration and sequencing of problems on student attention (e.g., Stevens and Bavelier (2012) *inter alia*), there is limited research on how these factors impact long-term student learning, particularly in group-based settings. Understanding this connection is crucial for grounding POSR in real contexts.

Additionally, POSR does not rigorously link the language content with the segment duration or ordering. This applies to other conversation domains as well, beyond education settings. Linking content and quality of the language with the time allocation and sequencing matters (Suresh et al., 2018): Are tutors soliciting student contributions, or talking all the time? Are they restating or engaging with student contributions? While our downstream applications illustrate one form of language analysis with a log odds analysis, future work should investigate using language categories, instead of unsupervised methods for understanding language patterns.

Another limitation is the absence of audio and visual inputs. Our current models rely solely on textual data and miss non-verbal cues that add to the full context in understanding conversations. We also only use the problem text, and ignore the problem's visual components such as graph information. Incorporating multimodal data, such as audio and visual inputs, could improve the accuracy of POSR systems.

10 Ethical Considerations

The purpose of this work is to promote and improve effective interactions, such as in the setting of education, using NLP techniques. The LessonLink dataset is intended for research purposes. The dataset should not be used for commercial purposes, and we ask that users of our dataset respect this restriction. As stewards of this data, we are committed to protecting the privacy and confidentiality of the individuals who contributed comments to the dataset. It is important to note that inferences drawn from the dataset should be interpreted with caution. The intended use case for this dataset is to further research on conversation interactions and education, towards the goal of improving interactions. Unacceptable use cases include any attempts to identify users or use the data for commercial gain. We additionally recommend that researchers who do use our dataset take steps to mitigate any risks or harms to individuals that may arise.

References

Akiko Aizawa and Michael Kohlhase. 2021. Mathematical information retrieval. *Evaluating Information Retrieval and Access Tasks: NTCIR's Legacy of Research Impact*, pages 169–185.

Tim Althoff, Kevin Clark, and Jure Leskovec. 2016. Large-scale analysis of counseling conversations: An application of natural language processing to mental health. *Transactions of the Association for Computational Linguistics*, 4:463–476.

Anthropic. 2024. Introducing the next generation of Claude. https://www.anthropic.com/news/claude-3-family. [Online; accessed 27-May-2024].

Doug Beeferman, Adam Berger, and John Lafferty. 1997. Text segmentation using exponential models. *arXiv preprint cmp-lg/9706016*.

Doug Beeferman, Adam Berger, and John Lafferty. 1999. Statistical models for text segmentation. *Machine learning*, 34:177–210.

Steven Bird, Ewan Klein, and Edward Loper. 2009. *Natural language processing with Python: analyzing text with the natural language toolkit*. " O'Reilly Media, Inc.".

James P Callan. 1994. Passage-level evidence in document retrieval. In *SIGIR'94: Proceedings of the Seventeenth Annual International ACM-SIGIR Conference on Research and Development in Information Retrieval, organised by Dublin City University*, pages 302–310. Springer.

Dan Carpenter, Andrew Emerson, Bradford W Mott, Asmalina Saleh, Krista D Glazewski, Cindy E Hmelo-Silver, and James C Lester. 2020. Detecting off-task behavior from student dialogue in game-based collaborative learning. In *Artificial Intelligence in Education: 21st International Conference, AIED 2020, Ifrane, Morocco, July 6–10, 2020, Proceedings, Part I 21*, pages 55–66. Springer.

Ciprian Chelba, Timothy J Hazen, and Murat Saraclar. 2008. Retrieval and browsing of spoken content. *IEEE Signal Processing Magazine*, 25(3):39–49.

Jiaao Chen and Diyi Yang. 2020. Multi-view sequence-to-sequence models with conversational structure for abstractive dialogue summarization. In *Proceedings of* the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 4106–4118.

Freddy YY Choi. 2000. Advances in domain independent linear text segmentation. *arXiv preprint cs/0003083*.

William G Cochran. 1950. The comparison of percentages in matched samples. *Biometrika*, 37(3/4):256–266.

Laura M Desimone. 2009. Improving impact studies of teachers' professional development: Toward better conceptualizations and measures. *Educational researcher*, 38(3):181–199.

Marjolein I Deunk, Annemieke E Smale-Jacobse, Hester de Boer, Simone Doolaard, and Roel J Bosker. 2018. Effective differentiation practices: A systematic review and meta-analysis of studies on the cognitive effects of differentiation practices in primary education. *Educational Research Review*, 24:31–54.

Chika Eze and Edward Misava. 2017. Lecture duration: A risk factor for quality teaching and learning in higher education. *Integrity Journal of Education and Training*, 1:1.

Barry J Fishman, Ronald W Marx, Stephen Best, and Revital T Tal. 2003. Linking teacher and student learning to improve professional development in systemic reform. *Teaching and teacher education*, 19(6):643–658.

Michael Fullan and Alan Pomfret. 1977. Research on curriculum and instruction implementation. *Review of educational research*, 47(2):335–397.

Michel Galley, Kathleen McKeown, Eric Fosler-Lussier, and Hongyan Jing. 2003. Discourse segmentation of multi-party conversation. In *Proceedings of the 41st Annual Meeting of the Association for Computational Linguistics*, pages 562–569.

Swapna Gottipati and Venky Shankararaman. 2018. Competency analytics tool: Analyzing curriculum using course competencies. *Education and Information Technologies*, 23:41–60.

Barbara Grosz and Julia Hirschberg. 1992. Some intonational characteristics of discourse structure. In *Second international conference on spoken language processing*.

Anthony Haynes. 2010. *The complete guide to lesson planning and preparation*. Bloomsbury Publishing.

Marti A Hearst. 1997. Text tiling: Segmenting text into multi-paragraph subtopic passages. *Computational linguistics*, 23(1):33–64.

Marti A Hearst and Christian Plaunt. 1993. Subtopic structuring for full-length document access. In *Proceedings of the 16th annual international ACM SIGIR conference on Research and development in information retrieval*, pages 59–68.

Sabine Heim and Andreas Keil. 2012. Developmental trajectories of regulating attentional selection over time. *Frontiers in Psychology*, 3:30493.

Maryumah Hejji Alanazi. 2019. A study of the preservice trainee teachers problems in designing lesson plans. *Arab World English Journal (AWEJ) Volume*, 10.

James Henderson. 1997. Transformative curriculum leadership. *Teaching Education*, 9(1):39–40.

Isabel Hilliger, Camila Aguirre, Constanza Miranda, Sergio Celis, and Mar Pérez-Sanagustín. 2020. Design of a curriculum analytics tool to support continuous improvement processes in higher education. In *Proceedings of the tenth international conference on learning analytics & knowledge*, pages 181–186.

Isabel Hilliger, Camila Aguirre, Constanza Miranda, Sergio Celis, and Mar Pérez-Sanagustín. 2022. Lessons learned from designing a curriculum analytics tool for improving student learning and program quality. *Journal of computing in higher education*, 34(3):633–657.

Julia Hirschberg and Christine H Nakatani. 1998. Acoustic indicators of topic segmentation. In *Fifth International Conference on Spoken Language Processing*.

Xiangji Huang, Fuchun Peng, Dale Schuurmans, Nick Cercone, and Stephen E Robertson. 2003. Applying machine learning to text segmentation for information retrieval. *Information Retrieval*, 6:333–362.

Paul Jaccard. 1912. The distribution of the flora in the alpine zone. 1. *New phytologist*, 11(2):37–50.

Albert Q Jiang, Alexandre Sablayrolles, Antoine Roux, Arthur Mensch, Blanche Savary, Chris Bamford, Devendra Singh Chaplot, Diego de las Casas, Emma Bou Hanna, Florian Bressand, et al. 2024. Mixtral of experts. *arXiv preprint arXiv:2401.04088*.

Shahab Kamali and Frank Wm Tompa. 2013. Retrieving documents with mathematical content. In *Proceedings of the 36th international ACM SIGIR conference on Research and development in information retrieval*, pages 353–362.

Omar Khattab and Matei Zaharia. 2020. Colbert: Efficient and effective passage search via contextualized late interaction over bert. In *Proceedings of the 43rd International ACM SIGIR conference on research and development in Information Retrieval*, pages 39–48.

Hideki Kozima. 1996. Text segmentation based on similarity between words. *arXiv preprint cmp-lg/9601005*.

Matthew A Kraft, David Blazar, and Dylan Hogan. 2018. The effect of teacher coaching on instruction and achievement: A meta-analysis of the causal evidence. *Review of educational research*, 88(4):547–588.

Zhengyuan Liu, Siti Umairah Md Salleh, Hong Choon Oh, Pavitra Krishnaswamy, and Nancy Chen. 2023. Joint dialogue topic segmentation and categorization: A case study on clinical spoken conversations. In *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing: Industry Track*, pages 185–193, Singapore. Association for Computational Linguistics.

Catalina Lomos, Roelande H Hofman, and Roel J Bosker. 2011. Professional communities and student achievement–a meta-analysis. *School effectiveness and school improvement*, 22(2):121–148.

Sander Martens and Brad Wyble. 2010. The attentional blink: Past, present, and future of a blind spot in perceptual awareness. *Neuroscience & Biobehavioral Reviews*, 34(6):947–957.

Meta. 2024. Introducing Meta Llama 3: The most capable openly available LLM to date. https://ai.meta.com/blog/meta-llama-3/. [Online; accessed 27-May-2024].

Burt L Monroe, Michael P Colaresi, and Kevin M Quinn. 2008. Fightin'words: Lexical feature selection and evaluation for identifying the content of political conflict. *Political Analysis*, 16(4):372–403.

Jane Morris and Graeme Hirst. 1991. Lexical cohesion computed by thesaural relations as an indicator of the structure of text. *Computational linguistics*, 17(1):21–48.

Rajesh Munavalli and Robert Miner. 2006. Mathfind: a math-aware search engine. In *Proceedings of the 29th annual international ACM SIGIR conference on Research and development in information retrieval*, pages 735–735.

Christine Nakatani, Julia Hirschberg, and Barbara Grosz. 1995. Discourse structure in spoken language: Studies on speech corpora. In *AAAI Spring Symposium on Empirical Methods in Discourse Interpretation and Generation (1995)*. Assocation for the Advancement of Artifical Intelligence.

Tam T Nguyen, Kuiyu Chang, and Siu Cheung Hui. 2012. A math-aware search engine for math question answering system. In *Proceedings of the 21st ACM international conference on Information and knowledge management*, pages 724–733.

Douglas W Oard, Dagobert Soergel, David Doermann, Xiaoli Huang, G Craig Murray, Jianqiang Wang, Bhuvana Ramabhadran, Martin Franz, Samuel Gustman, James Mayfield, et al. 2004. Building an information retrieval test collection for spontaneous conversational speech. In *Proceedings of the 27th annual international ACM SIGIR conference on Research and development in information retrieval*, pages 41–48.

OpenAI. 2024. GPT-4. https://openai.com/index/gpt-4-research/. [Online; accessed 27-May-2024].

Carol L O'Donnell. 2008. Defining, conceptualizing, and measuring fidelity of implementation and its relationship to outcomes in k–12 curriculum intervention research. *Review of educational research*, 78(1):33–84.

Regina M Panasuk and Jeffrey Todd. 2005. Effectiveness of lesson planning: Factor analysis. *Journal of Instructional Psychology*, 32(3):215.

Rebecca J Passonneau and Diane Litman. 1997. Discourse segmentation by human and automated means. *Computational Linguistics*, 23(1):103–139.

Lev Pevzner and Marti A. Hearst. 2002. A critique and improvement of an evaluation metric for text segmentation. *Computational Linguistics*, 28(1):19–36.

Marcela Pozas, Verena Letzel, and Christoph Schneider. 2020. Teachers and differentiated instruction: exploring differentiation practices to address student diversity. *Journal of Research in Special Educational Needs*, 20(3):217–230.

Vinodkumar Prabhakaran, Camilla Griffiths, Hang Su, Prateek Verma, Nelson Morgan, Jennifer L Eberhardt, and Dan Jurafsky. 2018. Detecting institutional dialog acts in police traffic stops. *Transactions of the Association for Computational Linguistics*, 6:467–481.

PyTesseract. 2017. Python Tesseract. https://github.com/madmaze/pytesseract. [Online; accessed 27-May-2024].

Nils Reimers and Iryna Gurevych. 2019. Sentence-bert: Sentence embeddings using siamese bert-networks. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing*. Association for Computational Linguistics.

Stephan Repp, Jörg Waitelonis, Harald Sack, and Christoph Meinel. 2007. Segmentation and annotation of audiovisual recordings based on automated speech recognition. In *Intelligent Data Engineering and Automated Learning-IDEAL 2007: 8th International Conference, Birmingham, UK, December 16-19, 2007. Proceedings 8*, pages 620–629. Springer.

Jeffrey C Reynar. 1999. Statistical models for topic segmentation. In *proceedings of the 37th Annual Meeting of the Association for Computational Linguistics*, pages 357–364.

Martin Riedl and Chris Biemann. 2012. Topictiling: a text segmentation algorithm based on lda. In *Proceedings of ACL 2012 student research workshop*, pages 37–42.

Alan Ritter, Colin Cherry, and Bill Dolan. 2010. Unsupervised modeling of twitter conversations.

Stephen Robertson, Hugo Zaragoza, et al. 2009. The probabilistic relevance framework: Bm25 and beyond. *Foundations and Trends® in Information Retrieval*, 3(4):333–389.

Ma Mercedes T Rodrigo, Ryan SJD Baker, and Lisa Rossi. 2013. Student off-task behavior in computer-based learning in the philippines: comparison to prior research in the usa. *Teachers College Record*, 115(10):1–27.

Gerard Salton and Chris Buckley. 1991a. Automatic text structuring and retrieval-experiments in automatic encyclopedia searching. In *Proceedings of the 14th annual international ACM SIGIR conference on Research and development in information retrieval*, pages 21–30.

Gerard Salton and Chris Buckley. 1991b. Global text matching for information retrieval. *Science*, 253(5023):1012–1015.

Gerard Salton, Amit Singhal, Chris Buckley, and Mandar Mitra. 1996. Automatic text decomposition using text segments and text themes. In *Proceedings of the the seventh ACM conference on Hypertext*, pages 53–65.

Claude Sammut and Geoffrey I Webb. 2011. *Encyclopedia of machine learning*. Springer Science & Business Media.

Mark Sanderson et al. 2010. Test collection based evaluation of information retrieval systems. *Foundations and Trends® in Information Retrieval*, 4(4):247–375.

Keshav Santhanam, Omar Khattab, Jon Saad-Falcon, Christopher Potts, and Matei Zaharia. 2021. Colbertv2: Effective and efficient retrieval via lightweight late interaction. *arXiv preprint arXiv:2112.01488*.

Moritz Schubotz, Alexey Grigorev, Marcus Leich, Howard S Cohl, Norman Meuschke, Bela Gipp, Abdou S Youssef, and Volker Markl. 2016. Semantification of identifiers in mathematics for better math information retrieval. In *Proceedings of the 39th International ACM SIGIR conference on Research and Development in Information Retrieval*, pages 135–144.

Petr Sojka and Martin Líška. 2011. The art of mathematics retrieval. In *Proceedings of the 11th ACM symposium on Document engineering*, pages 57–60.

Courtney Stevens and Daphne Bavelier. 2012. The role of selective attention on academic foundations: A cognitive neuroscience perspective. *Developmental cognitive neuroscience*, 2:S30–S48.

Bob Stradling and Lesley Saunders. 1993. Differentiation in practice: Responding to the needs of all pupils. *Educational Research*, 35(2):127–137.

Abhijit Suresh, Tamara Sumner, Isabella Huang, Jennifer Jacobs, Bill Foland, and Wayne Ward. 2018. Using deep learning to automatically detect talk moves in teachers' mathematics lessons. In *2018 IEEE International Conference on Big Data (Big Data)*, pages 5445–5447. IEEE.

Zeynep Baskan Takaoglu. 2017. Challenges faced by pre-service science teachers during the teaching and learning process in turkey. *Journal of Education and Training Studies*, 5(2):100–110.

Hugo Touvron, Louis Martin, Kevin Stone, Peter Albert, Amjad Almahairi, Yasmine Babaei, Nikolay Bashlykov, Soumya Batra, Prajjwal Bhargava, Shruti Bhosale, et al. 2023. Llama 2: Open foundation and fine-tuned chat models. *arXiv preprint arXiv:2307.09288*.

Rose Wang, Pawan Wirawarn, Omar Khattab, Noah Goodman, and Dorottya Demszky. 2024. Backtracing: Retrieving the cause of the query. In *Findings of the Association for Computational Linguistics: EACL 2024*, pages 722–735, St. Julian's, Malta. Association for Computational Linguistics.

Rose E Wang and Dorottya Demszky. 2024. Educonvokit: An open-source library for education conversation data.

Ross Wilkinson. 1994. Effective retrieval of structured documents. In *SIGIR'94: Proceedings of the Seventeenth Annual International ACM-SIGIR Conference on Research and Development in Information Retrieval, organised by Dublin City University*, pages 311–317. Springer.

Richard Zanibbi and Dorothea Blostein. 2012. Recognition and retrieval of mathematical expressions. *International Journal on Document Analysis and Recognition (IJDAR)*, 15:331–357.

A P_k and Time-P_k

The P_k metric is an established segmentation metric from Beeferman et al. (1999). Similar to WindowDiff, it uses a line-based sliding window approach that measures boundary mismatches within the window. Lower values is better. For example, P_k is computed as:

$$P_k(Y, Y^*) = \frac{1}{N-k} \sum_{j=1}^{N-k} \mathbb{1}\left(\mathbb{1}(b(s_{j:j+k}) > 0) \neq \mathbb{1}(b(s^*_{j:j+k}) > 0)\right)$$

where $b(\cdot)$ represents the number of boundaries within the $\cdot$ window and k is typically set to half of the average of the true segment line size.

Time-P_k is calculated as:

$$\text{Time-}P_k(Y, Y^*) = \frac{1}{N-k} \sum_{j=1}^{N-k} \mathbb{1}\left(\mathbb{1}(b(s_{t_j^{\text{start}}:t_j^{\text{end}}+\Delta_k}) > 0) \neq \mathbb{1}(b(s^*_{t_j^{\text{start}}:t_j^{\text{end}}+\Delta_k}) >\right.$$

where Δ_k, the time duration of the sliding window, is half of the average true segment duration (similar to k).

B Prompts

Recognizing that models are sensitive to prompt phrasing, we ran experiments on 15 transcripts to determine the best prompting approach for each task: independent segmentation, independent retrieval, and joint segmentation and retrieval. For each task, two authors collaboratively wrote a pool of prompt templates with varying phrasings. From these, we chose the top-performing template across all models to use for all transcripts.

Independent Segmentation Prompt

```
### System:
You are an assistant who will be given a transcript of an SAT math tutoring session
between a tutor and a group of students.  Each line in the transcript will contain the
line index, the speaker (tutor or student), and the utterance.  Your job is to read the
transcript and identify segments that each involve the discussion of an individual math
problem.  Note that each segment must involve the discussion of one math problem only.

Please then output the first line index and last line index of each segment as a list
of lists:
[[<first line index of segment 1>, <last line index of segment 1>], ...,
[<first line index of segment n>, <last line index of segment n>]].

Only output a list of lists.  Do not output any additional text or explanations.

### User:
Please read the transcript below and identify segments that each involve the discussion
of an individual math problem:
{transcript}

Please output the first line index and last line index of each segment as a list of lists:
[[<first line index of segment 1>, <last line index of segment 1>], ...,
[<first line index of segment n>, <last line index of segment n>]].

Only output a list of lists.  Do not output any additional text or explanations.
```

Figure 4: **Prompt for the independent segmentation task for LLM methods.** `{transcript}` is the placeholder for the entire tutoring transcript whose lines have the following format: `{idx} {speaker}: {utterance}`.

B.1 Independent segmentation

For the independent segmentation task, we designed three distinct prompt templates:

1. A template prompting the LLM to identify segments that each involve the discussion of an individual math problem, with an extra note emphasizing that each segment must involve the discussion of one math problem only;

2. A template prompting the LLM to segment the transcript into contiguous segments, where each segment either involves (a) the discussion of a single math problem or (b) anything else (such as small talks, the introduction of the tutoring session, and the conclusion of the tutoring session, which, if contiguous, must be part of the same segment);

3. A template prompting the LLM to detect lines where the tutor/students start transitioning to discussing a new math problem, as well as the line right after the tutor/students finish discussing the math problem, to mark the beginning of each segment

We found that the first prompt template, shown in Figure 4, performs best in terms of all segmentation metrics, i.e., WindowDiff and P_k scores.

B.2 Independent retrieval

For the independent retrieval task, we designed two distinct prompt templates:

1. A prompt template that retrieves for all segments in a transcript at once;

2. A prompt template that retrieves for one segment at a time, independently for each segment.

We found that both prompt templates perform comparably when given ground truth segments. However, when given imperfect, predicted segments, prompt template 2 performs significantly better in terms of SRS scores. We therefore choose to use prompt template 2, shown in Figure 5, for all transcripts.

B.3 Joint segmentation and retrieval

For the joint segmentation and retrieval task, we designed two distinct prompt templates:

1. Similar to template 1 for the independent segmentation task, this template prompts the LLM to identify segments that each involve

Independent Retrieval Prompt

```
### System:
You are an assistant who will be given (1) a segment of an SAT math tutoring session
between a tutor and a group of students and (2) the set of math problems that might be
discussed in the segment.  Your job is to read the segment's transcript and set of math
problems, then determine the math problem that was discussed in the segment, if any.  If
no math problem was discussed in the segment, please output "null".  If a math problem
was discussed in the segment but not found in the provided set of problems, please output
-1.  If a math problem was discussed in the segment and is found in the provided set of
problems, please output the ID of the problem.  Please do not output any additional text
or explanations.


### User:
Please read the segment's transcript, read the set of math problems that might be
discussed in the segment, and determine the math problem that was discussed in the
segment, if any.

Segment:
{transcript}

Math problems:
{problems}

If no math problem was discussed in the segment, please output "null".  If a math problem
was discussed in the segment but not found in the provided set of problems, please output
-1.  If a math problem was discussed in the segment and is found in the provided set of
problems, please output the ID of the problem.  Please do not output any additional text
or explanations.
```

Figure 5: **Prompt for the independent retrieval task for LLM methods.** `{transcript}` is the placeholder for a tutoring segment's transcript whose lines have the following format: `{speaker}: utterance`. `{problems}` is the placeholder for the worksheet problems relevant to the session that have the following format: `Problem ID {id}: problem string`.

the discussion of an individual math problem, then determine which math problem was discussed in each segment or indicate if a math problem was discussed but not found in the provided set of problems.

2. Similar to template 2 for the independent segmentation task, this template prompts the LLM to segment the transcript into contiguous segments, where each segment either involves (a) the discussion of a single math problem or (b) anything else (such as small talks, the introduction of the tutoring session, and the conclusion of the tutoring session, which, if contiguous, must be part of the same segment). It then requires determining if a math problem was discussed in each segment, and, if so, identifying the specific math problem or indicating if it can not be found in the provided set of problems.

We found that the first prompt template, shown in Figure 6, performs best in terms of all relevant metrics, i.e., WindowDiff, P_k scores, and SRS scores.

C Thresholds

A challenge in using traditional IR methods in our setting is specifying that nothing in the worksheet is linked to a segment, e.g., for informal or warm-up segments. For traditional IR methods, we must set a threshold to determine which scores indicate that a worksheet problem is relevant enough to a segment. We perform 5-fold cross-validation on the training set, testing threshold values from 0 to 1 in 0.01 intervals on ground truth segments, to determine the threshold that yields the highest retrieval accuracy on the held-out fold. We then average the best thresholds from each fold to obtain the final threshold for each method.

Note that for `BM-25` and `ColBERT`, which have unbounded relevance scores, we normalized the raw scores within the top 10 results for each query (as each worksheet has at least 10 problems to retrieve from). This normalization adjusts the scores relative to the top results, making them comparable across different queries and allowing us to set a threshold that would apply consistently across queries. Without this normalization, the scores would only be meaningful within the context of a single query and not comparable across different queries.

The threshold values for each traditional IR method are as follows:

- `Jaccard`: 0.11
- `tfidf`: 0.40
- `BM-25`: 0.19
- `ColBERT`: 0.14

D Extended Results

Table 6 shows the extended segmentation results where we used three pre-trained encoders from Sentence-Transformers (Reimers and Gurevych, 2019): the base-nli-stsb-mean-tokens (originally used in Chen and Yang (2020)), all-mpnet-base-v2, all-MiniLM-L12-v2. As the Table shows, the encoders did not vary much in segmentation performance.

E Extended Error Analysis

To assess why independently performing retrieval on top of segmentation does not perform as well as the joint POSR methods (rf. Table 2), we need to isolate and analyze the retrieval errors. Therefore, we additionally evaluate the retrieval performance conditioned on the ground truth segments in Table 7. We find that the LLM-based solutions typically perform better than traditional IR methods, and for `GPT-4` and `Claude-Opus` near ceiling. Interestingly, we find that `Haiku` performs similarly on retrieval as simpler methods such as using `Jaccard` similarity of `tfidf`. In our qualitative analysis, we find `Haiku`'s errors are due to retrieving incorrect worksheet problems on warm-up segments. This is also the most common error type of other LLM-based retrievers.

Method	P_k (↓) Sentence	P_k (↓) Time	WindowDiff (↓) Sentence	WindowDiff (↓) Time
Top-10	0.58 ± 0.04	0.28 ± 0.16	1.0 ± 0.01	1.0 ± 0.0
Top-20	0.58 ± 0.04	0.28 ± 0.16	1.0 ± 0.0	1.0 ± 0.0
TextTiling	0.58 ± 0.05	0.27 ± 0.16	0.90 ± 0.11	0.94 ± 0.06
Topic, mpnet	0.58 ± 0.04	0.27 ± 0.16	1.0 ± 0.02	0.99 ± 0.01
Topic, minilm	0.58 ± 0.04	0.27 ± 0.16	1.0 ± 0.02	1.0 ± 0.01
Topic, base	0.58 ± 0.04	0.27 ± 0.16	1.0 ± 0.02	1.0 ± 0.01
Stage, mpnet, avg	0.58 ± 0.05	0.28 ± 0.16	0.99 ± 0.03	1.0 ± 0.01
Stage, minilm, avg	0.58 ± 0.04	0.28 ± 0.16	1.0 ± 0.02	1.0 ± 0.01
Stage, base, avg	0.58 ± 0.04	0.28 ± 0.16	1.0 ± 0.0	1.0 ± 0.0
Stage, minilm, max	0.58 ± 0.04	0.28 ± 0.16	1.0 ± 0.00	1.0 ± 0.00
Stage, mpnet, max	0.58 ± 0.04	0.28 ± 0.16	1.0 ± 0.01	1.0 ± 0.00
Stage, base, max	0.58 ± 0.04	0.28 ± 0.16	1.0 ± 0.0	1.0 ± 0.0
GPT4	0.20 ± 0.10	0.25 ± 0.17	0.33 ± 0.09	0.52 ± 0.15
Haiku	0.29 ± 0.14	0.30 ± 0.17	0.39 ± 0.14	0.55 ± 0.16
Sonnet	0.24 ± 0.14	0.23 ± 0.18	0.37 ± 0.15	0.53 ± 0.17
Opus	0.15 ± 0.09	$\mathbf{0.11 \pm 0.10}$	0.31 ± 0.13	0.46 ± 0.17
POSR GPT4	0.16 ± 0.01	0.18 ± 0.17	0.32 ± 0.09	0.53 ± 0.17
POSR Haiku	0.24 ± 0.10	0.22 ± 0.13	0.35 ± 0.11	0.51 ± 0.17
POSR Sonnet	$\mathbf{0.13 \pm 0.08}$	$\mathbf{0.11 \pm 0.12}$	0.31 ± 0.09	0.49 ± 0.17
POSR Opus	$\mathbf{0.13 \pm 0.08}$	0.12 ± 0.13	$\mathbf{0.28 \pm 0.10}$	$\mathbf{0.44 \pm 0.17}$

Table 6: **Extended segmentation evaluations (↓ better).**

Segmentation and Retrieval Prompt

```
### System:
You are an assistant who will be given (1) a transcript of an SAT math tutoring session
between a tutor and a group of students and (2) the set of math problems that might be
discussed in the session. Each line in the transcript contains the line index, the
speaker (tutor or student), and the utterance. Each math problem corresponds to a
problem ID.

Your first job is to read the transcript and identify segments that each involve the
discussion of an individual math problem. Note that each segment must involve the
discussion of one math problem only. Your second job is to determine the math problem
that was discussed in each of the segments you identified. Please then output the
first line index and last line index of each segment, along with the ID of the problem
discussed in each segment as a list of JSON objects:
[{"start_line_idx": <first line index of segment 1>, "end_line_idx": <last line index of
segment 1>, "problem_id": <ID of problem discussed in segment 1>}, ..., {"start_line_idx":
<first line index of segment n>, "end_line_idx": <last line index of segment n>, "problem_id":
<ID of problem discussed in segment n>}].

If a math problem was discussed in a segment but not found in the provided set of
problems, let the problem_id be -1. Only output the list of JSON objects. Do not output
any additional text or explanations.

### User:
Please read the transcript, identify segments that each involve the discussion of an
individual math problem, and determine the math problem that was discussed in each of the
segments you identified.

Transcript:
{transcript}

Math problems:
{problems}

Please output the first line index and last line index of each segment, along with the
ID of the problem discussed in each segment as a list of JSON objects:
[{"start_line_idx": <first line index of segment 1>, "end_line_idx": <last line index of
segment 1>, "problem_id": <ID of problem discussed in segment 1>}, ..., {"start_line_idx":
<first line index of segment n>, "end_line_idx": <last line index of segment n>, "problem_id":
<ID of problem discussed in segment n>}].

If a math problem was discussed in a segment but not found in the provided set of
problems, let the problem_id be -1. Only output the list of JSON objects. Do not output
any additional text or explanations.
```

Figure 6: **Prompt for the joint segmentation and retrieval task for LLM methods.** `{transcript}` is the placeholder for the entire tutoring transcript whose lines have the following format: `{idx} {speaker}: {utterance}`. `{problems}` is the placeholder for the worksheet problems relevant to the session that have the following format: `Problem ID {id}: problem string`.

Method	**Accuracy** $\uparrow$
`Jaccard`	0.644 ± 0.196
`tfidf`	0.675 ± 0.205
`BM-25`	0.511 ± 0.216
`ColBERT`	0.577 ± 0.214
`GPT-4`	$\mathbf{0.965 \pm 0.066}$
`Claude Haiku`	0.688 ± 0.255
`Claude Sonnet`	0.863 ± 0.164
`Claude Opus`	0.947 ± 0.091

Table 7: **Independent retrieval evaluations on the ground truth segments.**