

ZERO-SHOT CUE-GROUNDED TOPIC SEGMENTATION OF SPOKEN DOCUMENTS

Suhwan Choi¹, Myeongho Jeon², Myungjoo Kang¹

¹Seoul National University, ²KAIST

ABSTRACT

Topic segmentation structures spoken documents into coherent sections, facilitating navigation and downstream understanding. The appropriate granularity can vary substantially, ranging from broad thematic shifts to fine-grained subtopics. Existing LLM-based segmenters, however, often struggle to adapt to this variation, causing them to either merge distinct subtopics or over-segment coherent themes. To address this, we introduce **Cue-Grounded Segmentation (CGS)**, a training-free framework that operates without any task-specific supervision. CGS first identifies phrases that explicitly signal the start of a new topic and uses their sentence positions as segment boundaries. When such cues are insufficient, it falls back to semantic segmentation, guided by the document structure inferred during cue extraction. Across six benchmarks and six LLM backbones, CGS consistently outperforms existing baselines, remains robust to noisy ASR transcripts, and achieves these gains with low API cost on proprietary models.

Index Terms— topic segmentation, spoken language processing, large language models, zero-shot learning, adaptive inference

1. INTRODUCTION

Topic segmentation organizes spoken documents into coherent sections by identifying where new topics begin. These topic-level units are essential for efficiently navigating, retrieving, and summarizing long-form recordings. Unlike written documents, however, spoken content rarely provides explicit structural cues such as headings or section titles, requiring topic boundaries to be inferred from the discourse itself, including transition phrases and shifts in subject matter.

Classical unsupervised methods place boundaries where lexical cohesion drops [1, 2], and speech segmentation methods use prosody and discourse markers [3, 4]. Representation-based methods learn boundary decisions from labeled corpora [5, 6], train per-document segment representations without labels [7], or hierarchically cluster transcript embeddings [8]. However, these methods determine section granularity through thresholds, supplied segment counts, or learned boundary rules, which can limit generalization across diverse document structures. Recent LLM systems predict topic boundaries through boundary lists, sentence-level topic labels, recursive splitting, table-of-contents planning, topic-shift predictions based on utterance intent, or iterative chunking [9, 10, 11, 12, 13]. These methods directly predict boundaries or segment structures *without explicitly grounding them* in topic-opening cues. To make these predictions, the model must decide whether each change in the discussion marks a new topic or is only a minor shift within the current topic. Treating minor shifts as new topics leads to over-segmentation, while overlooking meaningful topic changes leads to under-segmentation.

Spoken documents vary substantially in their topical structure. Some consist of short, clearly separated topics, whereas others sustain extended discussions around a single subject. An effective topic

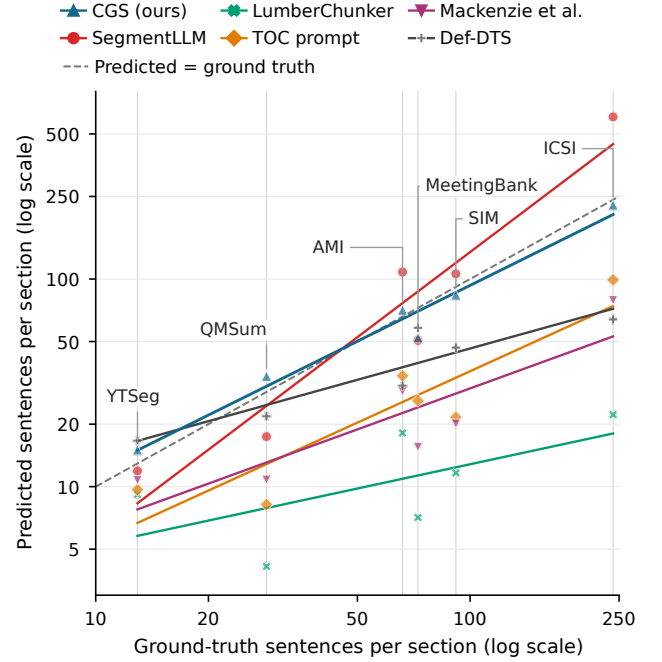


Fig. 1. Predicted versus ground-truth section length. CGS more closely tracks ground-truth lengths across corpora, while baselines often produce sections that are too short or too long. Lengths are defined in Table 1; predictions are averaged across six backbones. Vertical lines mark corpora; colored lines show log-log fits.

Table 1. Evaluated corpus subsets. n counts documents. For each document, we calculate the average ground-truth segment length in sentences. Length reports the median across documents.

Dataset	n	Length	Description
YTSeg [14]	1,448	12.9	Creator-authored chapters
ICSI [15, 16]	75	241.0	Research meetings
AMI [17]	129	66.0	Multi-party meetings
MeetingBank [18, 16]	28	72.6	City-council meetings
QMSum [19, 16]	20	28.6	Parliamentary committees
SIM [16]	100	91.6	Spliced meeting excerpts

segmenter must therefore adapt to a wide range of segment granularities. This variation is evident across six benchmarks. The median ground-truth segment length ranges from 12.9 sentences in chap-

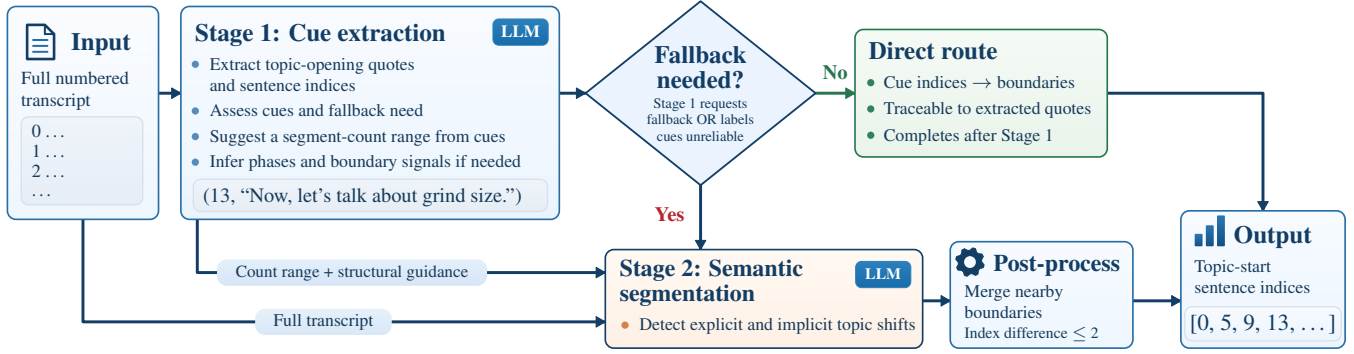


Fig. 2. CGS overview. Each cue pairs a topic-opening quote with its sentence index. Both stages use the same LLM with task-specific prompts. Structural guidance covers phases, boundary signals, and expected segment durations.

tered YouTube videos to 241 sentences in research-group meetings, an 18.6-fold difference. Despite this substantial variation, existing LLM-based segmenters often produce segments that are systematically too short or too long for the target corpus (Fig. 1).

In this regard, we propose **Cue-Grounded Segmentation (CGS)**¹, a zero-shot framework that grounds boundary prediction in topic-opening cues. Such cues can indicate which parts of a discussion are presented as distinct topics. When these cues are sufficiently informative, CGS uses their sentence positions as boundaries, allowing section lengths to reflect the spacing between topic openings. When cue evidence is insufficient, CGS falls back to semantic segmentation guided by the document structure inferred during cue extraction. This design enables CGS to better adapt to the wide variation in ground-truth segment lengths across the six benchmarks (Fig. 1).

More specifically, CGS is implemented in two stages (Fig. 2). Stage 1 reads the full numbered transcript, quotes topic-opening phrases, and reports their sentence indices. On the direct route, those indices define the boundaries, so each cue-derived boundary is paired with the transcript phrase that motivated it. When Stage 1 requests fallback or judges the cues unreliable, Stage 2 re-reads the full transcript, guided by the expected topic count, structural phases, and boundary signals inferred in Stage 1. It identifies topic changes from the transcript’s semantic content, without explicit transition phrases.

Our main contributions and findings are:

- We introduce CGS, a training-free, zero-shot method that uses quoted topic openings to locate boundaries and invokes structure-guided semantic segmentation when those cues are insufficient.
- Across six benchmarks and six LLM backbones, CGS achieves the best average performance on all metrics, with low API cost on proprietary models.
- Ablations show the benefit of combining cue extraction with semantic fallback and supplying segment-count guidance. Automatic speech recognition (ASR) experiments show that CGS retains its performance advantage under speech recognition noise.

¹Full prompts, output schemas, and code are available at <https://github.com/schoi828/CGS>.

2. METHODOLOGY

CGS consists of two stages. First, it identifies explicit transition cues in the transcript and determines whether they provide sufficient evidence for topic boundaries. When such cues are insufficient, CGS falls back to semantic segmentation guided by the document structure inferred during cue extraction (Fig. 2). The two stages are implemented as separate calls to the same LLM. We use a single shared set of prompts across all datasets, without providing dataset names, task-specific tuning, or labeled examples.

2.1. Stage 1: Indexed Transition Evidence

Stage 1 processes the full transcript and returns a structured JSON output. The prompt asks the model to identify and quote each topic-opening phrase together with its sentence index, forming the cue list C . In the same response, the model assigns an overall cue-reliability label—reliable, uncertain, or unreliable—and indicates whether semantic fallback is required. Stage 2 is invoked when fallback is explicitly requested or when the cue list is labeled unreliable; an uncertain label alone does not trigger fallback.

Stage 1 also predicts a segment-count range from the extracted cues. Given n cues, the lower and upper bounds are set to $\max(1, n - 1)$ and $n + 1$, respectively. When fallback is required, the model additionally produces structural guidance for Stage 2: phases summarizing the document’s major parts and their order, boundary signals describing likely indicators of topic transitions, and a duration range estimating plausible segment lengths in seconds from the transcript. When fallback is not required, these fields are omitted. Table 3 illustrates the Stage 1 output using an example from an AMI meeting.

The direct route completes segmentation after Stage 1. CGS discards out-of-range cue indices and returns $B_{\text{cue}} = \{0\} \cup \{i : (i, q) \in C\}$ over the remaining pairs.

2.2. Stage 2: Conditioned Semantic Fallback

Stage 2 reads the full numbered transcript together with the segment-count range and structural guidance from Stage 1. The LLM uses this guidance to identify coherent sections and places boundaries where the discussion moves to a new topic. It can identify these transitions without an explicit phrase such as “Next, let’s discuss staffing.” Stage 2 predicts a new set of topic-start sentence indices

Table 2. Performance on the six benchmark corpora. **Bold** and underline mark the best and second-best values per column. [†]Training-free, but uses self-supervised BERT pretraining. [‡]Def-DTS uses only valid outputs completed within the output-token limit of backbones.

		P_k ($\downarrow$)							WindowDiff ($\downarrow$)							Boundary F_1 ($\uparrow$)						
	Method	YT	IC	MB	SIM	AMI	QM	Avg.	YT	IC	MB	SIM	AMI	QM	Avg.	YT	IC	MB	SIM	AMI	QM	Avg.
Classical	TEXTTILING [1]	.513	.716	.648	.604	.611	.630	.620	.653	.996	.994	1.00	.816	.999	.910	.433	.036	.086	.057	.121	.129	.144
	BERT-TT [20] [†]	.402	.473	.420	.378	.420	<u>.407</u>	.417	.409	.514	.464	.434	.442	<u>.432</u>	.449	.189	.055	.133	.197	.102	.188	.144
LLM-based	LUMBERCHUNKER [12]	.378	.714	.641	.604	.540	.623	.583	.473	.983	.952	.994	.732	.972	.851	.539	.121	.182	.119	.291	.233	.247
	Mackenzie et al. [10]	.349	.533	.460	.540	.459	.511	.476	.421	.722	.692	.844	.599	.712	.665	.519	.182	.293	.141	.302	.226	.277
	TOC PROMPT [11]	.414	.516	.472	.574	.511	.495	.497	.483	.632	.648	.840	.606	.693	.650	.453	.098	.181	.090	.158	.322	.217
	DEF-DTS [‡] [13]	.338	.400	.287	.317	.304	.406	.340	.368	.490	.402	.430	.387	.536	.433	.325	.170	.272	.279	.358	.251	.289
	SEGMENTLLM [9]	<u>.271</u>	<u>.333</u>	<u>.206</u>	<u>.306</u>	<u>.309</u>	.416	<u>.307</u>	<u>.324</u>	<u>.402</u>	<u>.275</u>	<u>.370</u>	<u>.367</u>	.535	<u>.379</u>	<u>.590</u>	<u>.200</u>	<u>.618</u>	<u>.350</u>	<u>.330</u>	<u>.382</u>	<u>.412</u>
Ours	CGS	.245	.212	.069	.206	.216	.301	.208	.274	.272	.155	.241	.269	.333	.257	.604	.395	.751	.471	.438	.391	.508

from the transcript; it does not receive the Stage-1 cue list. The suggested segment-count range guides this prediction rather than imposing a strict constraint. To reduce the number of very short segments, CGS merges neighboring Stage 2 boundary predictions whose sentence indices differ by at most two.

Table 3. Stage-1 guidance supplied to Stage 2: an example from an AMI meeting. Phase and signal entries are excerpts; both ranges are shown in full.

Guidance	Saved example
Segment count	3–5 segments
Structural phases	<i>Prototype Discussion</i> : Follows the introduction; covers design ergonomics, materials, and cost-saving measures.
Boundary signals	Introduction of new documents or evaluation criteria
Segment duration	120–300 seconds

3. EXPERIMENTAL SETUP

Corpora. Table 1 summarizes the six benchmarks, covering 1,800 documents. We use their supplied transcripts, treating each sentence or utterance as one indexed sentence. SIM tests topic changes introduced by splicing unrelated meeting excerpts. YTSeg, AMI, and ICSI provide audio recordings and timestamped topic boundaries. For each evaluated configuration, prompts and settings remain fixed across corpora. No model is fine-tuned for segmentation or given labeled examples from the target corpora.

Metrics. P_k metric [21] measures how often sentences i and $i + k$ share a predicted segment but not a ground-truth segment, or vice versa. WindowDiff [22] measures how often predicted and ground-truth boundary counts differ within a window. Both use $k = \max(1, \lfloor \bar{L}/2 \rfloor)$, where $\bar{L}$ is mean ground-truth segment length per document. Boundary F_1 [14] is the harmonic mean of precision and recall. We count a predicted boundary as correct if it is within two sentences of a ground-truth boundary. Each boundary can be matched at most once. Macro scores average documents within corpora, then corpora equally.

Baselines. TEXTTILING [1] detects changes in lexical cohesion, while BERT-TT [20] uses contextual embeddings to detect drops in similarity between adjacent sentence blocks. The LLM baselines include a TOC PROMPT [11], which generates section headings and start indices; the recursive segmentation method of

Table 4. Performance by backbone, averaged equally over the six corpora. Table 2 averages LLM results over these backbones. **Bold** marks the better value.

Backbone	SEGMENTLLM			CGS (ours)		
	P_k ($\downarrow$)	WD ($\downarrow$)	F_1 ($\uparrow$)	P_k ($\downarrow$)	WD ($\downarrow$)	F_1 ($\uparrow$)
Gemini-3.1-FL	.234	.285	.539	.177	.207	.557
GPT-5.6-Terra	.328	.415	.525	.181	.218	.550
Qwen3.6-27B	.241	.278	.495	.175	.208	.543
Gemma-4-26B	.411	.605	.363	.221	.296	.486
Qwen3.5-9B	.309	.334	.299	.230	.279	.464
Qwen3.5-4B	.319	.356	.248	.264	.337	.452

Mackenzie et al. [10], which predicts boundary indices and recursively divides long segments using fixed illustrative examples; and LUMBERCHUNKER [12], which iteratively identifies topic shifts, adapted here to sentence units. SEGMENTLLM [9] uses a zero-shot prompt to group consecutive sentences by topic, listing every sentence index in each group. Segment boundaries are recovered from these groups after resolving gaps and overlaps. DEF-DTS [13] summarizes the preceding and following context of each utterance, classifies its intent, and predicts whether it starts a new topic. Generating these intermediate outputs for every utterance leads to substantial output-token usage on long transcripts. We apply Def-DTS to the full transcript, including speaker labels when available.

Backbones. We evaluate six LLM backbones: Gemini-3.1-Flash-Lite [23], GPT-5.6-Terra [24], Qwen3.6-27B [25], Gemma-4-26B-A4B [26], Qwen3.5-9B [27], and Qwen3.5-4B [27]. We use temperature $T=0.3$ for Flash-Lite and the open-weight models. Terra uses the API’s default temperature because non-default values were not supported.

4. RESULTS AND ANALYSIS

Table 2 shows that, averaged over six backbones, CGS outperforms all six published baselines on every corpus under all three metrics. Figure 1 summarizes section lengths from the same predictions. CGS also outperforms SEGMENTLLM, the strongest baseline in Table 2, on all three macro metrics for each backbone (Table 4). Across the six backbones, CGS completes segmentation after Stage 1 for 80–93% of documents.

Cost and token use. Table 5 shows that CGS combines the lowest segmentation error with the second-lowest API cost. CGS generates about one-quarter as many output tokens as SEGMENTLLM, averaged across six backbones. On the two proprietary

backbones, output tokens are priced six times higher than input tokens, allowing the output savings to offset the additional input cost.

Table 5. P_k from Table 2 and recorded token use (thousands per document). Per-document means are averaged across the six corpora and six backbones. Costs are calculated from recorded token usage at standard API rates and averaged over the two proprietary backbones (USD/1,000 documents). **Bold** and underline mark the best and second-best values.

Method	$P_k \downarrow$	Input (k) $\downarrow$	Output (k) $\downarrow$	Cost $\downarrow$
LUMBERCHUNKER	.583	32.36	0.825	35.59
Mackenzie et al.	.476	46.25	0.233	53.03
TOC PROMPT	.497	<u>14.94</u>	<u>0.306</u>	17.17
DEF-DTS	.340	16.37	42.221	317.48
SEGMENTLLM	<u>.307</u>	14.03	2.428	26.13
CGS	.208	19.63	0.603	<u>21.87</u>

CGS ablations. Table 6 reports ablations on Flash-Lite and Qwen3.6-27B, which achieve the strongest CGS performance among the evaluated proprietary and open-weight backbones, respectively.

Stage 1 only (FORCED CUES) places boundaries at the extracted topic openings and skips semantic fallback. *Stage 2 only* skips Stage 1 and segments every transcript directly. These comparisons test whether either stage can replace the complete pipeline. *Cue indices only* keeps the two-stage procedure but asks Stage 1 to report topic-opening sentence indices without quoting the corresponding phrases. This tests whether generating the quotes helps boundary prediction. CGS performs best among the variants in Table 6 on both backbones.

We also examine how much Stage-1 guidance improves Stage-2 segmentation. For the same documents routed to Stage 2, *Count-only fallback* supplies the transcript and suggested segment-count range, but removes phase descriptions, boundary signals, and duration estimates. *Transcript-only fallback* also removes the count range. Retaining the count range provides most of the improvement over transcript-only fallback. Structural guidance reduces P_k further, with a larger effect on Qwen-27B.

Table 6. CGS ablations ($P_k \downarrow$, averaged equally over six corpora; $T=.3$). **Bold** marks the lowest displayed value.

Configuration	Flash-Lite	Qwen-27B
CGS	.177	.175
Stage 1 only (FORCED CUES)	.197	.187
Stage 2 only	.306	.285
Cue indices only	.197	.180
Count-only fallback	.192	.183
Transcript-only fallback	.249	.218

Boundary errors. Table 7 summarizes the correct, missed, and incorrect boundary counts used to calculate Boundary F_1 for each document. CGS detects about as many ground-truth boundaries as SEGMENTLLM and makes the fewest incorrect boundary predictions overall. Some baselines detect more ground-truth boundaries but also make many more incorrect predictions.

Table 7. Counts are normalized to 100 ground-truth boundaries. Incorrect predictions can exceed 100 when a method predicts too many boundaries. Correct counts predictions that match a ground-truth boundary within two sentences. Each boundary is matched at most once. Missed reports how many ground-truth boundaries are not detected. Incorrect counts wrong boundary predictions. Values average six corpora and, for LLMs, six backbones.

Method	Correct $\uparrow$	Missed $\downarrow$	Incorrect $\downarrow$
TEXTTILING	61.6	38.4	1361.8
BERT-TT	13.3	86.7	71.6
LUMBERCHUNKER	79.3	20.7	750.6
Mackenzie et al.	62.4	37.6	476.2
TOC PROMPT	41.4	58.6	262.4
DEF-DTS	46.4	53.6	351.0
SEGMENTLLM	54.6	45.4	148.3
CGS	55.5	44.5	58.2

CGS retains its advantage on ASR transcripts. Speech recognition errors and missing punctuation can make topic boundaries harder to locate. We use YTSeg’s released Whisper-Large transcripts and generate AMI and ICSI transcripts with Whisper large-v3 [28]. For AMI and ICSI, we compare the audio timestamp of each ground-truth boundary with Whisper’s chunk start timestamps. A chunk starting at the boundary timestamp begins the new segment. If there is no exact timestamp match, the new segment begins with the first chunk after the boundary timestamp. Averaged across six backbones, CGS outperforms SEGMENTLLM on all metrics for all three corpora (Table 8).

Table 8. ASR segmentation performance, averaged over six backbones. Sentence divisions differ between the original and ASR transcripts.

Corpus	Method	$P_k \downarrow$	WD $\downarrow$	$F_1 \uparrow$
AMI	SegmentLLM	.278	.359	.420
	CGS	.202	.263	.512
ICSI	SegmentLLM	.310	.391	.355
	CGS	.244	.307	.456
YTSeg	SegmentLLM	.277	.335	.577
	CGS	.253	.286	.592

5. CONCLUSION

CGS grounds topic boundaries in quoted transition cues and invokes structure-guided semantic segmentation when these cues are insufficient. Averaged across the six corpora, it outperforms the evaluated baselines on all metrics with each of the six backbones. Ablations show the benefit of combining cue extraction with selective semantic fallback and identify segment-count guidance as an important contributor to fallback performance. CGS also retains its performance advantage on noisy ASR transcripts, supporting robustness to speech recognition errors. By combining accurate segmentation with low API cost, CGS offers a practical approach to segmenting spoken documents without task-specific training.

6. ACKNOWLEDGMENTS

This work was supported by (1) the National Research Foundation of Korea (NRF) grant funded by the Korea government (MSIT) (RS-2026-25477522), (2) Institute of Information & communications Technology Planning & Evaluation (IITP) grant funded by the Korea government (MSIT) [NO.RS-2021-II211343, Artificial Intelligence Graduate School Program (Seoul National University)], and (3) the Starting growth Technological R&D Program (RS-2024-00506994) funded by the Ministry of SMEs and Startups (MSS, Korea).

7. COMPLIANCE WITH ETHICAL STANDARDS

This study uses publicly released research corpora under their respective terms and involves no new human-subject data collection.

8. REFERENCES

- [1] M. A. Hearst, “TextTiling: Segmenting text into multi-paragraph subtopic passages,” *Computational Linguistics*, vol. 23, no. 1, pp. 33–64, 1997.
- [2] J. Eisenstein and R. Barzilay, “Bayesian unsupervised topic segmentation,” in *Proceedings of EMNLP*, 2008, pp. 334–343.
- [3] E. Shriberg, A. Stolcke, D. Hakkani-Tür, and G. Tür, “Prosody-based automatic segmentation of speech into sentences and topics,” *Speech Communication*, vol. 32, no. 1–2, pp. 127–154, 2000.
- [4] M. Galley, K. R. McKeown, E. Fosler-Lussier, and H. Jing, “Discourse segmentation of multi-party conversation,” in *Proceedings of ACL*, 2003, pp. 562–569.
- [5] M. Lukasik, B. Dadachev, K. Papineni, and G. Simões, “Text segmentation by cross segment attention,” in *Proceedings of EMNLP*, 2020, pp. 4707–4716.
- [6] K. Lo, Y. Jin, W. Tan, M. Liu, L. Du, and W. Buntine, “Transformer over pre-trained transformer for neural text segmentation with enhanced topic coherence,” in *Findings of EMNLP*, 2021, pp. 3334–3340.
- [7] K. Wang, X. Zhao, Y. Li, and W. Peng, “M3Seg: A maximum-minimum mutual information paradigm for unsupervised topic segmentation in ASR transcripts,” in *Proceedings of EMNLP*, 2023, pp. 7928–7934.
- [8] D. C. Gklezakos, T. Misiak, and D. Bishop, “TreeSeg: Hierarchical topic segmentation of large transcripts,” *arXiv preprint arXiv:2407.12028*, 2024.
- [9] Y. Fan, F. Jiang, P. Li, and H. Li, “Uncovering the potential of ChatGPT for discourse analysis in dialogue: An empirical study,” in *Proceedings of LREC-COLING*, 2024, pp. 16998–17010.
- [10] P. Mackenzie, M. Shah, and P. Frenett, “Topic segmentation using generative language models,” *arXiv preprint arXiv:2601.03276*, 2025.
- [11] S. Freisinger, P. Seeberger, T. Ranzenberger, T. Bocklet, and K. Riedhammer, “Towards multi-level transcript segmentation: LoRA fine-tuning for table-of-contents generation,” in *Proceedings of Interspeech*, 2025, pp. 276–280.
- [12] A. V. Duarte, J. D. S. Marques, M. Graça, M. Freire, L. Li, and A. L. Oliveira, “LumberChunker: Long-form narrative document segmentation,” in *Findings of EMNLP*, 2024, pp. 6473–6486.
- [13] S. Lee, Y. Yoo, M. Jung, and M. Song, “Def-DTS: Deductive reasoning for open-domain dialogue topic segmentation,” in *Findings of ACL*, 2025, pp. 20736–20753.
- [14] F. Retkowski and A. Waibel, “From text segmentation to smart chaptering: A novel benchmark for structuring video transcriptions,” in *Proceedings of EACL*, 2024, pp. 406–419.
- [15] A. Janin, D. Baron, J. Edwards, D. Ellis, D. Gelbart, N. Morgan, B. Peskin, T. Pfau, E. Shriberg, A. Stolcke, and C. Wooters, “The ICSI meeting corpus,” in *Proceedings of ICASSP*, 2003, vol. 1, pp. 364–367.
- [16] Y. Fan, J. Pool, S. Filipi, and R. Cutler, “Topic-conversation relevance (TCR) dataset and benchmarks,” in *Advances in Neural Information Processing Systems, Datasets and Benchmarks Track*, 2024, vol. 37, pp. 140159–140174.
- [17] J. Carletta, “Unleashing the killer corpus: experiences in creating the multi-everything AMI meeting corpus,” *Language Resources and Evaluation*, vol. 41, no. 2, pp. 181–190, 2007.
- [18] Y. Hu, T. Ganter, H. Deilamsalehy, F. Dernoncourt, H. Foroosh, and F. Liu, “MeetingBank: A benchmark dataset for meeting summarization,” in *Proceedings of ACL*, 2023, pp. 16409–16423.
- [19] M. Zhong, D. Yin, T. Yu, A. Zaidi, M. Mutuma, R. Jha, A. H. Awadallah, A. Celikyilmaz, Y. Liu, X. Qiu, and D. Radev, “QMSum: A new benchmark for query-based multi-domain meeting summarization,” in *Proceedings of NAACL-HLT*, 2021, pp. 5905–5921.
- [20] A. Solbiati, K. Heffernan, G. Damaskinos, S. Poddar, S. Modi, and J. Cali, “Unsupervised topic segmentation of meetings with BERT embeddings,” *arXiv preprint arXiv:2106.12978*, 2021.
- [21] D. Beeferman, A. Berger, and J. Lafferty, “Statistical models for text segmentation,” *Machine Learning*, vol. 34, no. 1–3, pp. 177–210, 1999.
- [22] L. Pevzner and M. A. Hearst, “A critique and improvement of an evaluation metric for text segmentation,” *Computational Linguistics*, vol. 28, no. 1, pp. 19–36, 2002.
- [23] Google DeepMind, “Gemini 3.1 Flash-Lite model card,” <https://deepmind.google/models/model-cards/gemini-3-1-flash-lite/>, Mar. 2026.
- [24] OpenAI, “GPT-5.6 Terra,” <https://developers.openai.com/api/docs/models/gpt-5.6-terra>, 2026, API model documentation, accessed September 17, 2026.
- [25] Qwen Team, “Qwen3.6-27B: Flagship-level coding in a 27B dense model,” <https://qwen.ai/blog?id=qwen3.6-27b>, Apr. 2026.
- [26] Gemma Team, “Gemma 4 technical report,” *arXiv preprint arXiv:2607.02770*, 2026.
- [27] Qwen Team, “Qwen3.5: Towards native multimodal agents,” <https://qwen.ai/blog?id=qwen3.5>, Feb. 2026.
- [28] A. Radford, J. W. Kim, T. Xu, G. Brockman, C. McLeavey, and I. Sutskever, “Robust speech recognition via large-scale weak supervision,” in *Proceedings of ICML*, 2023, pp. 28492–28518.