International Journal on Science and Technology (IJSAT)
E-ISSN: 2229-7677 ● Website: www.ijsat.org ● Email: editor@ijsat.org

MOMent: Intelligent Audio Summarization & Audio Tracking System

Sarthak Atkare[1], Vishakha Nandeshwar[2], Swejal Gujwar[3], Ivan Mendonca[4], Prof. Sanjay Kalamdhad[5]

[1,2,3,4]Department of Computer Science and Business Systems, [5]Guide, Assistant Professor, Department of CSBS
St. Vincent Pallotti College of Engineering & Technology, Nagpur, India
An Autonomous Institution Affiliated to Rashtrasant Tukadoji Maharaj Nagpur University

Abstract
Meetings are places where decisions are made and work is assigned, but the information they produce is trapped in unstructured speech. Handwriting things is slow and error-prone, and it's rarely practical to listen to an entire recording to try to find a single promise. This paper presents MOMent, a modular system that utilizes a meeting recording and produces a speaker-labelled transcript, an abstractive summary highlighting the key discussion points, and a structured list of action items with an owner, deadline, and priority when specified. The pipeline consists of a large-scale weakly supervised speech recognizer, a neural speaker diarization stage, a transformer-based summarizer, and a prompt-driven large language model (LLM) extractor that outputs schema-constrained task records persisted in a tracking store. We describe the architecture, the design decisions behind each stage, a chunking strategy for long transcripts, and a rigorous evaluation protocol based on word error rate, diarization error rate, ROUGE, BERTScore, and field-level precision and recall for task extraction. The paper presents the system design and the planned evaluation; the quantitative results will be added after the experiments on public meeting corpora are completed. MOMent aims to show that the usual individual tools for transcription, summarization, and task management can be integrated into one workflow that reduces the number of steps from conversation to follow-up.

Keywords: speech-to-text, speaker diarization, meeting summarization, action item extraction, large language models, natural language processing, task tracking

1. Introduction

Meetings are a standard part of academic, corporate and organizational life. Ideas are proposed, decisions are made and responsibilities are assigned, almost always in the course of spoken conversation. But spoken conversation is a poor storage device. It is linear, it jumps from one subject to another, and it leaves no structured record of who would do what by when. Two habits are dominant in current practice. Participants listen and write notes or listen and record the recording later. The first divides attention and tends to lose detail; the second takes as long as the meeting itself. In both cases, accountability suffers because the mapping between a task, its owner and its deadline lives in people's memory rather than in a record. Recent advances in speech and language technology enable improved workflow. [1] ASR is now robust to accents and recording conditions. Neural diarization can assign speech segments to specific speakers [2], [4].

Pretrained sequence-to-sequence and decoder-only language models can summarize long text, and follow structured instructions [3], [10]. All of these capabilities are mature in their own right but are usually available as separate products.

A user wanting a transcript, summary, and task list has to stitch together several tools and manually re-enter the results.

In this paper we introduce MOMent, a system that integrates these capabilities end to end. The contributions of this work can be summarized as follows:

1. A modular architecture for converting meeting audio/video into a speaker-attribute transcript, a summary with key points and structured action items.
2. A schema-constrained extraction approach where an LLM returns task records (description, assignee, deadline, priority) that are validated before storage, with explicit handling of missing fields.
3. A hierarchical chunking strategy which enables long transcripts to be summarized without surpassing model context limits.
4. A protocol for evaluation that measures each stage individually as well as the pipeline as a whole, so that error propagation can be observed, rather than hidden.

The rest of the paper is organized as follows. Section II reviews the related work. Section III formalizes the problem. The system is described in Section IV. Implementation is discussed in Section V. The evaluation protocol is described in Section VI. Expected behavior and limitations are discussed in Section VII. Section VIII concludes.

2. Related Work

A. Speech Recognition

Traditional ASR systems are trained on meticulously transcribed corpora and perform poorly on out-of-domain audio [8]. Radford et al. trained an encoder- decoder transformer on hundreds of thousands of hours of weakly supervised web audio and showed that the resulting model generalizes well to unseen accents, noise and domains without fine-tuning [1]. Such robustness is especially desirable for meetings where microphone quality and speaking styles are very diverse.

B. Speaker Diarization

Diarization answers the question of who spoke and when. Classical systems relied on clustering of hand-crafted acoustic features, as surveyed by Anguera et al. [7], and later work replaced them by learned embeddings and end-to-end segmentation networks, reviewed by Park et al. [8]. The pyannote.audio toolkit includes neural building blocks for segmentation, embedding and clustering [2]. Further work improved the handling of overlapping speech with overlap-aware resegmentation [4]. Overlapping speech is still the primary source of errors in multi-party meeting.

C. Meeting Summarization

backchanneled and disseminates information over many speaker turns. Datasets like SAMSum [11] for chat dialog and QMSum [12] for query-based meeting summarization have fueled progress. Denoising sequence-to-sequence models like BART [3] have demonstrated strong baselines for abstractive dialog summarization. Instruction-following LLMs [10] are now providing an alternative that does not require task- specific training, but needs to be carefully controlled to avoid unsupported statement.

D. Action Item Detection

Commitments extraction has been studied on meeting corpora such as AMI [5] and ICSI [6] where decisions and action related dialog acts are annotated. Early approaches used supervised classifiers on lexical and prosodic features. Recent systems have cast the problem as structured generation . A persistent difficulty is that task information is often scattered or implicit: the owner might be mentioned several turns before the deadline is stated, or a deadline might be relative ("by next Friday") and require a reference date to be resolved.

E. Positioning of This Work

Commercial and research tools today do a pretty good job of covering individual stages. Less common is an open pipeline that (i) preserves speaker attribution available to the extraction stage, (ii) treats missing task fields as first class outcomes rather than filling them with guesses, and (iii) closes the loop by storing tasks for follow-up. These three properties are the foundation of MOMent.

3. Problem Formulation

Let A be an input recording of length T with K unknown participants. The system must output three times.

- Transcript. An ordered sequence of segments $S = \{(s_i, t_i^{start}, t_i^{end}, w_i)\}$, where s_i is a speaker label and w_i the recognized words.
- Summary. A short text Y together with a set of key points P, both faithful to S.
- Action items. A set of records $R = \{(d_j, o_j, e_j, p_j)\}$, where d_j is the task description, o_j is the owner, e_j is the deadline and p_j is the priority. If not specified by the conversation, o_j, e_j, p_j can all be null.

The objective is to reduce the effort required for the user to reach an accurate understanding of the meeting and its obligations, while minimizing the amount of fabricated content, such as invented owners or deadlines. We consider an incorrectly filled field to be
more harmful than a missing one, because a missing field is visible and forces a human to check, while a plausible but false value does not.

4. System Architecture And METHODOLOGY

MOMent is organized as seven sequential stages, each consuming the output of the previous one (Fig. 1)
Fig. 1. Pipeline Overview

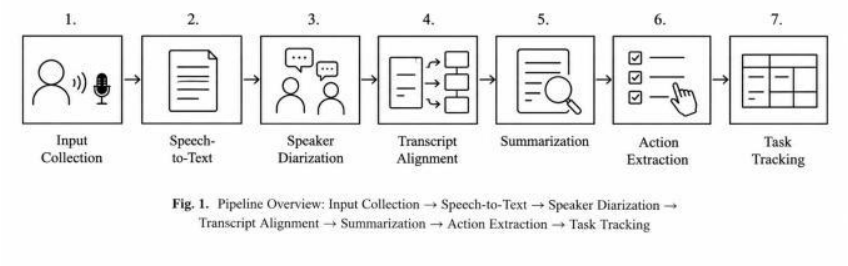


Fig. 1. Pipeline Overview: Input Collection → Speech-to-Text → Speaker Diarization → Transcript Alignment → Summarization → Action Extraction → Task Tracking

A. Input Collection and Preprocessing The system takes in audio files and meeting video. For video, the audio track is extracted and all inputs are resampled to a single mono, 16 kHz representation, which is the expected format for the recognition model. There are optional steps such as loudness normalization and silence trimming. The original file is kept so that each generated item can be traced to a time position in the source.

B. Speech-to-Text

Transcription uses an encoder-decoder model of the Whisper family [1]. We request word-level time stamps because long recordings are processed in overlapping windows and we want to be able to match text against speaker segments later. The use of a weakly supervised multilingual model is in tune with the practical reality of Indian academic and workplace meetings where accent variation and occasional code-mixing is the norm. You can configure the model size to trade-off between accuracy and the hardware you have.

C. Speaker Diarization

We employed a pyannote.audio pipeline [2], [4] for speaker segmentation and clustering. The outcome is a collection of time segments labeled with anonymous speaker ids. Where the number of participants is known beforehand, this can be provided as a constraint on clustering. Users can later rename anonymous tags to real names and this then propagates into the summary and task records.

The recognizer and the diarizer are running independently, so their output has to be merged. Each identified word is assigned to the speaker whose segment has the maximum temporal overlap with the word. Consecutive words of the same speaker are combined into turns, and very short turns, e.g. acknowledgments, are retained but marked for downstream stages to discount. Words in overlapping regions are tagged with the dominant speaker and marked as uncertain.

E. Summarization

The full transcript of the meeting is often longer than the input length of the summarization models. MOMent therefore has a two level procedure:

1. Segment-level summaries. The transcript is segmented at topic or pause boundaries into bounded length chunks with a small overlap and each chunk is summarized independently.
2. Global summary. The chunk summaries are concatenated and summarized again to generate the final short summary and a list of key discussion points.
3. The summarizer can be a BART [3] fine-tuned sequence-to-sequence model or an instruction-following LLM [10]. For the latter, the prompts tell the model to use only information in the transcript, to attribute statements to speakers when appropriate, and not to add recommendations. Speaker labels must be preserved in the input as decisions only make sense if you know who is making them.

F. Action Item Extraction

The extraction stage is cast as structured generation. For each chunk, the LLM is prompted to return a JSON array, where each element has the fields description, assignee, deadline, priority, and evidence, the latter being the transcript span that supports the task. The instructions say to set the unspecified fields to null, not to guess. Three safety measures follow generation:

- Schema validation. Malformed output is rejected and regenerated once with an error message; If it still fails, the chunk will be flagged for review.
- Date resolution. Relative expressions are translated to calendar dates using the meeting date as the reference date and the original phrase is preserved with the resolved date.
- Deduplication. Tasks discussed in multiple chunks are combined based on description similarity and assignee equality.

Inferred from explicit cues (“urgent”, “top priority”, “when you get time”) and mapped to a small ordinal scale, default null when not present.

G. Task Tracking Validated tasks are saved to a persistent store with a status field (open, in progress, done), a link to the source meeting, and the evidence span with its time stamp. The easy interface lets users filter by assignee, due date or status, edit any field, and go from a task to the point in the recording where it was assigned. Human editing is intentional: extraction is a proposal that someone verifies, not an unquestioned record.

5. Implementation Considerations

Modularity. Modularity means that every stage offers an interface, like audio coming in and segments coming out or transcript coming in and summary coming out. Because of this Modularity any component can be swapped. For example the recognizer can be. The large language model can be moved from a hosted service to a local model and this change will not disturb the other components.

Privacy. Privacy means that meeting audio often holds information. The system allows local processing for recognition and diarization. Privacy also demands user consent and clear notice, to all participants before any recording is processed. If an external large language model service is used transcripts should be kept to an sensitive meetings should be sent to a local model.

Latency and cost. Recognition and diarization drive the compute cost. Grow as recording length increases. Therefore processing is done asynchronously with progress reporting. Intermediate artefacts such, as transcript and diarization are cached, so that re- summarization or extraction does not repeat expensive audio steps.

Language coverage. Although the initial target is English the recognizer is multilingual. The downstream prompts can be extended to other languages after evaluation

6. Evaluation Protocol

Mistakes in the early stages propagate downstream, so each stage is evaluated alone and then the pipeline is evaluated end to end.

A. Datasets

The AMI Meeting Corpus [5] and the ICSI Meeting Corpus [6] are multi-party meeting recordings with transcripts and annotations, and they are the main benchmarks for transcription, diarisation and action-related content. The summarization component is evaluated against SAMSum [11] and QMSum [12]. Also, a small internal set of recorded student-project meetings will be annotated by two independent annotators regarding tasks, owners, deadlines and priorities, providing a domain-specific test set that reflects the target users.

B. Metrics

Table I. Evaluation metrics by stage

Stage Metric Purpose
Transcription accuracy
Speech-to- Text
Word Error Rate (WER)
Diarization Diarization Error Rate (DER)
Speaker attribution accuracy, including missed speech, false alarm, and confusion
Summarization ROUGE-1/2/L [9],
Lexical and semantic overlap with references

BERTScore [13]
Summarization Human rating of
Detects unsupported statements that automatic metrics miss
faithfulness, coverage, and conciseness (1–5)
Precision, recall, and F1 for task detection
Task extraction
Whether tasks are found
Field-level accuracy for assignee, deadline, priority
Task extraction
Whether details are correct
Hallucination rate (fields filled but unsupported by the transcript)
Safety of the null- handling policy
Task extraction
System End-to-end time-to- understanding in a small user study
Practical benefit over manual notes

C. Baselines and Ablations Summarization will be compared to an extractive baseline and a single-pass LLM with no chunking. The task extraction will be compared with a rule-based baseline with keyword and date patterns. Ablation studies will quantify the contribution of speaker labels by removing them from the extraction input. Ablation studies will quantify the error propagation from ASR and diarization by replacing the reference transcripts with noisy transcripts.

D. User Study

A small group of participants will be granted access to recorded meetings, and will be asked to generate meeting notes and task lists either manually or using
MOMent. Completion time, completeness and perceived effort will be recorded with a within-subjects design to limit individual variation.

7. Expected Outcomes, Discussion, And Limitations

A. Status of Results

The paper presents a system design and evaluation plan. No measured results are reported at this stage. The final numbers when the experiments are finished will be held in the template in Table II.

Table II. Results template (to be completed after
experimentation)
Component Metric Dataset Result
ASR WER (%) AMI Low WER
Diarization DER (%) AMI Low DER
High ROUGE-L / BERTScore
QMSum / SAMSum
Summarization ROUGE-L /
BERTScore
Task detection F1 In-house set High F1-
score
In-house set High

Task fields Assignee / Deadline / Priority accuracy
accuracy for assignee, deadline, and priority
% of filled fields unsupported
Hallucination rate
In-house set Low
percentage of unsupported fields

B. Anticipated Behaviour The paper presents a system design and evaluation plan. No measured results are reported at this stage. The final numbers when the experiments are finished will be held in the template in Table II. Several patterns are expected given what is known about the components involved. The quality of the transcription should be high for close-talk audio and degrade for far-field recordings. Diarization errors are expected to be concentrated in overlapped speech and in meetings with many similar sounding voices. The extraction should be best for assignees and worst for implicit deadlines and priorities. The summaries should reliably capture the main topics but may leave out minor items which are often never explicitly said. These are expectations to be tested, not findings.

C. Limitations

- Error propagation. A diarization error can lead to a task being assigned to the wrong person, and there is no direct mechanism for the extraction stage to detect this. Evidence spans and human confirmation help but do not solve the problem.
- Implicit information. Commitments are often indirect ("I can probably look into that") and the line between a task and a casual remark is inherently subjective, which will limit achievable agreement even among human annotators.
- Faithfulness of LLM output. Generative models are able to generate fluent but unsupported statements. This risk is mitigated by the null-field policy and the evidence requirement, but human review is still required for consequential decisions.
- Domain shift. Public corpora have particular meeting styles. The performance on technical, multilingual or noisy meetings may vary.

D. Ethical Considerations Recording and analyzing conversations raises issues of consent and surveillance. Deployment should ensure that all the participants are aware, that the data is securely stored and that access to transcripts and task records is limited. Task tracking needs to foster teamwork and clarity, and should not be used for covert monitoring of individuals.

8. Conclusion and Future Work

MOMent solves a common and costly problem where valuable commitments are made in meetings but lost in the free flow of speech. The system converts a conversation into actionable information that is quickly digestible by integrating transcription, speaker diarization, hierarchical summarization, schema-constrained action extraction, and persistent tracking into a single modular pipeline. It is meant to be traceable back to the source audio, conservative with missing information, and have a human-in-the-loop confirmation step.

Next immediate step is to implement the full pipeline and run the evaluation protocol as specified in Section VI, then filling in the results template and revising the design based on any errors encountered. Future directions include processing live meetings in real time, integration with calendar and project-

management tools, support for additional Indian languages and code- mixed speech, voice-based speaker enrollment for
stable identification across meetings, and detection of decisions and open questions in addition to tasks.

References

1. A. Radford, J. W. Kim, T. Xu, G. Brockman, C. McLeavey, and I. Sutskever, "Robust Speech Recognition via Large-Scale Weak Supervision," arXiv preprint arXiv:2212.04356, 2022.
2. H. Bredin et al., "pyannote.audio: Neural Building Blocks for Speaker Diarization," in Proc. IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP), 2020.
3. M. Lewis et al., "BART: Denoising Sequence-to-Sequence Pre-training for Natural Language Generation, Translation, and Comprehension," in Proc. 58th Annual Meeting of the Association for Computational Linguistics (ACL), pp. 7871– 7880, 2020.
4. H. Bredin and A. Laurent, "End-to-End Speaker Segmentation for Overlap-Aware Resegmentation," in Proc. Interspeech, 2021.
5. J. Carletta et al., "The AMI Meeting Corpus: A Pre-announcement," in Machine Learning for Multimodal Interaction, vol. 3869, pp. 28–39, Springer, 2005.
6. A. Janin et al., "The ICSI Meeting Corpus," in Proc. IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP), 2003.
7. X. Anguera et al., "Speaker Diarization: A Review of Recent Research," IEEE Transactions on Audio, Speech, and Language Processing, vol. 20, no. 2, pp. 356–370, 2012.
8. T. J. Park et al., "A Review of Speaker Diarization: Recent Advances with Deep Learning," Computer Speech & Language, vol. 72, 2022.
9. C.-Y. Lin, "ROUGE: A Package for Automatic Evaluation of Summaries," in Text Summarization Branches Out, pp. 74–81, 2004.
10. T. B. Brown et al., "Language Models are Few-Shot Learners," in Advances in Neural Information Processing Systems (NeurIPS), vol. 33, 2020.
11. B. Gliwa, I. Mochol, M. Biesek, and A. Wawer, "SAMSum Corpus: A Human-Annotated Dialogue Dataset for Abstractive Summarization," in Proc. 2nd Workshop on New Frontiers in Summarization, 2019.
12. M. Zhong et al., "QMSum: A New Benchmark for Query-Based Multi-Domain Meeting Summarization," in Proc. NAACL-HLT, 2021.
13. T. Zhang et al., "BERTScore: Evaluating Text Generation with BERT," in Proc. International Conference on Learning Representations (ICLR), 2020.
14. M. Bain, J. Huh, T. Han, and A. Zisserman, "WhisperX: Time-Accurate Speech Transcription of Long-Form Audio," in Proc. Interspeech, 2023.
15. C. Zhu, R. Xu, M. Zeng, and X. Huang, "A Hierarchical Network for Abstractive Meeting Summarization with Cross-Domain Pretraining," in Findings of EMNLP, 2020.
16. M. Purver, J. Dowding, J. Niekrasz, P. Ehlen, S. Noorbaloochi, and S. Peters, "Detecting and Summarizing Action Items in Multi-Party Dialogue," in Proc. 8th SIGdial Workshop on Discourse and Dialogue, pp. 18–25, 2007.
17. W. Morgan, P.-C. Chang, S. Gupta, and J. M. Brenier, "Automatically Detecting Action Items in Audio Meeting Recordings," in Proc. 7th SIGdial Workshop on Discourse and Dialogue, pp. 96–103, 2006.

18. I. Beltagy, M. E. Peters, and A. Cohan,"Longformer: The Long-Document Transformer," arXiv preprint arXiv:2004.05150, 2020.
19. M. Purver, P. Ehlen, and J. Niekrasz, "Detecting Action Items in Multi-Party Meetings: Annotation and Initial Experiments," in Machine Learning for Multimodal Interaction, pp. 200–211, Springer, 2006.
20. P. Tardy, D. Janiszek, Y. Estève, and V. Nguyen, "Align then Summarize: Automatic Alignment Methods for Summarization Corpus Creation," in Proc. 12th International Conference on Language Resources and Evaluation (LREC), 2020.