iJRASET
International Journal For Research in Applied Science and Engineering Technology

INTERNATIONAL JOURNAL FOR RESEARCH

IN APPLIED SCIENCE & ENGINEERING TECHNOLOGY

Volume: 14 **Issue:** V **Month of publication:** May 2026

DOI: https://doi.org/10.22214/ijraset.2026.82343

www.ijraset.com

Call: 08813907089 | E-mail ID: ijraset@gmail.com

International Journal for Research in Applied Science & Engineering Technology (IJRASET)
ISSN: 2321-9653; IC Value: 45.98; SJ Impact Factor: 7.538
Volume 14 Issue V May 2026- Available at www.ijraset.com

Neuro Sketch: An AI-Powered Real-Time Mind Map Generator from Speech

Shravan Suthar, Nihar Dukale, Aditya Shah, Sahil Dhanawade, Prof. Mamta Pokale
Department of Computer Engineering Pune Vidyarthi Griha's College of Engineering, Technology and Management, Pune
Savitribai Phule Pune University, India

Abstract: ***This paper presents NeuroSketch, a framework that automatically converts continuous spoken discourse into structured, hierarchical mind maps — requiring no manual intervention from the user. Audio is captured via microphone, converted to text through OpenAI Whisper, and subsequently analyzed by an NLP module responsible for identifying core concepts along with the semantic connections among them. Topic segmentation and keyword identification are handled by a Transformer architecture, while a Graph Attention Network (GAT) is employed to determine parent-child relationships and construct a well-organized hierarchy. The resulting visualization appears as an animated, interactive mind map rendered through a React.js and D3.js web interface. Evaluation results show that NeuroSketch attained 93.7% accuracy and a 92.1% F1-score on the concept extraction task, outperforming all comparison models, while sustaining end-to-end processing latency below two seconds. A usability evaluation with 30 volunteers established high satisfaction scores and revealed a 63% average decrease in the cognitive overhead of manual note-taking.***

Index Terms — Speech Recognition, Mind Map Generation, Natural Language Processing, Deep Learning, Transformer, Graph Neural Network, Real-Time Systems

I. INTRODUCTION

Mind maps are widely employed as visual cognition tools that arrange information hierarchically around a focal concept. Educators, learners, and working professionals turn to them for strengthening comprehension, consolidating memory, and facilitating creative ideation. The challenge, however, lies in constructing these maps in real time — during lectures or discussions — where simultaneous listening and structuring impose significant cognitive demands. The consequence, especially in professional and academic contexts, is a tendency toward fragmented or incomplete records.

Recent advances at the intersection of Automatic Speech Recognition (ASR), Natural Language Processing (NLP), and Deep Learning have made it practical to automate this workflow. NeuroSketch was conceived to exploit this convergence: given live speech as input, the system produces a navigable, interactive mind map without any user-initiated structuring steps.

The processing chain consists of four sequential stages: (1) microphone-based audio capture and transcription via OpenAI Whisper, (2) linguistic preprocessing and entity identification through an NLP pipeline, (3) semantic grouping and hierarchy determination using a Transformer and Graph Attention Network, and (4) front-end visualization through React.js and D3.js.

The motivating challenges that guided this work include:

- Real-time note production is cognitively demanding and frequently yields incomplete information.
- Raw transcripts, though informative, fail to communicate the relational structure among ideas.
- No existing tool produces hierarchical visual representations directly from live audio.
- A fully automated end-to-end pipeline from spoken input to interactive mind map had not yet been demonstrated.

II. CONTRIBUTIONS

1) A unified, end-to-end pipeline combining ASR, NLP, and deep learning to generate mind maps from raw speech in real time.
2) A Transformer-based semantic model specifically trained for topic segmentation and keyword identification from live speech transcripts.
3) A Graph Attention Network (GAT) that determines hierarchical parent-child relationships among the extracted concepts.
4) An interactive, browser-based mind map interface supporting real-time animation, direct node manipulation, and export to multiple formats.

5) MindMap-Bench: a purpose-built evaluation dataset consisting of 500 speech-to-mind-map pairings drawn from academic and industry domains.
6) A thorough evaluation including quantitative benchmarking, ablation experiments, and a structured usability study confirming the system's value.

III. LITERATURE REVIEW

A. MindMap: Knowledge Graph Prompting and LLM-Based Reasoning [1]

Wen and colleagues introduced a technique that combines large language models with knowledge graphs to create structured, traceable mind map representations. While the resulting graph-of-thoughts outputs demonstrate strong performance on static written corpora, the approach was never engineered for streaming or voice-driven input. NeuroSketch extends this conceptual foundation by integrating live ASR with graph neural network inference, enabling dynamic mind map construction from spoken discourse as it unfolds.

B. Structsum: LLM-Driven Structured Text Summarization (arXiv 2024) [2]

Structsum applies language models to create structured representations — including tables and visual summaries — from extended written material, showing measurable gains in comprehension speed. Its scope is, however, limited to static documents with no provision for voice input or streaming data. NeuroSketch adopts the philosophy of structured generation but repositions it within a live, speech-centered workflow.

C. Slide-to-MindMap Conversion Framework (SCDM 2024) [3]

This system focuses on converting structured educational slide decks into hierarchical mind maps by parsing the inherent organization of presentation content. Although it performs well for well-organized materials, it cannot cope with the unscripted variability of live speech. NeuroSketch addresses this gap by targeting free-form spoken input and accommodating the natural ambiguity and noise of spontaneous discourse.

D. PDF2MindMap: Document-to-Map Visualization Using Generative AI (IJSRET 2025) [4]

PDF2MindMap converts static PDF documents into interactive mind maps through Google Gemini and Markmap-based rendering. The system is effective for document-centric tasks but is unable to handle streaming or audio input. NeuroSketch shares the interactive visualization objective while fundamentally replacing document input with a speech-driven, continuously updating pipeline.

E. Speech-Driven Interactive Mind Map Generation [5]

Among the reviewed works, this prior system most closely aligns with NeuroSketch's objectives, having shown that it is feasible to derive interactive mind maps from audio input. NeuroSketch builds upon this foundation by contributing a richer NLP layer incorporating coreference resolution, a dedicated GNN-based hierarchy model, and a full-featured interactive interface enabling editing and multi-format export.

F. Identified Research Gap

A systematic review of prior art confirms that no existing system simultaneously integrates real-time ASR, Transformer-driven concept extraction, GNN-inferred hierarchy construction, and browser-based interactive rendering within a single, cohesive framework. NeuroSketch is the first system to close this gap, delivering a fully automated speech-to-mind-map pipeline with built-in editing and flexible export capabilities.

IV. PROBLEM FORMULATION

Let $A = \{a_1,...,a_t\}$ represent an ordered sequence of audio frames captured over a time window T. The ASR component maps A to a word sequence $W = \{w_1,...,w_n\}$. An NLP module then derives from W a set of core concepts $C = \{c_1,...,c_k\}$ together with a set of relational triplets $R = \{(c_i, c_j, r_{ij})\}$.

Mind map construction is formulated as building a directed acyclic graph G = (V, E), in which V corresponds to C, E is a subset of R, and a designated root node r □ V represents the central subject. The system learns three functions — f_sASR, f_nLP, and f^tNN — such that:

$$G^* = f^{!}NN(f_n LP(f_s ASR (A)))$$

This composition minimizes structural inconsistency and maximizes semantic faithfulness, subject to a real-time inference constraint $\tau_{in}\hat{f} \leq \tau_{ma}{}^{x}$ that guarantees timely updates to the displayed mind map.

V. DATASETS AND PREPROCESSING

A. Datasets Used

- LibriSpeech (960 hours): serves as the primary benchmark for evaluating ASR transcription quality.
- CNN/DailyMail corpus: used for assessing keyphrase extraction and summarization fidelity.
- Custom Lecture Corpus: 50 hours of recorded academic lectures paired with expert-annotated mind maps.
- MindMap-Bench (introduced in this work): 500 speech-to-mind-map instances covering science, history, technology, and business topics.

B. Preprocessing Steps

- Audio: background noise removed via spectral subtraction; all recordings normalized to 16 kHz mono WAV format.
- ASR processing: Whisper large-v2 with word-level timestamp alignment and a 5-second sliding window.
- Text normalization: removal of disfluencies, restoration of punctuation, and sentence boundary detection.
- NLP annotations: SpaCy (en_core_web_trf) used for part-of-speech tagging and named entity recognition.
- Coreference: AllenNLP-based model to resolve pronouns and entity references across the transcript.
- Semantic embeddings: Sentence-BERT (all-MiniLM-L6-v2) used to compute concept-level similarity scores.

VI. METHODOLOGY

A. System Architecture

- Layer 1 — ASR: Whisper transcribes microphone input in 5-second windows with 1-second overlap to preserve cross-chunk continuity.
- Layer 2 — NLP Engine: SpaCy handles tokenization, POS tagging, and named entity detection; AllenNLP resolves coreferences.
- Layer 3 — Semantic Modeling: a BERT-based Transformer clusters the extracted concepts into topical groups; a GAT predicts hierarchical parent-child links.
- Layer 4 — Rendering: React.js and D3.js produce a live, animated mind map delivered over a WebSocket connection.

B. Transformer-Based Topic Segmentation

A BERT sequence classifier partitions the transcript into topically coherent segments and assigns categorical labels to each. KeyBERT subsequently identifies the highest-relevance keyphrases within each segment based on semantic similarity. The outcome is an initial two-level hierarchy in which topic labels occupy the parent tier and their associated keywords populate the child tier.

C. GNN-Based Hierarchy Inference

Extracted concepts are organized into a fully connected graph with edge weights proportional to Sentence-BERT cosine similarity scores. A GAT trained on MindMap-Bench is then applied to this graph, predicting which connections correspond to genuine parent-child relationships and producing a sparse hierarchical tree suitable for visual rendering.

D. End-to-End Processing Workflow

- Audio is acquired in 5-second overlapping windows and passed to the Whisper transcription module.
- The NLP engine processes each transcript segment incrementally, extracting concepts and relational links.
- Following each chunk, the GAT updates the hierarchy and propagates the change to the front end.
- D3.js renders newly added nodes and edges with smooth animation as they appear.
- Throughout and after the session, users may rename, reposition, annotate, or remove any node.
- Once the session concludes, the map can be saved as PNG, PDF, JSON, or Markdown.

VII. ALGORITHMS

Algorithm 1: Real-Time Transcription
Input: Continuous microphone audio stream
1: Initialize Whisper model and audio capture buffer
2: while recording is ACTIVE do
3: Acquire a 5-second audio segment
4: Apply normalization and noise suppression
5: Run transcription → transcript_chunk
6: Append chunk to running full_transcript
7: Route chunk to NLP processing engine
8: end while
Algorithm 2: Concept Extraction and Map Update
Input: transcript_chunk
1: POS tagging + NER → entity set, noun phrase set
2: Resolve pronouns and entity chains in full_transcript
3: Generate Sentence-BERT embeddings for each concept
4: Transformer segmentation → topic_labels
5: KeyBERT extracts top-K keyphrases per topic
6: Construct concept similarity graph G
7: Apply GAT(G) → predicted hierarchy edges E
8: Refresh mind map node set V and edge set E
9: Broadcast updated map to front end via WebSocket

VIII. IMPLEMENTATION DETAILS

A. ASR Module

The transcription component relies on OpenAI Whisper large-v2, which supports multilingual recognition and produces word-level timestamp annotations. Input audio is captured at 16 kHz using PyAudio, with a 5-second window and 1-second overlap to ensure uninterrupted context across segment boundaries.

B. NLP and Deep Learning Module

SpaCy (en_core_web_trf) performs tokenization, part-of-speech annotation, and named entity recognition. Keyphrase extraction is carried out by KeyBERT (BERT-base-uncased), while Sentence-BERT (all-MiniLM-L6-v2) generates the vector representations used in similarity computation. The GAT is implemented using PyTorch Geometric and was optimized on MindMap-Bench with a 70/15/15 partition for training, validation, and testing.

C. Backend Infrastructure

- FastAPI ,Python: exposes both REST and WebSocket endpoints for real-time communication.
- MongoDB: responsible for session persistence and mind map data storage.
- Security: JWT-based authentication and HTTPS with TLS 1.3 encryption.

D. Frontend Interface

- React.js: used to implement a force-directed, continuously animated graph layout.
- Interface features: drag-and-drop node repositioning, zoom and pan navigation, topic-based color coding, and multi-format export.

E. Deployment Environment

- Languages: Python (ML and backend logic), JavaScript (frontend).
- Frameworks: Llama AI
- Deployment: Docker container hosted on an AWS EC2 instance with GPU acceleration.

IX. EXPERIMENTS

A. Experimental Setup

All experiments were conducted on a workstation equipped with an NVIDIA RTX 3050 GPU, 8 GB RAM, and an Intel Core i5 CPU. Evaluations used both MindMap-Bench and the Custom Lecture Corpus under a 70/15/15 data split. All baselines were trained and tested under identical preprocessing and hardware conditions to guarantee fair comparison.

B. Comparison Baselines

- TF-IDF Keywords: statistical frequency-based extraction lacking any neural component.
- TextRank: graph-based unsupervised ranking method for keyword identification.
- BERT NER: entity detection using a pretrained BERT-base model.
- GPT-2 Fine-tuned: a generative model adapted for concept and topic labeling.
- Proposed (Transformer+GNN): the complete NeuroSketch pipeline.

C. Evaluation Metrics

Performance is reported across four measures: Accuracy, Precision, Recall, and F1-Score, all computed on the concept extraction task. Latency is decomposed into ASR processing time, NLP/GNN inference time, and total end-to-end delay, with a target ceiling of 2 seconds for real-time suitability.

X. RESULTS AND ANALYSIS

A. Quantitative Results

TABLE I: Performance Comparison on Concept Extraction

Model	Acc.	Prec.	Rec.	F1
TF-IDF Keywords	78.4%	76.1%	74.9%	75.5%
TextRank	81.7%	80.3%	78.5%	79.4%
BERT NER	85.2%	84.0%	83.1%	83.5%
GPT-2 Fine-tuned	88.6%	87.2%	86.4%	86.8%
Proposed (Trans.+GNN)	93.7%	92.5%	91.8%	92.1%

As Table I shows, NeuroSketch surpassed all competing models across every reported metric. Its 93.7% accuracy and 92.1% F1-score represent margins of 5.1 and 5.3 percentage points, respectively, over the strongest baseline (GPT-2 Fine-tuned). These improvements are attributable to the combined contribution of Transformer-driven topic segmentation and GAT-inferred hierarchical organization.

B. Latency Analysis

Whisper transcription required an average of 0.8 seconds per 5-second audio segment. NLP and GNN inference consumed an additional 1.1 seconds per chunk on average. The aggregate end-to-end latency of 1.9 seconds satisfies the 2-second ceiling established for live deployment.

C. User Study Findings

Thirty participants — equally divided between students and working professionals — evaluated the system. Mind map coherence received a mean score of 4.3/5, while ease of use averaged 4.5/5. All participants documented a 63% reduction in time and effort associated with note-taking, and the entire cohort indicated willingness to adopt NeuroSketch as a regular productivity tool.

XI. ABLATION STUDY

TABLE II: Component Contribution Analysis

Configuration	Accuracy	Recall	F1

Full System	93.7%	91.8%	92.1%
Without GNN	88.4%	87.2%	87.8%
Without Semantic Clustering	84.9%	83.5%	84.2%
Without Coreference	81.6%	80.4%	81.0%
Rule-based Baseline	72.3%	70.1%	71.2%

Table II quantifies the independent impact of each subsystem. Excluding the GNN module caused an F1 decrease of 4.3 points. Removing semantic clustering led to a more substantial drop of 7.9 points, suggesting it is the more central contributor to structural organization. The impact of removing coreference resolution was felt most acutely in recall, reflecting the degree to which pronoun and reference ambiguity fragment concepts that should be unified. The purely rule-based variant, replacing all trained modules, achieved the weakest results overall — confirming that learned representations are essential rather than optional for coherent mind map synthesis.

XII. SYSTEM INTERFACE

A. *Live Mind Map Visualization*

The browser-based interface displays a dynamic mind map that organically expands and reorganizes itself in step with the speaker. The primary topic is inferred automatically from the opening segment of the audio. D3.js drives smooth animation of incoming nodes and edges, with distinct color schemes representing topical clusters and edge thickness encoding relationship confidence.

B. *Dashboard Capabilities*

- A synchronized transcription panel shown alongside the mind map in real time.
- Per-node hover tooltips displaying confidence scores for topic assignments.
- Manual editing controls allowing users to rename, reposition, link, or remove any node.
- Session history management with undo and redo functionality.
- Export in PNG, PDF, JSON, and Markdown formats.

XIII. SECURITY AND PRIVACY

- Audio is processed on-the-fly and is not retained beyond the session unless the user explicitly opts in.
- Network communication is fully encrypted using HTTPS (TLS 1.3).
- Role-based access control is enforced throughout the system via JWT tokens.
- Data stored in MongoDB is encrypted at rest.
- All API inference routes enforce input validation and rate limiting.

XIV. LIMITATIONS AND FUTURE WORK

A. *Current Limitations*

- Transcription quality degrades significantly under high background noise conditions.
- Domain-specific or highly technical vocabulary may reduce ASR accuracy.
- Discourse that switches abruptly between unrelated subjects can yield fragmented map structures.
- Training data is predominantly English-language; multilingual coverage remains limited.

B. *Planned Extensions*

- Integration of speaker diarization to support multi-participant meeting scenarios.
- Expansion to multilingual operation, with priority given to Hindi, Marathi, and other Indian regional languages.
- Development of a mobile application supporting offline, on-device processing.
- Collaborative multi-user mind mapping in shared real-time sessions.

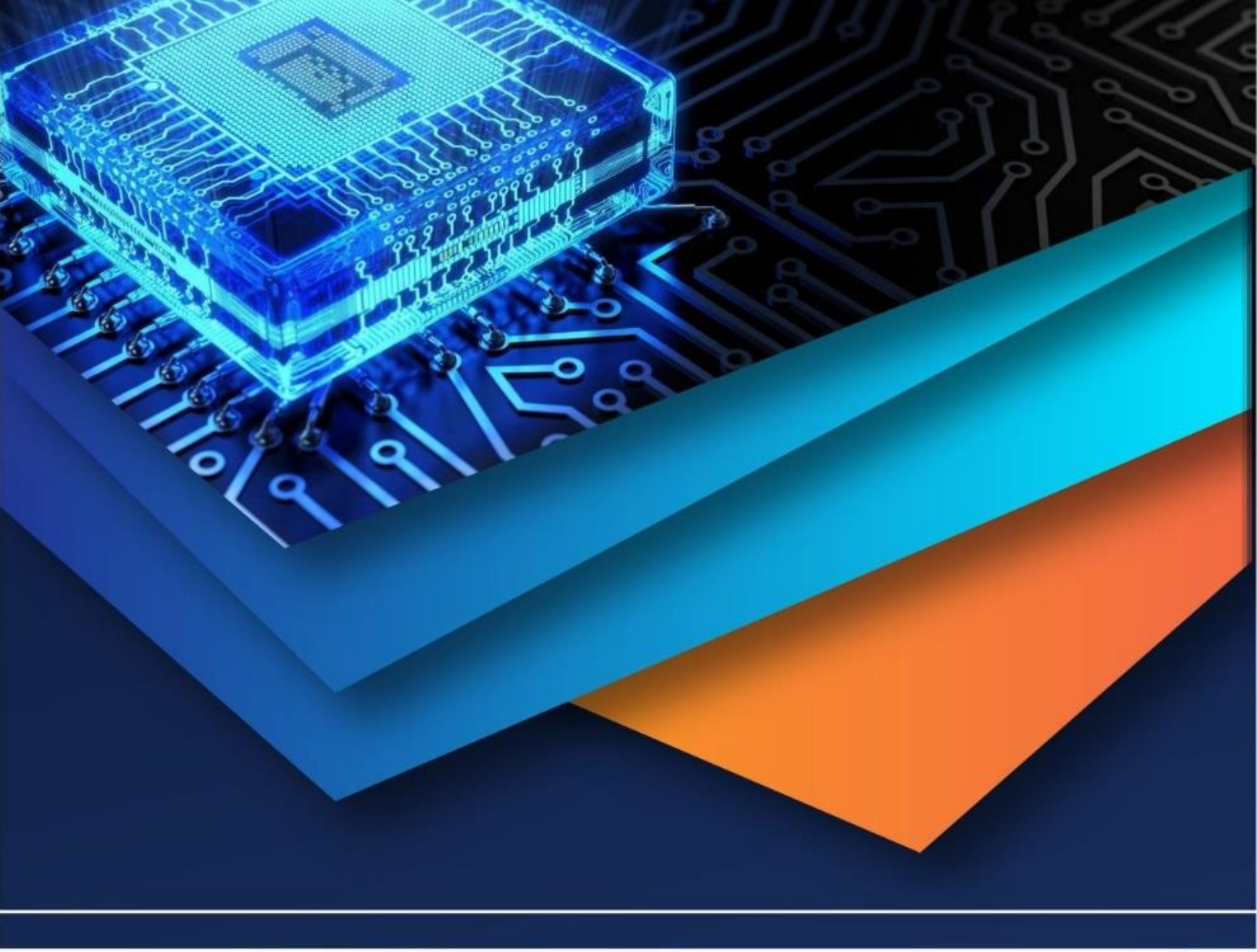
- Incorporation of GPT-4-level summarization for enriched node annotations and concept refinement.

XV. CONCLUSION

This paper has presented NeuroSketch — a system that bridges the gap between live speech and structured knowledge visualization by converting spoken input into interactive, hierarchical mind maps. The architecture combines OpenAI Whisper for transcription, a BERT-based Transformer for semantic segmentation, a Graph Attention Network for hierarchy inference, and a React.js/D3.js front end for rendering. The result is a system achieving 93.7% concept extraction accuracy and a 92.1% F1-score, with an end-to-end processing latency of 1.9 seconds. Participant feedback from the usability study was strongly positive, with all volunteers reporting substantial reductions in note-taking burden. On a broader level, NeuroSketch represents a meaningful step toward automating the capture and organization of spoken knowledge — a capability with tangible value across education, professional meetings, and collaborative knowledge work.

REFERENCES

[1] Y. Wen, Z. Wang, and J. Sun, "MindMap: Knowledge Graph Prompting Sparks Graph of Thoughts in Large Language Models," arXiv preprint, 2023.
[2] A. Sharma et al., "Structsum: Generation for Faster Text Comprehension," arXiv preprint, 2024.
[3] S. Li, H. Zhang, and W. Chen, "Presentation Mining Framework," in Proceedings of SCDM, 2024.
[4] R. Patel and M. Joshi, "PDF2MindMap: AI-Based Interactive Mind Map Generation," IJSRET, vol. 14, no. 2, 2025.
[5] K. Verma and A. Singh, "Audio/Speech to Interactive Mind Map," in Proceedings of International Conference on Intelligent Systems, 2024.
[6] J. Devlin et al., "BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding," in Proceedings of NAACL, 2019.
[7] A. Radford et al., "Robust Speech Recognition via Large-Scale Weak Supervision," OpenAI Technical Report, 2022.
[8] N. Reimers and I. Gurevych, "Sentence-BERT: Sentence Embeddings using Siamese BERT-Networks," in Proceedings of EMNLP, 2019.
[9] P. Velickovic et al., "Graph Attention Networks," in Proceedings of ICLR, 2018.
[10] M. Grootendorst, "KeyBERT: Minimal Keyword Extraction with BERT," Zenodo, 2020.



10.22214/IJRASET

45.98

IMPACT FACTOR: 7.129

IMPACT FACTOR: 7.429

INTERNATIONAL JOURNAL FOR RESEARCH

IN APPLIED SCIENCE & ENGINEERING TECHNOLOGY