Published by :
https://www.ijert.org/
An International Peer-Reviewed Journal

International Journal of Engineering Research & Technology (IJERT)
ISSN: 2278-0181
Vol. 15 Issue 05 , May - 2026

Automatic Podcast Summarization and Topic Classification Using ASR and Transformer-Based Models

Gayatri Nandkumar Panchal
Department of Data Science
UMIT SNDT Women's University
Mumbai-49, India
Gayatripanchal.1204gmail.com
\

Gitanjali Sanjay Salunke
Department of Data Science
UMIT SNDT Women's University
Mumbai-49, India
gitanjalisalunke18@gmail.com

Sujata Hanmanlu Sunkewar
Department of Data Science
UMIT SNDT Women's University
Mumbai-49, India
sujatasunkewar2204@gmail.com

Prof. Mohan Bond
Department of Data Science
UMIT SNDT Women's University
Mumbai-49, India

***Abstract*—Podcasts have emerged as a widely adopted medium for information dissemination across domains such as technology, education, health, and entertainment. However, their extended duration and unstructured conversational format make efficient content discovery challenging. This paper presents an automated Podcast Summarization and Topic Classification system that integrates Automatic Speech Recognition (ASR) and transformer-based Natural Language Processing (NLP) models to convert raw audio into structured textual insights. The proposed framework employs Whisper ASR for accurate speech-to-text transcription, followed by preprocessing techniques including tokenization, stop-word removal, and normalization. Abstractive summarization is performed using the BART transformer model to generate concise and semantically coherent summaries. Topic classification is achieved using supervised machine learning algorithms, including Logistic Regression, Decision Tree, Random Forest, and contextual modeling BERT. Experimental evaluation demonstrates reliable transcription accuracy and improved classification performance across multiple podcast genres. The system significantly reduces listening time, enhances accessibility, and improves content discoverability. The modular architecture ensures scalability and provides a practical solution for intelligent audio content analysis in academic and real-world applications.**

Keywords*—Podcast Summarization, Automatic Speech Recognition, Transformer Models, Topic Classification, Natural Language Processing, Machine Learning.

I. INTRODUCTION

Podcasts have emerged as one of the fastest-growing digital platforms for knowledge sharing and entertainment, offering on-demand audio content across domains such as technology, education, health, business, and news. Their flexibility and accessibility have contributed to widespread adoption. However, podcast episodes are often long and unstructured, typically ranging from 30 minutes to several hours, making it difficult for users to quickly determine their relevance. Most platforms provide only brief titles and descriptions, which often fail to capture the full context of the discussion, leading to challenges in content discovery and user engagement.

With the rapid growth of podcast content, users increasingly face issues such as information overload, inefficient browsing, and limited accessibility. These challenges are especially significant for time-constrained individuals and non-native language speakers who prefer concise textual summaries over lengthy audio formats. As a result, there is a strong need for automated systems that can convert lengthy audio content into structured, easy-to-understand summaries, enabling users to quickly grasp key insights without listening to entire episodes.

Recent advancements in Artificial Intelligence (AI), Automatic Speech Recognition (ASR), and Natural Language Processing (NLP) have enabled the development of intelligent systems for audio analysis. Transformer-based models such as BERT and BART have shown strong performance in contextual understanding and abstractive summarization. This paper proposes an AI-based Podcast Summarization and Topic Classification system that transcribes audio using ASR, preprocesses text using NLP techniques, generates concise summaries, and classifies content into predefined categories. The modular framework improves accessibility, reduces listening time, and enhances podcast discoverability through efficient and scalable content analysis.

II. LITERATURE SURVEY

Several research studies have explored the application of Artificial Intelligence (AI) and Natural Language Processing (NLP) techniques for podcast transcription and summarization. Early approaches to text summarization mainly relied on extractive methods, where key sentences were selected based on statistical features such as word frequency and sentence position. While these methods were computationally efficient, they often lacked semantic

International Journal of Engineering Research & Technology (IJERT)
ISSN: 2278-0181
Vol. 15 Issue 05 , May - 2026

coherence and failed to capture the contextual meaning of conversational audio. As a result, the generated summaries were often fragmented and less informative for users.

With the advancement of deep learning, transformer-based architectures such as BERT, BART, T5, and PEGASUS have become dominant in abstractive summarization tasks. These models significantly improve contextual understanding and generate more human-like summaries compared to traditional extractive techniques. Studies such as Zhang et al. (2020) demonstrated the effectiveness of PEGASUS in handling long-form content, achieving state-of-the-art performance in summarization benchmarks. However, these models require large computational resources and are not always optimized for real-time podcast summarization.

To address transcription challenges, Automatic Speech Recognition (ASR) systems such as Whisper have been widely adopted to convert raw audio into text. These systems support multilingual transcription and handle diverse accents with improved accuracy. Research by Jones and Soboroff (2022) evaluated summarization quality using metrics like ROUGE and BERT Score, highlighting improvements in linguistic quality but also identifying limitations such as reliance on transcript accuracy and lack of real-time capabilities. Similarly, systems like Podcast Summarizer using Machine Learning (Islam et al., 2025) integrate ASR with models like BERT and PEGASUS, but their performance is affected by noisy audio and domain-specific terminology.

Recent studies focus on integrating summarization with topic classification and keyword extraction using machine learning techniques such as Logistic Regression, Support Vector Machines, and Random Forest, along with contextual embeddings like BERT. Frameworks such as AI-Based Podcast Summarizer and Keyword Extractor (Priyanka et al., 2025) combine Whisper-based ASR with transformer models like BART and T5, and utilize TF-IDF, Key BERT, and Named Entity Recognition (NER) for improved keyword extraction. Other systems like PodSumm (Vartak et al., 2021) and query-based summarization approaches (Spina et al., 2016) demonstrate the usefulness of ASR-integrated summarization but remain limited by extractive techniques and lack of deep contextual understanding. Overall, despite significant progress, most existing solutions lack a unified and scalable architecture that integrates transcription, summarization, and classification, highlighting a research gap addressed by the proposed system.

III. System Architecture

The overall architecture of the proposed Podcast Summary and Topic Tracker system is designed to provide an end-to-end automated solution for podcast transcription, summarization, and topic classification. The system follows a modular architecture, ensuring scalability, maintainability, and seamless integration with external APIs and machine learning models. This design enables efficient handling of long-form audio content while supporting future enhancements such as real-time processing and multilingual capabilities.

The workflow of the system begins with the user uploading a podcast audio file through a web-based interface or retrieving podcast data via external APIs. The uploaded audio is processed by the backend server, where preprocessing techniques such as format conversion and noise handling are applied using tools like FFmpeg. The processed audio is then passed to the Whisper Automatic Speech Recognition (ASR) model, which performs speech-to-text conversion and generates a structured transcript of the podcast. The transcript may also include timestamp alignment to support segment-level analysis.

Once the transcript is generated, the system performs text preprocessing, including tokenization, stop-word removal, and normalization using NLP libraries such as spaCy. The cleaned transcript is then passed to the summarization module, where both extractive and transformer-based approaches are applied. Models such as BERT and BART are used to generate concise, coherent, and context-aware summaries that capture the key insights of the podcast while reducing redundancy and preserving semantic meaning.

To further enhance content understanding, the system performs topic classification using machine learning algorithms such as Logistic Regression, Decision Tree, Random Forest, and BERT-based classifiers. Feature extraction techniques such as TF-IDF are used to convert textual data into numerical representations for accurate prediction. The system categorizes podcasts into predefined domains such as Technology, Health, Education, Business, and Entertainment, improving content discoverability and organization.

Finally, the processed outputs, including transcripts, generated summaries, and predicted topic labels, are stored in a structured database such as SQLite or CSV for efficient retrieval and historical tracking. The results are displayed on an interactive frontend dashboard, allowing users to view, analyse, and download summarized content. This integrated architecture enhances accessibility, reduces listening time, and provides an intelligent and efficient solution for podcast content analysis.

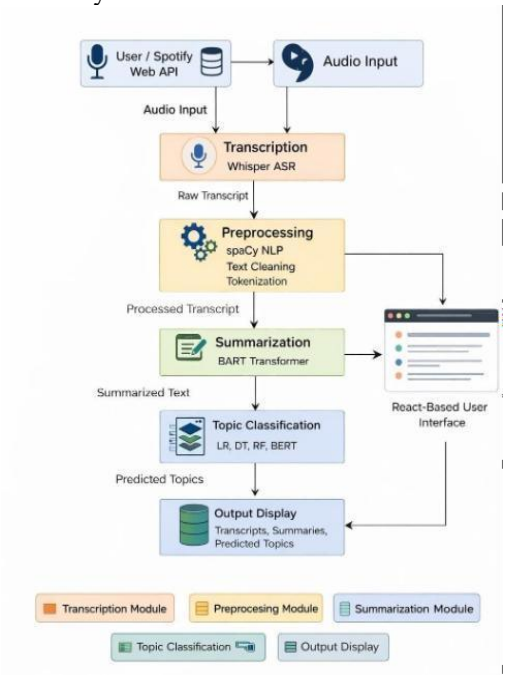

Fig 1. Architecture of the podcast summarization and topic classification

International Journal of Engineering Research & Technology (IJERT)
ISSN: 2278-0181
Vol. 15 Issue 05 , May - 2026

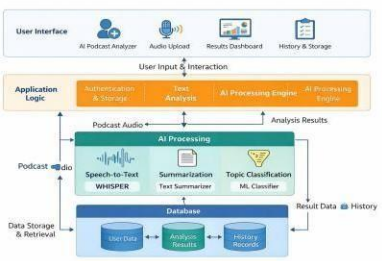

Fig 2. System Workflow of podcast summary and topic tracker system

IV. METHODOLOGY

A. Speech-to-Text Conversion Using Whisper ASR

The proposed system employs the Whisper Automatic Speech Recognition (ASR) model to convert podcast audio into textual format. Prior to transcription, the input audio is preprocessed using FFmpeg for format conversion, normalization, and noise handling to ensure compatibility with the model. Whisper is capable of handling long-form audio, diverse accents, and conversational speech patterns, making it suitable for podcast transcription. The model generates a structured transcript that accurately represents the spoken content and serves as the foundational input for subsequent Natural Language Processing (NLP) tasks.

B. Text Preprocessing and Feature Extraction

The generated transcript undergoes several preprocessing steps to improve data quality and consistency. These steps include conversion to lowercase, removal of punctuation and special characters, tokenization, sentence segmentation, and lemmatization using NLP libraries such as spaCy. After preprocessing, the textual data is transformed into numerical representations using the TF-IDF (Term Frequency–Inverse Document Frequency) technique.

$$TF\text{-}IDF(t,d) = TF(t,d) \times \log\left(\frac{N}{DF(t)}\right)$$

where $TF(t,d)$represents the frequency of term $t$in document d, $DF(t)$ denotes the number of documents containing the term, and $N$is the total number of documents. Additionally, n-grams (unigrams and bigrams) are incorporated to capture contextual and phrase-level information, improving model performance in downstream tasks.

C. Podcast Summarization Using Hybrid Approach

The system adopts a hybrid summarization approach that combines extractive and transformer-based techniques. Extractive summarization identifies and selects important sentences from the transcript based on statistical relevance and positional importance. To enhance contextual understanding and coherence, transformer-based models such as BERT and BART are used for abstractive summarization. These models generate human-like summaries by capturing semantic relationships within the text. For efficient processing of long podcast transcripts, the text is segmented into smaller chunks before summarization. The final output is a concise and meaningful summary that preserves the key insights of the podcast.

D. Topic Classification Using Machine Learning

To categorize podcast content into predefined domains, the system utilizes supervised machine learning algorithms such as Logistic Regression, Decision Tree, Random Forest, and BERT-based classifiers. The TF-IDF feature vectors serve as input to these models, enabling them to learn patterns associated with different topics. During prediction, the system assigns the most relevant category, such as Technology, Health, Education, Business, or Entertainment, to each podcast episode. This classification improves content organization and enhances discoverability for users.

E. System Integration and Output

All system components are integrated using FastAPI, which enables efficient communication between the backend processing modules and the React-based frontend interface. The system follows a pipeline architecture where audio input is sequentially processed through transcription, preprocessing, summarization, and classification stages. The final outputs, including the transcript, generated summary, and predicted topic label, are stored in a structured database such as SQLite or CSV. These results are then displayed on an interactive dashboard, allowing users to easily access, analyze, and download the processed information. This integrated approach ensures scalability, efficiency, and an improved user experience.

V. SYSTEM IMPLEMENTATION

The proposed Podcast Summary and Topic Tracker system is implemented as a web-based platform to ensure accessibility, scalability, and efficient processing of podcast content. The system integrates deep learning models, Natural Language Processing (NLP) techniques, machine learning algorithms, and database management into a unified architecture. The implementation focuses on seamless interaction between the frontend interface and backend processing modules, enabling automated transcription, summarization, and topic classification of podcast audio data.

A. Backend Framework

The backend of the system is developed using the FastAPI framework, which provides a high-performance and scalable environment for handling HTTP requests, audio uploads, and model inference. FastAPI enables efficient communication between frontend and backend through RESTful APIs.

The Whisper Automatic Speech Recognition (ASR) model is deployed on the server side to convert podcast audio into textual transcripts. The uploaded audio is first preprocessed using FFmpeg to ensure format compatibility and improved quality. The backend further performs text preprocessing, summarization using transformer models such as BERT and BART, and topic classification using machine learning algorithms. The modular design ensures flexibility and efficient execution of each processing stage.

B. Database Management

The system utilizes SQLite and CSV-based storage for efficient and lightweight data management. The database stores podcast metadata, transcripts, generated summaries, and predicted topic labels in a structured format.

The database schema is organized into logical components such as podcast information, transcript data, summary outputs, and classification results. This structure supports

International Journal of Engineering Research & Technology (IJERT)
ISSN: 2278-0181
Vol. 15 Issue 05 , May - 2026

efficient querying, historical tracking, and easy integration with machine learning workflows. The lightweight design ensures fast performance while allowing future scalability to advanced database systems.

C. API Integration and Processing Pipeline

The system integrates multiple processing modules using FastAPI-based REST APIs, ensuring smooth data flow across all components. The workflow begins with audio upload, followed by transcription using Whisper ASR. The transcript is then preprocessed using NLP techniques such as tokenization and lemmatization.

Feature extraction is performed using TF-IDF, and the processed data is passed to the summarization module and classification module. Extractive and transformer-based methods generate summaries, while machine learning models predict the podcast topic. Asynchronous processing is implemented to efficiently handle long audio inputs and improve system responsiveness.

D. Machine Learning and NLP Modules

The system incorporates advanced NLP and machine learning models to perform summarization and topic classification. Transformer-based models such as BERT and BART are used to generate context-aware and coherent summaries. For classification, supervised learning models including Logistic Regression, Decision Tree, Random Forest, and BERT-based classifiers are employed.

Feature extraction techniques such as TF-IDF and n-grams are used to represent textual data numerically. These models are trained on labeled datasets and deployed in the backend to perform real-time predictions with improved accuracy and reliability.

E. User Interface

The frontend of the system is developed using React.js and TypeScript, providing a modern, responsive, and user-friendly interface. The dashboard allows users to upload podcast audio files, view transcripts, analyze summaries, and explore topic classifications.

Key features include transcript visualization, summary display, topic categorization, and history tracking. The frontend communicates with the backend through API calls and presents results in a structured and visually intuitive manner, enhancing overall user experience and accessibility.

V. Result and Analysis

The proposed Podcast Summary and Topic Tracker system successfully converts long podcast audio into structured transcripts, concise summaries, and categorized topic insights. The system was tested on podcasts from domains such as Health and Education, demonstrating effective performance in transcription, summarization, and topic classification.

The Whisper ASR model accurately generated textual transcripts while preserving key contextual information. The hybrid summarization module effectively reduced lengthy podcast content into meaningful key points and AI-generated insights, significantly improving content accessibility and reducing listening time. Using TF-IDF with machine learning models such as Logistic Regression, Random Forest, and BERT, the topic classification module accurately categorized podcasts into relevant domains,

Using TF-IDF with machine learning models such as Logistic Regression, Random Forest, and BERT, the topic classification module accurately categorized podcasts into relevant domains, improving discoverability and user understanding. AI functionalities such as Download, Read Aloud, Save, Share, and Translation.

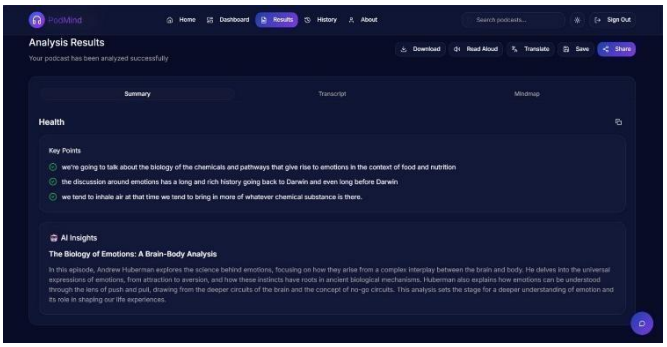

Fig 3. Generate Podcast Summary And Ai Insights

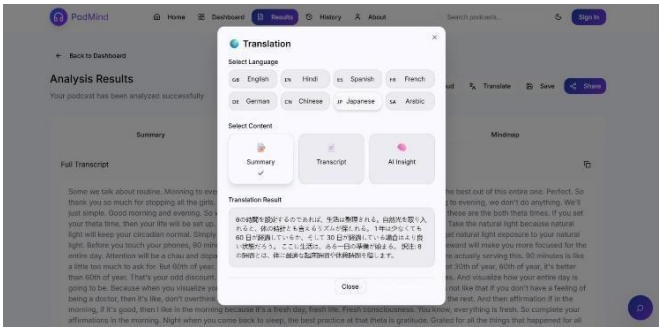

Fig 4. Multilingual Translation Output

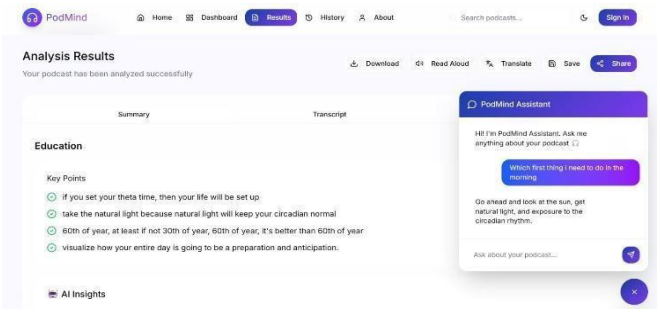

Fig 5. AI Chatbot Interaction

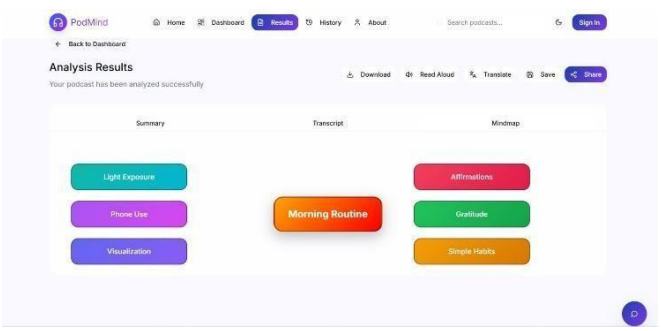

Fig 6. Generated Podcast Mindmap

International Journal of Engineering Research & Technology (IJERT)
ISSN: 2278-0181
Vol. 15 Issue 05 , May - 2026

Additional features such as multilingual translation, mindmap visualization, read-aloud, and AI chatbot assistance enhanced user interaction and accessibility. Overall, the PodMind system effectively integrates ASR, NLP, summarization, and topic classification into a unified framework, demonstrating strong potential for intelligent podcast analysis and content discover.

VI. Conclusion

This paper presented PodMind, an AI-powered Podcast Summary and Topic Tracker system that integrates Automatic Speech Recognition (ASR), Natural Language Processing (NLP), transformer-based summarization, and machine learning into a unified platform. The proposed system effectively automates podcast transcription, summarization, and topic classification, addressing the limitations of lengthy and unstructured podcast content. By utilizing the Whisper ASR model, the system accurately converts podcast audio into transcripts, while hybrid summarization techniques generate concise and meaningful summaries that improve accessibility and reduce listening time.

The system is implemented as a web-based platform using React.js, FastAPI, and SQLite, ensuring scalability and practical usability. Machine learning–based topic classification accurately categorizes podcasts into relevant domains, while additional features such as multilingual translation, mindmap visualization, and AI chatbot integration enhance user interaction and content discoverability. Experimental observations indicate that PodMind delivers reliable performance in podcast analysis, making it a practical solution for intelligent audio content understanding in academic, educational, and real-world applications.

VII. Conclusion

The proposed Podcast Summary and Topic Tracker system demonstrates effective performance in transcription, summarization, and topic identification; however, several enhancements can further improve its capabilities. Future work may include enabling multilingual support to process podcasts in diverse languages using advanced ASR and transformer-based models. Real-time processing can be introduced to support live podcast analysis and streaming content. Additionally, integrating deep learning-based topic modeling techniques may enhance thematic clustering and trend detection across large datasets. The incorporation of sentiment analysis can provide deeper insights into the emotional tone of discussions. Furthermore, personalized recommendation systems based on extracted topics can improve user engagement. From a deployment perspective, future implementations may leverage cloud-based infrastructure and optimized lightweight models to support large-scale processing with reduced latency, thereby transforming the system into a comprehensive intelligent podcast analytics platform suitable for academic and industry applications.

ACKNOWLEDGMENT

The authors would like to express their sincere gratitude to the project guide for their continuous support, valuable guidance, and constructive feedback throughout this research work. Their expertise and insights significantly contributed to the successful development of the proposed system. The authors also extend their appreciation to the faculty members and the department for providing essential academic support and technical resources. The availability of institutional infrastructure played a key role in the implementation and validation of the system.

Furthermore, the authors would like to thank all individuals who contributed podcast audio samples for testing and evaluation, enabling practical system development. Special thanks are extended to peers and colleagues for their encouragement and assistance during the research process. Finally, the authors acknowledge the support and motivation received from family and well-wishers, which contributed to the successful completion of this work.

REFERENCES

[1] L. Islam, A. Ahmed, A. Furqan, U. Mulla, and S. Awez, “Podcast summarizer using machine learning,” Dept. Comput. Eng., M. H. Saboo Siddik College of Eng., Mumbai, India.

[2] A. Vartakavi, A. Garg, and Z. Rafii, “Audio summarization for podcasts,” Gracenote, Emeryville, CA, USA.

[3] M. Vashisht, “Podcast insights: Transcription and summarization,”Maharshi Dayanand Univ., Rohtak, Haryana, India.

[4] K. Song, C. Li, X. Wang, D. Yu, and F. Liu, “Automatic summarization of open-domain podcast episodes,” Comput. Sci. Dept., Univ. Central Florida and Tencent AI Lab, Bellevue, WA, USA.

[5] P. V. Kashid, A. Chourashiya, D. Ugalmugale,H. Dalvi, and P. Dumbare,“Podcast transcription and summarization with speech synthesis,”Dept. Inf. Technol., Sir Visvesvaraya Inst. Technol., Nashik, Maharashtra, India.

[6] R. Rezapour, S. Reddy, R. Jones, and I. Soboroff, “What makes a good podcast summary?” in Proc. 45th Int. ACM SIGIR Conf. Res. Develop. Inf. Retrieval (SIGIR), Madrid, Spain, July 2022.

[7] K. E. K. and D. Viswanath, “The rise of podcasting and its influence on communication patterns,” Dept. Vocational Studies, St. Mary’s College, Thrissur, India.

[8] S. Reddy, M. Lazarova, Y. Yu, and R. Jones, “Modeling language usage and listener engagement in podcasts.”

[9] C. Zheng, H. J. Wang, K. Zhang, and L. Fan, “A baseline analysis for podcast abstractive summarization,” in Proc. PodRecs: Workshop on Podcast Recommendations, Sept. 25, 2020.

[10] D. Spina, J. R. Trippas, L. Cavedon, and M. Sanderson,“Extracting audio summaries to support effective spoken document search,” J. Assoc. Inf. Sci. Technol., Sept. 16, 2017.

[11] N. Garg, B. Favre, K. Reidhammer, and D. Hakkani-Tür, ClusterRank: A graph-based method for meeting summarization,” in Proc. Tenth Annu. Conf. Int. Speech Commun. Assoc., Feb. 18, 2009.

[12] Y. Keneshloo, T. Shi, N. Ramakrishnan, and C. K. Reddy, “Deep reinforcement learning for sequence-to-sequence models,” IEEE Trans. Neural Netw. Learn. Syst., vol. 31, July 7, 2019.

[13] J. Devlin, M.-W. Chang, K. Lee, and K. Toutanova, “BERT: Pre-training of deep bidirectional transformers for language understanding,” in Proc. 17th Annu. Conf. North Amer. ChapterAssoc. Comput. Linguistics: Hum. Lang. Technol.,Minneapolis, MN, USA, June 2, 2019

[14] C.-Y. Lin, “ROUGE: A package for automatic evaluation of summaries,” in Proc. Workshop Text Summarization Branches Out, Barcelona, Spain, July 26, 2004.