國立臺灣大學管理學院資訊管理研究所

碩士論文

Department of Information Management

College of Management

National Taiwan University

Master's Thesis

大型語言模型在語音到文字摘要生成任務中的評估: 基於會議語料庫的研究

Evaluation of LLM on Verbal to Text Summarization: A Study Using the Meeting Corpus

邱語謙

Yu-Chien Chiu

指導教授: 曹承礎博士

Advisor: Seng-Cho Chou, Ph.D.

中華民國 113 年 7 月

July, 2024

doi:10.6342/NTU202402787

國立臺灣大學碩士學位論文
口試委員會審定書

（論文中文題目：大型語言模型在語音到文字摘要生成任務中的評估: 基於會議語料庫的研究）

（論文英文題目：Evaluation of LLM on Verbal to Text Summarization: A Study Using the Meeting Corpus）

本論文係邱語謙君（學號 r11725016）在國立臺灣大學資訊管理學系、所完成之碩士學位論文，於民國 113 年 7 月 25 日承下列考試委員審查通過及口試及格，特此證明

口試委員：

所　　長：

doi:10.6342/NTU202402787

誌謝

碩士兩年的時光一下子就過了，首先要感謝曹承礎老師的細心指導，每當研究卡關時找老師討論，總能給出很多方面的指點，讓我在做研究時沒有那麼迷茫。也要謝謝口試委員陳建錦老師與杜志挺老師對論文的建議，讓我的論文更加完整。

感謝實驗室的同學們：庭秀、佳真、佳莉、立倫、冠宇、柏言。雖然大家很少同時待在實驗室，但仍然很開心有你們的陪伴。感謝實驗室的學弟妹們主動邀約我們這些碩二生一起吃飯、唱歌，讓我們在寫論文的日子裡能夠有所放鬆。還有其他好友以及實習生夥伴們，時常傾聽我不論是論文或是找工作時的煩惱，是你們給於我很大的支持跟鼓勵。在無數的夜晚 Seventeen 總帶給我很多歡笑，讓我獨處時不再無聊，七月的克拉島是我完成論文的一大動力。

最後是家人們，謝謝爸爸、媽媽、姑姑、叔叔還有阿姨在這段日子的關心，提醒我要準時吃飯跟早點睡覺，還有謝謝妹妹幫我處理很多事情，讓我能更專心的做研究。這兩年在台大的日子過得很充實且開心，一路上也有許多人的幫助跟陪伴，是各位讓我有繼續前進的動力，未來我會更加努力的!

doi:10.6342/NTU202402787

摘要

隨著現代企業和組織日益依賴會議來進行溝通和決策，會議記錄和摘要的生成變得尤為重要。手動撰寫會議記錄和摘要過程繁瑣且容易出錯，並且品質往往依賴於紀錄者的能力，因此過去便有相關研究使用語言模型進行會議摘要。而近年來，大型語言模型崛起，如 GPT、Gemini 和 Llama 系列，目前已經有先關研究證明大型語言模型在自然語言處理相關任務如: 文本生成、翻譯、問答系統等有優異的表現。會議記錄文本通常較長且結構複雜，包含多個發言者的對話，涉及話題廣泛，導致會議摘要的生成與一般文章摘要有所不同。然而目前針對使用大型語言模型生成會議摘要的研究仍然相對較少，因此本研究旨在填補這一空缺。

本研究評估多種大型語言模型在生成會議摘要方面的效果。通過使用 AMI 會議語料庫，並結合不同的預處理方法（如 Google 語音識別和 Whisper Base 轉錄）及不同的 Prompt 設計，來比較這些模型的表現。研究結果顯示，GPT-4 在大多數情況下表現最佳，但在需要高度精確率的情境中，GPT-3.5 更具優勢。而 Gemini 1.5 Pro 在召回率方面表現突出。本研究提供使用不同大型語言模型在實際生成會議摘要時的建議，可以依據不同需求選擇相應解決方案，期望這些發現能夠幫助企業和組織選擇適合的技術來提高會議記錄和摘要的效率與準確性。

關鍵字：大型語言模型、會議摘要、自動語音識別、提示工程

doi:10.6342/NTU202402787

Abstract

Enterprises and organizations increasingly rely on meetings for communication and decision-making, the generation of meeting summaries has become particularly important. The manual process of writing meeting summaries is cumbersome and prone to errors, with the quality often depending on the recorder's ability. Consequently, there has been related research using language models for meeting summarization in the past. In recent years, the rise of large language models such as GPT, Gemini, and Llama series has demonstrated exceptional performance in natural language processing tasks like text generation, translation, and question-answering systems. Meeting transcripts are usually lengthy and complex, involving multiple speakers' dialogues and covering a wide range of topics, making meeting summarization different from general article summarization. However, there is relatively little research on using large language models for meeting summarization, and this study aims to fill this gap.

This study evaluates the effectiveness of various large language models in generating

doi:10.6342/NTU202402787

meeting summaries. By using the AMI meeting corpus and combining different preprocessing methods (such as Google Speech Recognition and Whisper transcription) with different prompt designs, and compare the performance of these models. The research results show that GPT-4 performs best in most cases, but GPT-3.5 is more advantageous in situations requiring high precision. On the other hand, Gemini 1.5 Pro excels in recall rate. This study provides recommendations for using different large language models in practical meeting summarization scenarios, allowing for the selection of appropriate solutions based on specific needs. It is hoped that these findings can help enterprises and organizations choose suitable technologies to improve the efficiency and accuracy of meeting minutes and summaries.

Keywords: Large Language Models, Meeting Summarization, Automatic Speech Recognition, Prompt Engineering

doi:10.6342/NTU202402787

Contents

doi:10.6342/NTU202402787

doi:10.6342/NTU202402787

List of Figures

doi:10.6342/NTU202402787

List of Tables

doi:10.6342/NTU202402787

Chapter 1 Introduction

1.1 Background and Motivation

In contemporary enterprises and organizations, meetings have emerged as a crucial part of daily operations. With the rapid advancement of technology, the summarization of meeting has become increasingly essential for accurately documenting discussion content, decisions, and action items. Manually writing meeting minutes and summaries, however, is time-consuming, prone to errors, and the quality and detail of these summaries often hinge on the recorder's proficiency. Consequently, the development of technology for the automatic generation of meeting summaries has become an urgent necessity for enterprises.

In recent years, the development of large language models (LLMs) has significantly advanced, particularly in tasks related to natural language processing(NLP), such as text generation, translation, and question-answering systems. Since the introduction of the Transformer in 2017 [28], numerous LLMs have emerged. For instance, OpenAI GPT models, from GPT-1 [19] in 2018 to GPT-3 [5] in 2020, and the latest GPT-4 [3]. Concurrently, major technology companies have developed their own LLMs, such as Meta AI team, which released LLaMA [1] [26] in 2023. Google has also released the Gemini series, from Gemini 1.0 in 2022 [25] to the latest Gemini 1.5 [21]. These models have

doi:10.6342/NTU202402787

demonstrated progressively enhanced performance across a variety of real-world NLP tasks, including the processing of legal documents [23] and medical domain consultations [16].

Currently, there is limited research on using LLMs for meeting summarization tasks. Meeting records are typically long and complex, containing dialogues from multiple speakers and covering a wide range of topics. In the past, using language models to generate meeting summaries often required extensive pre-training and fine-tuning for specific tasks to adapt to the particular domain of the meeting. Today, LLMs can perform zero-shot and few-shot learning, completing specified tasks with little or no examples, without needing domain-specific fine-tuning. This makes large language models more widely and conveniently applicable to various NLP tasks.

Therefore, this paper aims to explore the effectiveness of LLMs in generating meeting summaries and compare them with traditional language models. This study hopes to understand the current performance of commonly used LLMs in automatically generating meeting summaries, as well as the advantages and limitations of each model in different scenarios, and provide valuable application recommendations for enterprises and organizations with meeting summarization needs.

1.2 Research Objectives

The primary objective of this study is to investigate the performance of meeting summarization tasks using different large language models (LLMs) in conjunction with various speech-to-text models applied to meeting audio files. The specific research objectives are as follows:

doi:10.6342/NTU202402787

1. **Evaluate the effectiveness of LLMs in generating meeting summaries**: This study will use several mainstream LLMs, including GPT-3.5, GPT-4, Llama 2, Llama 3, and Google's Gemini 1.0 Pro and Gemini 1.5 Pro models, to generate meeting summaries. By comparing the performance of these models, can understand their effectiveness with different data sources and prompts.

2. **Compare the differences between LLMs and traditional language models**: Traditional language models typically require extensive pre-training and fine-tuning for specific tasks to achieve optimal results. In contrast, LLMs can complete tasks with little or no examples without domain-specific fine-tuning. This study will compare the performance of these two types of models in generating meeting summaries.

3. **Provide practical application recommendations**: This research aims to offer recommendations on how to use LLMs for automatic meeting summarization in practical applications, thereby helping enterprises and organizations enhance their operational efficiency.

doi:10.6342/NTU202402787

Chapter 2 Literature Review

2.1 Automatic Speech Recognition

Automatic Speech Recognition (ASR) and Speech-to-Text Recognition (STR) are technologies that convert speech into text. The development of speech recognition technology has seen significant progress, evolving from Hidden Markov Models (HMM) and Gaussian Mixture Models (GMM) to deep learning models. Initially, speech recognition systems primarily relied on these statistical models to process the temporal characteristics of speech signals. However, with the rise of deep learning, speech recognition technology has increasingly adopted Deep Neural Networks (DNN), Convolutional Neural Networks (CNN), Long Short-Term Memory (LSTM) networks, and Transformer models. These advancements have greatly enhanced the accuracy and efficiency of speech recognition compared to earlier statistical models.

Before the emergence of machine learning and deep learning models, speech recognition was mainly based on Gaussian Mixture Model-Hidden Markov Model (GMM-HMM). GMM was used to model the probability distribution of acoustic features, while HMM captured the temporal characteristics of speech signals, enabling continuous speech recognition for large vocabularies. Due to its simplicity, GMM-HMM was the mainstream acoustic model in ASR systems[20]. However, these methods had limitations, such as

doi:10.6342/NTU202402787

difficulty in handling long-term dependencies between words. As computational power increased and datasets expanded, deep learning technologies gradually became the mainstream approach in speech recognition.

With the rise of deep learning, GMM was replaced by DNN, evolving into DNN-HMM models. Subsequently, CNNs, RNNs, and LSTMs have been applied in the field of speech recognition[22]. DNN extracts speech features through multiple layers of nonlinear transformations, performing better than GMM-HMM models[11][24], but it has limitations in handling sequential data. To address this, CNN was introduced to handle local features in speech signals[2], while RNN can handle long sequences, effectively capturing the temporal dependencies of speech but facing gradient vanishing or exploding issues during training[9]. LSTM, a variant of RNN, addresses this issue by introducing memory cells, making speech recognition systems more stable and accurate when processing long sequences, achieving notable results in speech recognition[8].

In recent years, Transformer-based models have also been applied to automatic speech recognition tasks. With self-attention mechanism, it can infer the representation of words in context and capture long-term dependencies between words, making it one of the current methods in speech recognition technology[6].

In current applications of speech recognition technology, Google Speech-to-Text API and Whisper model are two commonly used methods. Google Speech-to-Text API is a commercial speech recognition service, supporting multiple languages and accents. In multilingual recognition, it can automatically detect the language and perform corresponding recognition processing.

Whisper, developed by OpenAI, is a Transformer-based speech recognition model

doi:10.6342/NTU202402787

capable of handling multiple languages and accents. Whisper utilizes a large amount of unlabeled speech data, significantly reducing the reliance on labeled data and enhancing the model's generalization capability. The Whisper model performs multi-task learning, which enables the model to share knowledge across different tasks. Multi-task learning also helps the model better understand the diversity of speech signals [18].

2.2 Text Summarization

Text summarization is a significant part in natural language processing, referring to the techniques used to extract and summarize important information from a text to encapsulate the primary content. The techniques for text summarization can be classified into two categories: extractive methods and abstractive methods.

Extractive summarization involves selecting key words, sentences, or paragraphs from the original text to form a summary. Unlike abstractive summarization, extractive methods focus on directly picking important content from the source text rather than rephrasing or generating new language to express the summary. However, the drawback of extractive summarization is that the generated summaries may lack coherence. Major extractive summarization methods include TextRank, LexRank, and models based on deep learning.

TextRank is an extractive summarization method based on the PageRank algorithm. It calculates the similarity between sentences to determine the TextRank value, with higher scores indicating key sentences in the text. Because the summary is formed from sentences in the original text, the quality relies heavily on the source material[15]. Similarly, LexRank computes sentence similarity to create a weighted graph, with sentences as nodes

doi:10.6342/NTU202402787

and similarity values as edge weights, selecting the highest-scoring sentences to form the summary[7].

Recent research in extractive summarization involves deep learning and neural networks, using encoder-decoder frameworks to score sentences or classify them as important or unimportant. Methods such as Bert[14], RNN[32], and GNN[29] evaluate and extract key sentences to create summaries.

Abstractive summarization aims to address the limitations of extractive methods, such as potential lack of coherence and difficulty in controlling summary length. It allows for the inclusion of new vocabulary not present in the original text, resulting in summaries that are more fluent and clearer. Abstractive summarization methods can be categorized into structure-based, semantic-based, and deep learning and neural network-based approaches[10].

The structure-based approach identifies the most important information in the text and uses various structures, such as templates or tree structures, to generate the summary. These methods are often combined with extractive, semantic, or deep learning approaches. Template-based methods extract key phrases and fit them into predefined templates to form the summary. However, this approach is time-consuming and labor-intensive due to the need for manual template creation[4]. Tree-based methods involve organizing sentences from the text into a tree structure, which helps represent the content of a document. Although these methods effectively structure content, they may still lack a deep semantic understanding[10].

Semantic-based methods represent each text as a semantic graph, where nouns and verbs are nodes, and relationships are edges, capturing the relationships between words

doi:10.6342/NTU202402787

and phrases[12].

The deep learning and neural network-based approach is the most prominent method. Compared to extractive or other abstractive summarization methods, deep learning models capture both semantic and syntactic structures, resulting in superior performance. These models use an encoder-decoder architecture, where the encoder processes the input information into a context vector that retains contextual relationships, and the decoder generates the summary from this vector.

Transformers, a deep learning model based on the encoder-decoder architecture, improve on traditional models by using attention mechanisms to capture information across the entire input sequence, addressing the issue of information loss in longer sequences. Research utilizing such methods includes self-supervised pre-trained models based on Transformers[31].

doi:10.6342/NTU202402787

Chapter 3 Methodology

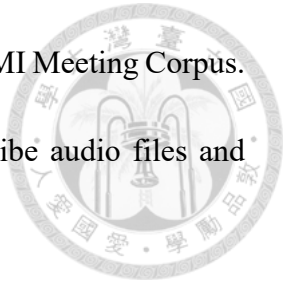
The objective of this study is to evaluate the effectiveness of various LLMs in generating meeting summaries. The AMI Meeting Corpus is selected as the evaluation dataset, which includes transcripts provided by AMI, as well as transcripts generated from original meeting audio files using Google Speech Recognition and Whisper. Comparative experiments are conducted on multiple LLMs, including models without fine-tuning and those with fine-tuning, to explore their performance in summarizing meeting records.

3.1 Research Framework

The research framework is designed to assess the performance of various LLMs in generating meeting summaries. The AMI Meeting Corpus serves as the source of experimental data, and comparative experiments are performed on various LLMs and pre-trained model for long dialogue summarization to explore their performance under different data pre-processing methods. The research framework mainly includes the following steps:

1. Data Collection and Preprocessing: Simulate different methods of recording meeting notes to comprehensively evaluate the summarization performance of various LLMs under different transcript generation scenarios.

doi:10.6342/NTU202402787



- Collect transcripts, audio files, and summaries from the AMI Meeting Corpus.
- Use Google Speech Recognition and Whisper to transcribe audio files and generate transcripts.

2. Model Selection:

   - The following models are selected for the experiments: GPT-3.5, GPT-4, Gemini 1.0 Pro, Gemini 1.5 Pro, LlaMA 2 - 7b, LlaMA 3 - 8b, and the pre-trained DialogLED model.

3. Summary result Evaluation:

   - Evaluate the effectiveness of the generated summaries using ROUGE scores, including ROUGE-N and ROUGE-L.

4. Result Analysis and Discussion:

   - Analyze the performance of different models and data preprocessing methods, and provide recommendations for practical applications in enterprises.

3.2 Dataset

The dataset utilized in this study is the multi-modal AMI Meeting Corpus, which includes approximately 100 hours of meeting recordings and associated textual information. The meetings in the AMI Meeting Corpus predominantly feature participants acting in various roles within a design team, collaboratively discussing the development of a remote control. In this scenario, each meeting participant is assigned a specific role. The Project Manager (PM) is responsible for coordinating the project and chairing meetings, ensuring the project is completed within time and budget constraints, producing and distributing

doi:10.6342/NTU202402787

meeting minutes, and writing a report at the end of the trial. The Marketing Expert (ME) is tasked with identifying user requirements and monitoring market trends. The User Interface Designer (UI) is responsible for the technical features and user interface of the remote control. The Industrial Designer (ID) is in charge of designing the functionality of the remote control.

In addition to meeting recordings, the dataset includes meeting transcripts and various annotations. These annotations provide detailed records of conversational behaviors, including participants' head and hand movements, gaze directions, room movements, emotional states, etc., as well as extractive and abstractive summaries of each meeting. The primary data used in this study are:

- **Meeting Recordings**: Audio files recorded by microphones in the meeting room.
- **Meeting Transcripts**: Time-stamped, word-level transcripts for each speaker. An example of a transcript is as follows:

doi:10.6342/NTU202402787

```
<w nite:id="EN2001d.A.words4" starttime="169.73" endtime="170.01">Mm-hmm</w>
<w nite:id="EN2001d.A.words5" starttime="170.01" endtime="170.01" punc="true">.</w>
<w nite:id="EN2001d.A.words6" starttime="224.83" endtime="225.04">So</w>
<w nite:id="EN2001d.A.words7" starttime="225.04" endtime="225.3">uh</w>
<w nite:id="EN2001d.A.words8" starttime="225.3" endtime="225.71">which</w>
<w nite:id="EN2001d.A.words9" starttime="225.71" endtime="226.38">version</w>
<w nite:id="EN2001d.A.words10" starttime="226.38" endtime="226.6">of</w>
<w nite:id="EN2001d.A.words11" starttime="226.6" endtime="226.82">my</w>
<w nite:id="EN2001d.A.words12" starttime="226.82" endtime="227.2">data</w>
<w nite:id="EN2001d.A.words13" starttime="227.2" endtime="227.43">was</w>
<w nite:id="EN2001d.A.words14" starttime="227.43" endtime="227.63">that</w>
<w nite:id="EN2001d.A.words15" starttime="227.63" endtime="227.63" punc="true">,</w>
<w nite:id="EN2001d.A.words16" starttime="227.63" endtime="227.74">the</w>
<w nite:id="EN2001d.A.words17" starttime="227.74" endtime="228.14">very</w>
<w nite:id="EN2001d.A.words18" starttime="228.14" endtime="228.61">first</w>
<w nite:id="EN2001d.A.words19" starttime="228.61" endtime="229.28">one</w>
<w nite:id="EN2001d.A.words20" starttime="229.28" endtime="229.28" punc="true">?</w>
<w nite:id="EN2001d.A.words21" starttime="229.28" endtime="230.64">Yeah</w>
<w nite:id="EN2001d.A.words22" starttime="230.64" endtime="230.64" punc="true">.</w>
<w nite:id="EN2001d.A.words23" starttime="230.64" endtime="230.87">Okay</w>
<w nite:id="EN2001d.A.words24" starttime="230.87" endtime="230.87" punc="true">.</w>
<w nite:id="EN2001d.A.words25" starttime="233.96" endtime="234.2">Yeah</w>
<w nite:id="EN2001d.A.words26" starttime="234.2" endtime="234.2" punc="true">,</w>
<w nite:id="EN2001d.A.words27" starttime="234.2" endtime="234.4" trunc="true">v</w>
```

Figure 3.1: AMI Meeting Corpus Transcripts

- **Meeting Summaries**: Manually written content that includes the abstract, actions,

doi:10.6342/NTU202402787

decisions, and problems discussed during the meeting.

```
<nite:root xmlns:nite="http://nite.sourceforge.net/">
<abstract nite:id="ES2002a.rdhillon.abstract.1">
<sentence nite:id="ES2002a.rdhillon.s.1">The project manager introduced the
    upcoming project to the team members and then the team members participated
    in an exercise in which they drew their favorite animal and discussed what
    they liked about the animal.</sentence>
<sentence nite:id="ES2002a.rdhillon.s.2">The project manager talked about the
    project finances and selling prices.</sentence>
<sentence nite:id="ES2002a.rdhillon.s.3">The team then discussed various features
     to consider in making the remote.</sentence>
</abstract>
<actions nite:id="ES2002a.rdhillon.actions.1">
<sentence nite:id="ES2002a.rdhillon.s.4">The industrial designer will work on the
     working design of the remote.</sentence>
<sentence nite:id="ES2002a.rdhillon.s.5">The user interface designer will work on
     the technical functions of the remote.</sentence>
<sentence nite:id="ES2002a.rdhillon.s.6">The marketing executive will work on
    what requirements the remote has to fulfill</sentence>
</actions>
<decisions nite:id="ES2002a.rdhillon.decisions.1">
<sentence nite:id="ES2002a.rdhillon.s.7">The remote will sell for 25 Euro.</
    sentence>
<sentence nite:id="ES2002a.rdhillon.s.8">The remote will be sold on an
    international scale.</sentence>
<sentence nite:id="ES2002a.rdhillon.s.9">The production costs cannot exceed 12.50
     Euro.</sentence>
</decisions>
<problems nite:id="ES2002a.rdhillon.problems.1">
<sentence nite:id="ES2002a.rdhillon.s.10">Whether the remote will be used
    exclusively for televisions.</sentence>
</problems>
</nite:root>
```

Figure 3.2: Example of Meeting Summaries from the AMI Meeting Corpus

doi:10.6342/NTU202402787

3.3 Data Preprocessing

In this study, three different data preprocessing methods were applied to the AMI Meeting Corpus to compare the effectiveness of different LLMs in generating summaries under various input scenarios. These preprocessing methods aim to simulate different meeting recording techniques and evaluate the impact of different transcription methods on summary generation.

3.3.1 Using Transcripts Provided by the AMI Meeting Corpus

In this study, the transcripts provided by the AMI Meeting Corpus are used as the input data. These transcripts are manually transcribed by professional transcribers and include word-level time stamps for each speaker, allowing the transcripts to accurately reflect the content and sequence of the meeting speeches.

The specific data processing steps are as follows:

1. Extracting Transcripts from the AMI Meeting Corpus: First, extract the transcript files from the AMI Meeting Corpus. These files are stored in XML format and include complete transcripts with time stamps.

2. Parsing the Transcript Files: Extract the speech content and time stamps for each speaker. Consolidate the content described by all speakers into a single file according to the time stamps. Each file represents the conversation of a single meeting.

doi:10.6342/NTU202402787

3.3.2 Using Transcripts Transcribed by Google Speech Recognition

The audio files from the AMI Meeting Corpus are transcribed into text using Google Speech Recognition. Google Speech Recognition is a commercial speech recognition service that can swiftly convert speech into text. This method simulates the automatic transcription services that enterprises might use in real-world scenarios. Although these services are convenient and fast, they may have limitations in transcription quality and accuracy.

The specific processing steps are as follows:

1. Extract the audio files from the AMI Meeting Corpus.

2. Use the Google Speech Recognition API to transcribe each meeting audio file, processing the audio in segments (each segment being 100 seconds) to generate the transcripts.

3.3.3 Using Transcripts Transcribed by Whisper

The final method employs Whisper for speech-to-text conversion. Whisper, developed by OpenAI, is an open-source speech recognition tool used to simulate the speech transcription services that might be employed in enterprises.

The specific processing steps are as follows:

1. Extract the audio files from the AMI Meeting Corpus.

2. Use Whisper Base to transcribe each meeting audio file.

These three data preprocessing methods enable a comparative analysis of the sum-

doi:10.6342/NTU202402787

marization performance of various LLMs under different scenarios, providing a comprehensive evaluation of these models in real-world applications.

3.4 Model Selection

This study utilizes various mainstream LLMs to assess their effectiveness and suitability for the task of summarizing meeting transcripts. The following is a detailed introduction of each model used in this study:

3.4.1 GPT-3.5

GPT-3.5, released by OpenAI in 2022, represents a significant advancement over its predecessor, GPT-3, particularly in its multi-task understanding capabilities. GPT-3.5 retains the Transformer architecture from the previous GPT models, which consists of a stack of decoder layers. The decoder processes the input sequence and generates the output sequence. The decoder consists of self-attention mechanisms and feed-forward neural networks. The self-attention mechanisms enable the model to focus on different parts of the input sequence, capturing semantic relationships and dependencies. The feed-forward neural networks process this information to produce relevant and coherent outputs in context.

The training methodology for GPT-3.5 involves pre-training on large-scale textual data, followed by fine-tuning using Reinforcement Learning from Human Feedback (RLHF). GPT-3.5 contains approximately 175 billion parameters.

In terms of performance, GPT-3.5 demonstrates capabilities in zero-shot and few-shot

doi:10.6342/NTU202402787

prompting. Zero-shot prompting refers to the model's ability to be applied to new tasks without any specific task examples. Few-shot prompting refers to the model's capability to learn and perform new tasks with only a few examples. This means that GPT-3.5 can be applied to various downstream tasks, such as translation, question answering, and text generation, through simple textual interactions specifying tasks and examples, without the need for additional fine-tuning. Moreover, it performs better than GPT-2 in these tasks [30].

3.4.2 GPT-4

GPT-4, released by OpenAI in March 2023, is the latest member of the GPT series, featuring multimodal capabilities that extend its application range by processing both text and image inputs [17].

GPT-4's multimodal abilities enable it to handle not only text but also image inputs simultaneously. Moreover, GPT-4 has demonstrated human-level performance in various professional and academic benchmark tests, such as achieving a top 10 % score in simulated bar exams, showcasing its capability to handle complex cognitive tasks.

In terms of architecture and training, GPT-4 inherits the Transformer architecture from earlier GPT models. It pre-trained on a large corpus of textual data, including both public data and third-party licensed data, GPT-4 is subsequently fine-tuned using Reinforcement Learning from Human Feedback (RLHF) and a Rule-Based Reward Model (RBRM) to adapt to different application scenarios and tasks.

The scale and number of parameters of GPT-4 are extraordinarily large, with an estimated parameter count of around one trillion, enabling it to process and generate detailed

doi:10.6342/NTU202402787

and context-rich outputs. GPT-4's practical applications are extensive, suitable for a wide range of NLP tasks such as dialogue, content generation, automated writing, natural language understanding, and complex visual processing tasks.

3.4.3 Gemini 1.0

Gemini 1.0 Pro, a multimodal LLM developed by Google, is designed to process images, audio, video, and text inputs concurrently. The model architecture of Gemini 1.0 Pro is built using a Transformer decoder-only framework [27], and it utilizes Google's Tensor Processing Units (TPUs) for large-scale model training, enabling efficient handling of large datasets. Its pre-training data includes text from web documents, books, and code, as well as image, audio, and video data.

In terms of performance, Gemini 1.0 Pro excels in multiple benchmark tests, achieving human expert levels in various professional and academic evaluations. The Massive Multitask Language Understanding (MMLU) dataset is a benchmark designed to assess a language model's abilities across a wide range of tasks and domains. MMLU covers 57 subjects, including STEM, humanities, and social sciences, with difficulty levels ranging from elementary to advanced professional, testing both world knowledge and problem-solving skills. In the MMLU test, Gemini 1.0 Pro achieved an accuracy of 71.8%, slightly lower than GPT-4's 86.4% but significantly outperforming GPT-3.5's 70% and LLAMA-2 70B's 68.0% [25].

doi:10.6342/NTU202402787

3.4.4 Gemini 1.5

The Gemini 1.5 series, recently announced by Google, includes two main versions: Gemini 1.5 Pro and Gemini 1.5 Flash. Gemini 1.5 Pro is a Mixture of Experts (MoE) model based on the Transformer architecture. It replaces the original Transformer's Feedforward Neural Networks (FNN) with MoE layers, which allows pre-training with significantly fewer computational resources than dense models and enables a substantial increase in the number of parameters. In terms of performance, Gemini 1.5 Pro surpasses Gemini 1.0 in most capabilities and benchmark tests, capable of reasoning over multiple long documents and several hours of video and audio files. In one practical application, when provided with a grammar manual of Kalamang, a small language with only 500 speakers, Gemini 1.5 Pro was able to learn to translate from English to Kalamang, achieving a level comparable to that of a human learning from the same material. In the MMLU test, the Gemini 1.5 Pro model achieved an accuracy of 85.9% [21].

3.4.5 Llama 2

Llama 2, released by Meta AI in July 2023, is a LLM featuring versions with parameters ranging from 7 billion to 70 billion. Llama 2 introduces several significant improvements over its predecessor, Llama. It extends the context length from 2048 tokens to 4096 tokens and integrates Grouped-Query Attention (GQA), an advanced attention mechanism, into each layer. Additionally, it replaces traditional Layer Norm with Root Mean Square Layer Normalization (RMSQLN). Distinct from typical Transformer models, Llama 2 uses the Swish-Gated Linear Unit (SwiGLU) activation function. The training data for Llama 2 consists of open-source data, totaling up to 2 trillion tokens, including

doi:10.6342/NTU202402787

web content, Wikipedia articles in 20 languages, and scientific papers from ArXiv [26].

3.4.6 Llama 3

Llama 3, developed by Meta AI, is the latest generation of LLMs, released in 2024, featuring pre-trained models with 8 billion and 70 billion parameters. Compared to Llama 2, the smaller version has increased from 7 billion to 8 billion parameters. In terms of training data, Llama 3's pre-training dataset includes 15 trillion tokens from public resources, more than seven times the dataset used for Llama 2. This dataset also includes four times more code, covering over 30 languages. In the MMLU test, the Llama 3 8B model achieved 68.4% accuracy, while the model with 70B parameters reached 82.0% [1].

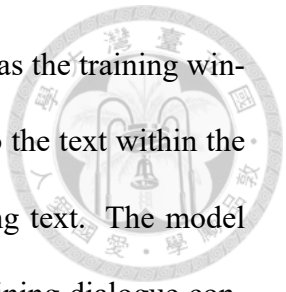
Model	MMLU (5-shot) Score
GPT-3.5	70.0%
GPT-4	86.4%
Gemini 1.0 Pro	71.8%
Gemini 1.5 Pro	85.9%
Llama 2 7B	45.3%
Llama 2 13B	54.8%
Llama 2 34B	62.6%
Llama 2 70B	68.9%
Llama 3 8B	68.4%
Llama 3 70B	82.0%

Table 3.1: Performance of Different LLMs on the MMLU Test

3.4.7 DialogLED-large

DialogLED-large is a pre-trained model specifically designed for understanding and summarizing long conversations. It currently represents the state-of-the-art (SOTA) model for meeting summarization tasks. This model utilizes a window-based denoising pre-training method, wherein a continuous segment of text (approximately 10% of the total

doi:10.6342/NTU202402787



conversation length) is randomly selected from long dialogues to act as the training window. During the pre-training phase, five types of noise are applied to the text within the window, including masking parts of sentences, merging, and splitting text. The model is then trained to reconstruct the original window based on the remaining dialogue content. This method enhances the model's ability to comprehend the structure and content of dialogues more effectively.

3.5 Experimental Design

The large language models in this study demonstrate excellent performance in zero-shot prompting. However, the existing large language models have limited capabilities in handling long texts, which is a common feature of meeting transcripts. Previous studies have addressed this issue by experimenting with the AMI meeting corpus, dividing the long texts into two categories for preprocessing: the first method involves directly using the transcripts but only taking the first n words. The second method splits a meeting transcript into multiple sections, each containing n words, and then summarizing each section using a large language model. These section summaries are then further summarized by the model to produce the final meeting summary. However, the results of these experiments showed that, regardless of the large language model used, the second method did not consistently outperform the first method when evaluated using ROUGE scores. Moreover, the second method requires more prompt processing, resulting in higher time and cost. Therefore, this study directly inputs the entire transcript into the large language models to evaluate their effectiveness.

For model selection, the large language models chosen for the experiments include

doi:10.6342/NTU202402787

GPT-3.5, GPT-4, Gemini 1.0 Pro, Gemini 1.5 Pro, LlaMA 2 7B, LlaMA 3 8B, and the pre-trained model: DialogLED-large, mainly accessed via API. These models will be used to generate summaries of meeting transcripts, with the generation time recorded.

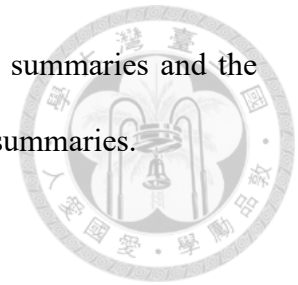
The experiment adjusts model versions and prompts. Since the summaries provided by the AMI meeting corpus include abstract , actions, decisions, and problems, this study designs two prompts. The first prompt is: "Please summarize the following transcript: transcript, highlighting the abstract concepts, actions, decisions, and problems discussed." The second prompt only asks the model to generate a summary without specifying specific key points based on the reference summary: "Please summarize the following transcript: transcript." The maximum generation length is set to 200, and the generation temperature is set to 0.7.

Temperature is a parameter that controls the randomness of the generated text. In natural language generation tasks, adjusting the temperature affects the distribution and diversity of the generated words. When the temperature is set to a lower value (close to 0), the model tends to select the highest probability words, resulting in more conservative and predictable text with higher repetition but better consistency and coherence. When the temperature is set to a higher value (close to 1 or higher), the model generates more creative and diverse text, which may lead to incoherent or illogical content. This study sets the generation temperature to 0.7, aiming to retain a certain level of creativity and diversity without making the generated content too random or incoherent.

During the summary generation process, the transcripts are input into each model using the same prompts. The generated summaries are evaluated using ROUGE scores, including specific metrics such as ROUGE-N and ROUGE-L. ROUGE-1, ROUGE-2, and

doi:10.6342/NTU202402787

ROUGE-L are used to measure the overlap between the generated summaries and the reference summaries, thereby assessing the quality of the generated summaries.



3.6 Evaluation Method

This study uses the ROUGE (Recall-Oriented Understudy for Gisting Evaluation) metric to evaluate the effectiveness of the generated summaries. The ROUGE metric is commonly used for summarization evaluation by measuring the similarity between the generated summary and the reference summary. This study will use two sub-metrics of ROUGE: ROUGE-N and ROUGE-L. Below is a detailed description of the ROUGE metrics [13]:

3.6.1 ROUGE-N

ROUGE-N (N-Gram Overlap): This metric calculates the overlap of N-grams (continuous sequences of N words) between the generated summary and the reference summary. It measures the proportion of N-grams in the generated text that match those in the reference text. The formula is as follows:

$$\text{ROUGE-N} = \frac{\sum_{S \in \text{ReferenceSummaries}} \sum_{gram_n \in S} \text{Countmatch}(gram_n)}{\sum S \in \text{ReferenceSummaries} \sum_{gram_n \in S} \text{Count}(gram_n)} \tag{3.1}$$

Here, n represents the length of the N-gram.

doi:10.6342/NTU202402787

3.6.1.1 Confusion Matrix

The Confusion Matrix is a tool used to describe the performance of a classification model, representing the distribution of prediction results. It comprises four types of outcomes, and the following figure is an example of a confusion matrix, illustrating these four types of outcomes:

- **True Positive, TP**：The number of samples correctly predicted as positive by the model.
- **False Positive, FP**：The number of samples incorrectly predicted as positive by the model, but actually negative.
- **True Negative, TN**：The number of samples correctly predicted as negative by the model.
- **False Negative, FN**：The number of samples incorrectly predicted as negative by the model, but actually positive.

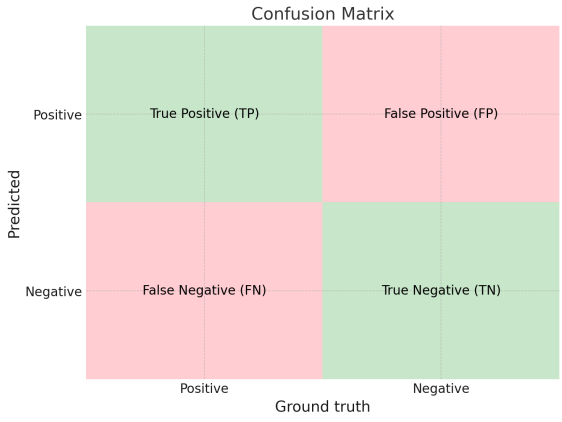


Figure 3.3: Confusion Matrix

doi:10.6342/NTU202402787

3.6.1.2 Precision, Recall and F1-Score

Common metrics for evaluating model accuracy include Precision, Recall, and F1-Score. These metrics are introduced as follows:

- **Precision**：The proportion of true positive samples among the samples predicted as positive by the model. The formula is:

$$\text{Precision} = \frac{TP}{TP + FP} \tag{3.2}$$

- **Recall**：The proportion of actual positive samples that are correctly predicted as positive by the model. The formula is:

$$\text{Recall} = \frac{TP}{TP + FN} \tag{3.3}$$

- **F1-Score**：The harmonic mean of Precision and Recall. The formula is:

$$\text{F1-Score} = 2 \cdot \frac{\text{Precision} \cdot \text{Recall}}{\text{Precision} + \text{Recall}} \tag{3.4}$$

When applied to the ROUGE-N formula, it becomes:

$$\text{Precision} = \frac{\text{Count}_{\text{match}}}{\text{Count}_{\text{generated}}} \tag{3.5}$$

$$\text{Recall} = \frac{\text{Count}_{\text{match}}}{\text{Count}_{\text{reference}}} \tag{3.6}$$

Thus, it can be concluded that the ROUGE-N formula is equivalent to calculating the

doi:10.6342/NTU202402787

Recall between the reference summary and the generated summary.

3.6.2 ROUGE-L（Longest Common Subsequence）

ROUGE-L calculates the longest common subsequence (LCS) between the generated summary and the reference summary. The formulas are as follows:

$$R_{LCS} = \frac{LCS(X,Y)}{m} \tag{3.7}$$

$$P_{LCS} = \frac{LCS(X,Y)}{n} \tag{3.8}$$

$$F_{LCS} = \frac{(1+\beta^2) \cdot R_{LCS} \cdot P_{LCS}}{R_{LCS} + \beta^2 \cdot P_{LCS}} \tag{3.9}$$

Where $LCS(X,Y)$ is the length of the longest common subsequence between sequences X and Y, m is the length of the reference summary, n is the length of the generated summary, and β is usually set to a larger value.

3.6.2.1 Summary-Level ROUGE

When applied to the summary level, the joint LCS match between reference summary sentences r_i and each generated summary sentence c_j is computed. For a reference summary with u sentences and a generated summary with v sentences, the summary-level LCS based on the F-Measure can be calculated as follows:

doi:10.6342/NTU202402787

$$R_{LCS} = \frac{\sum_{i=1}^{u} LCS(r_i, C)}{m} \tag{3.10}$$

$$P_{LCS} = \frac{\sum_{i=1}^{u} LCS(r_i, C)}{n} \tag{3.11}$$

$$F_{LCS} = \frac{(1+\beta^2) \cdot R_{LCS} \cdot P_{LCS}}{R_{LCS} + \beta^2 \cdot P_{LCS}} \tag{3.12}$$

Here, $LCS(r_i, C)$ is the length of the joint longest common subsequence between reference sentence r_i and generated summary C, m is the total length of the reference summary, and n is the total length of the generated summary.

doi:10.6342/NTU202402787

Chapter 4 Experiments

4.1 Dataset

In this experiment, the AMI Meeting Corpus was utilized, comprising 142 meeting transcripts and summaries, each meeting lasting approximately 30 to 40 minutes. The following table presents the average word count of transcripts generated through three different preprocessing methods:

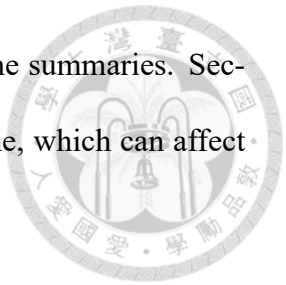
Data Source	**Average Transcript Word Count**
Google Speech Recognition	1667.30
Whisper Base	3799.77
Reference Transcripts	4820.25

Table 4.1: Average word count of transcripts generated by different methods

The table reveals a discrepancy in the word counts of transcripts generated by different methods.Transcripts from Google Speech Recognition show a significantly lower average word count compared to the original transcripts, indicating that many words were not successfully transcribed. The impact of word count on summary generation can be attributed to the following factors: First, the accuracy and completeness of the summaries: Longer transcripts may contain more information and context, enabling the model to generate more accurate and complete summaries. However, excessively long transcripts

doi:10.6342/NTU202402787

might overwhelm the model, potentially diminishing the quality of the summaries. Second, processing time: Longer transcripts require more processing time, which can affect the practicality of the system.



4.2 Experimental Analysis: Impact of Prompts

In this study, summarization experiments were conducted using transcripts from different data sources to evaluate the impact of different prompts on summary generation. The data sources include the original transcripts from the AMI Meeting Corpus, transcripts transcribed using Google Speech Recognition, and transcripts transcribed using Whisper. The models employed in this study include GPT-3.5, GPT-4, Gemini 1.0 Pro, Gemini 1.5 Pro, Llama 2 - 7B, and Llama 3 - 8B.

4.2.1 Experimental Analysis of Targeted Summary Prompt

The design of the prompt is based on the summaries provided by the AMI meeting corpus, which are annotated with: abstract, actions, decisions, and problems.

> **Prompt:** Please summarize the following transcript: {transcript}, highlighting the abstract concepts, actions, decisions, and problems discussed.

In the study, ROUGE-1, ROUGE-2, and ROUGE-L metrics are employed to evaluate the effectiveness of the generated summaries. The results of the summaries generated using the targeted summary prompt are as follows:

doi:10.6342/NTU202402787

Data Source / Model	ROUGE-1			ROUGE-2			ROUGE-L		
	Precision (%)	Recall (%)	F1-score (%)	Precision (%)	Recall (%)	F1-score (%)	Precision (%)	Recall (%)	F1-score (%)
Google Speech Recognition									
GPT-3.5	44.33	32.86	36.93	8.57	6.46	7.22	23.63	17.45	19.64
GPT-4	35.89	37.70	36.13	6.08	6.39	6.13	17.88	19.07	18.12
Gemini 1.0 Pro	35.35	25.46	28.45	5.53	4.07	4.50	18.23	13.15	14.67
Gemini 1.5 Pro	33.18	25.62	27.84	5.35	4.34	4.61	17.43	13.60	14.68
Llama 2 - 7B	34.83	42.73	37.49	6.82	8.30	7.33	17.81	22.16	19.26
Llama 3 - 8B	30.72	43.15	35.15	5.59	8.02	6.47	15.58	22.32	17.95
Whisper Base									
GPT-3.5	46.10	33.46	37.73	9.75	7.29	8.11	25.11	18.28	20.55
GPT-4	46.37	36.31	39.54	9.99	8.14	8.67	24.44	19.33	20.90
Gemini 1.0 Pro	33.23	28.00	29.33	5.55	4.71	4.93	17.07	14.44	15.07
Gemini 1.5 Pro	35.39	28.81	30.61	6.48	5.46	5.71	17.51	14.53	15.26
Llama 2 - 7B	32.72	36.99	33.03	6.04	6.97	6.16	17.25	19.77	17.43
Llama 3 - 8B	31.40	38.71	33.40	5.15	6.54	5.57	16.19	20.25	17.27
Reference Transcripts									
GPT-3.5	47.82	34.81	39.06	10.23	7.52	8.38	25.65	18.74	20.97
GPT-4	38.74	41.52	39.37	7.43	8.10	7.60	19.07	20.71	19.46
Gemini 1.0 Pro	34.87	31.55	32.14	6.09	5.66	5.70	17.15	15.72	15.90
Gemini 1.5 Pro	28.19	50.98	35.67	5.52	10.17	7.04	13.40	24.64	17.05
Llama 2 - 7B	33.90	39.07	35.02	6.32	7.16	6.46	17.55	20.46	18.15
Llama 3 - 8B	30.55	38.94	33.42	5.07	6.49	5.56	15.34	19.97	16.88

Table 4.2: ROUGE Score for Various Data Sources and Model Combinations, using targeted summary prompt

Data Source/Model	Model	Average Summary Generation Time (Seconds)
Google Speech Recognition	GPT-3.5	2.65
	GPT-4	8.68
	Gemini 1.0 Pro	4.54
	Gemini 1.5 Pro	4.69
	Llama 2 - 7B	7.39
	Llama 3 - 8B	17.52
Whisper Base	GPT-3.5	4.27
	GPT-4	4.30
	Gemini 1.0 Pro	6.21
	Gemini 1.5 Pro	6.14
	Llama 2 - 7B	7.21
	Llama 3 - 8B	18.11
Reference Transcripts	GPT-3.5	5.26
	GPT-4	11.89
	Gemini 1.0 Pro	6.40
	Gemini 1.5 Pro	10.51
	Llama 2 - 7B	7.32
	Llama 3 - 8B	10.08

Table 4.3: Time Required for Summary Generation with Various Data Sources and Model Combinations, using targeted summary prompt

doi:10.6342/NTU202402787

4.2.1.1 Comparison Using the Same Dataset

Using Google Speech Recognition Transcripts: The transcripts generated by Google Speech Recognition had the lowest average word count (1667.30 words). However, overall, the performance of the models did not show significant differences compared to the other two data sources. The GPT and Llama series performed better in terms of F1-score, whereas the performance of Gemini 1.0 Pro and Gemini 1.5 Pro was relatively worse.

Using Whisper Base Transcripts: In the ROUGE scores for transcripts generated using Whisper across various models, the GPT series exhibited the best F1-score, followed by the Llama series, and then the Gemini series. There was no significant improvement in the overall F1-score compared to the Google Speech Recognition transcripts. This suggests that despite the Google Speech Recognition transcripts having the lowest average word count, they effectively captured key terms, enabling the completion of meeting summaries even with fewer words.

Using the Original Transcripts Provided by the AMI Meeting Corpus: When using the original transcripts, the meeting content is recorded accurately, and the ROUGE scores are generally superior to those from transcripts generated by Whisper Base or Google Speech Recognition. However, there are exceptions in some metric. For instance, in the experiment with Gemini 1.5 Pro, the Recall was 50.98%, indicating that compared to summaries generated by other models, it included a larger amount of content matching the reference summaries, but it also contained many additional words not present in the reference summaries.

doi:10.6342/NTU202402787

4.2.1.2 Comparison Based on the Same Model

GPT-3.5 and GPT-4: Among all data sources, the GPT series exhibits superior performance. However, GPT-3.5 often demonstrates higher Precision and lower Recall, indicating that GPT-3.5 can more accurately capture key information when generating summaries, but lacks comprehensive coverage. Therefore, it is evident that GPT-3.5 is more suitable for summarization tasks requiring high precision, where concise and accurate but incomplete summaries are acceptable.

Gemini 1.0 Pro and Gemini 1.5 Pro: These two models generally have the lowest F1-scores for ROUGE-1, ROUGE-2, and ROUGE-L across all data sources, with both models also showing lower Precision, indicating deficiencies in their performance. A distinct feature of Gemini 1.5 Pro is its high Recall, meaning most of the important content is included in the reference summaries, though the generated summaries may also contain some irrelevant content.

Llama 2 - 7B and Llama 3 - 8B: These two models exhibit the best ROUGE-1 F1-score performance with the Google Speech Recognition data source. However, a primary disadvantage is the longer time required to generate summaries.

4.2.1.3 Overall Analysis

Overall, when generating summaries using the targeted summary prompt, GPT-4 exhibited the best F1 score across all data sources, with GPT-3.5 achieving higher Precision. The Gemini series models showed relatively weaker performance across various metrics, except for Gemini 1.5 Pro, which had good Recall. The Llama series performed the second-best, with an advantage in average Recall, but lower Precision compared to

doi:10.6342/NTU202402787

other models, and it also required a longer time to generate summaries.

4.2.2 Experimental Analysis of General Summary Prompt

In this experiment, the prompt design only asks the model to generate a summary without specifying specific focus points based on the reference summary. The prompt design is as follows:

Prompt: Please summarize the following transcript: {text}.

In the study, ROUGE-1, ROUGE-2, and ROUGE-L metrics are employed to evaluate the effectiveness of the generated summaries. The results of the summaries generated using the general summary prompt are as follows:

Data Source / Model	ROUGE-1			ROUGE-2			ROUGE-L		
	Precision (%)	Recall (%)	F1-score (%)	Precision (%)	Recall (%)	F1-score (%)	Precision (%)	Recall (%)	F1-score (%)
Google Speech Recognition									
GPT-3.5	45.75	31.69	36.43	7.95	5.64	6.41	24.93	17.19	19.76
GPT-4	38.62	38.16	37.65	6.80	6.83	6.68	19.48	19.30	19.01
Gemini 1.0 Pro	36.48	33.59	34.08	6.11	5.78	5.81	19.09	17.39	17.71
Gemini 1.5 Pro	29.89	45.12	35.27	4.99	7.70	5.95	14.65	22.26	17.33
Llama 2 - 7B	34.87	42.13	36.89	7.06	8.54	7.50	17.95	21.68	18.98
Llama 3 - 8B	36.35	46.07	39.81	7.34	9.38	8.06	17.74	22.77	19.51
Whisper Base									
GPT-3.5	47.33	30.64	36.15	9.85	6.78	7.78	26.03	16.94	19.93
GPT-4	40.27	40.34	39.45	8.04	8.33	8.02	20.47	20.65	20.08
Gemini 1.0 Pro	32.94	38.01	33.75	6.48	7.88	6.88	17.68	19.86	17.68
Gemini 1.5 Pro	29.42	47.18	34.92	5.42	9.15	6.61	14.75	23.32	17.30
Llama 2 - 7B	35.78	37.29	34.81	6.62	6.82	6.44	18.64	19.51	18.04
Llama 3 - 8B	34.47	41.30	36.48	6.20	7.53	6.64	17.07	20.51	18.03
Reference Transcripts									
GPT-3.5	49.09	34.11	38.83	11.07	7.96	8.91	26.41	18.46	20.95
GPT-4	40.73	41.63	40.23	8.35	8.65	8.29	20.79	21.50	20.62
Gemini 1.0 Pro	31.74	43.36	35.87	6.22	8.72	7.10	16.10	22.05	18.20
Gemini 1.5 Pro	30.64	50.53	37.22	6.29	10.64	7.70	14.95	24.97	18.22
Llama 2 - 7B	34.58	38.76	34.91	6.43	7.16	6.46	17.43	19.66	17.59
Llama 3 - 8B	34.62	43.01	37.75	6.32	7.68	6.84	16.96	21.29	18.56

Table 4.4: ROUGE Score for Various Data Sources and Model Combinations, using general summary prompt

doi:10.6342/NTU202402787

Data Source/Model	Model	Average Summary Generation Time (Seconds)
Google Speech Recognition	GPT-3.5	2.13
	GPT-4	7.75
	Gemini 1.0 Pro	4.91
	Gemini 1.5 Pro	9.30
	Llama 2 - 7B	7.18
	Llama 3 - 8B	7.61
Whisper Base	GPT-3.5	4.28
	GPT-4	11.76
	Gemini 1.0 Pro	6.99
	Gemini 1.5 Pro	10.41
	Llama 2 - 7B	6.08
	Llama 3 - 8B	7.05
Reference Transcripts	GPT-3.5	5.23
	GPT-4	13.29
	Gemini 1.0 Pro	8.02
	Gemini 1.5 Pro	10.45
	Llama 2 - 7B	6.47
	Llama 3 - 8B	7.15

Table 4.5: Time Required for Summary Generation with Various Data Sources and Model Combinations, using general summary prompt

4.2.2.1 Comparison Using the Same Dataset

Using Google Speech Recognition Transcripts: Similar to the results from the targeted summary prompt experiments, the GPT and Llama series performed better in terms of ROUGE-1 and ROUGE-L F1-scores, while Gemini 1.0 Pro and Gemini 1.5 Pro showed relatively weaker performance. However, compared to the targeted summary prompt, all models showed improvements. The ROUGE-1 F1-score for Gemini 1.0 Pro increased from 28.45% to 34.08%, and for Gemini 1.5 Pro, it increased from 27.84% to 35.27%.

Using Whisper Base Transcripts:As with the targeted summary prompt experiments, the GPT series achieved the best F1-scores among all models. In terms of ROUGE-2 F1-score, there was no significant difference between the Gemini and Llama series. Compared to the F1-scores from Google Speech Recognition transcripts, although the

doi:10.6342/NTU202402787

word count of Whisper Base transcripts is higher, there was no notable difference in the effectiveness of the summaries as measured by F1-score.

Using the Original Transcripts Provided by the AMI Meeting Corpus: In the general summary prompt experiments, models using the original transcripts provided by the AMI Meeting Corpus generally performed better than those using transcripts generated by Google Speech Recognition or Whisper Base.

4.2.2.2 Comparison Based on the Same Model

GPT Series: The GPT series exhibited superior performance in Precision across all data sources, indicating that the generated summaries' content was largely included in the reference information. Compared to the targeted summary prompt, both Recall and F1-score improved, particularly for GPT-4, which showed performance improvements across all data sources. Nevertheless, GPT-3.5 had a higher Precision score than GPT-4.

Gemini Series and Llama Series: In the targeted summary prompt experiments, the performance of the Gemini series was significantly lower than that of the GPT and Llama series. However, in this general summary prompt experiment, the gap between the Gemini and Llama series narrowed, with the Gemini series outperforming the Llama series in terms of Recall.

4.2.2.3 Overall Analysis

When generating summaries using the general summary prompt, the GPT series performed the best across all data sources, particularly in terms of Precision. This makes the GPT series a better choice in scenarios where accurate and precise information is re-

doi:10.6342/NTU202402787

quired. GPT-4 generally had higher Precision and F1-scores than other models, indicating that most of the generated content was correct and relevant. However, its relatively lower Recall suggests it did not capture all important information. The Gemini series had higher Recall scores, especially Gemini 1.5 Pro, which generated more important content but also included many irrelevant details due to its lower Precision.

Overall, when using the targeted summary prompt for summary generation, GPT-4 demonstrated the best performance across all data sources, particularly in Recall scores, indicating it covered more content from the reference summaries. However, its Precision was slightly lower than GPT-3.5, suggesting more redundant information in the generated summaries. The Gemini series models had relatively weaker performance across various metrics. The Llama series performed the second-best, with certain advantages in Recall, but lower Precision compared to other models, and required a longer time to generate summaries.

4.2.3 Comparison Analysis of Targeted and General Summary Prompts

In this experiment, the effectiveness of generating summaries using different prompts, data sources, and models was compared. The detailed analysis is as follows:

4.2.3.1 ROUGE Metrics Discussion

Recall: The general summary prompt improved Recall for most models and data sources, particularly for the GPT and Llama series models. This means that the simplified instructions resulted in summaries that covered more of the reference summaries, although they also included many irrelevant words.

doi:10.6342/NTU202402787

Precision: There was no significant difference in Precision between the target and general summary prompts.

F1-score: F1-score: Overall, the general summary prompt improved the F1-score for most experimental combinations by increasing Recall without sacrificing too much Precision.

4.2.3.2 Discussion on Prompt Design

In this study, two different prompts were designed to evaluate their impact on summary generation. The targeted summary prompt was based on the structure of summaries provided by the AMI meeting corpus, which included specific elements such as abstract, actions, decisions, and problems. The purpose of this design was to guide the model to focus on specific key information, making the generated summaries more structured and specific. However, this could also increase the model's burden, making it difficult to capture all information comprehensively when dealing with longer instructions.

The general summary prompt only required the model to generate a summary without specifying specific focus points. This design reduced the model's burden, allowing it to more flexibly process and capture various key information, thereby improving Recall.

4.2.3.3 Comparison of Summary Generation Time

Comparing the time required to generate summaries using the two prompts reveals that the general summary prompt generally required less time. For example, this was most evident with Llama 3 - 8B, where generating a summary with the Targeted Summary Prompt took an average of 17.52 seconds per meeting, while using the General Summary

doi:10.6342/NTU202402787

Prompt took an average of 7.61 seconds per meeting.

Among different models, GPT-3.5 and GPT-4 both performed well in terms of F1-score, with GPT-4 outperforming GPT-3.5. However, GPT-3.5 generally had shorter generation times, making it suitable for scenarios requiring high efficiency. GPT-4 took longer to generate summaries but had the best performance, making it suitable for applications requiring high accuracy in summaries.

4.2.3.4 Summary

In this experiment, using the original transcripts from the AMI meeting corpus resulted in better performance compared to using Whisper-generated transcripts, while summaries generated from Google Speech Recognition transcripts had the poorest model performance. If Recall is considered, the best performance was achieved by the Gemini 1.5 Pro model using the General Summary Prompt. For Precision, the best performance was by the GPT-3.5 model using the General Summary Prompt. In terms of F1-score, the GPT-4 model performed the best. Therefore, the most appropriate model combination can be selected based on the specific requirements of the meeting summaries.

4.3 Comparison of Language Model and Large Language Models

In this experiment, the current state-of-the-art (SOTA) model for meeting summarization tasks, the pre-trained DialogLED model without fine-tuning, was utilized. By comparing the performance of the pre-trained model without fine-tuning with that of LLMs, the performance differences between them were examined.

doi:10.6342/NTU202402787

Below are the ROUGE scores and the time taken by the DialogLED-large model across different data sources:

Data Source / Model	ROUGE-1 Recall (%)	ROUGE-2 Recall (%)	ROUGE-L Recall (%)
Google Speech Recognition			
Llama 3 - 8B	43.15	8.02	22.32
Pre-Trained-DialogLED-large-5120	24.39	3.49	12.61
Whisper Base			
Gemini 1.5 Pro	28.81	5.46	14.53
Pre-Trained-DialogLED-large-5120	23.42	2.69	12.81
Reference Transcripts			
Gemini 1.5 Pro	50.98	10.17	24.64
Pre-Trained-DialogLED-large-5120	22.36	2.19	12.38
Base line: Fine-Tuned-DialogLED-large-5120	54.80	20.37	52.26

Table 4.6: ROUGE Score for Various Data Sources and Model Combinations

Data Source / Model	ROUGE-1 Recall (%)	ROUGE-2 Recall (%)	ROUGE-L Recall (%)
Reference Transcripts			
Gemini 1.5 Pro	50.98	10.17	24.64
Pre-Trained-DialogLED-large-5120	22.36	2.19	12.38
Base line: Fine-Tuned-DialogLED-large-5120	54.80	20.37	52.26

Table 4.7: Comparison of Summary Performance of Different Models on Reference Transcripts

Data Source/Model	Model	Average Summary Generation Time (Seconds)
Google Speech Recognition	Llama 3 - 8B	7.61
	Pre-Trained-DialogLED-large-5120	23.80
Whisper Base	Gemini 1.5 Pro	10.41
	Pre-Trained-DialogLED-large-5120	21.97
Reference Transcripts	Gemini 1.5 Pro	10.45
	Pre-Trained-DialogLED-large-5120	27.42

Table 4.8: Time Required for Summary Generation with Various Data Sources and Model Combinations

The experimental results demonstrate that the ROUGE Recall scores of the pre-trained DialogLED model without fine-tuning are notably lower across all data sources when compared to LLMs. Nevertheless, after the pre-trained DialogLED model undergoes fine-tuning, it surpasses the performance of LLMs. The fine-tuning process allows the model to acquire domain-specific knowledge and features, thereby improving the per-

doi:10.6342/NTU202402787

formance of the generated summaries. The fine-tuned DialogLED model generates more accurate meeting summaries and enhances ROUGE metrics. Experimental findings show that the Fine-Tuned-DialogLED-large-5120 model attains a ROUGE-1 Recall of 54.80%, significantly higher than LLMs such as Gemini 1.5 Pro. Therefore, for the task of meeting summarization, pre-trained models necessitate fine-tuning to acquire relevant domain knowledge.LLMs, while capable of zero-shot and few-shot learning, can generate meeting summaries without domain-specific training but still fall short in summary quality compared to the fine-tuned pre-trained DialogLED model.

Additionally, in terms of the time required to generate meeting summaries, the pre-trained DialogLED model without fine-tuning takes considerably longer than other LLMs, resulting in relatively lower efficiency in practical applications.

4.4 Evaluation of Practical Applications of Language Models and Large Language Models for Meeting Summarization

In practical applications, selecting the appropriate model for meeting summarization requires a comprehensive consideration of the cost of generating summaries, the speed of generation, the accessibility of each model, and security considerations. This section will analyze the performance of different models in these aspects.

GPT-3.5 and GPT-4 OpenAI provides APIs for GPT-3.5 and GPT-4, which can be directly utilized by enterprises. However, using these APIs requires uploading internal data to OpenAI's servers, posing a risk of data leakage. Additionally, usage is billed

doi:10.6342/NTU202402787

based on the number of tokens processed, which could lead to high costs for enterprises with large-scale application needs. Nevertheless, an advantage is the relatively fast speed of summary generation, particularly with GPT-3.5, which takes less time to generate summaries, making it suitable for scenarios requiring real-time meeting summaries. In terms of model performance, GPT-3.5 using a general summary prompt achieves the best results when considering Precision, while GPT-4 performs better based on F1-score. For meeting scenarios that require highly accurate and reliable information, such as in legal or medical fields, ensuring that every piece of information in the summary is correct is crucial. In such cases, the GPT-3.5 model with high Precision and using a general summary prompt can be chosen.

Gemini 1.0 and Gemini 1.5 Google provides free API usage, though the summary generation speed is slower compared to GPT models. However, when considering the Recall of meeting summaries, the Gemini 1.5 Pro model using a general summary prompt performs the best. This is suitable for scenarios where it is important to record all discussion points, even if the summary includes some irrelevant information in the summary.

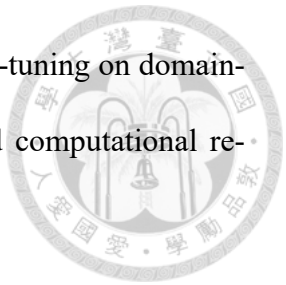
Llama 2 and Llama 3

These LLMs are provided by Meta and are available for free download and use. This experiment utilized the smallest versions of these models. Both Llama 2 and Llama 3 also offer models trained with higher parameters. For enterprises with the necessary internal technical and computational resources, deploying higher-tier Llama models can improve both summary quality and generation speed. Additionally, local deployment ensures a lower risk of data leakage.

DialogLED This model offers both an API and a downloadable pre-trained model

doi:10.6342/NTU202402787

for self-deployment. However, to improve the model's accuracy, fine-tuning on domain-specific data is necessary, which requires additional manpower and computational resources, resulting in higher costs.



doi:10.6342/NTU202402787

Chapter 5 Conclusion

This study evaluated the performance of various LLMs in generating meeting summaries, including GPT-3.5, GPT-4, Gemini 1.0, Gemini 1.5, Llama 2, and Llama 3. Using the verbatim data from the AMI Meeting Corpus, different preprocessing methods (Google Speech Recognition and Whisper Base transcription) and different prompts were applied to compare the effectiveness of these models in generating meeting summaries.

The main findings of this study are as follows:

Model Performance Comparison: Among all LLMs, GPT-4 demonstrated the best performance in generating meeting summaries, with higher Precision and F1-scores in most cases. However, when Precision is the primary consideration, GPT-3.5 outperformed in the general summary prompt design, indicating that GPT-3.5 can generate more precise summary content but with less coverage. Conversely, Gemini 1.5 Pro achieved the best Recall, especially in the general summary prompt design, indicating that it generated more content matching the reference summary, although with some redundant information.

Impact of Prompt Design: The prompt comparison experiment revealed that simplified prompts can improve the model's Recall but may also increase redundancy in the generated summary. Therefore, in practical applications, the appropriate prompt design should be chosen based on specific needs. Targeted summary prompts guide the model

doi:10.6342/NTU202402787

to focus on specific summary elements, making the generated meeting summaries more detailed, whereas general summary prompts allow the model to more flexibly capture key information.

Comparison of Traditional and LLMs: The pre-trained DialogLED model without fine-tuning had significantly lower ROUGE scores across all data sources compared to LLMs. However, language models fine-tuned with domain-specific knowledge produced more accurate summaries than LLMs. Therefore, fine-tuning is still necessary for applications requiring high-precision summary generation.

Although LLMs did not perform as well as fine-tuned traditional language models in this meeting summarization experiment, current technologies like Retrieval-Augmented Generation (RAG) and Low-Rank Adaptation (LoRA) can improve the summary generation quality of LLMs. RAG allows LLMs to retrieve relevant information to assist generation, enhancing the model's understanding of specific content. LoRA fine-tunes the model through low-rank matrix decomposition, reducing computational resource requirements. Enterprises can leverage these two techniques to potentially improve the accuracy of large language model-generated meeting summaries based on their specific meeting content.

doi:10.6342/NTU202402787

References

[1] Introducing meta llama 3: The most capable openly available llm to date. `https://ai.meta.com/blog/meta-llama-3/`. 2024.

[2] O. Abdel-Hamid, A.-r. Mohamed, H. Jiang, L. Deng, G. Penn, and D. Yu. Convolutional neural networks for speech recognition. *IEEE/ACM Transactions on audio, speech, and language processing*, 22(10):1533–1545, 2014.

[3] J. Achiam, S. Adler, S. Agarwal, L. Ahmad, I. Akkaya, F. L. Aleman, D. Almeida, J. Altenschmidt, S. Altman, S. Anadkat, et al. Gpt-4 technical report. *arXiv preprint arXiv:2303.08774*, 2023.

[4] S. Alshaina, A. John, and A. G. Nath. Multi-document abstractive summarization based on predicate argument structure. In *2017 IEEE International Conference on Signal Processing, Informatics, Communication and Energy Systems (SPICES)*, pages 1–6. IEEE, 2017.

[5] T. Brown, B. Mann, N. Ryder, M. Subbiah, J. D. Kaplan, P. Dhariwal, A. Neelakantan, P. Shyam, G. Sastry, A. Askell, et al. Language models are few-shot learners. *Advances in neural information processing systems*, 33:1877–1901, 2020.

[6] L. Dong, S. Xu, and B. Xu. Speech-transformer: a no-recurrence sequence-to-

doi:10.6342/NTU202402787

sequence model for speech recognition. In *2018 IEEE international conference on acoustics, speech and signal processing (ICASSP)*, pages 5884–5888. IEEE, 2018.

[7] G. Erkan and D. R. Radev. Lexrank: Graph-based lexical centrality as salience in text summarization. *Journal of Artificial Intelligence Research*, 22:457–479, Dec. 2004.

[8] A. Graves, N. Jaitly, and A.-r. Mohamed. Hybrid speech recognition with deep bidirectional lstm. In *2013 IEEE workshop on automatic speech recognition and understanding*, pages 273–278. IEEE, 2013.

[9] A. Graves, A.-r. Mohamed, and G. Hinton. Speech recognition with deep recurrent neural networks. In *2013 IEEE international conference on acoustics, speech and signal processing*, pages 6645–6649. Ieee, 2013.

[10] S. Gupta and S. K. Gupta. Abstractive summarization: An overview of the state of the art. *Expert Systems with Applications*, 121:49–65, 2019.

[11] G. Hinton, L. Deng, D. Yu, G. E. Dahl, A.-r. Mohamed, N. Jaitly, A. Senior, V. Vanhoucke, P. Nguyen, T. N. Sainath, and B. Kingsbury. Deep neural networks for acoustic modeling in speech recognition: The shared views of four research groups. *IEEE Signal Processing Magazine*, 29(6):82–97, 2012.

[12] A. Khan, N. Salim, H. Farman, M. Khan, B. Jan, A. Ahmad, I. Ahmed, and A. Paul. Abstractive text summarization based on improved semantic graph approach. *International Journal of Parallel Programming*, 46:992–1016, 2018.

[13] C.-Y. Lin. Rouge: A package for automatic evaluation of summaries. In *Text summarization branches out*, pages 74–81, 2004.

doi:10.6342/NTU202402787

[14] Y. Liu. Fine-tune bert for extractive summarization. *arXiv preprint arXiv:1903.10318*, 2019.

[15] R. Mihalcea and P. Tarau. Textrank: Bringing order into text. In *Proceedings of the 2004 conference on empirical methods in natural language processing*, pages 404–411, 2004.

[16] H. Nori, N. King, S. M. McKinney, D. Carignan, and E. Horvitz. Capabilities of gpt-4 on medical challenge problems. *arXiv preprint arXiv:2303.13375*, 2023.

[17] OpenAI, J. Achiam, S. Adler, S. Agarwal, L. Ahmad, I. Akkaya, F. L. Aleman, D. Almeida, J. Altenschmidt, S. Altman, S. Anadkat, R. Avila, I. Babuschkin, S. Balaji, V. Balcom, P. Baltescu, H. Bao, M. Bavarian, J. Belgum, I. Bello, J. Berdine, G. Bernadett-Shapiro, C. Berner, L. Bogdonoff, O. Boiko, M. Boyd, A.-L. Brakman, G. Brockman, T. Brooks, M. Brundage, K. Button, T. Cai, R. Campbell, A. Cann, B. Carey, C. Carlson, R. Carmichael, B. Chan, C. Chang, F. Chantzis, D. Chen, S. Chen, R. Chen, J. Chen, M. Chen, B. Chess, C. Cho, C. Chu, H. W. Chung, D. Cummings, J. Currier, Y. Dai, C. Decareaux, T. Degry, N. Deutsch, D. Deville, A. Dhar, D. Dohan, S. Dowling, S. Dunning, A. Ecoffet, A. Eleti, T. Eloundou, D. Farhi, L. Fedus, N. Felix, S. P. Fishman, J. Forte, I. Fulford, L. Gao, E. Georges, C. Gibson, V. Goel, T. Gogineni, G. Goh, R. Gontijo-Lopes, J. Gordon, M. Grafstein, S. Gray, R. Greene, J. Gross, S. S. Gu, Y. Guo, C. Hallacy, J. Han, J. Harris, Y. He, M. Heaton, J. Heidecke, C. Hesse, A. Hickey, W. Hickey, P. Hoeschele, B. Houghton, K. Hsu, S. Hu, X. Hu, J. Huizinga, S. Jain, S. Jain, J. Jang, A. Jiang, R. Jiang, H. Jin, D. Jin, S. Jomoto, B. Jonn, H. Jun, T. Kaftan, Łukasz Kaiser, A. Kamali, I. Kanitscheider, N. S. Keskar, T. Khan, L. Kilpatrick, J. W. Kim, C. Kim, Y. Kim, J. H. Kirchner, J. Kiros, M. Knight, D. Kokotajlo, Łukasz Kondraciuk,

doi:10.6342/NTU202402787

A. Kondrich, A. Konstantinidis, K. Kosic, G. Krueger, V. Kuo, M. Lampe, I. Lan, T. Lee, J. Leike, J. Leung, D. Levy, C. M. Li, R. Lim, M. Lin, S. Lin, M. Litwin, T. Lopez, R. Lowe, P. Lue, A. Makanju, K. Malfacini, S. Manning, T. Markov, Y. Markovski, B. Martin, K. Mayer, A. Mayne, B. McGrew, S. M. McKinney, C. McLeavey, P. McMillan, J. McNeil, D. Medina, A. Mehta, J. Menick, L. Metz, A. Mishchenko, P. Mishkin, V. Monaco, E. Morikawa, D. Mossing, T. Mu, M. Murati, O. Murk, D. Mély, A. Nair, R. Nakano, R. Nayak, A. Neelakantan, R. Ngo, H. Noh, L. Ouyang, C. O'Keefe, J. Pachocki, A. Paino, J. Palermo, A. Pantuliano, G. Parascandolo, J. Parish, E. Parparita, A. Passos, M. Pavlov, A. Peng, A. Perelman, F. de Avila Belbute Peres, M. Petrov, H. P. de Oliveira Pinto, Michael, Pokorny, M. Pokrass, V. H. Pong, T. Powell, A. Power, B. Power, E. Proehl, R. Puri, A. Radford, J. Rae, A. Ramesh, C. Raymond, F. Real, K. Rimbach, C. Ross, B. Rotsted, H. Roussez, N. Ryder, M. Saltarelli, T. Sanders, S. Santurkar, G. Sastry, H. Schmidt, D. Schnurr, J. Schulman, D. Selsam, K. Sheppard, T. Sherbakov, J. Shieh, S. Shoker, P. Shyam, S. Sidor, E. Sigler, M. Simens, J. Sitkin, K. Slama, I. Sohl, B. Sokolowsky, Y. Song, N. Staudacher, F. P. Such, N. Summers, I. Sutskever, J. Tang, N. Tezak, M. B. Thompson, P. Tillet, A. Tootoonchian, E. Tseng, P. Tuggle, N. Turley, J. Tworek, J. F. C. Uribe, A. Vallone, A. Vijayvergiya, C. Voss, C. Wainwright, J. J. Wang, A. Wang, B. Wang, J. Ward, J. Wei, C. Weinmann, A. Welihinda, P. Welinder, J. Weng, L. Weng, M. Wiethoff, D. Willner, C. Winter, S. Wolrich, H. Wong, L. Workman, S. Wu, J. Wu, M. Wu, K. Xiao, T. Xu, S. Yoo, K. Yu, Q. Yuan, W. Zaremba, R. Zellers, C. Zhang, M. Zhang, S. Zhao, T. Zheng, J. Zhuang, W. Zhuk, and B. Zoph. Gpt-4 technical report, 2024.

[18] A. Radford, J. W. Kim, T. Xu, G. Brockman, C. McLeavey, and I. Sutskever. Robust

doi:10.6342/NTU202402787

speech recognition via large-scale weak supervision, 2022.

[19] A. Radford, K. Narasimhan, T. Salimans, I. Sutskever, et al. Improving language understanding by generative pre-training. 2018.

[20] M. Razavi, R. Rasipuram, and M. Magimai-Doss. On modeling context-dependent clustered states: Comparing hmm/gmm, hybrid hmm/ann and kl-hmm approaches. In *2014 IEEE international conference on acoustics, speech and signal processing (ICASSP)*, pages 7659–7663. IEEE, 2014.

[21] M. Reid, N. Savinov, D. Teplyashin, D. Lepikhin, T. Lillicrap, J.-b. Alayrac, R. Soricut, A. Lazaridou, O. Firat, J. Schrittwieser, et al. Gemini 1.5: Unlocking multimodal understanding across millions of tokens of context. *arXiv preprint arXiv:2403.05530*, 2024.

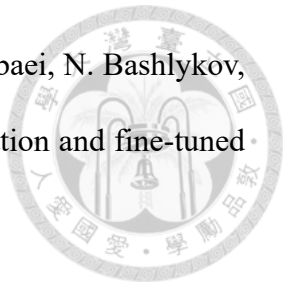
[22] H. Sak, A. Senior, and F. Beaufays. Long short-term memory based recurrent neural network architectures for large vocabulary speech recognition. *arXiv preprint arXiv:1402.1128*, 2014.

[23] J. Savelka. Unlocking practical applications in legal domain: Evaluation of gpt for zero-shot semantic annotation of legal texts. In *Proceedings of the Nineteenth International Conference on Artificial Intelligence and Law*, pages 447–451, 2023.

[24] P. Swietojanski, A. Ghoshal, and S. Renals. Revisiting hybrid and gmm-hmm system combination techniques. In *2013 IEEE International Conference on Acoustics, Speech and Signal Processing*, pages 6744–6748. IEEE, 2013.

[25] G. Team, R. Anil, S. Borgeaud, Y. Wu, J.-B. Alayrac, J. Yu, R. Soricut, J. Schalkwyk, A. M. Dai, A. Hauth, et al. Gemini: a family of highly capable multimodal models. *arXiv preprint arXiv:2312.11805*, 2023.

doi:10.6342/NTU202402787



[26] H. Touvron, L. Martin, K. Stone, P. Albert, A. Almahairi, Y. Babaei, N. Bashlykov, S. Batra, P. Bhargava, S. Bhosale, et al. Llama 2: Open foundation and fine-tuned chat models. *arXiv preprint arXiv:2307.09288*, 2023.

[27] A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A. N. Gomez, Ł. Kaiser, and I. Polosukhin. Attention is all you need. *Advances in neural information processing systems*, 30, 2017.

[28] A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A. N. Gomez, L. u. Kaiser, and I. Polosukhin. Attention is all you need. In I. Guyon, U. V. Luxburg, S. Bengio, H. Wallach, R. Fergus, S. Vishwanathan, and R. Garnett, editors, *Advances in Neural Information Processing Systems*, volume 30. Curran Associates, Inc., 2017.

[29] D. Wang, P. Liu, Y. Zheng, X. Qiu, and X. Huang. Heterogeneous graph neural networks for extractive document summarization. *arXiv preprint arXiv:2004.12393*, 2020.

[30] J. Ye, X. Chen, N. Xu, C. Zu, Z. Shao, S. Liu, Y. Cui, Z. Zhou, C. Gong, Y. Shen, J. Zhou, S. Chen, T. Gui, Q. Zhang, and X. Huang. A comprehensive capability analysis of gpt-3 and gpt-3.5 series models, 2023.

[31] J. Zhang, Y. Zhao, M. Saleh, and P. Liu. Pegasus: Pre-training with extracted gap-sentences for abstractive summarization. In *International Conference on Machine Learning*, pages 11328–11339. PMLR, 2020.

[32] Q. Zhou, N. Yang, F. Wei, S. Huang, M. Zhou, and T. Zhao. Neural document summarization by jointly learning to score and select sentences. *arXiv preprint arXiv:1807.02305*, 2018.

doi:10.6342/NTU202402787