國立臺灣師範大學理學院資訊工程學系

碩士論文

Department of Computer Science and Information Engineering
College of Science
National Taiwan Normal University

Master's Thesis

應用兩階段生成模型於會議摘要之研究

A Study of Extract-then-Generate Model for Meeting Summarization

黃怡萍

Huang, Yi-Ping

指導教授： 陳柏琳 博士

Advisor: Berlin Chen, Ph.D.

中華民國 112 年 7 月

July 2023

致謝

兩年的碩士時光轉瞬即逝，這兩年間學習和接觸到的遠比我想像的多，讓我豐富了許多寶貴的學術知識和人生經驗。我很慶幸兩年前的自己選擇走上研究所這條路，從一開始的模糊和不確定到漸漸熟悉並明瞭研究方向，這一切都是值得珍惜的回憶。

首先，我要感謝我的指導教授陳柏琳老師，總是以耐心和關心的態度幫助我解決問題。在研究上遇到困難時，適當的給予想法和目標，也經常在每週的會議中關懷和勉勵我們。讓我們無論在學術研究上或是生活的待人處事上，都學習到了許多。此外，也很感謝讓我有機會參與各類的產學合作和計畫，透過每次的會議精進自己的專業能力，與實際產業的直接接觸也拓展了我的視野。

感謝洪志偉博士、陳冠宇博士和曾厚強博士願意來擔任我的口試委員，並給予我論文和實驗方面的建議及指導，讓我在撰寫論文和實驗設計上能夠有所改進。

接著是實驗室的成員們，覺得自己很幸運能夠進入這間實驗室，我要感謝實驗室的學長姐天宏、必成、馨偉、沁穎、俊廷、又升、詩諺，總是無私的分享研究上的經驗和想法，每次的討論都能夠從你們身上吸收到新的知識。謝謝同屆的孟庭、姿儀、盈慈、詣承、子霆、泓壬、皓天、冠勳，每次的一起修課、做專題和討論研究，都是在歡樂的氣氛中度過的，我們都專注於不同的研究領域，透過討論可以從你們身上獲得不同領域的知識。還有學弟妹恩倫、孟欣、玟瑄、憶婷、俞華，有你們的加入讓實驗室更加充滿活力，也讓我意識到自己應該更加努力。

最後，要感謝一直給予我鼓勵和支持的家人，在我遇到困難時，總是盡自己所能幫助我，讓我知道未來就算再艱難，我也有最堅強的後盾。因此，我也想把此論文的成果獻給我的家人。

怡萍 謹誌

摘　要

近年來，由於疫情的影響和遠端工作的普及，線上會議和視訊交流平台的使用變得更加廣泛。但隨之而來的問題是，會議記錄往往包含許多分散的資訊，要在大量的對話中擷取和理解關鍵資訊是困難的，且隨著會議越來越頻繁，意味著參與者需要在有限的時間內掌握會議的要點，以便在忙碌的日程中做出明智的決策。在這樣的情境下，能夠從會議紀錄中自動辨識和摘要出關鍵資訊的技術變得更為重要。

自動文件摘要主要分為擷取式 (Extractive) 和重寫式 (Abstractive) 兩種方法，擷取式摘要透過計算原始文件中每個句子的重要性分數，選擇得分高的句子並將它們組合起來成為摘要。重寫式摘要透過對原始文件的理解重新改寫句子，生成出一個簡潔且包含原始文件中核心內容的摘要。由於對話中的話語經常是不流暢且資訊分散的，使用擷取式摘要容易擷取出不完整的句子，造成可讀性不高。目前在會議摘要任務中，主要的應用是能夠將原始語句改寫的重寫式摘要。雖然已有許多相關的研究被提出，重寫式的方法應用在會議摘要中仍面臨幾個普遍性的限制，包括輸入長度問題、複雜的對話結構，以及缺乏訓練資料與事實不一致，而這些問題也是提高會議摘要模型效能的關鍵。

本論文專注在「輸入長度問題」和「對話式結構」的研究，提出了一個先擷取後生成的會議摘要模型架構，在擷取階段設計了三種方法來選擇重要的文本片段，分別是異質圖神經網路模型、對話語篇剖析和文本相似度。在生成階段使用先進的生成式預訓練模型。實驗結果顯示，提出的方法透過微調基線模型，可以達到效果提升。

關鍵詞：會議摘要，自動文件摘要，自然語言處理，異質圖神經網路，對話語篇剖析，生成式模型

Abstract

In recent years, the use of online meetings and video communication platforms has become more widespread due to the impact of the pandemic and the popularity of remote work. However, this trend brings along certain challenges. Meeting transcripts often contain scattered information, making it difficult to extract and understand key details from a large volume of conversations. Additionally, as meetings become increasingly frequent, participants need to grasp the main points of the discussions within limited time to make informed decisions amidst their busy schedules. In such a context, the ability to automatically identify and summarize crucial information from meeting transcripts becomes even more important.

Automatic document summarization can be categorized into two main approaches: extractive and abstractive. Extractive summarization calculates the importance scores of each sentence in the original document and selects high-scoring sentences to form the summary. On the other hand, abstractive summarization involves understanding the original document and rewriting sentences to generate a concise summary that captures the core content. Extractive summarization is prone to extracting incomplete sentences due to the often disjointed and scattered nature of dialogues, leading to reduced readability. Currently, the primary application in meeting summarization tasks is abstractive summarization, which involves rewriting the original sentences. Despite the numerous related studies, the application of abstractive methods in meeting summarization still faces several common limitations, including input length constraints, complex dialogue structures, the lack of training data, and consistency with facts. Addressing these issues is crucial for improving the performance of meeting summarization models.

This paper focuses on the research of "input length constraints" and "dialogue-style structures" and proposes a meeting summarization model architecture that follows an extract-then-generate approach. In the extraction phase, three methods are designed to select important text segments: heterogeneous graph neural network model, dialogue discourse parsing, and cosine similarity. Advanced generative pre-training model are employed in the generation phase. Experimental results demonstrate that the proposed

approach, through fine-tuning the baseline model, achieves performance improvements.

Keywords: Meeting Summarization, Automatic Document Summarization, Natural Language Processing, Heterogeneous Graph Neural Network, Dialogue Discourse Parsing, Generative Model

目　錄

第一章　緒論 1

1.1　研究背景與動機 1

1.2　研究內容 3

1.3　研究貢獻 6

1.4　論文架構 9

第二章　文獻探討 10

2.1　文件摘要背景概述 10

2.2　會議摘要背景概述 13

2.3　會議摘要方法分類 14

2.3.1　擷取式方法 15

2.3.2　重寫式方法 16

2.4　先擷取後生成方法 18

2.5　對話語篇剖析 20

2.5.1　語篇結構分類 21

2.5.2　評估方法與資料集 22

2.5.3　對話語篇剖析方法 24

2.6　文本相似度的應用 25

第三章　研究方法 27

3.1　問題定義與假設 27

3.2 兩階段摘要模型 27
3.2.1 模型架構 27
3.2.2 擷取式摘要 28
3.2.3 對話語篇剖析 30
3.2.4 文本相似度 31
3.2.5 重寫式摘要 31
第四章 實驗設計與結果 33
4.1 實驗語料 33
4.2 評估方法 33
4.3 實驗結果 35
4.3.1 基礎實驗 35
4.3.2 擷取式摘要結果 37
4.3.3 對話語篇剖析實驗 38
第五章 結論與未來展望 39
參考文獻 40

表　目　錄

表 1. 1 處理輸入長度問題方法分類.....4

表 2. 1 深度學習方法分類.....18

表 2. 2 對話語篇剖析模型分類.....25

表 3. 1 對話相關的雜訊.....32

表 4. 1 AMI 資料集的統計數據.....33

表 4. 2 微調方法實驗結果.....35

表 4. 3 兩階段方法初步結果.....36

表 4. 4 評估模型泛化能力.....36

表 4. 5 兩階段 ROUGE 分數比較.....37

圖　目　錄

圖 1. 1 會議摘要系統示意圖.....2

圖 1. 2 對話語篇剖析示意圖.....5

圖 1. 3 單詞節點和句子節點的迭代更新.....6

圖 1. 4 四種類型的依賴結構示意圖.....7

圖 2. 1 RNN 架構示意圖.....11

圖 2. 2 TRANSFORMER 架構示意圖.....13

圖 2. 3 會議摘要的時間線圖.....14

圖 2. 4 對話語篇剖析與樹狀結構.....20

圖 2. 5 STAC 語料庫中關係類別統計圖.....23

圖 3. 1 系統架構圖.....28

圖 3. 2 異質圖神經網路模型.....28

圖 4. 1 混淆矩陣.....34

圖 4. 2 預測結果畫成生成樹.....38

第一章　緒論

1.1　研究背景與動機

隨著資訊時代的來臨，人們對於資訊的需求量越加龐大，如何有效地從大量的資訊中擷取並整理出關鍵資訊已成為一個迫切的需求。此外，隨著網路技術的發展，多人的會議和對話變得更加普遍。近年來，由於疫情的影響和遠端工作的普及，線上會議和視訊交流平台的使用也變得更加廣泛。但隨之而來的問題是，會議記錄往往包含許多分散的資訊，要在大量的對話中擷取和理解關鍵資訊是困難的，且隨著會議越來越頻繁，意味著參與者需要在有限的時間內掌握會議的要點，以便在忙碌的日程中做出明智的決策。在這樣的情境下，能夠從會議紀錄中自動辨識和摘要出關鍵資訊的技術變得更為重要。

在自然語言處理 (Natural Language Processing, NLP) 中，自動文件摘要 (Automatic Document Summarization) 是一項重要的任務，它可以從長文件中擷取出重要資訊，並產生出簡短且關鍵的內容。自動文件摘要主要分為擷取式 (Extractive) 和重寫式 (Abstractive) 兩種方法：

- **擷取式摘要**：透過計算原始文件中每個句子的重要性分數，選擇得分高的句子並將它們組合起來成為摘要。
    - 優點：保留了原始文件的句子或段落，能夠確保內容的真實性，且相對於重寫式摘要更加高效，因為它僅需要選擇和組合現有的句子作為摘要。
    - 缺點：缺乏創造性，因為它只是從原始文件中選擇現有的內容，無法生成全新的語句。其次，由於擷取摘要只使用原始文本中的內容，可能導致摘要中的句子來自不同的上下文，資訊連貫性不足。
- **重寫式摘要**：透過對原始文件的理解重新改寫句子，生成出一個簡潔且包含原始文件中核心內容的摘要。
    - 優點：具有更高的創造性和靈活性。通過重新組織和改寫原始文本的句子來生成全新的摘要，這種方法能夠產生結構化且更具流

暢性的摘要。此外，它可以去除冗長的句子、刪除細節或使用更簡單的表達方式來傳達相同的意思，從而提供更精簡的摘要。

- 缺點：通常需要大量的訓練資料和計算資源，以便生成高質量的摘要。其次，生成一個有邏輯且流暢的摘要是一項困難的任務，因為它涉及到理解原始文本、抽取關鍵資訊和生成新的句子。另外，生成模型也容易帶來不準確的資訊。

由於對話中的話語經常是不流暢且資訊分散的，使用擷取式摘要容易擷取出不完整的句子，造成可讀性不高。目前在會議摘要任務中，主要的應用是能夠將原始語句改寫的重寫式摘要。重寫式摘要應用於會議中又可以大致分為基於圖 (Graph-based) 、基於模板 (Template-based) 和基於深度學習 (Deep Learning-based) 的方法 [1]。其中，深度學習方法表現最佳。在深度學習方法中，DialogLM [2] 為目前會議摘要任務的 State-of-the-art 模型，透過設計多個與對話相關的預訓練任務，讓模型學習對話結構，進而理解和預測對話內容。

雖然已有許多相關的研究被提出，重寫式的方法應用在會議摘要中仍面臨幾個普遍性的限制，包括輸入長度問題、複雜的對話結構，以及缺乏訓練資料與事實不一致，而這些問題也是提高會議摘要模型效能的關鍵。本論文針對於上前兩個問題「輸入長度問題」和「複雜的對話結構」，分別提出改進的方法。對於長度問題，提出了先擷取後生成的兩階段模型，並且在擷取階段也探討了幾種不同的方式。對於對話結構，額外引入了對話語篇剖析。

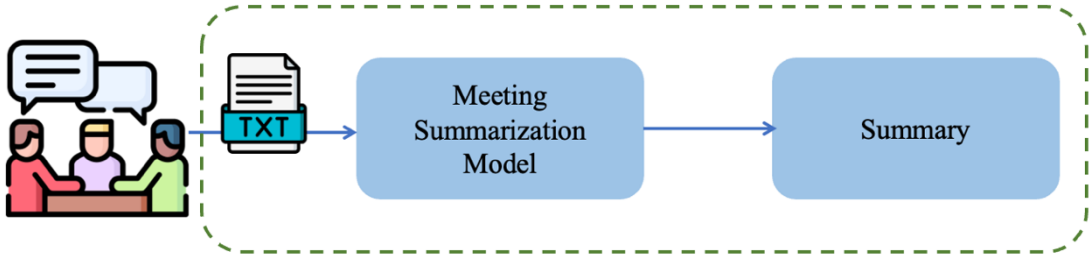

圖 1. 1 會議摘要系統示意圖

圖 1.1 為會議摘要系統之示意圖，會議摘要的任務是要將經過人工轉錄的會議文本，透過會議摘要模型，生成出此會議的摘要內容。

1.2 研究內容

會議摘要與傳統文件摘要面臨著不同的挑戰，其中一些與缺乏訓練資料或模型限制有關，而其他與會議或多人對話之間的互動有關 [3]，以下列舉四個常見的挑戰：

1. **輸入長度問題**：會議轉錄文件的長度通常比傳統新聞文件長許多，並且往往超過了基於 Transformer [4] 的模型能夠處理的長度。平均而言，會議語料 AMI [5] 的長度為 4,339 個單詞，而新聞語料 CNN/DailyMail 則為 781 個單詞。
2. **對話式結構**：會議中的發言通常是即興的，可能包含片段或不合語法的話語、自我更正、重複語句或停頓等等，導致資訊分散。此外，多人對話也容易出現同時說話和每個人的說話風格不同。
3. **缺乏訓練資料**：雖然自動語音辨識的進步使得高質量的會議語音轉錄變得容易獲得，但是生成摘要和會議紀錄需要龐大的人力成本。此外，會議內容的隱私也是一大問題，通常涉及到公司或單位的機密，因此無法公開使用。
4. **事實不一致**：幻覺 (Hallucination) 是生成式模型普遍會遇到的問題，若又在缺乏訓練資料的情況下，就更容易產生。會導致生成出的摘要包含不正確的訊息，降低了摘要的可性度和可用性。

本論文主要關注於上述問題中的前兩項：「輸入長度問題」和「對話式結構」。 [6] 將先前提出用來處理會議摘要長度問題的模型分成四種，如表 1.1 所示。第一種是採用**稀疏注意力 (Sparse Attention) 機制**，降低原本自注意力 (Self Attention) 機制模型的複雜度，通常基於注意力權重的分佈性質，將輸入序列分成多個子集，只對子集進行注意力計算，這樣可以減少模型的計算量，進而能夠關注更多的上下文。第二種是**先擷取後生成 (Extract-then-Generate)** 的方法，先從輸入文本中擷取出重要的部分或關鍵資訊，然後再使用生成模型進行摘要生成，擷取過程可以基於各種技術，例如關鍵詞擷取、實體辨識或句子相似度排序等，這樣的方法能夠減少生成模型需要處理的資訊量，提高生成摘要

的效率和準確性。第三種是**分而治之 (Divide-and-Conquer)**，分別對每個片段進行摘要生成，然後再將摘要合併成整體的會議摘要，這樣的分段式方法能夠確保對每個片段的充分理解和摘要生成。第四種是**階層式模型 (Hierarchical Model)**，對話語的不同結構做建模來改進摘要模型，例如 [7] 構建了一個包含語篇級別資訊和語者角色的分層結構。

表 1. 1 處理輸入長度問題方法分類

稀疏注意力機制 (Sparse Attention Mechanism)	• 降低原本自注意力 (Self Attention) 機制模型的複雜度 • 通常基於注意力權重的分佈性質，將輸入序列分成多個子集，只對子集進行注意力計算
先擷取後生成 (Extract-then-Generate)	• 先從輸入文本中擷取出重要的部分或關鍵資訊，然後再使用生成模型進行摘要生成 • 減少生成模型需要處理的資訊量，提高生成摘要的效率和準確性
分而治之 (Divide-and-Conquer)	• 分別對每個片段進行摘要生成，然後再將摘要合併成整體的會議摘要 • 確保對每個片段的充分理解和摘要生成
階層式模型 (Hierarchical Model)	• 將輸入文本分成不同的階層或段落，分別對不同階層的結構建模 • 這些階層可以是段落、句子、語篇、語者等

其中，稀疏注意力機制雖有效透過減少模型的複雜度來關注更多上下文，但這種方法犧牲了部分可關注的區域，並且減弱了預訓練的好處。而將輸入文本分成多個片段分別摘要的方法，沒有捕捉到部分之間的上下文依賴關係，並且假設輸入具有一定的結構。階層式模型則是較關注於模型的效能，而不是減少記憶體和計算成本。相比之下，先擷取後生成的方法能夠沿用傳統做摘要的擷取式技術，再透過新穎的生成式摘要模型達到目的。先擷取後生成方法具有較高的可解釋性且較接近人類一般在做摘要的模式，因為我們通常會先從整篇

文章中找出關鍵字詞，再用自己的話改寫成摘要。此外，它同時結合了擷取式和重寫式兩種摘要的方法。因此，本論文選擇了先擷取後生成的方法作為模型的架構。

在對話式結構的部分，因為會議是一個動態的資訊交流過程，相比於傳統文件較不正式、冗長且結構性較弱 [8]，採用順序地建模會議文本會使得話語之間豐富的互動關係受到限制。對話語篇剖析能夠提供話語之間的預定義關係 [9] ，明確標示出話語之間的資訊流動和互動，進而建構出整個會議的結構。本論文透過對話語篇剖析來選擇較具有架構的文本片段。

圖 1.2 展示了對話資料集 STAC [10] 在經過對話語篇剖析後所畫出的結果，其中包含兩個話語間的鏈結，以及鏈結之間的關係，得到的結果可以再進一步建構成生成樹。

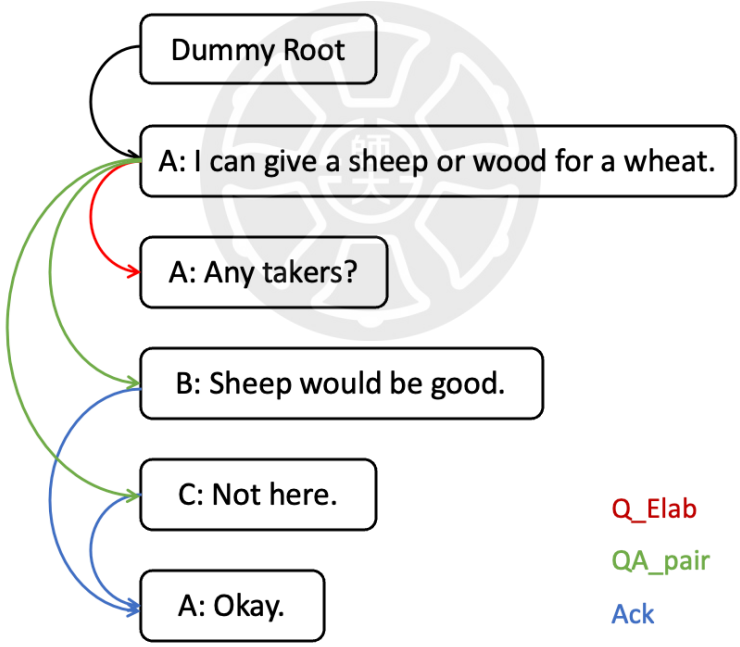

圖 1. 2 對話語篇剖析示意圖

最後，計算句子與整個會議的餘弦相似度是一種基於文本相似度的句子選取方法。這種方法將每個句子表示為向量，並通過計算句子向量與整個會議向量之間的餘弦相似度來評估句子的重要性。相似度越高的句子被認為是與整個

會議主題相關性較高的句子，因此被選取作為摘要的內容。這樣的方法可以捕捉到句子與整體會議之間的相似性，選取具有代表性的句子。

1.3 研究貢獻

本論文提出的模型為先擷取後生成的兩階段架構。在擷取模型的部分，為了要分析對話結構和文本相似度對於摘要模型的重要性，本論文嘗試了三種選取句子的方法，分別是圖神經網路模型、對話語篇剖析和透過計算句子與整個會議的餘弦相似度來選擇重要的句子。在擷取式摘要任務上，建模句間關係的方法主要分為兩大類，序列模型和以圖為核心架構的模型，其中序列模型較難捕捉到句子級別的長距離依賴關係，且容易過於依賴上下文的局部資訊，這對於有冗長文本和複雜架構的會議而言是不利的。相較之下，基於全局資訊的圖結構更加適用於會議摘要任務。

首先，HeterSumGraph [11] 是異質圖神經網路架構的擷取式摘要模型，透過引入單詞節點來增加句子間的關係，讓單詞節點和句子節點之間反覆迭代更新，共同出現的單詞可以更新句子間關係，而出現在越多句子中的單詞也會變得越重要，最後會得到更新完後的句子表示，其中擁有越多相同單詞的句子間關係會越緊密。迭代更新的方法如圖 1.3 所示，每一輪會更新一個單詞或句子的節點表示。

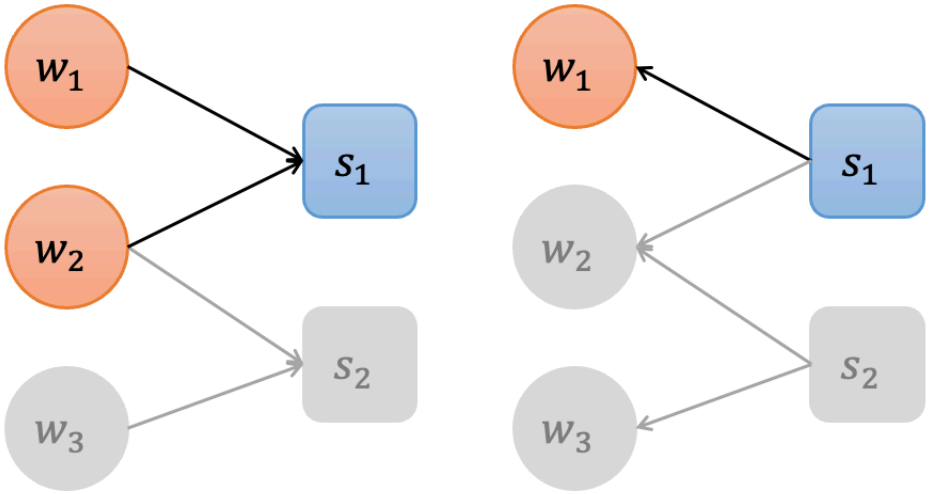

圖 1. 3 單詞節點和句子節點的迭代更新

接著，對話語篇剖析模型 SDDP [12] 通過在對話開始處添加一個虛擬根節點 (Dummy Root)，將整體結構建構成帶標籤的多根非投射生成樹 (Labeled Multi-Root Non-Projective Spanning Tree) ，也是第一個提出將話語 (Utterances，用 V 表示) 、鏈結 (Links，用 E 表示) 和鏈結關係 (Link Labels，用 R 表示) 建構成三維空間 $G(V,E,R)$，聯合預測語篇鏈結和關係的方法。其中，編碼器是完全端到端的，並且可以同時保持結構資訊。解碼器在統一的鏈結和關係空間上運行修改後的生成樹解碼算法。

圖 1.4 展示了四種類型的依賴結構，單根 (Single-Root) 指生成樹僅有一個根節點，所有節點都是從此根節點延伸出來，建構出的生成樹會是單一的，且所有節點之間的路徑都是唯一的。多根 (Multi-Root) 則可以有多個根節點，這些根節點分別連接到不同的子樹，建構出的生成樹可能不是單一的。在依存句法剖析 (Dependency Parsing) 中，投射性 (Projective) 指沒有交叉或重疊的弧線，每個詞彙之間的依存關係是清晰且無歧義的，使得句子中的所有詞彙在句法樹中形成了一個連續的投射性結構。若句子中的詞彙形成了交叉或重疊的弧線，則稱為非投射性 (Non-Projective) ，這表示存在某些詞彙之間的依存關係是無法以一個簡單的投射性結構來表示的，而需要更複雜的樹狀結構。

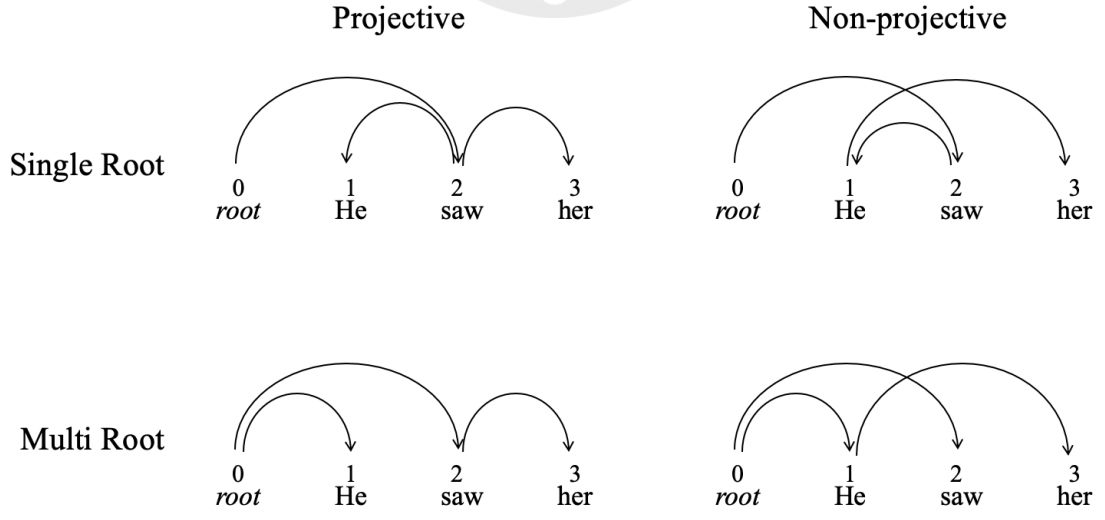

圖 1. 4 四種類型的依賴結構示意圖

最後，文本相似度在傳統文件的擷取式摘要任務上是常見的做法，餘弦相似度 (Cosine Similarity) 是一種衡量兩個向量之間相似度的指標，透過計算兩個向量之間夾角的餘弦值來求得相似度，並且相似度衡量的是兩個向量的方向是否相似，而不考慮大小或長度。在本論文中，運用餘弦相似度計算每個分割出的文本片段與原始整個會議的相似度，從中選擇相似度高的片段作為結果。

在生成模型的部分，DialogLM 是一個用於長對話理解和摘要的預訓練模型，基於統一語言模型 UniLM [13] 做改良。從網路架構來看，UniLM 是和 BERT [14] 相同的編碼器架構，但在預訓練任務上引入了編碼器—解碼器 (Encoder-Decoder) 架構，因此可以像遮罩語言模型 (Masked Language Model) 一樣利用遮罩標識 [MASK] 的上下文進行訓練，也可以像編碼器—解碼器架構的模型一樣先對輸入文本進行編碼，再從左到右生成序列。有鑒於 UniLM 在會議摘要任務上存在兩個限制，分別是在預訓練時沒有用到對話格式的訓練資料和預訓練過程中使用的皆為短文本。作者根據這兩個限制做了改良，提出一種基於窗口 (Window-based) 的去噪方法來進行生成預訓練，從整個會議記錄中隨機選取 10% 會議長度的連續文本作為窗口，在預訓練時，根據對話的性質設計了多個任務，包含對於語者、說話內容的遮罩 (Masking)、話語間的合併和分割，及對話順序的打亂。透過預測出被遮住和改變順序的語者或話語來學習對話結構。因為 DialogLM 在會議摘要上達到很好的效能，本論文使用 DialogLM 作為生成模型，並分析了模型對於對話架構和連貫性的理解。

本論文的貢獻：

1. 提出了一個適用於會議摘要的重寫式模型，運用兩階段先擷取後生成的架構。
2. 在擷取階段分析了三種不同的方法來選擇重要的文本片段。分別為圖神經網路模型、對話語篇剖析和計算句子與整個會議的餘弦相似度。
3. 探討了會議摘要中的「輸入長度問題」和「對話式結構」兩個限制。透過兩階段模型改進「輸入長度問題」，對於「對話式結構」額外引入了對話語篇剖析。
4. 結果顯示，透過微調基線模型達到效果提升。

1.4 論文架構

本論文之章節安排如下：

第二章　回顧過去會議摘要領域上相關的研究，包含背景和方法分類。背景會從文件摘要和對話摘要再延伸到會議摘要。方法會分為擷取式和重寫式以及各自作法的細分。另外，還包含本論文使用的「先擷取後生成」架構之相關研究，和對話語篇剖析及文本相似度在摘要領域的應用。

第三章　介紹本論文的問題定義和提出的方法架構。在方法架構的地方會就擷取式摘要、對話語篇剖析、文本相似度和重寫式摘要分別說明。

第四章　本論文的實驗設定與結果。包含實驗語料和實驗結果，並探討各種方法的差異及與過去的研究和基線模型的效能做比較。

第五章　本論文的總結及未來展望。

第二章　文獻探討

2.1　文件摘要背景概述

自動文件摘要 (Automatic Document Summarization) 可以根據不同的方式分類，根據輸入的類型分為單文件摘要跟多文件摘要，單文件摘要從單個文件中擷取重要的資訊，適用於新聞文章等單一來源的文件，多文件摘要針對多個相關文件或文件集合進行摘要生成，根據某個主題從多個文件中擷取重要資訊，適用於網路搜索結果、研究領域的文獻回顧等。根據摘要方法分為擷取式摘要和重寫式摘要。擷取式摘要是一種從原始文件中直接擷取重要句子或段落的摘要方法。透過計算句子或段落的重要性得分，並選擇得分最高的內容來構成摘要。重寫式摘要則是一種基於自然語言生成的方法，它通過理解原始文件的內容，並根據一定的摘要目標重新組織和重寫文本來生成摘要。

最初，最大邊際相關性（Maximal Marginal Relevance, MMR）被提出應用於擷取式文件摘要的方法 [15] ，透過最大邊際相關性來選擇最具相關性和多樣性的句子，同時達到減少摘要的冗餘 (Redundancy)。

$$MMR(Q, C, R) = Arg \max_{D_i \in R \setminus S} [\lambda(Sim_1(D_i, Q) - (1 - \lambda) \max_{D_j \in S} Sim_2(D_i, D_j))] \quad (1)$$

隨後，整數線性規劃法（Integer Linear Programming, ILP）也被引入到自動文件摘要中， [16] 透過 ILP 的條件約束 (Constraint) 讓摘要能夠在相關性和冗餘性之間取得平衡。

$$\begin{aligned} \text{Maximize} \quad & c^T x \\ \text{Subject to} \quad & Ax \leq b, \\ & x \geq 0, \\ \text{And} \quad & x \in \mathbb{Z}^n, \end{aligned} \quad (2)$$

在評估句子重要性方面，傳統的基於圖的方法有 LexRank [17] 和 TextRank [18]，在 TextRank 中每個句子會被表示為一個節點，根據詞的相似性建立兩個

句子之間的無向邊。將一個句子表示為一組詞：$S_i = w_1^i, w_2^i, \dots, w_{|S_i|}^i$，則兩個句子 S_i 和 S_J 之間的相似性定義如式 (3)：

$$Sim(S_i, S_j) = \frac{|w_k : w_k \in S_i \wedge w_k \in S_j|}{\log\left(|S_i| + log(|S_j|)\right)} \tag{3}$$

接著，許多基於神經網路的模型被提出，例如 [19] [20] 分別使用循環神經網路（Recurrent Neural Network, RNN）於重寫式和擷取式摘要任務上。

$$s(t) = f\left(Vx(t) + Us(t-1)\right) \tag{4}$$

$$o(t) = Ws(t) \tag{5}$$

$$o(t) = f_1\left(Ws(t)\right) = f_1\left(Wf_2\left(Vx(t) + Us(t-1)\right)\right), \forall t \tag{6}$$

上面為 RNN 的核心算法。其中，$x(t)$ 為第 t 時間步的輸入，$s(t)$ 為第 t 時間步的隱藏狀態，$o(t)$ 為第 t 時間步的輸出。V 是輸入到隱藏狀態的權重矩陣，W 是隱藏狀態到輸出的權重矩陣，U 是 t 到 $t+1$ 時間步的權重矩陣。算式 (3) 為隱藏狀態 $s(t)$ 的更新公式，其中 $f()$ 為激活函數，通常為 $tanh$ 或 $ReLU$。算式 (4) 為輸出 $o(t)$ 的計算方法。算式 (6) 則為將 (4) 帶入到 (5)，並加上激活函數的最終輸出結果。

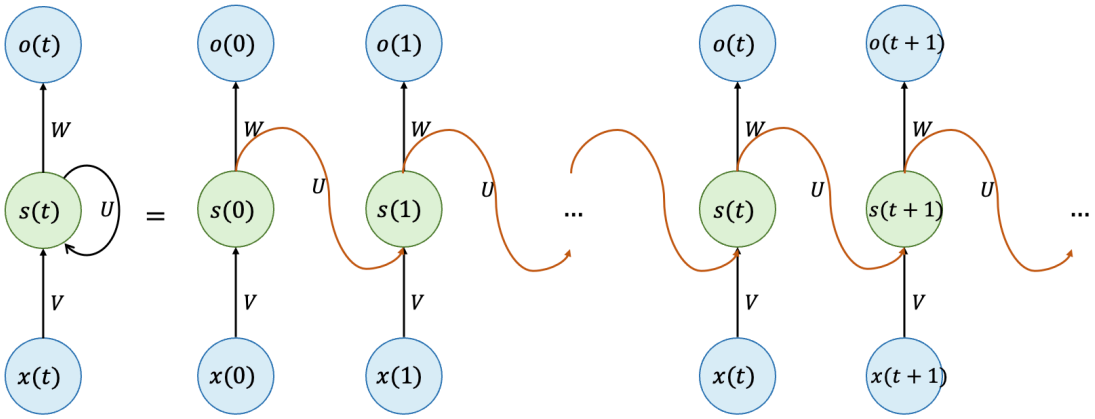

圖 2. 1 RNN 架構示意圖

然而，隨著 Transformer [4] 架構的引入，預訓練語言模型在文件摘要任務中的應用取得了重大突破。Transformer 架構通過在未標記的原始語料上進行語言模型的預訓練，並通過微調來適應特定的下游任務，實現了自然語言處理的遷移學習。像是 BERT 、GPT [21]、BART [22] 、UniLM [13] 和 Pegasus [23] 等預訓練語言模型在文件摘要領域上都取得了巨大成功。這些模型也是目前文件摘要研究中的主要方法。

Transformer 是一種基於自注意力 (Self-Attention) 的神經網路架構，核心是透過自注意力機制來捕捉輸入序列中不同位置之間的關係，由一個編碼器和一個解碼器組成。注意力機制的公式如下：

$$q_i = x_i W_Q \tag{7}$$

$$k_i = x_i W_K \tag{8}$$

$$v_i = x_i W_V \tag{9}$$

首先，對於一個輸入序列 x ，透過三個矩陣 W_Q 、W_k 和 W_v ，分別去計算 Query 向量、Key 向量和 Value 向量。

$$\alpha_{ij} = \frac{q_i \cdot k_j}{\sqrt{d_k}} \tag{10}$$

此時，每個 x 都會有三個向量。注意力權重的算法為將 x_i 的 Query 向量和所有 k_j 的 Key 向量做內積，再除以向量維度的開根號 $\sqrt{d_k}$ ，對內積進行正歸化以避免出現極值，因為當 K 的維度越大，內積就會越大，這個方法被稱為 Scaled Dot-Product Attention。

$$x_i = \sum_j softmax(\alpha_{ij}) v_j \tag{11}$$

x_i 的輸出向量即為把所有 v 透過注意力權重加權的總和。最後，自注意力機制的算法如下：

$$Attention(Q, K, V) = softmax\left(\frac{QK^T}{\sqrt{d_k}}\right)V \tag{12}$$

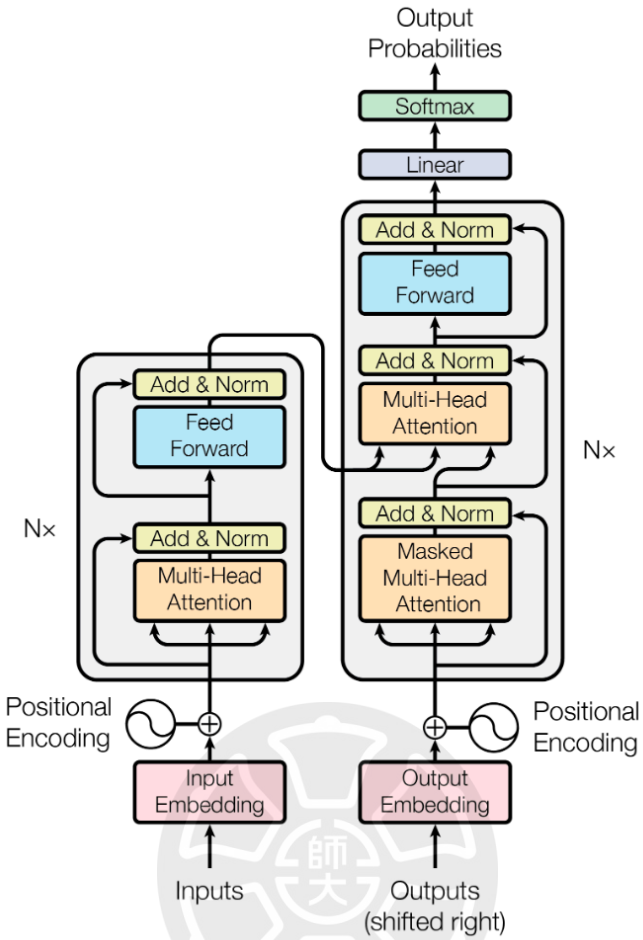

圖 2. 2 Transformer 架構示意圖

2.2 會議摘要背景概述

在探討會議摘要之前，首先要提到對話摘要 (Dialogue Summarization)，對話摘要是一個近年越來越受歡迎的領域，原因在於現今人們無時無刻都在使用通訊軟體交流，此種對話方式已經成為生活中不可或缺的一部分。

對話摘要的領域包括會議、線上聊天、客服或醫療對話等等。對於不同領域也有各自適合的摘要方法，因此有不同的語料庫被提出，像是 SAMSum [24] 是一個常見用於對話摘要的語料庫，由多位有著流利英語的語言學家模擬生活中自然的訊息對話建構而成的，其中可能包含表情符號或是打字錯誤等等，再以第三人稱方式提供每個對話一個參考摘要。MediaSUM [25] 也是值得一提的對話摘要語料庫，它是一個大型的媒體採訪資料集，建構方式是從廣播新聞台 NPR 和 CNN 收集採訪記錄，並使用其主題描述作為摘要，有包含來自多個領

域的複雜多人對話。客服領域有 TODSum [26] 是專注於任務導向 (Task-oriented) 的對話摘要，因為客服對話通常為一對一且用戶具有明確的意圖，作者引入對話狀態 (Dialog State) 來增強摘要。醫療對話則有 [27] 根據醫生和患者之間的對話記錄生成臨床摘要，他們資料集包含現實生活中患者與醫生就診的記錄，其中姓名等敏感資訊已被去除，作者在探索從擷取式到重寫式的方法後，提出了 CLUSTER2SENT 算法，先將相關話語聚集在一起，再為每個聚集生成一個摘要句子。

會議摘要是對話摘要領域中的一種類別，先前的研究主要集中在擷取式會議摘要。採用各種特徵來檢測重要的話語，如關鍵詞、主題和語者特徵等等。然而，由於會議具有多參與者的特點，資訊在會議中分散且不連貫，使得擷取式方法不適合會議摘要。近年來，重寫式的會議摘要越來越受到關注 [28]。在接下來的 2.3 節中，將根據會議摘要的方法進行分類，包含早期的擷取式到近期的重寫式，以及兩種類別中方法的更細部分類。

2.3 會議摘要方法分類

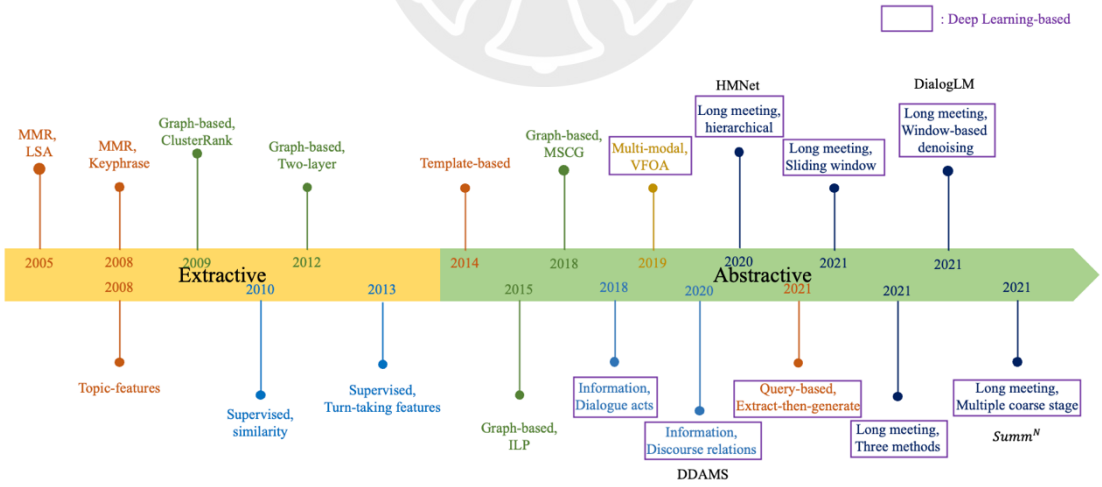

圖 2. 3 會議摘要的時間線圖

2.3.1 擷取式方法

會議摘要早期的研究主要為擷取式，可以分為基於 MMR (MMR-based)、基於圖 (Graph-based) 和監督學習 (Supervised Learning) 三種類別。

- **基於 MMR (MMR-based) 的方法**：最早的擷取式會議摘要 [29] 研究了如 MMR 和潛在語義分析(Latent Semantic Analysis, LSA) 的文本摘要技術，並表明提出的方法優於基於特徵的作法。

$$A = USV^T \tag{13}$$

LSA 是一種將詞—文件矩陣投影到向量空間表示的方法，基於 $m \times n$ 詞—文件矩陣 A 的奇異值分解 (Singular Value Decomposition, SVD) ，即將一個矩陣分解成三個矩陣的乘積，如式 (13) ，V^T 的行可以視為定義的主題，列則表示來自文件的句子。接著，[30] 進而使用關鍵詞作為 MMR 的查詢。[31] 則是在基於特徵的方法上額外引入與主題相關的特徵。

$$ITF(w_i) = \log\left(NT/NT_i\right) \tag{14}$$

與主題相關的特徵是基於主題詞頻 (Topic Term Frequency, TTF) 和逆主題頻率 (Inverse Topic Frequency, ITF) 而得。TTF 是主題內的詞出現的頻率，而 ITF 的計算如式 (14) 。其中，NT_i 是包含詞 w_i 的主題個數，NT 是會議中的主題數。

- **基於圖 (Graph-based) 的方法**：ClusterRank [32] 改進了 TextRank 算法，並在表現上優於基於 MMR 的方法，ClusterRank 的節點是根據相似度分組的相鄰話語的集群，每個話語的評分都基於其所屬的集群，能夠處理冗餘和與主題無關的資訊。

$$sim(X,Y) = \frac{\sum_{\{w:w\in X \wedge w\in Y\}} W_X(w)W_Y(w)}{\sqrt{\sum_{w\in X} W_X^2(w)}\sqrt{\sum_{w\in Y} W_Y^2(w)}} \tag{15}$$

ClusterRank 方法以每個句子作為單獨的集群開始，如果相鄰的集群在某一閾值以上相似，則將它們合併。並定義了集群 X 中詞 w 的權重為 $W_X(w) = freq(w) \times IDF(w)$，式 (15) 使用兩個集群包含詞的餘弦相似性來計算兩個集群之間的相似度。

[33] 建構了一個具有話語 — 話語、語者 — 語者和語者 — 話語三種關係的兩層圖，每個話語和每個語者分別在圖的話語層和語者層中表示為節點，兩個節點之間的邊權重是根據兩個話語之間的相似度、兩個語者之間的相似度或話語與語者之間的相似度進行加權。

- **監督學習 (Supervised Learning) 的方法**：[34] 文件中的每個句子都通過多種特徵進行表示，包含詞彙、語篇、主題和語者資訊。使用統計分類器來決定是否將該句子包含在摘要中，根據與摘要句子的相似性，給每個訓練句子分配一個數值權重，在這個任務中使用回歸模型而非二元分類的方法。[35] 研究了參與者參與度和輪次相關的特徵與擷取式會議對話摘要之間的關係。使用回歸模型，利用輪次相關的特徵，如參與度平等性、話語之間輪次自由度、中斷率等，來捕捉對話的不同特徵。

2.3.2 重寫式方法

雖然擷取式摘要在傳統文件上取得了不錯的表現，但對於會議等對話式結構的文件上卻難以取得流暢的摘要，歸因於對話中的話語通常較不流暢且資訊經常分散在多回合對話中，使得擷取出來的句子不完整、可讀性不高。因此，後來的研究傾向於使用重寫式摘要。

重寫式摘要方法可以大致分為基於圖 (Graph-based) 的方法、基於模板 (Template-based) 的方法和基於深度學習 (Deep Learning-based) 的方法。基於圖的方法有 [36] 通過整數線性規劃將在每個主題段中重要的話語聚合起來，並為每個主題段生成一個句子摘要，[37] 將對應於會議中討論的同一個主題或子主題的句子分組在一起為社群，然後使用 Multi-Sentence Compression Graph（MSCG）[38] 為每個社群單獨生成一個抽象句子。基於模板的方法概念是從人工撰寫的摘要中生成模板，再將輸入文本中取得的重要資訊填入模板，[39]

首先使用多句子融合演算法 (Multi-Sentence Fusion Algorithm) 從人工撰寫的摘要中獲取模板，接著根據主題對記錄進行分段，並從中擷取重要的詞語，最後通過參考人類撰寫的摘要與其來源之間的關係來選擇模板，並用擷取出的詞語填充模板以建立摘要。而目前最主要的研究方向是基於深度學習 (Deep Learning-based) 的方法，[1] 將基於深度學習的方法分為以下幾種類別：

- **基於查詢 (Query-based)**：[40] 將摘要分成兩個步驟進行，先根據查詢定位會議紀錄中的相關片段，再利用先進的重寫式模型如 Pointer-Generator Network [41] 、BART [22] 、HMNet [7] 生成摘要。
- **基於多模態 (Multi-Modal-based)**：[42] 使用視覺關注焦點 (Visual Focus of Attention, VFOA) 作為特徵，如果一個演講者獲得較多的注意力，在生成摘要時，應該給予他的話語更高的權重。
- **長會議摘要模型 (Summarization Models for Long Meetings)**：[7] 提出了使用層次結構來處理長會議的方法，對會議對話的詞級和輪級進行注意力，並在輪級編碼中添加了講話者的角色向量。[43] 研究了三種處理長輸入的方法：(1) 使用 Longformer [44] 模型 (2) 先檢索後生成模型 (3) 使用 HMNet 等階層對話編碼模型，表明了先檢索後生成是最有效的方法。[45] 提出了動態滑動窗口 (Dynamic Sliding Window) 的方法，解碼器除了生成摘要句子外，還預測上下文邊界。$Summ^N$ [46] 框架包含粗粒度和細粒度兩個階段，先將數據樣本進行拆分，在多個階段生成粗粒度摘要，然後根據它生成最終的細粒度摘要。DialogLM [2] 設計了一種基於窗口的去噪方法對模型進行預訓練，並根據對話的特點引入五種類型的雜訊，訓練模型能夠將帶有雜訊的窗口回復。
- **基於補充資訊 (Supplementary Information-based)**：[47] 和 [48] 將語篇結構納入， 其中 [47] 提出了一種基於零樣本學習的抽象式摘要方法，使用語篇關係重構對話成為文檔。[48] 將具有語篇關係的會議發言轉換為會議圖，然後使用圖編碼器對其進行建模。[49] 利用語者之間的對話行為進行對話摘要。

表 2. 1 深度學習方法分類

基於查詢 (Query-based)	• 能夠根據感興趣的領域獲得相關的會議摘要 • 先根據查詢找到相關範圍，再使用摘要模型生成摘要
基於多模態 (Multi-Modal-based)	• 使用文字外的其他資訊做輔助，像是參考會議錄影中，參與者的眼球注視方向向量和頭部姿態等等
長會議摘要模型 (Summarization Models for Long Meetings)	• 提出方法改進長輸入的限制，主要為三種：(1) Longformer 模型 (2) 先檢索後生成模型 (3) 階層式模型
基於補充資訊 (Supplemnetary Information-based)	• 利用額外的資訊作為摘要模型的指導 • 包含語篇結構、對話行為、引入領域術語等等

2.4 先擷取後生成方法

先擷取後生成的方式，結合了兩種摘要的方法，先從輸入文本中選取重要的文本片段，再從擷取出的片段生成摘要。先前方法大多是分別訓練擷取器和生成器 [50] [51] [52] [53] [43] 。

[50] 設計了兩階段的解碼步驟，第一個階段使用基於 Transformer 的解碼器生成一個初步的輸出序列。第二階段對初步序列的每個單詞進行遮罩，並輸入到 BERT 中，然後通過將輸入序列和 BERT 生成的初步表示結合起來，使用基於 Transformer 的解碼器為每個遮罩位置預測詞語。[51] 透過將單個句子壓縮，並將句子對合併的方式來生成摘要。[52] 先從文件中選擇句子，基於組成分析辨識可能的壓縮方式，並使用神經模型對這些壓縮進行評分以生成最終的摘要。[53] 通過使用基於 GPT-2 語言模型困惑度 (Perplexity) 得分的算法，壓縮這些長文檔，從而辨識出在源文檔中最能鞏固摘要的關鍵句子，並應用在平均文

檔長度為 4,268 個詞的長篇法律簡報。[43] 在擷取步驟使用了 TF-IDF、BM25 和 Locator [40] 三種方法，其中 Locator 為基於 BERT 的卷積神經網路。

TF-IDF (Term Frequency-Inverse Document Frequency) 是一種常用於評估單詞在文件中重要性的指標，它結合了詞頻 (Term Frequency, TF) 和逆文件頻率 (Inverse Document Frequency, IDF)。

詞頻 (TF) 的計算方法：

$$tf_{i,j} = \frac{n_{i,j}}{\sum_k n_{k,j}} \tag{16}$$

假設文件 d_j 中共有 k 個單詞，分母是所有單詞出現次數的總和，分子 n_{ij} 則是單詞 t_i 在文件 d_j 中出現的次數。

逆文件頻率 (IDF) 的計算方法：

$$idf_i = log\frac{|D|}{|\{j: t_i \in d_j\}|} \tag{17}$$

$|D|$ 是語料庫中的文件總數，$\left|\{j: t_i \in d_j\}\right|$ 代表包含單詞 t_i 的文件個數。

接著就會得到單詞的重要性指標：

$$tfidf_{i,j} = tf_{i,j} \times idf_i \tag{18}$$

BM25 的一般公式：

$$Score(Q,d) = \sum_i^n W_i R(q_i,d) \tag{19}$$

其中，Q 表示查詢 (Query)，q_i 表示查詢中的單詞，d 為要評估的文件，W_i 是單詞的權重。

因為先擷取後生成的方式，在擷取階段是直接將部分資訊捨棄，一些方法嘗試透過強化學習 (Reinforcement Learning) 來降低這種損失。[54] 使用句子級別的策略梯度方法以分層方式連接兩個神經網絡之間的不可微分計算。[55] 使用摘要級別的 ROUGE 分數作為強化學習目標。

2.5 對話語篇剖析

對話語篇剖析 (Dialogue Discourse Parsing) 是一種自然語言處理的技術，旨在將文本分析為不同的語篇結構關係，透過確定對話與對話之間的鏈結和相應的關係來建構多人參與的對話中的內部結構。

對話語篇剖析可以被視為一系列的子任務。首先，一個長句子會被切割成數個基本話語單元 (Elementary Discourse Unit, EDU)，基本話語單元是語篇剖析中的基本單元，用於表示文本中具有意義的獨立語義單元，涵蓋的資訊量通常與一般完整的句子相當。不過，基本話語單元的定義可能因特定語篇剖析模型或任務而有所不同，不同的研究和系統可能對基本話語單元的界定和表示有不同的做法。接著，要辨識和分類基本話語單元之間的關係，這些關係包括問答對 (QA_Pair)、接續 (Continuation)、闡述 (Elaboration) 等各種預先定義好的語義關係。對話語篇剖析通常被用作下游任務的第一步，像是情感分析、對話系統、機器翻譯、對話摘要等。

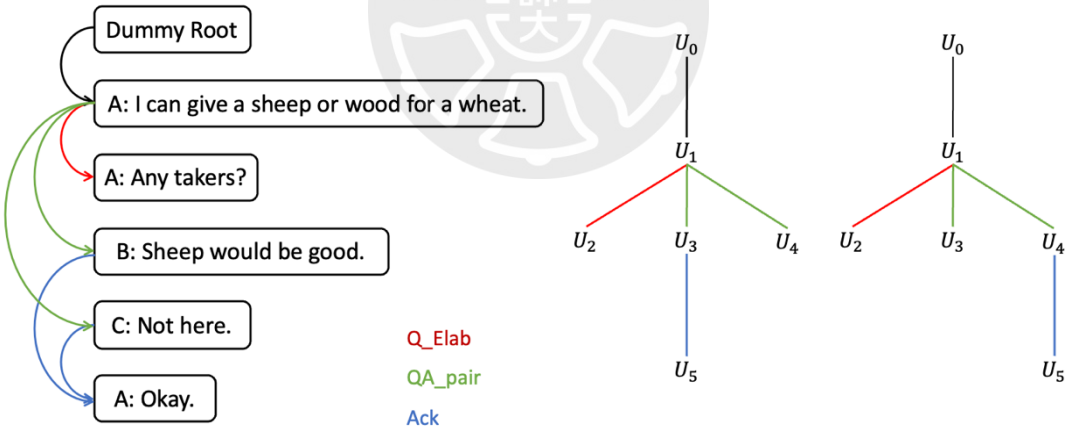

圖 2. 4 對話語篇剖析與樹狀結構

圖 2.4 解釋了在經過對話語篇剖析畫出的話語間關係後，可以透過計算出的最大生成樹作為最終的預測結果。

2.5.1 語篇結構分類

首先，要先提到語篇結構的範式，語篇結構範式是一種描述和解析文本結構和關係的理論框架，主要有三種架構：RST [56] 、 PDTB [57] 和 SDRT [58] 。修辭結構理論 (Rhetorical Structure Theory, RST) 是用來描述語篇結構和語篇元素之間關係的理論。在 RST 中，語篇的結構是一個二分樹，每個節點都有一個語篇功能，如講述原因、結果、例證等，並且節點之間存在一種「核心 -- 衛星」關係，即一個節點 (核心) 的資訊對理解整個語篇的意義比另一個節點 (衛星) 的資訊更為重要。Penn 語篇樹庫 (Penn Discourse Treebank, PDTB) 是一種基於語料庫的語篇理論，它使用多種語篇連接詞如「因為」、「然而」等，和指示詞如「這個」、「那個」等來描繪文本中的語篇結構和關係。在 PDTB 中，語篇的結構是一種圖結構，而不是樹結構，因此可以表現出語篇中更為複雜的結構和關係。分段語篇表示理論 (Segmented Discourse Representation Theory, SDRT) 是一種結合了語篇結構和語篇語義的理論。在 SDRT 中，語篇的結構是一個有向圖，每個節點都有一個語義表示，並且節點之間的關係也有相應的語義表示。SDRT 將語篇語義的問題視為尋找一種最佳解的問題，去解釋語篇結構和節點之間的關係，因此它可以用來處理語篇中的推理、語境、對應等問題。

考慮到對話的動態性和內容之間的緊密關聯性，RST 主要關注文本的邏輯結構，但對於對話中涉及多種節點之間的互動關係的研究而言，RST 的能力有限。此外，RST 只允許相鄰話語之間的關係，對於常常存在跨語句關係的對話來說，這限制了其應用的效果。而 PDTB 的描述主要基於語料庫，因此也不適用於需要深入理解語篇的深層結構和語篇元素之間語義關係的對話任務。PDTB 主要關注句子和句子之間的連接關係，而在對話中，涉及到更廣泛的語篇結構和對話元素的交互作用。SDRT 是最適合本論文任務的語篇範式。主要原因為：

1. **對話上下文理解**：對話中的每一句話都與前文和後文有關。SDRT 可以解釋對話中的這種連接性，並且能夠解析並理解這種語境關係。
2. **複雜語篇結構處理**：在多輪對話中，資訊的交換和語篇結構可能會變得相對複雜。SDRT 能夠處理這種複雜的結構，並提供深層的語篇理解。

3. **對應和推理**：對話中經常存在大量的對應和需要進行推理的情況。例如，一個代詞可能指的是前面提到的某個名詞，或者某個隱含的訊息需要從前面的對話中推理出來。SDRT 在這方面有強大的處理能力。

2.5.2 評估方法與資料集

語篇剖析有兩個主要的評估指標來評估模型的效能：

- Unlabeded Attachment Score (UAS) ：指標只關注話語之間的鏈結是否有被準確預測出來。
- Labeled Attachment Score (LAS) ：指標除了要求話語之間的鏈結有被預測出來外，話語之間的關係也要預測正確。

UAS 和 LAS 是常用於評估自然語言處理任務中語篇鏈結和語篇關係的指標。UAS 衡量的是語篇鏈結的準確性，而 LAS 則綜合考慮了語篇鏈結和語篇關係的準確性。

在訓練對話語篇剖析任務中，有兩個常見的資料集被廣泛使用，它們分別是 STAC [10] 和 Molweni [59] 。STAC 是一個來自線上遊戲的多人對話語料庫。它包含了來自遊戲中多個玩家之間的對話文本，以及與之相關的動作和遊戲情境。STAC 的特點是對話具有結構化的文本對齊，可以幫助模型理解對話參與者之間的關係和交互作用。另一個常用的資料集是 Molweni，它是從 Linux 技術論壇中收集而來的。Molweni 包含了大量關於 Linux 系統和技術的討論文和回覆。這個資料集的特點是包含了真實世界中的技術對話，其中涉及到問題的提出、解決方案的討論以及對話參與者之間的交流。

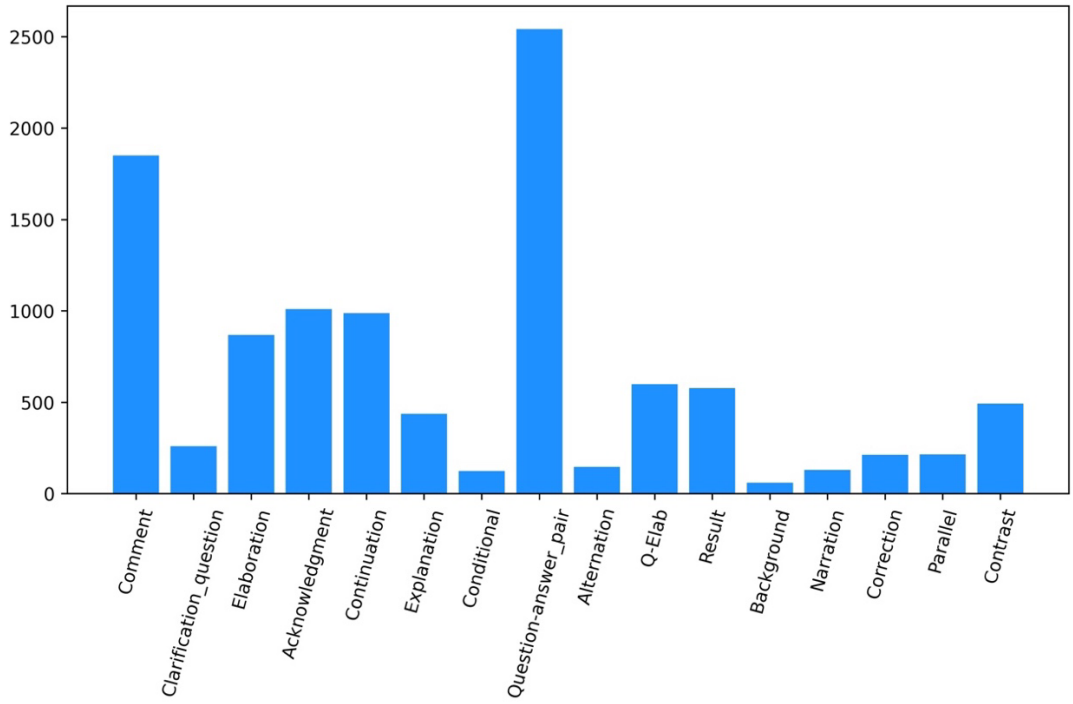

圖 2. 5 STAC 語料庫中關係類別統計圖

圖 2.5 統計了在對話語篇剖析模型訓練時所使用的資料集 STAC 中，每個預先定義好的鏈結關係的個數。

根據文獻 [60] 中的分析，對 STAC 和 Molweni 這兩個資料集進行了比較，得出了以下結論：

1. 平均 EDU 數量：STAC 和 Molweni 的平均 EDU 數量之間沒有顯著差異。兩個資料集在語篇結構的大小方面相對相似，提供了類似的語篇長度和複雜性。
2. 詞彙分佈：研究發現， STAC 和 Molweni 兩個資料集使用的詞彙有很大的差異，只有一小部分詞彙是共同的。這可能是因為 STAC 和 Molweni 是來自兩個不同領域的資料集，STAC 包含了特定的遊戲術語和專業詞彙，而 Molweni 則是包含了與 Linux 相關的專有詞彙和術語。
3. 關係分佈：STAC 和 Molweni 之間的關係分佈相似。這表明在這兩個資料集中，語篇關係的類型和比例相對一致。這對於開發和評估語篇鏈結和語篇關係辨識模型是有益的，因為它們可以利用這種相似性來建立共享的特徵和模型。

2.5.3 對話語篇剖析方法

對話語篇剖析任務除了要預測話語之間的鏈結外，還包含預測代表這些鏈結的關係。目前有六種主要的對話語篇剖析模型，分別為 MST [61]、ILP [62] 、Deep-Seq [63] 、Struct-Aware [64] 、Hierarchical [60] 和 SDDP [12] 。根據不同步驟的實作方法， [70] 將這些模型進行分類比較：

- **編碼 (Encoding)** ：MST 與 ILP 使用 MaxEnt [65] 模型對兩個話語之間的局部成對分數進行編碼，這種方法只考慮兩個話語之間的關係，並沒有考慮整體和上下文資訊。Hierarchical 模型進而透過引入一個階層式的編碼器來建模上下文資訊。 Deep-Seq 則是透過先預測所有的鏈結後，再使用全局的編碼模組注入更多的結構訊息，但是這種預測的方式是離散的，表示它只能預測鏈結的存在或不存在，無法提供鏈結的具體資訊或分數，使得這種兩階段解決方法無法進行端到端的訓練。接著，Struct-Aware 為了解決這個問題而引入了全連接圖來建模所有話語之間的關係，雖然這種方法是完全端到端的，但它無法充分捕捉和利用與結構相關的重要特徵。基於這些先前解碼器的缺點，SDDP 提出了一種完全端到端編碼器，運用兩個雙向 LSTM 來編碼上下文資訊，同時保持了結構資訊的完整性。
- **解碼 (Decoding)**：Deep-Seq、Struct-Aware 和 Hierarchical 三個模型中，鏈結和關係的解碼任務被視為多選問題且鏈結和關係獨立預測，在這個模式下，一個鏈結的存在與其他的鏈結無關。ILP 是透過整數線性規劃 (Integer Linear Programming) 來找到結構，整數線性規劃雖然在設計上較具有彈性，但它需要複雜的人工設計的解碼條件限制。而 MST 和 SDDP 都是使用最大生成數 (Maximum Spanning Tree) 解碼算法，差別在於 MST 只在預測的鏈結上進行計算，而 SDDP 將鏈結和關係建構成一個三維矩陣空間，同時執行生成樹的解碼。
- **鏈結和關係的預測：**MST、ILP、Deep-Seq、Struct-Aware 和 Hierarchical 模型都是將鏈結和關係的預測視為兩階段的訓練任務，首先預測鏈結是否存在，只有當鏈結存在時才會預測關係。SDDP 透過將話語、鏈結和關係建構成一個統一的空間，達到鏈結和關係的聯合學習。

- **人工特徵的使用**：MST、ILP、Deep-Seq 和 Struct-Aware 都使用了一些人工設計的特徵，像是明確地標示兩個話語是否為同一個語者、是否位於同一個話語輪次，或是話語對之間的距離等。這種特徵能夠提供額外的資訊，但在訓練時，可能導致模型的深度耦合，也就是過度依賴於這些人工設計的特徵，當運用到新的資料集時，容易造成效能下降。此外，要從一個資料集取得這些特徵是困難的。相比之下，Hierarchical 和 SDDP 模型不依賴於這種明確的特徵。

表 2. 2 對話語篇剖析模型分類

Models	**Encoding**	**Decoding**	**Link & relation prediction**	**Use features**
MST	*Local, edge-wise*	Partial MST	*Separate*	Y
ILP	*Local, edge-wise*	ILP	*Separate*	Y
Deep-Seq	*Global, two-staged*	Indp. Multiple choice	*Separate*	Y
Struct-Aware	*Global, fully-connected*	Indp. Multiple choice	*Separate*	Y
Hierarchical	*Hierarchical*	Indp. Multiple choice	*Separate*	N
SDDP	*Global, structured*	Fully MST	*Joint*	N

2.6 文本相似度的應用

在傳統文件的擷取式摘要任務上，文本相似度也是蠻常見的做法之一。早期，許多通用文本摘要方法使用餘弦相似度向量空間模型 [66] 來直接計算句子之間的相關性，或者使用 TF-IDF 修正的餘弦相似度 [17] 和基於潛在語義分析 (Latent Smantic Analysis, LSA) 的餘弦相似度 [67]。LSA 方法的提出是為了改進餘弦相似度將每個詞視為獨立實體的問題，例如一段話中提到「政府」和「機構」，可能代表同樣意思，卻會被視為不同的實體，LSA 能夠使用詞語的語境含義來找到句子之間的相似性。

[68] 使用餘弦相似度來尋找文本內的關係，接著使用中心性測量來對句子進行排名。將文件中的詞建構成一個稀疏矩陣，稱為文件—詞語矩陣 (Document-term Matrix, DTM)，表示文件中不同詞語出現的頻率。接者使用 TF-IDF 進行加權處理，以考慮到詞語的重要性。

加權後，矩陣 DTM 形成多維度的向量空間模型 (Vector Space Model, VSM) ，每個文件可以表示為一個特徵向量，其中每個不同詞語作為特徵，詞語的權重作為該特徵的值。這樣在向量空間中表示時，每個詞語形成 VSM 中的一個新維度，每個文件形成多維向量空間中的一個向量。

餘弦相似度的計算方法如下：

$$a \cdot b = \|a\| \|b\| \cos\theta \tag{20}$$

式 (20) 為計算兩個向量 a 和 b 的內積的公式。內積的結果為兩個向量長度的乘積再乘以它們之間夾角的餘弦值。

$$similarity = \cos\theta = \frac{a \cdot b}{\|a\| \|b\|} \tag{21}$$

餘弦相似度 $\cos\theta$ 透過將內積除以兩個向量的長度乘積而得。

第三章　研究方法

3.1　問題定義與假設

給定一個會議文件 $D = \{u_1, \ldots, u_L\}$，包含 L 個話語輪次。接著，跟一般擷取式摘要不同的是，會先把輸入文本分成多個塊 (Chunk)，而塊的大小考量到生成模型 DialogLM 的預訓練，其一次參考足夠量的連續對話文本，將整個會議 10% 長度的文本做為窗口，從中做隨機遮罩和預測。為了要讓第一階段擷取出的文字在送到生成模型時能夠盡量是完整且連續的，這邊與 DialogLM 相同，也將塊的大小設為 10% 的輸入長度。經過話語級別的切分後，每個會議文本會有 10 個句數相近的塊 $C = \{c_0, \ldots, c_9\}$，每個塊的話語數為 $l = \left\lceil \frac{L}{10} \right\rceil$。

擷取階段的目標是要預測機率 $p(y_i \mid c_i, D, \theta)$，其中 $y_i \in \{0, 1\}$ 表示話語 $c_i \in C$ 是否應該包含在摘要裡，當 y_i 為 1 時即為預測的摘要句。最後，根據分數 $p(1 \mid c_i, D, \theta)$ 做排序，取前 K 個作為最終摘要。

用 C_K 表示擷取模型選出來的結果，要作為生成模型的輸入。在生成階段的目標是要生成一個長度為 T 的摘要 $W = \{w_1, w_2, \ldots, w_T\}$，給定輸入 C_K 和先前生成的文字 $w_{<t}$，計算 $P(w \mid C_K) = \prod_{t=1}^{T} P(w_t \mid C_K, w_{<t})$。

3.2　兩階段摘要模型

3.2.1　模型架構

本論文採用兩階段先擷取後生成架構，在擷取階段比較了三種方法：異質圖神經網路模型 HeterSumGraph、對話語篇剖析和文本相似度。對話語篇剖析是去排序和選擇包含較長鏈的文本片段，文本相似度則是選擇與整個會議相似度較高的文本片段。

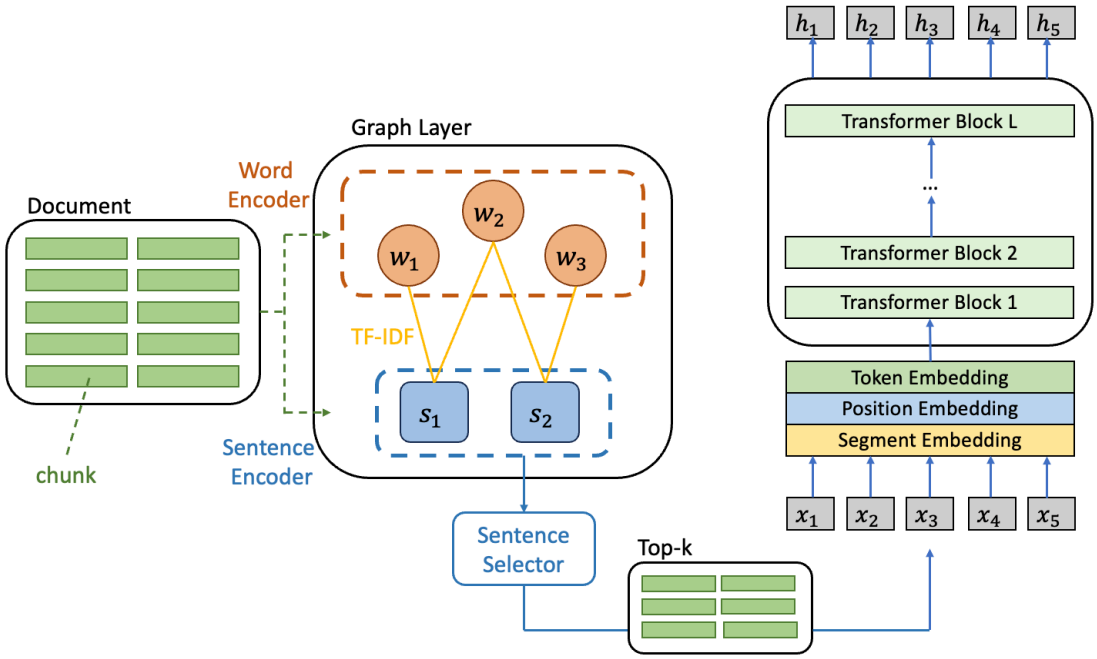

圖 3. 1 系統架構圖

圖 3.1 為系統架構的示意圖，首先會將長輸入會議文本平均分成多個塊(Chunk)，每個塊包含連續的文本片段並且作為一個候選句子，在經過擷取器後會選擇出 Top-K 個重要的文本片段，這些被選擇出的文本片段作為生成器的輸入，最後生成出最終的摘要。其中，擷取階段使用了三種方法，此處僅使用異質圖神經網路作為示意圖。

3.2.2 擷取式摘要

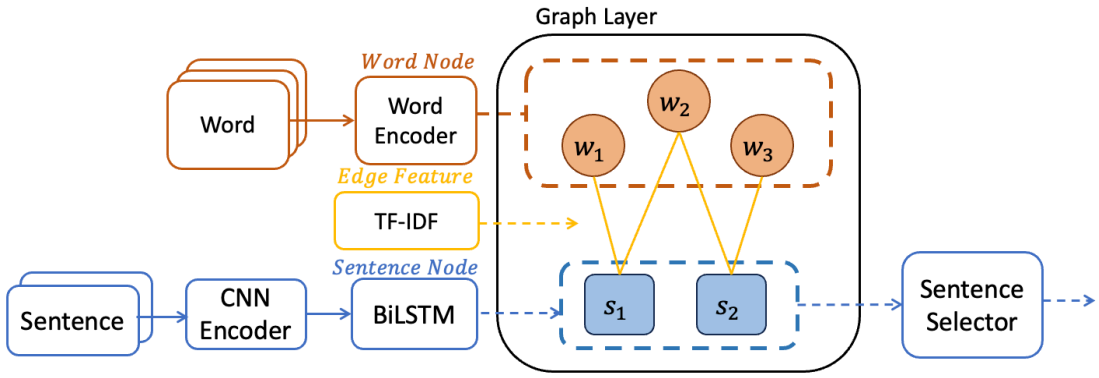

圖 3. 2 異質圖神經網路模型

擷取階段使用的模型是 HeterSumGraph，是一個異質圖神經網路，透過單詞節點的資訊來豐富句子間的關係。每個句子節點與其中包含的單詞節點相連接。透過這種關係，句子節點可以獲取與其相關的單詞資訊，同時單詞節點也可以聚合來自句子的資訊並進行更新。

將單詞節點特徵 X_w 、句子節點特徵 X_s 及邊特徵 E 建構成圖 G ，使用圖注意網路 (Graph Attention Network, GAT) [69] 來更新節點的表示：

$$z_{ij} = LeakyReLU(W_a[W_q h_i; W_k h_j]) \tag{22}$$

計算節點 i 和節點 j 之間的注意力得分 z_{ij}。由矩陣 W_a 對兩個部分 $[W_q h_i; W_k h_j]$ 的線性組合經過 $LeakyReLU$ 激活函數得到的。其中，W_a 、W_q 和 W_k 是可學習的權重矩陣，h_i 和 h_j 是節點 i 和節點 j 的特徵表示。

$$\alpha_{ij} = \frac{exp\,(z_{ij})}{\sum_{l \in N_i} exp(z_{ij})} \tag{23}$$

計算節點 i 對節點 j 的注意力權重 a_{ij}。注意力權重由 z_{ij} 的指數形式 $\exp(z_{ij})$ 除以節點 i 的鄰域 N_i 中所有 $\exp(z_{ij})$ 的和而得，接著使用 $softmax$ 函數，確保注意力權重總和為 1 。

$$u_i = \sigma(\sum_{j \in N_i} \alpha_{ij} W_v h_j) \tag{24}$$

計算節點 i 的更新表示 u_i 。由節點 i 的鄰域 N_i 中所有節點 j 的特徵表示 h_j 加權總和而得。權重是注意力權重 a_{ij} 乘以學習權重矩陣 W_v ，再經過 $sigmoid$ 函數進行非線性轉換。

$$z_{ij} = LeakyReLU(W_a[W_q h_i; W_k h_j; e_{ij}]) \tag{25}$$

接著，算式 (24) 修改了 GAT 層，將邊權重 e_{ij} 納入其中以獲得注意力得分 z_{ij}。

$$U_{w\leftarrow s}^{t+1} = GAT(H_w^t, H_s^t, H_s^t) \quad (26)$$

$$H_w^{t+1} = FFN(U_{w\leftarrow s}^{t+1} + H_w^t) \quad (27)$$

$$U_{s\leftarrow w}^{t+1} = GAT(H_s^t, H_w^{t+1}, H_w^{t+1}) \quad (28)$$

$$H_s^{t+1} = FFN(U_{s\leftarrow w}^{t+1} + H_s^t) \quad (29)$$

在第 t 次迭代中，將 H_s^t 作為注意力查詢，單詞節點表示 H_w^t 作為鍵和值，通過 GAT 層計算句子節點和單詞節點之間的注意力分數。使用這些注意力分數對相鄰的單詞節點進行加權平均，得到新的句子節點表示 H_s^{t+1}。然後，將 H_s^{t+1} 作為新的句子節點表示，再次用於獲得下一次迭代的單詞節點表示 H_w^{t+1}。多次執行這樣的迭代過程，直到達到收斂或指定的迭代次數。

3.2.3 對話語篇剖析

對話語篇剖析使用的模型是 SDDP。首先，將話語、鏈結和鏈結關係建構成三維的空間表示：

$$\{V_{h,t}^r\}_{t=h+1}^n = LSTM(\{V_{h,t}\}_{t=h+1}^n) \quad (30)$$

使用 Bi-LSTM 來建構上下文資訊，在算式 (5) 中，對於第 h 行，使用 LSTM 獲取時間步 $h+1$ 到 n 的所有隱藏狀態，並表示為 $\{V_{h,t}^r\}_{t=h+1}^n$。這樣做的目的是為了判斷話語 U_t 是否應該指向後面的話語。

$$\{V_{t,m}^c\}_{t=0}^{m-1} = LSTM(\{V_{t,m}\}_{t=0}^{m-1}) \quad (31)$$

同樣的，在算式 (6) 中，對於第 m 列，使用 LSTM 獲取時間步 0 到 $m-1$ 的所有隱藏狀態，並表示為 $\{V_{t,m}^c\}_{t=0}^{m-1}$。為了收集前面話語連接到 U_m 的資訊。

$$\tilde{V}_{h,m} = V_{h,m}^r + V_{h,m}^c \quad (32)$$

算式 (7) 定義了最終的上下文感知潛在得分，即將行和列的隱藏狀態相加。其中，$\tilde{V} \in \mathbb{R}^{(n+1)\times(n+1)\times 2d}$ 為話語對之間的成對分數，n 表示語篇數量，d 表示維度。

$$\theta_{h,m} = Linear(\tilde{V}) \tag{33}$$

接著要將 $\tilde{V}$ 轉換為對應 17 種語篇關係的分數，透過線性轉換來實現，其中 $\theta \in \mathbb{R}^{(n+1)\times(n+1)\times 17}$，如式 (32) 所示。

3.2.4 文本相似度

首先將候選句子和解答句子分別使用 Python 的 SpaCy 套件做詞項量化，得到每個句子的表示，接著透過計算候選句子和解答句子的餘弦相似度，最後選出前 Top-K 個最相似的候選句子。

3.2.5 重寫式摘要

DialogLM 是一個基於窗口去噪的預訓練語言模型，給定一個包含 n 個話語輪次的長對話文本 $D = (x_1, x_2, \dots, x_n)$，話語輪次表示一個語者–話語對 (Speaker-Utterance Pair) $x_i = (s_i; u_i)$。會隨機選擇一個包含多個輪次的窗口 $W = (x_j, x_{j+1}, \dots, x_{j+m})$，將設計的 5 種與對話相關的雜訊加入，成為新的帶雜訊窗口 $\widetilde{W} = (\tilde{x}_j, \tilde{x}_{j+1}, \dots, \tilde{x}_{j+m})$。

在預訓練階段，用帶雜訊的窗口取代原本的並與其他所有話語輪次連接成一個長序列，作為模型的輸入。解碼器要能夠透過帶雜訊的窗口和對話其餘部分來重建成原始窗口。

表 3. 1 對話相關的雜訊

Noise Type	**Description**	**Example**
Speaker Mask	Randomly mask 50% of the speakers.	[MASK]: Good morning! How are you today?
Turn Splitting	Split a single turn into multiple turns. Keep the speaker of the first turn and mask the rest.	Tom: Good morning! [MASK]: How are you today?
Turn Merging	Merge multiple turns into one turn. Keep the first speaker and remove the rest.	Tom: Good morning! How are you today? I'm doing well, thank you. How about yourself?
Text Infilling	Mask the content of the dialogue.	Tom: Good morning! How are you [MASK]?
Turn Permutation	Shuffle the order of the turns within the dialogue.	Bob: I'm doing well, thank you. How about yourself? Tom: Good morning! How are you today?

第四章　實驗設計與結果

4.1　實驗語料

在會議摘要領域中 AMI [5] 和 ICSI [70] 是目前最被廣泛應用的英語會議語料庫，除了包含每次會議的摘要，還提供了每次會議的錄音和影像記錄，支持不同的研究工作。AMI 的會議為一個設計團隊，其中有四名參與者分別為：專案經理 (Project Manager, PM)、營銷專家 (Marketing Expert, ME)、使用者介面設計師 (User Interface Designer, UI) 和 工業設計師 (Industrial Designer, ID) ，共同討論設計和開發一個新的電視遙控器。ICSI 是由國際計算機科學研究所 (International Computer Science Institute) 記錄的會議組成，包含研究小組的成員討論專業和技術主題，每次會議平均為六名參與者。

表 4. 1 AMI 資料集的統計數據

AMI	Counts	Max Len.	Mean Len.	Med Len.	Min Len.
Traing	97	7222	3772.86	4025	572
Validation	20	10269	5378.7	4998	1354
Testing	20	10041	6043.25	6512.5	2362
Total	137	10269	4338.73	4378	572

4.2　評估方法

ROUGE (Recall-Oriented Understudy for Gisting Evaluation) [71] 是評估自動文件摘要常用的方法。以召回率 (Recall) 作為和核心評估指標，用於衡量自動生成的摘要與參考摘要之間的字詞相似程度。

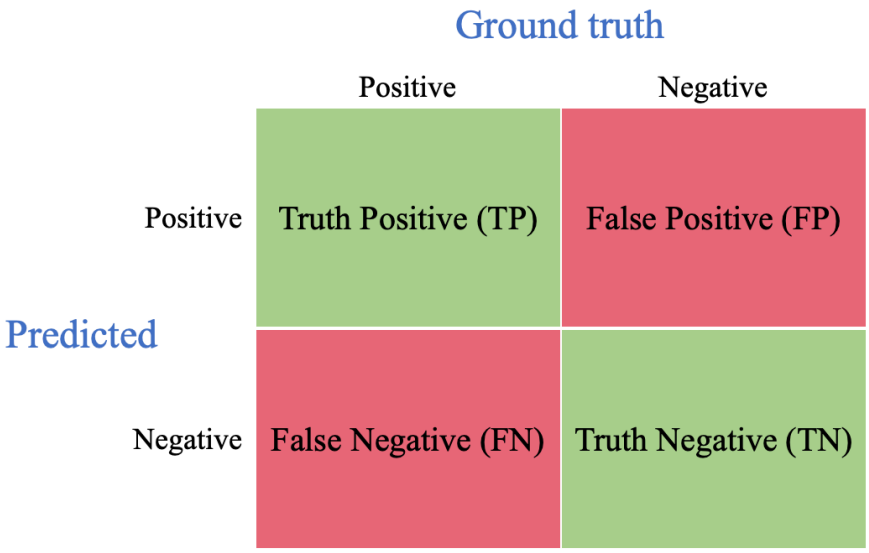

圖 4. 1 混淆矩陣

$$Recall = \frac{TP}{TP+TN} \tag{21}$$

召回率 (Recall) 表示預測結果為正樣本 $(TP + TN)$ 中，實際也為正樣本 (TP) 的比例。召回率高代表模型能夠找到更多的正樣本。

論文中的 $Rouge - N$ 算法：

$$Rouge - N = \frac{\sum_{S \in ReferenceSummaries} \sum_{gram_n} Count_{match}(gram_n)}{\sum_{S \in ReferenceSummaries} \sum_{gram_n \in S} Count(gram_n)} \tag{22}$$

在摘要評估中，我們看 ROUGE-1、 ROUGE-2 和 ROUGE-L。ROUGE-1 是基於單詞級別的評估，計算模型生成的摘要和參考摘要之間單詞的重疊率。ROUGE-2 是基於雙連詞 (Bigram) 級別的評估，考慮了相鄰單詞的組合。ROUGE-L 則是基於最長共同子序列 (Longest Common Subsequence) 級別的評估，會考慮到單詞的順序。

$$R_{lcs} = \frac{LCS(X,Y)}{m} \tag{23}$$

$$P_{lcs} = \frac{LCS(X,Y)}{n} \tag{24}$$

$$F_{lcs} = \frac{(1+\beta^2)R_{lcs}P_{lcs}}{R_{lcs}+\beta^2 P_{lcs}} \tag{25}$$

公式中，X 表示參考摘要，Y 表示生成的摘要。m 為 X 的長度，n 為 Y 的長度。$LCS(X,Y)$ 是 X 和 Y 之間的最長共同子序列，β 是一個超參數，通常會設定比較大的數值，提高召回率的權重。

4.3 實驗結果

表 4. 2 微調方法實驗結果

	AMI		
	ROUGE-1	**ROUGE-2**	**ROUGE-L**
Baseline	51.24	16.97	30.47
Finetune baseline with HSG	50.6	17.39	30.47
Finetune baseline with DDP	51.04	17.73	31.24
Finetune baseline with CS	51.69	17.49	30.64

表 4.2 為實驗的最終結果，將透過三種方式選擇出來的文本去微調基線模型，在評估單詞重疊率的 ROUGE-1 指標上沒有帶來提升，然而對於評估上下文的 ROUGE-2 和 ROUGE-L 都能帶來有效的提升。

4.3.1 基礎實驗

表 4.3 為先擷取後生成模型的初步實驗結果，可以看出在擷取階段將不同片段選擇出來後，再重新組成一份文本的作為生成模型輸入，這種方式效果不好，可能的原因為 DialogLM 會考慮到完整的對話，透過先擷取的方式破壞了對話的位置資訊。

表 4.3 兩階段方法初步結果

	AMI		
	ROUGE-1	**ROUGE-2**	**ROUGE-L**
Baseline	51.24	16.97	30.47
Extractive ground truth	51.19	16.58	30.4
HSG (5 chunks)	49.45	15.3	29.18
HSG (6 chunks)	50.08	15.54	29.13
HSG (7 chunks)	49.99	15.56	29.08
HSG (8 chunks)	49.71	15.6	29.01
DDP (5 chunks)	50.34	16.41	29.28
DDP (6 chunks)	49.78	16.37	29.13
DDP (7 chunks)	50.26	16.28	29.66
DDP (8 chunks)	50.69	16.7	29.88
CS (5 chunks)	48.92	14.99	28.64
CS (6 chunks)	49.03	15.45	29.21
CS (7 chunks)	49.71	16.05	29.53
CS (8 chunks)	51.39	16.79	29.4

表 4.4 評估模型泛化能力

	Pre 2560 tokens			**HSG(6)**		
	R-1	**R-2**	**R-L**	**R-1**	**R-2**	**R-L**
Baseline	51.24	16.97	30.47	49.12	16.19	28.81
HSG(6)	48.42	14.58	28.78	50.08	15.54	29.13
DDP(6)	49.28	15.88	28.99	49.42	15.86	28.99
CS(6)	49.46	15.28	28.71	48.64	16.46	28.98
	DDP(6)			**CS(6)**		
	R-1	**R-2**	**R-L**	**R-1**	**R-2**	**R-L**
Baseline	51.08	17.79	30.38	51.46	16.64	29.75
HSG(6)	48.94	13.80	28.38	48.20	14.53	27.87
DDP(6)	49.78	16.37	29.13	48.67	14.36	28.13
CS(6)	50.25	15.91	28.94	49.03	15.45	29.21

表 4.4 為將不同的測試集測試在不同訓練集訓練出來的生成模型的結果。結果表明基線模型的泛化能力較佳。

4.3.2 擷取式摘要結果

表 4. 5 兩階段 ROUGE 分數比較

		HeterSumGraph			DialogLM		
		R-1	**R-2**	**R-L**	**R-1**	**R-2**	**R-L**
Ext. ground truth		-	-	-	51.19	16.58	30.40
5 chunks	Ext.	75.47	69.47	73.19	49.45	15.30	29.18
	Abs.	22.31	4.79	19.91			
6 chunks	Ext.	78.91	69.83	75.06	50.08	15.54	29.13
	Abs.	19.49	4.61	17.52			
7 chunks	Ext.	78.86	70.69	75.35	49.99	15.56	29.08
	Abs.	17.30	4.40	15.62			
8 chunks	Ext.	78.27	71.71	75.53	49.71	15.60	29.01
	Abs.	15.15	4.15	13.69			

將兩個階段的結果分別跟參考摘要算 ROUGE 分數，其中擷取式的參考摘要是從重寫式參考摘要中，每個句子跟會議文本中的句子算 ROUGE 分數選擇出來的。

4.3.3 對話語篇剖析實驗

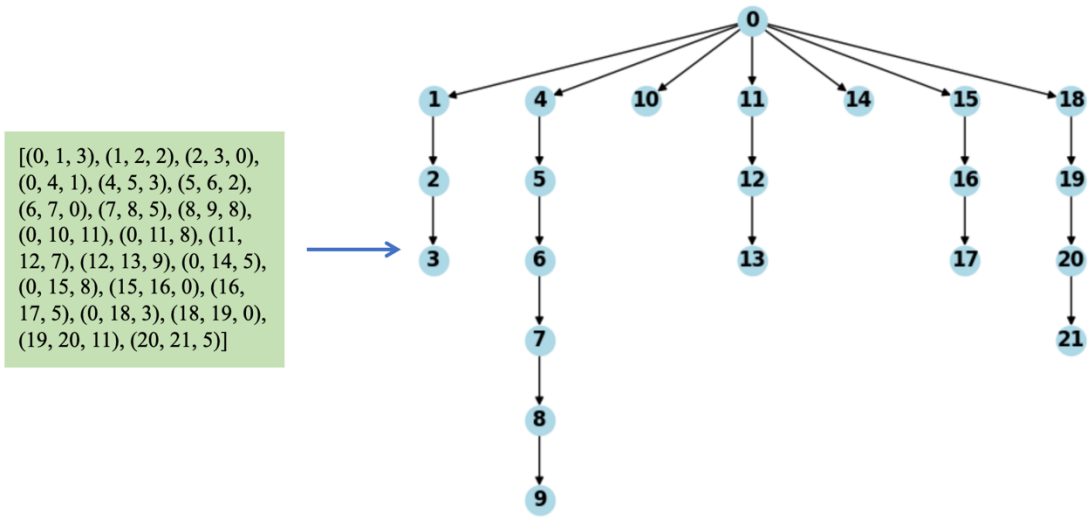

圖 4. 2 預測結果畫成生成樹

對話語篇剖析任務預測出的結果會是一個三元組 (u, v, r)，如圖 4.2 所示，其中每個三元組分別代表兩個節點的標號和它們之間的語篇關係，可以再將結果化成樹狀結構。

第五章　結論與未來展望

本研究提出了一個適用於會議摘要的重寫式摘要模型，該模型使用兩階段的先擷取後生成架構。在擷取階段，我們探討了三種方法來選擇重要的文本片段，包括異質圖神經網路、對話語篇剖析和計算句子與整個會議的餘弦相似度。對於現今會議摘要普遍會遇到的「輸入長度問題」和「對話式結構」兩個限制分別提出方法改進。透過兩階段的先擷取後生成方式，我們能夠解決生成模型無法一次讀取過長序列的問題。同時，引入對話語篇剖析來選擇文本片段有助於提供生成模型更具結構的輸入訓練文本。我們還研究了在擷取階段選擇與整個會議相似度較高的文本片段是否可以提升摘要的結果。實驗結果表明，這三種方法在需要對上下文有理解的 ROUGE-2 和 ROUGE-L 評分中都取得了有效的提升。

在未來的研究中，我們希望能夠引入更多不同的外部知識到摘要模型中，例如對話行為和對話語篇剖析的關係類別，以及聲音、影像等多種模態。此外，大型語言模型(Large Language Model, LLM) 是未來研究的趨勢，因此如何有效地微調和運用大型語言模型也是一個值得探討的課題。在評估方面，我們也需要引入更接近人類評估的方式，以更全面地評估摘要模型的效能。

參考文獻

[1] L. P. Kumar and A. Kabiri, “Meeting Summarization: A Survey of the State of the Art.” arXiv, Dec. 15, 2022.

[2] M. Zhong, Y. Liu, Y. Xu, C. Zhu, and M. Zeng, “DialogLM: Pre-trained Model for Long Dialogue Understanding and Summarization.” arXiv, Jan. 06, 2022.

[3] V. Rennard, G. Shang, J. Hunter, and M. Vazirgiannis, “Abstractive Meeting Summarization: A Survey.” arXiv, Apr. 25, 2023.

[4] A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A. N. Gomez, Ł. Kaiser, and I. Polosukhin, “Attention is All you Need,” in *Advances in Neural Information Processing Systems*, Curran Associates, Inc., 2017.

[5] J. Carletta, S. Ashby, S. Bourban, M. Flynn, M. Guillemot, T. Hain, J. Kadlec, V. Karaiskos, W. Kraaij, M. Kronenthal, G. Lathoud, M. Lincoln, A. Lisowska, I. McCowan, W. Post, D. Reidsma, and P. Wellner, “The AMI meeting corpus: a pre-announcement,” in *Proceedings of the Second international conference on Machine Learning for Multimodal Interaction*, in MLMI’05. Berlin, Heidelberg: Springer-Verlag, 11 2005, pp. 28–39.

[6] Z. Mao, C. H. Wu, A. Ni, Y. Zhang, R. Zhang, T. Yu, B. Deb, C. Zhu, A. Awadallah, and D. Radev, “DYLE: Dynamic Latent Extraction for Abstractive Long-Input Summarization,” in *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, Dublin, Ireland: Association for Computational Linguistics, May 2022, pp. 1687–1698.

[7] C. Zhu, R. Xu, M. Zeng, and X. Huang, “A Hierarchical Network for Abstractive Meeting Summarization with Cross-Domain Pretraining,” in *Findings of the Association for Computational Linguistics: EMNLP 2020*, Online: Association for Computational Linguistics, Jan. 2020, pp. 194–203.

[8] H. Sacks, E. Schegloff, and G. Jefferson, “A Simplest Systematics for the Organization of Turn-Taking for Conversation,” vol. 50, no. 4, 1974.

[9] M. Stone, U. Stojnic, and E. Lepore, “Situated Utterances and Discourse Relations,” in *Proceedings of the 10th International Conference on Computational Semantics (IWCS 2013) – Short Papers*, Potsdam, Germany: Association for Computational Linguistics, Mar. 2013, pp. 390–396.

[10] N. Asher, J. Hunter, M. Morey, B. Farah, and S. Afantenos, “Discourse Structure and Dialogue Acts in Multiparty Dialogue: the STAC Corpus,” in *Proceedings of*

the Tenth International Conference on Language Resources and Evaluation (LREC'16), Portorož, Slovenia: European Language Resources Association (ELRA), May 2016, pp. 2721–2727.

[11] D. Wang, P. Liu, Y. Zheng, X. Qiu, and X. Huang, "Heterogeneous Graph Neural Networks for Extractive Document Summarization," in *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, Online: Association for Computational Linguistics, Jul. 2020, pp. 6209–6219.

[12] T.-C. Chi and A. Rudnicky, "Structured Dialogue Discourse Parsing," in *Proceedings of the 23rd Annual Meeting of the Special Interest Group on Discourse and Dialogue*, Edinburgh, UK: Association for Computational Linguistics, Sep. 2022, pp. 325–335.

[13] L. Dong, N. Yang, W. Wang, F. Wei, X. Liu, Y. Wang, J. Gao, M. Zhou, and H.-W. Hon, "Unified Language Model Pre-training for Natural Language Understanding and Generation," in *Advances in Neural Information Processing Systems*, Curran Associates, Inc., 2019.

[14] J. Devlin, M.-W. Chang, K. Lee, and K. Toutanova, "BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding." arXiv, May 24, 2019.

[15] J. Carbonell and J. Goldstein, "The use of MMR, diversity-based reranking for reordering documents and producing summaries," in *Proceedings of the 21st annual international ACM SIGIR conference on Research and development in information retrieval*, in SIGIR '98. New York, NY, USA: Association for Computing Machinery, Spring 1998, pp. 335–336.

[16] R. McDonald, "A study of global inference algorithms in multi-document summarization," in *Proceedings of the 29th European conference on IR research*, in ECIR'07. Berlin, Heidelberg: Springer-Verlag, Summer 2007, pp. 557–564.

[17] G. Erkan and D. R. Radev, "LexRank: Graph-based Lexical Centrality as Salience in Text Summarization," *J. Artif. Intell. Res.*, vol. 22, pp. 457–479, Dec. 2004,

[18] R. Mihalcea and P. Tarau, "TextRank: Bringing Order into Text," in *Proceedings of the 2004 Conference on Empirical Methods in Natural Language Processing*, Barcelona, Spain: Association for Computational Linguistics, Jul. 2004, pp. 404–411.

[19] R. Nallapati, B. Zhou, C. dos Santos, Ç. Gùlçehre, and B. Xiang, "Abstractive Text Summarization using Sequence-to-sequence RNNs and Beyond," in *Proceedings of the 20th SIGNLL Conference on Computational Natural Language*

Learning, Berlin, Germany: Association for Computational Linguistics, Aug. 2016, pp. 280–290.

[20] R. Nallapati, F. Zhai, and B. Zhou, "SummaRuNNer: A Recurrent Neural Network Based Sequence Model for Extractive Summarization of Documents," *Proc. AAAI Conf. Artif. Intell.*, vol. 31, no. 1, Art. no. 1, Feb. 2017,

[21] T. Brown, B. Mann, N. Ryder, M. Subbiah, J. D. Kaplan, P. Dhariwal, A. Neelakantan, P. Shyam, G. Sastry, A. Askell, S. Agarwal, A. Herbert-Voss, G. Krueger, T. Henighan, R. Child, A. Ramesh, D. Ziegler, J. Wu, C. Winter, C. Hesse, M. Chen, E. Sigler, M. Litwin, S. Gray, B. Chess, J. Clark, C. Berner, S. McCandlish, A. Radford, I. Sutskever, and D. Amodei, "Language Models are Few-Shot Learners," in *Advances in Neural Information Processing Systems*, Curran Associates, Inc., 2020, pp. 1877–1901.

[22] M. Lewis, Y. Liu, N. Goyal, M. Ghazvininejad, A. Mohamed, O. Levy, V. Stoyanov, and L. Zettlemoyer, "BART: Denoising Sequence-to-Sequence Pre-training for Natural Language Generation, Translation, and Comprehension," in *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, Online: Association for Computational Linguistics, Jul. 2020, pp. 7871–7880.

[23] J. Zhang, Y. Zhao, M. Saleh, and P. J. Liu, "PEGASUS: pre-training with extracted gap-sentences for abstractive summarization," in *Proceedings of the 37th International Conference on Machine Learning*, in ICML'20, vol. 119. JMLR.org, 13 2020, pp. 11328–11339.

[24] B. Gliwa, I. Mochol, M. Biesek, and A. Wawer, "SAMSum Corpus: A Human-annotated Dialogue Dataset for Abstractive Summarization," in *Proceedings of the 2nd Workshop on New Frontiers in Summarization*, Hong Kong, China: Association for Computational Linguistics, Jan. 2019, pp. 70–79.

[25] C. Zhu, Y. Liu, J. Mei, and M. Zeng, "MediaSum: A Large-scale Media Interview Dataset for Dialogue Summarization," in *Proceedings of the 2021 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, Online: Association for Computational Linguistics, Jun. 2021, pp. 5927–5934.

[26] L. Zhao, F. Zheng, K. He, W. Zeng, Y. Lei, H. Jiang, W. Wu, W. Xu, J. Guo, and F. Meng, "TODSum: Task-Oriented Dialogue Summarization with State Tracking." arXiv, Oct. 25, 2021.

[27] K. Krishna, S. Khosla, J. Bigham, and Z. C. Lipton, “Generating SOAP Notes from Doctor-Patient Conversations Using Modular Summarization Techniques,” in *Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing (Volume 1: Long Papers)*, Online: Association for Computational Linguistics, Aug. 2021, pp. 4958–4972.

[28] X. Feng, X. Feng, and B. Qin, “A Survey on Dialogue Summarization: Recent Advances and New Frontiers.” arXiv, Apr. 27, 2022.

[29] G. Murray, S. Renals, and J. Carletta, “Extractive summarization of meeting recordings.,” in *Proceedings of the Annual Conference of the International Speech Communication Association, INTERSPEECH*, 2005.

[30] K. Riedhammer, B. Favre, and D. Hakkani-Tur, “A keyphrase based approach to interactive meeting summarization,” in *2008 IEEE Spoken Language Technology Workshop*, Feb. 2008, pp. 153–156.

[31] S. Xie, Y. Liu, and H. Lin, “Evaluating the effectiveness of features and sampling in extractive meeting summarization,” in *2008 IEEE Spoken Language Technology Workshop*, Feb. 2008, pp. 157–160.

[32] N. Garg, B. Favre, K. Riedhammer, and D. Hakkani-Tur, “ClusterRank: A Graph Based Method for Meeting Summarization,” in *Proceedings of the Annual Conference of the International Speech Communication Association, INTERSPEECH*, Sep. 2009, pp. 1499–1502.

[33] Y.-N. Chen and F. Metze, “Two-layer mutually reinforced random walk for improved multi-party meeting summarization,” in *2012 IEEE Spoken Language Technology Workshop (SLT)*, Feb. 2012, pp. 461–466.

[34] S. Xie and Y. Liu, “Improving supervised learning for meeting summarization using sampling and regression,” *Comput. Speech Lang.*, vol. 24, no. 3, pp. 495–514, Spring 2010,

[35] C. Lai, J. Carletta, and S. Renals, “Detecting Summarization Hot Spots in Meetings Using Group Level Involvement and Turn-Taking Features,” in *Proceedings of the Annual Conference of the International Speech Communication Association, INTERSPEECH*, 2015.

[36] S. Banerjee, P. Mitra, and K. Sugiyama, “Abstractive Meeting Summarization UsingDependency Graph Fusion.” arXiv, Sep. 22, 2016.

[37] G. Shang, W. Ding, Z. Zhang, A. Tixier, P. Meladianos, M. Vazirgiannis, and J.-P. Lorré, "Unsupervised Abstractive Meeting Summarization with Multi-Sentence Compression and Budgeted Submodular Maximization," in *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, Melbourne, Australia: Association for Computational Linguistics, Jul. 2018, pp. 664–674.

[38] K. Filippova, "Multi-Sentence Compression: Finding Shortest Paths in Word Graphs," in *Proceedings of the 23rd International Conference on Computational Linguistics (Coling 2010)*, Beijing, China: Coling 2010 Organizing Committee, Aug. 2010, pp. 322–330.

[39] T. Oya, Y. Mehdad, G. Carenini, and R. Ng, "A Template-based Abstractive Meeting Summarization: Leveraging Summary and Source Text Relationships," in *Proceedings of the 8th International Natural Language Generation Conference (INLG)*, Philadelphia, Pennsylvania, U.S.A.: Association for Computational Linguistics, Jun. 2014, pp. 45–53.

[40] M. Zhong, D. Yin, T. Yu, A. Zaidi, M. Mutuma, R. Jha, A. H. Awadallah, A. Celikyilmaz, Y. Liu, X. Qiu, and D. Radev, "QMSum: A New Benchmark for Query-based Multi-domain Meeting Summarization," in *Proceedings of the 2021 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, Online: Association for Computational Linguistics, Jun. 2021, pp. 5905–5921.

[41] A. See, P. J. Liu, and C. D. Manning, "Get To The Point: Summarization with Pointer-Generator Networks." arXiv, Apr. 25, 2017.

[42] M. Li, L. Zhang, H. Ji, and R. J. Radke, "Keep Meeting Summaries on Topic: Abstractive Multi-Modal Meeting Summarization," in *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, Florence, Italy: Association for Computational Linguistics, Jul. 2019, pp. 2190–2196.

[43] Y. Zhang, A. Ni, T. Yu, R. Zhang, C. Zhu, B. Deb, A. Celikyilmaz, A. H. Awadallah, and D. Radev, "An Exploratory Study on Long Dialogue Summarization: What Works and What's Next," in *Findings of the Association for Computational Linguistics: EMNLP 2021*, Punta Cana, Dominican Republic: Association for Computational Linguistics, Jan. 2021, pp. 4426–4433.

[44] I. Beltagy, M. E. Peters, and A. Cohan, "Longformer: The Long-Document Transformer." arXiv, Dec. 02, 2020.

[45] Z. Liu and N. F. Chen, “Dynamic Sliding Window for Meeting Summarization.” arXiv, Aug. 31, 2021.

[46] Y. Zhang, A. Ni, Z. Mao, C. H. Wu, C. Zhu, B. Deb, A. Awadallah, D. Radev, and R. Zhang, “Summ^N: A Multi-Stage Summarization Framework for Long Input Dialogues and Documents,” in *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, Dublin, Ireland: Association for Computational Linguistics, May 2022, pp. 1592–1604.

[47] P. Ganesh and S. Dingliwal, “Restructuring Conversations using Discourse Relations for Zero-shot Abstractive Dialogue Summarization.” arXiv, Oct. 13, 2020.

[48] X. Feng, X. Feng, B. Qin, and X. Geng, “Dialogue Discourse-Aware Graph Model and Data Augmentation for Meeting Summarization,” presented at the Twenty-Ninth International Joint Conference on Artificial Intelligence, Aug. 2021, pp. 3808–3814.

[49] C.-W. Goo and Y.-N. Chen, “Abstractive Dialogue Summarization with Sentence-Gated Modeling Optimized by Dialogue Acts.” arXiv, Sep. 29, 2018.

[50] H. Zhang, J. Cai, J. Xu, and J. Wang, “Pretraining-Based Natural Language Generation for Text Summarization,” in *Proceedings of the 23rd Conference on Computational Natural Language Learning (CoNLL)*, Hong Kong, China: Association for Computational Linguistics, Jan. 2019, pp. 789–797.

[51] L. Lebanoff, K. Song, F. Dernoncourt, D. S. Kim, S. Kim, W. Chang, and F. Liu, “Scoring Sentence Singletons and Pairs for Abstractive Summarization,” in *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, Florence, Italy: Association for Computational Linguistics, Jul. 2019, pp. 2175–2189.

[52] J. Xu and G. Durrett, “Neural Extractive Text Summarization with Syntactic Compression,” in *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*, Hong Kong, China: Association for Computational Linguistics, Jan. 2019, pp. 3292–3303.

[53] A. Bajaj, P. Dangati, K. Krishna, P. Ashok Kumar, R. Uppaal, B. Windsor, E. Brenner, D. Dotterrer, R. Das, and A. McCallum, “Long Document Summarization in a Low Resource Setting using Pretrained Language Models,” in *Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing:*

Student Research Workshop, Online: Association for Computational Linguistics, Aug. 2021, pp. 71–80.

[54] Y.-C. Chen and M. Bansal, "Fast Abstractive Summarization with Reinforce-Selected Sentence Rewriting," in *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, Melbourne, Australia: Association for Computational Linguistics, Jul. 2018, pp. 675–686.

[55] S. Bae, T. Kim, J. Kim, and S. Lee, "Summary Level Training of Sentence Rewriting for Abstractive Summarization." arXiv, Sep. 26, 2019.

[56] W. Mann and S. Thompson, "Rhetorical Structure Theory: A Framework for the Analysis of Texts," in *IPRA (International Pragmatics Association) Papers in Pragmatics*, 1987.

[57] R. Prasad, N. Dinesh, A. Lee, E. Miltsakaki, L. Robaldo, A. Joshi, and B. Webber, "The Penn Discourse TreeBank 2.0.," in *Proceedings of the Sixth International Conference on Language Resources and Evaluation (LREC'08)*, Marrakech, Morocco: European Language Resources Association (ELRA), May 2008.

[58] A. Lascarides and N. Asher, "Segmented Discourse Representation Theory: Dynamic Semantics With Discourse Structure," in *Computing*, 2007, pp. 87–124.

[59] J. Li, M. Liu, M.-Y. Kan, Z. Zheng, Z. Wang, W. Lei, T. Liu, and B. Qin, "Molweni: A Challenge Multiparty Dialogues-based Machine Reading Comprehension Dataset with Discourse Structure," in *Proceedings of the 28th International Conference on Computational Linguistics*, Barcelona, Spain (Online): International Committee on Computational Linguistics, Feb. 2020, pp. 2642–2652.

[60] Z. Liu and N. Chen, "Improving Multi-Party Dialogue Discourse Parsing via Domain Integration," in *Proceedings of the 2nd Workshop on Computational Approaches to Discourse*, Punta Cana, Dominican Republic and Online: Association for Computational Linguistics, Jan. 2021, pp. 122–127.

[61] S. Afantenos, E. Kow, N. Asher, and J. Perret, "Discourse parsing for multi-party chat dialogues," in *Proceedings of the 2015 Conference on Empirical Methods in Natural Language Processing*, Lisbon, Portugal: Association for Computational Linguistics, Sep. 2015, pp. 928–937.

[62] J. Perret, S. Afantenos, N. Asher, and M. Morey, "Integer Linear Programming for Discourse Parsing," in *Proceedings of the 2016 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language*

Technologies, San Diego, California: Association for Computational Linguistics, Jun. 2016, pp. 99–109.

[63] Z. Shi and M. Huang, "A Deep Sequential Model for Discourse Parsing on Multi-Party Dialogues." arXiv, Dec. 01, 2018.

[64] A. Wang, L. Song, H. Jiang, S. Lai, J. Yao, M. Zhang, and J. Su, "A Structure Self-Aware Model for Discourse Parsing on Multi-Party Dialogues," presented at the Twenty-Ninth International Joint Conference on Artificial Intelligence, Aug. 2021, pp. 3943–3949.

[65] A. L. Berger, S. A. Della Pietra, and V. J. Della Pietra, "A Maximum Entropy Approach to Natural Language Processing," *Comput. Linguist.*, vol. 22, no. 1, pp. 39–71, 1996.

[66] G. Sidorov, A. Gelbukh, H. Gomez Adorno, and D. Pinto, "Soft Similarity and Soft Cosine Measure: Similarity of Features in Vector Space Model," *Comput. Sist.*, vol. 18, Sep. 2014,

[67] J. Steinberger and K. Jezek, "Using Latent Semantic Analysis in Text Summarization and Summary Evaluation," in *Proceedings of the 7th International Conference ISIM*, Apr. 2004, pp. 93–100.

[68] M. Jain and H. Rastogi, "Automatic Text Summarization using Soft-Cosine Similarity and Centrality Measures," in *2020 4th International Conference on Electronics, Communication and Aerospace Technology (ICECA)*, Jan. 2020, pp. 1021–1028.

[69] P. Veličković, G. Cucurull, A. Casanova, A. Romero, P. Liò, and Y. Bengio, "Graph Attention Networks." arXiv, Feb. 04, 2018.

[70] A. Janin, D. Baron, J. Edwards, D. Ellis, D. Gelbart, N. Morgan, B. Peskin, T. Pfau, E. Shriberg, A. Stolcke, and C. Wooters, "The ICSI Meeting Corpus," in *2003 IEEE International Conference on Acoustics, Speech, and Signal Processing, 2003. Proceedings. (ICASSP '03).*, Apr. 2003.

[71] C.-Y. Lin, "ROUGE: A Package for Automatic Evaluation of Summaries," in *Text Summarization Branches Out*, Barcelona, Spain: Association for Computational Linguistics, Jul. 2004, pp. 74–81.