The 34th Conference on Computational Linguistics and Speech Processing (ROCLING 2022)
Taipei, Taiwan, November 21-22, 2022. The Association for Computational Linguistics and Chinese Language Processing

利用監督式對比學習來建構增強型的自迴歸文件檢索器
Building an Enhanced Autoregressive Document Retriever Leveraging Supervised Contrastive Learning

Yi-Cheng Wang[1], Tzu-Ting Yang[1], Hsin-Wei Wang[1], Yung-Chang Hsu[2], Berlin Chen[1]
[1]National Taiwan Normal University
[2]EZ-AI Inc.
[1]{yichengwang, tzutingyang, hsinweiwang, berlin}@ntnu.edu.tw
[2]mic@ez-ai.com.tw

摘要

資訊檢索系統的目標是從大量的文件中，找出與使用者查詢 (Query) 最相關的文件。在傳統的資訊檢索流程中，需要經過多次的比對許多文件才能找出最相關的文件。近期，有一種基於可微搜索索引 (Differentiable Search Index, DSI) 的新穎資訊檢索策略被提出，並展現相當優異的效能。DSI 透過單一個 Transformer 模型先將文件集中所有的資訊編碼在模型的參數中；在應用時，使用者可以將查詢輸入 Transformer，再由 Transformer 以自迴歸的方式直接地產生其相關文件的編號 (Document IDs)，因而能大幅地簡化與加速整個檢索過程。先前的研究指出，DSI 是以文件編號作爲橋梁來建立查詢與文件之間的關係，但在訓練資料中並不是每篇文件都會有相關的查詢，這將導致這些文件沒辦法被順利的建立起關係。有鑑於此，在模型訓練階段，我們提出先使用監督式對比學習來增強查詢與文件在潛在語意空間中的對應關係，並在模型推理階段時，透過最鄰近搜尋法來進一步的輔助模型產生文件編號。因此，我們提出的方法能有效增強 DSI 中文件與查詢薄弱的對應關係，在公開的語料集 Nature Question 上也驗證了它的成效。

Abstract

The goal of an information retrieval system is to retrieve documents that are most relevant to a given user query from a huge collection of documents, which usually requires time-consuming multiple comparisons between the query and candidate documents so as to find the most relevant ones. Recently, a novel retrieval modeling approach, dubbed Differentiable Search Index (DSI), has been proposed. DSI dramatically simplifies the whole retrieval process by encoding all information about the document collection into the parameter space of a single Transformer model, on top of which DSI can in turn generate the relevant document identities (IDs) in an autoregressive manner in response to a user query. Although DSI addresses the shortcomings of traditional retrieval systems, previous studies have pointed out that DSI might fail to retrieve relevant documents because DSI uses the document IDs as the pivotal mechanism to establish the relationship between queries and documents, whereas not every document in the document collection has its corresponding relevant and irrelevant queries for the training purpose. In view of this, we put forward to leveraging supervised contrastive learning to better render the relationship between queries and documents in the latent semantic space. Furthermore, an approximate nearest neighbor search strategy is employed at retrieval time to further assist the Transformer model in generating document IDs relevant to a posed query more efficiently. A series of experiments conducted on the Nature Question benchmark dataset confirm the effectiveness and practical feasibility of our approach in relation to some strong baseline systems.

關鍵字：資訊檢索、自迴歸檢索系統、對比學習

Keywords: Information Retrieval, Autoregressive Retrieval System, Contrastive Learning

1 緒論

爲了滿足使用者的資訊需求 (Information Needs)，資訊檢索 (Information Retrieval) 系統需要依據使用者的查詢，從大量的語料庫中找出相關的文件。文件檢索的方式分別是以詞匹配 (Term-matching) 與語意匹配 (Semantic-matching) 作爲基礎。TF-IDF 和 BM25 (Robertson et al., 2009) 爲詞匹配類中常見的作法，它們將使用者的查詢 (Query) 與文件 (Document) 用高維的稀疏向量來表徵，透過簡單的向量相似度計算，可以快速的匹配關鍵詞。雖然詞匹配的方法能簡單且快速的找出相關的文件，但它卻無法考慮到文字與

文字之間的順序和語意上的關聯。爲此，許多以語意匹配作爲基礎的檢索系統被紛紛提出，包含潛藏語意分析 (Latent Semantic Analysis, LSA) (Deerwester et al., 1990) 主題模型 (Topic Model) (Hofmann, 1999; Papadimitriou et al., 2000; Blei et al., 2003) 等。與詞匹配不同的是語意匹配用了密集向量來表徵，可以看做是將查詢與文件投影到潛在的語意空間 (Latent Semantic Space)，同義詞與一詞多義的關聯就能有效的在這個空間中被捕捉到。

隨著深度學習的蓬勃發展，近年來自監督 (Self-supervised Learning) 的預訓練語言模型 (Pre-trained Language Model) 在許多自然語言處理的任務上都有突破性的表現。其中 (Dense Passage Retriever, DPR) (Karpukhin et al., 2020) 爲一個著名的語意匹配檢索系統，它使用兩個不同的 BERT (Devlin et al., 2019) 預訓練語言模型，來將查詢及文件分別以密集的向量表徵，並使用簡單的向量相似度計算，來求得相關的文件。由於 DPR 有著 BERT 強大的語意表示能力及雙編碼器 (Dual Encoder) 的設計，讓它可以預先計算全部文件的向量表示，在效果及速度上都能超越被視爲標準的 BM25 檢索系統。最近，另外一種資訊檢索的方法被提出，如圖1所示，(Differentiable Search Index, DSI) (Tay et al., 2022) 有別於以往的檢索系統需要進行大量的向量比對，來找出相關的文件，這個方法利用序列到序列 (Sequence-to-sequence) 的預訓練語言模型，將所有的文件資訊編碼進一個 Transformer (Vaswani et al., 2017) 的參數中。DSI 在訓練時分爲兩個步驟，第一步驟爲索引 (Indexing Phase)，第二步驟爲檢索 (Retrieval Phase)。在索引的階段，模型學習如何將文件的內容 (Document Texts) 對應到文件的編號 (Document Identifiers)。在檢索的階段，模型學習如何將查詢 (Query) 對應到相關文件的編號。最後在模型推理 (Inference) 時，使用者只需要輸入查詢，這個檢索模型就會自迴歸 (Autoregressive) 的產生潛在相關的文件編號，大幅的簡化整個檢索過程。DSI 的作者也顯示當使用更大的預訓練語言模型時，檢索的效果也會跟著上升，在現今模型參數量快速成長的趨勢下，這類的方法展現了十足的潛力。

強大的檢索系統通常需表現出以下幾點能力 (Lewis et al., 2021)，從最簡單的記憶訓練時看過的查詢，到使用訓練時看過的答案來回答新的查詢表述，最後是使用完全新的答案表述來回答新的查詢表述。 (Lewis et al., 2021) 指出，將回答查詢所需的所有知識都儲存在模型參數中的閉卷模型 (Closed-book Model)，在遇到新的查詢表述時，容易傾向回答訓練時看過的答案，這個現象顯示出此類模型的泛化能力不足。而 DSI 同爲閉卷模型的一種，在 (Zhuang et al., 2022) 的研究中也指出它的泛化能力不足。 (Zhuang et al., 2022) 提到 DSI 模型在學習建立查詢與相關文件之間的關聯時，是透過索引與檢索的兩個階段來學習，但是在大部分的訓練資料中，並不是每篇文件都會有對應的查詢，所以若有文件編號未被任何查詢對應過，而導致模型無法爲該文件與相關的查詢建立關聯，模型在推理時就會傾向回答有被順利對應過的文件編號。(Differentiable Search Index With Query Generation, DSI-QG) (Zhuang et al., 2022) 提出了一個簡單直覺的解決方法，它使用另一個序列到序列的預訓練語言模型，先將所有的文件產生出對應的查詢，確保每篇文件都有對應的查詢後，再進行 DSI 模型的訓練，經過這個方法訓練後，模型在泛化能力上的表現大幅提升。

然而，我們認爲原本的 DSI 模型只使用文件編號來在建立查詢與文件之間的關聯太爲薄弱，爲了增強這之間的對應關係，本研究提出使用 (Supervised Contrastive Learning, SCL) (Khosla et al., 2020) 將查詢與文件先在語意空間中建立關聯。實驗在公開語料集 Nature Question (Kwiatkowski et al., 2019) 上，我們提出的 (Building an Enhanced Autoregressive **D**ocument **R**etriever Leveraging **S**upervised **C**ontrastive **L**earning, DR-SCL) 能進一步的改善模型的泛化能力，在與 DSI-QG 結合後，我們的模型能超越強大的 DPR 檢索模型，讓此類模型向實際應用又邁進了一步。

2 相關研究

2.1 Dense Retriever

隨著深度學習在自然語言處理上的突破，預訓練語言模型 BERT (Devlin et al., 2019)、RoBERTa (Liu et al., 2019) 帶來了強大的語意表示能力，在過去幾年中各式各樣的深度檢索模型也相繼被提出。依據不同查詢與文件的表徵方式及不同計算相似度的方式， (Zhu et al., 2021) 將深度檢索模型分成了三類:Representation-based, Interaction-based, Representation-interaction Retriever。

Representation-based Retriever 也稱做雙編碼器 (Dual Encoder) 檢索模型，這類模型使用兩個不同的編碼器來將查詢與文件以密集向量的方式表徵，透過向量相似度計算來評估查詢與文件的相關程度。其中，

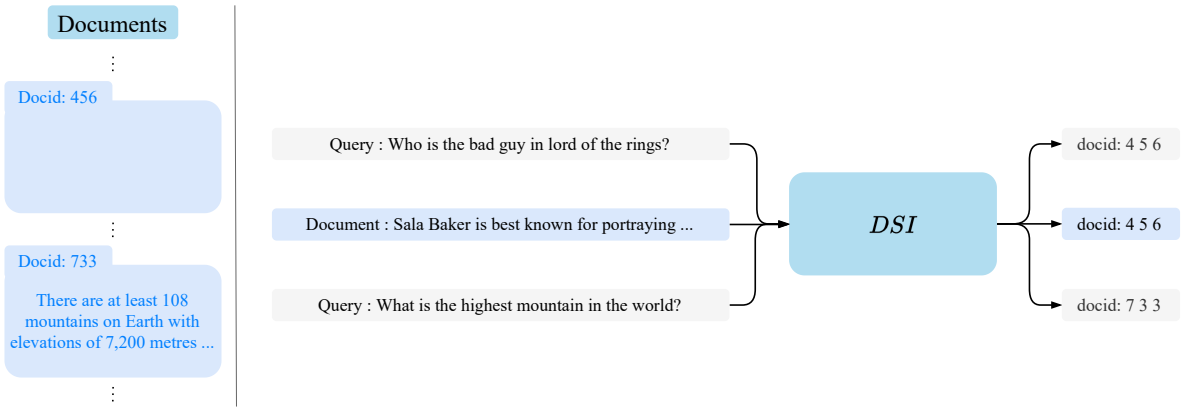

圖 1. (左) 爲許多文件所組成的語料庫。(右) 爲 DSI 的模型架構圖。在 DSI 訓練時，模型透過索引階段 (Indexing Phase) 學習如何將文件對應到它的文件編號，並再透過檢索階段 (Retrieval Phase) 學習如何將查詢對應到與它相關的文件的文件編號。在使用者出入一個查詢後，DSI 會自迴歸的產生相關的文件編號。如果需要，可以使用束搜索來產生潛在相關文件編號的排序列表。另外，爲了幫助模型分辨查詢與文件，在文字輸入進模型前，會先在前面加上任務提示 (Task Prompt)，例如文件會加上 *Document:* 、查詢會加上 *Query:* 。

DPR (Karpukhin et al., 2020) 爲這類模型的著名的方法，它使用了兩個參數不共享的 BERT 編碼器來分別將查詢及文件表徵。在訓練階時，使用對比學習 (Contrastive Learning) 讓目前的查詢在語意空間中和相關的文件拉近，並同時推遠不相關的文件。此類模型有著優良的檢索速度，因爲所有文件的向量表示都能事先被計算好並儲存在記憶體中，當模型進行推理時，只需要計算查詢的向量表示，並將它與儲存在記憶體中的文件向量計算相似度，就能快速的找到相關的文件。但也因爲查詢與文件的表示是獨立獲得，透過簡單的向量相似度計算，兩者之間只有淺層的互動，導致模型犧牲掉不少檢索的效果。

Interaction-based Retriever 也稱做跨編碼器 (Cross Encoder)，這類模型將查詢與文件同時輸入進一個編碼器中，通過它們之間的字符層級 (Token-level) 互動，模型能捕捉查詢與文件間的豐富資訊。 (Nie et al., 2019; Nogueira and Cho, 2019) 使用了 BERT 做爲跨編碼器，它們將 Dense Retrieval 視爲二元分類的問題，若查詢與文件相關爲 1，反之則爲 0。因爲查詢與文件能有深度的互動，所以此類方法有著非常良好的檢索效果，但因查詢與文件必須同時計算，所以在檢索的效率上有很大的限制。

Representation-interaction Retriever 爲了在速度與準確度上取得平衡，這類模型結合了 representation-based 與 interaction-based 兩者的特點。ColBERT (Khattab and Zaharia, 2020) 是此類模型中著名的方法，它延伸了雙編碼器的做法，先使用不同的編碼器分別取得查詢與文件的向量表示，再由字符層級的相似度計算方法來求得兩者之間的關聯程度。值得注意的是，雙編碼器是用一個向量來表徵整個查詢或文件，而此類方法是使用多個字符層級的向量來表示，雖然能取得更好的檢索效果，但在儲存文件向量表示時會需要更大的空間。

2.2 Autoregressive Retriever

另一種檢索系統設計的方式是使用序列到序列 (Sequence-to-sequence) 的自迴歸 (Autoregressive) 模型。此類模型利用預訓練語言模型 T5 (Raffel et al., 2020)、BART (Lewis et al., 2020) 對語意理解的強大能力，將所有回答查詢所需的知識都儲存在模型的參數中，並使用自迴歸的解碼器 (Decoder) 來產生答案。在此小節中，我們預計介紹三個使用自迴歸的檢索系統，分別爲 Autoregressive Entity Retrieval (Cao et al., 2021), DSI (Tay et al., 2022), DSI-QG (Zhuang et al., 2022)

Autoregressive Entity Retrieval 是一個使用序列到序列的預訓練語言模型 BART 來預測實體連結 (Entity Linking) 的系統。使用者將句子輸入到模型後，如果句子中可能存在有實體，模型就會將其對應到語料庫中預設的同義實體，並透過自迴歸的方式輸出。在此研究中，作者使用 Wikipedia 作爲語料庫，而每篇維基百科中的文章的標題則當作是實體名稱。此方法可以看作是特殊類別的檢索系統。

Differentiable Search Index 使用了一個序列到序列的預訓練語言模型 T5 來進行文件檢索，如圖1所示。它先將所有文件資訊編碼進模型的參數中，並在使用者輸入查詢後，自迴歸的產生相對應的文件編號。如果需要，可以使用束搜索 (Beam Search) 來產生潛在相關

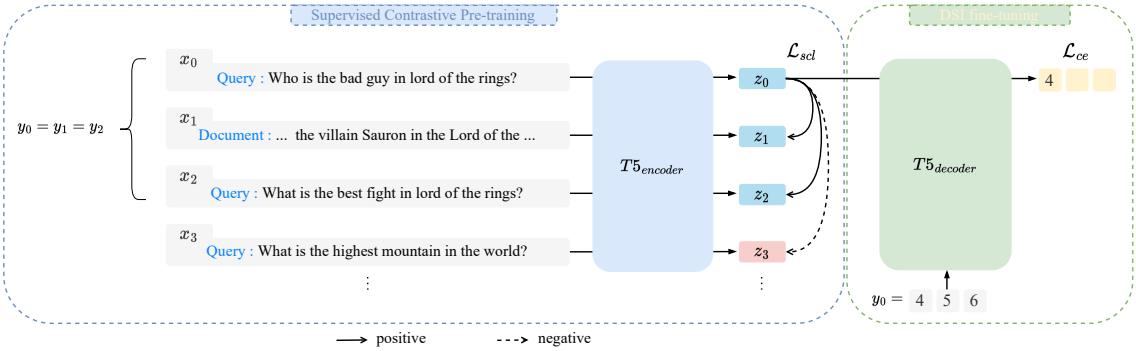

圖 2. 爲本研究提出的 DR-SCL 模型架構。DR-SCL 的模型架構是由一個 T5 預訓練語言模型組成。在訓練階段時，模型爲了將查詢與相關的文件在語意空間中拉近，反之則推遠，首先使用監督式對比學習來預訓練編碼器。在預訓練好模型的編碼器後，我們使用與 DSI 相同的訓練方式來微調模型的解碼器。

文件編號的排序列表。在訓練模型時分爲兩個步驟，第一步驟爲索引 (Indexing Phase)，第二步驟爲檢索 (Retrieval Phase)。在索引的階段，模型學習如何將語料庫中的每一篇文件 $x_d \in X_d$ 對應到它的文件編號 $y_d \in Y_d$。模型將文件內容當作輸入而文件編號當爲輸出，與一般序列到序列的模型訓練方式相同，損失函數爲交叉熵 (Cross Entropy)。以下爲索引階段模型使用的損失計算公式：

$$\mathcal{L}_{indexing} = -\sum_{x_d \in X_d} \log p(y_d|T5(x_d)). \quad (1)$$

如果只有第一階段的訓練，模型只學會文件與文件編號的對應，並不知道如何將查詢對應到它相關的文件內容。所以在第二階段時，模型利用訓練資料中的每一個查詢 $x_q \in X_q$ 與人工標記的相關文件的文件編號 $y_q \in Y_q$ 來建立對應關係。下面爲檢索階段模型使用的總損失計算公式：

$$\mathcal{L}_{retrieval} = -\sum_{x_q \in X_q} \log p(y_q|T5(x_q)). \quad (2)$$

(Tay et al., 2022) 提到如果先訓練索引階段再訓練檢索階段，模型會產生災難性遺忘 (Catastrophic Forgetting) 的現象。爲此，作者改用與 T5 預訓練方式相同的多任務訓練 (Multi-task Learning) 來同時訓練這兩個階段。此外，爲了讓模型能區分查詢及文件的表示，在輸入到模型之前，查詢與文件會分別在開頭加上任務提示 (Task Prompt) 的字串。以下爲 DSI 模型使用的總損失計算公式：

$$\mathcal{L}_{DSI} = -\sum_{x \in X} \log p(y|T5(x)), \quad (3)$$

其中 $x \in X = \{X_d \cup X_q\}, y \in Y = \{Y_d \cup Y_q\}$。

在模型推理 (Inference) 的階段中，輸入一個查詢 x 後，模型會自迴歸的產生對應的文件編號 y。給定查詢 x 模型產生文件編號 y 的機率，可以使用下面的公式來描述：

$$p(y|x) = \prod_{m=1}^{M} p(y_m|T5(x, y_{1:m-1})). \quad (4)$$

作者提到了非常多種產生文件編號 y 的方式，從簡單的流水號 (Atomic Docid) 到使用隨機字符組成的字串編號 (String Docid)，最後是有語意結構的字串編號 (Semantically Structured Docid) 也是其中效果最好的表示方法。爲了產生有語意結構的文件編號，如圖3所示，需先經由一個 BERT 編碼器將所有的文件投影到語意空間中，再使用階層式群集 (Hierarchical Clustering) 演算法，將語意相近的文件歸類到同個群，最後只需要搜尋產生出的樹狀結構，即可指派每篇文件的文件編號。

Differentiable Search Index With Query Generation 提到 DSI 模型是使用索引階段時文件對應到文件編號的訓練，及檢索階段時查詢對應到文件編號的訓練，來建立查詢與相關文件之間的關聯，但在訓練資料中並不是每一篇文件都會有對應到的查詢，所以 X_q 可能爲很小的集合甚至是空集合。爲了解決上述的問題，DSI-QG 使用了一個簡單直覺的解決方式，它先利用了另一個序列到序列的語言模型 T5，學習如何在給定一篇文件後產生其對應的查詢內容。使用此查詢生成模型幫助所有文件產生完對應的查詢後，再使用這些資料與原訓練資料來訓練 DSI 模型。此方法能有效的改善模型無法順利對應到文件的問題。

3 研究方法

3.1 Overview

本篇研究專注於如何增強在 DSI 模型中，查詢與文件之間的對應關係。爲此，我們提出了一個新穎的方法，先使用對比學習 (Contrastive Learning) 來預訓練模型的編碼器，再來進行原本的 DSI 模型訓練。我們使用的模型架構與 DSI 相同，是一個由編碼器與解碼器組成的 T5 (Raffel et al., 2020) 預訓練語言模型。

其中，查詢與文件在輸入模型前會先加上個別的任務提示 (Task Prompt)，如圖2所示。輸入文字序列 x_ℓ 後，模型的編碼器會產生一串序列的向量表示，依據 (Ni et al., 2022) 的做法我們使用平均池化來取得表示整個序列的向量 z_ℓ ，以下面的公式表示：

$$z_\ell = MeanPooling(T5_{encoder}(x_\ell)). \quad (5)$$

3.2 Contrastive Pre-training

爲了增強模型在查詢與文件之間的對應關係，我們先使用對比學習來將查詢與相關的文件在語意空間中拉近，並推遠不相關的文件。由此一來，在新的查詢表述進來時，這個查詢在語意空間中的表示就會更容易的與它相關的文件表示相近，因此也會對後續的自迴歸文件編號生成有所幫助。爲了設計適合的對比學習方法來增強 DSI 模型中查詢與文件的對應關係，我們嘗試了使用與 DPR 模型相同的對比學習方法。給定一個包含 N 筆資料的訓練集 $B_{cl} = \{(x_{q,v}, x_{d,v})\}_{v=1..N}$，其中 $x_{d,v} \in X_d$ 爲查詢 $x_{q,v} \in X_q$ 相關的一篇文件。

訓練資料中的查詢與文件在經過第 (5) 式後，得到向量表示 $\{(z_{q,v}, z_{d,v})\}_{v=1..N}$。令 $i \in I = \{1, .., N\}$ 爲訓練資料的索引。接著，DPR 所用的對比學習可以使用以下的式子來描述：

$$\mathcal{L}_{cl} = -\sum_{i \in I} \log \frac{\exp(z_{q,i} \cdot z_{d,i})}{\sum_{a \in I} \exp(z_{q,i} \cdot z_{d,a})}, \quad (6)$$

其中 $\cdot$ 表示使用內積運算，而文件 $z_{d,i}$ 爲查詢 $z_{q,i}$ 的正樣本，在訓練資料中除了 $z_{d,i}$ 以外的 $N-1$ 筆文件則當作是 $z_{q,i}$ 查詢的負樣本。

考量到後續的自迴歸文件編號生成任務，類似於序列分類的問題，如果我們能將同一類 (有著相同的文件編號) 的查詢或文件拉近，並將不同類別的推遠，那對於序列的分類必定會有更大的幫助。監督式對比學習 (Khosla et al., 2020) 將同類別的資料都視爲正樣本，不同類別的資料視爲負樣本，由此一來如果我們使用監督式對比學習就可以比 (6) 式有著更多的正樣本。在訓練時也不會侷限於只能以查詢作爲錨點 (Anchor) 來拉近與推遠文件，而是能任意讓查詢或文件來當作錨點拉近相同類別與推遠不相同的類別。給定一個包含 N 筆資料的訓練集 $B_{drscl} = \{(x_u, y_u)\}_{u=1..N}$，其中 $x_u \in X, y_u \in Y$。在訓練資料中的 x_u 經過 (5) 式後，得到向量表示 $\{(z_v, y_v)\}_{u=1..N}$。

令 $A(i) = I \setminus \{i\}$ 爲索引 I 扣除掉 i 後的集合，$S(i) = \{s \in A(i) : y_s = y_i\}$ 爲索引 i 的所有正樣本的索引。監督式對比學習可以使用下面的公式來描述：

$$\mathcal{L}_{scl} = -\sum_{i \in I} \frac{1}{|S(i)|} \sum_{s \in S(i)} \log \frac{\exp(z_i \cdot z_s)}{\sum_{a \in A(i)} \exp(z_i \cdot z_a)}, \quad (7)$$

其中 $|S(i)|$ 爲集合 $S(i)$ 的大小。

3.3 DSI Fine-tuning

爲了不讓建立好的語意空間受到後續訓練序列分類的影響，在預訓練完模型的編碼器後，我

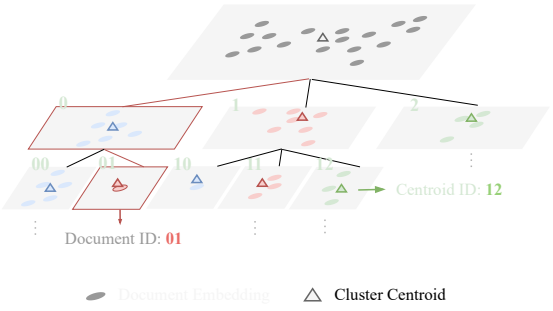

圖 3. 爲產生有語意結構的文件編號時用的階層式分群的示意圖。

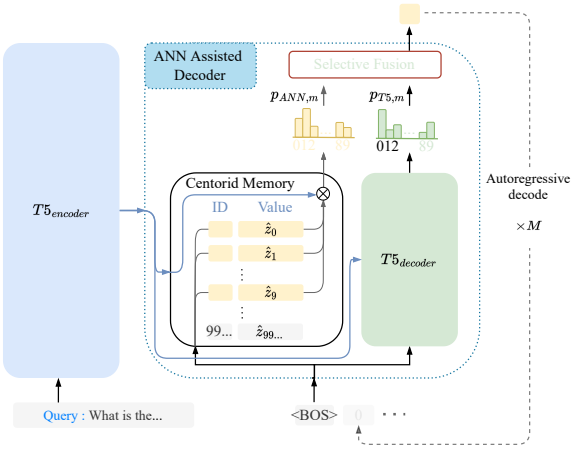

圖 4. 爲 DR-SCL 在模型推理時的架構圖。模型會使用最鄰近搜尋法 ANN 來輔助模型解碼器自迴歸的生成文件編號。

們的作法與 (Khosla et al., 2020) 相同，先將模型編碼器的參數凍結住，之後再進行模型解碼器的微調 (Fine-tune)。在微調模型的解碼器時，訓練方式與 DSI 相同，使用多任務學習來同時訓練文件對應到文件編號的索引階段 (Indexing Phase)，與查詢對應到文件編號的檢索階段 (Retrieval Phase)。給定訓練資料 B_{drscl}，模型的損失函數爲：

$$\mathcal{L}_{ce} = -\sum_{i \in I} \log p(y_i | T5(x_i)). \quad (8)$$

3.4 Document IDs

有鑑於在 DSI 中使用結構化語意 (Semantically Structured) 的文件編號模型會有最好的表現，所以我們也使用語意結構化的方法來表示文件編號。在 DSI 中，作者使用 BERT 預訓練語言模型來抽取出所有文件的語意向量，在取完所有文件向量後，DSI 使用階層式的 k-means 分群法將相近的文件分在一起 (在這裡 $k = 10$)，經過階層式分群後會產生一個十元樹 (Decimal Tree)，從樹根走訪到文件所在的葉節點的過程中，經過的節點編號所組成的字串就爲文件的編號，如圖3 所示。與 DSI 不同的是，我們使用已經預訓練好的模型編碼器，並經過第 (5) 式來產生所有文件的語意向量，如此一來也不需要有額外的模型加入。

3.5 Inference

經過了預訓練後的編碼器，本身就具備著文件檢索的能力。我們利用這項優勢，將 DRSCL 的編碼器透過最鄰近搜尋法 (Approximate Nearest Neighbor Search, ANN) 來輔助模型解碼器的自迴歸文件編號生成。在模型進行推理前我們先使用在 3.4 小節中，產生好的文件向量與十元樹，來計算樹上每一群的中心點的向量，計算完的中心點向量以 $\hat{z}_c \in \hat{Z}$ 表示，而群中心點的編號 c 則以樹根走訪到 $\hat{z}_c$ 所在的節點，當中經過的節點編號所組成的字串來表示，如圖3所示。在模型推理時，ANN Assisted Decoder 同時參考模型編碼器得到的 ANN 分數與模型解碼器得到的分數，來決定下一個時間點要產生的文件編號字符。

T5 Decoder 在第 m 個時間點文件編號字符 $y_{i,m}$ 產生的機率可以由下面的式子來描述：

$$P_{T5,m} = p(y_{i,m} | T5(x_i, y_{i,1:m-1})), \quad (9)$$

ANN Search 在第 m 個時間點文件編號字符 $y_{i,m}$ 產生的機率可以由下面的式子來描述：

$$\begin{aligned} P_{ANN,m} &= p(y_{i,m} | ANN(x_i, y_{i,1:m-1})) \\ &= \frac{\exp(z_i \cdot \hat{z}_{y_{i,1:m}})}{\sum_{c \in Child(y_{i,1:m-1})} \exp(z_i \cdot \hat{z}_c)}, \end{aligned} \quad (10)$$

其中 z_i 爲 x_i 經過第 (5) 式後得到的向量表示，$C = Child(c)$，C 是由中心點編號 c 的所有子節點的中心點編號所組成的集合。

Selective Fusion 我們使用另外兩個超參數 α, β 來控制 $P_{T5,m}$ 與 $P_{ANN,m}$ 的結合：

$$P_{Fusion,m} = \exp(\alpha \log P_{T5,m} + (1-\alpha) \log P_{ANN,m}). \quad (11)$$

$$p(y_i | x_i) = \prod_{m=1}^{M} \left(\begin{cases} P_{Fusion,m} & m \leq \beta \\ P_{T5,m} & m > \beta \end{cases} \right). \quad (12)$$

其中 α 控制在產生第 m 個時間點的文件編號字符機率時，$P_{ANN,m}$ 所佔的權重。β 則是控制計算時要往下參考 $P_{ANN,m}$ 幾個時間點，如果 $\beta = 0$，模型的輸出就是正常的 T5 模型的輸出，若 $\beta = \infty$ 則代表使用 ANN 輔助整個文件編號的生成。最後模型在推理階段看到查詢 x_i 後，產生文件編號 y_i 的機率，可以由第 (12) 式來描述。值得注意的是，使用 ANN 能有效減少搜尋的次數，生成一筆完整的文件編號所需的最大搜尋次數爲 $10 \times M$。

Dataset	$\|D\|$	Train Pairs	Test Pairs
NQ10k	10k	8k	2k
NQ100k	100k	80k	20k

表 1. 爲我們實驗中使用的 Nature Question Dataset 的語料集統計資訊。$|D|$ 表示語料集中，文件的總數。

Dataset	Document Overlap
NQ10k	20.65%
NQ100k	60.89%

表 2. 爲 Test-train 語料集中的重疊比例。Document Overlap 是計算有多少爲答案的文件同時出現在測試資料與訓練資料中。

4 實驗與討論

4.1 語料集

本研究使用的語料集爲 (Nature-Question, NQ) (Kwiatkowski et al., 2019)，NQ 是專爲端到端 (End-to-end) 的開域問答系統 (Open-domain Question Answering System) 所設計

Model	NQ10k						NQ100k					
	Total		Overlap		No Overlap		Total		Overlap		No Overlap	
	Hit@1	**Hit@10**	**Hit@1**	**Hit@10**	**Hit@1**	**Hit@10**	**Hit@1**	**Hit@10**	**Hit@1**	**Hit@10**	**Hit@1**	**Hit@10**
BM25	51.60	73.70	49.39	73.84	52.17	73.66	35.31	59.99	32.03	59.48	40.41	60.77
DPR	61.25	**83.80**	62.71	**88.13**	**60.86**	**82.67**	**53.84**	**79.18**	53.78	81.00	**53.94**	**76.33**
DSI	13.10	30.10	44.06	67.79	5.04	20.28	33.09	50.71	52.11	72.55	3.47	16.69
DSI+QG	56.35	73.90	63.92	82.08	54.37	71.77	51.62	72.13	59.42	80.33	39.48	59.36
DR-SCL (β=0)	25.80	52.20	56.17	77.23	17.89	45.68	37.58	57.47	56.06	75.93	8.79	28.71
DR-SCL+QG (β=0)	56.75	75.40	62.95	86.19	55.13	72.58	52.14	73.34	**60.00**	81.29	39.90	60.96
DR-SCL+QG (α=0.7, β=∞)	**61.70**	79.80	**65.85**	87.89	59.16	77.63	53.40	77.12	58.78	**83.31**	45.00	67.48

表 3. 爲基準模型與我們提出的方法，在 NQ10k 與 NQ100k 的測試語料集上的結果。Overlap 指的是有多少爲答案的文件同時出現在測試資料與訓練資料中。

的語料集，其中總共有 307k 筆的訓練資料，每一筆訓練資料中含有一個眞實 Google 搜尋的查詢及一篇人工標記爲相關的 Wikipedia 文件。給定一個查詢，檢索系統必須找出一篇與這個查詢相關的 Wikipedia 文件。我們將 NQ 語料集切成兩個大小 NQ10K 與 NQ100k，來觀測模型對於不同大小語料集的表現，相關的統計資訊如表1所示。

爲了檢視我們提出來的模型是否能改善在遇到新的查詢表述後，模型傾向於回答在訓練過程中看過的答案文件的查詢。我們依據 (Lewis et al., 2021) 將測試語料集再細分成重疊 (Overlap) 與沒有重疊 (No Overlap)，其中重疊代表在測試語料集爲答案的文件，也出現在訓練資料中，而沒有重疊就是沒有在訓練資料出現過，表2爲文件在訓練語料集中重疊的統計資訊。

4.2 實驗設置

我們的模型使用 Huggingface (Wolf et al., 2019) 開源的 T5-base 模型，在進行模型編碼器的預訓練前，我們先使用有著較好語意表示能力的 Sentence-T5 (Ni et al., 2022) 來初始化模型編碼器的參數。在我們提出的 DR-SCL 中，輸入查詢的長度最長爲 32 個字符，而文件的長度我們則是依照 DSI (Tay et al., 2022) 的做法，只保留文件前 32 個字符，並在最前面並接上文件的標題。模型訓練時的批次大小 N 設定爲 64，並訓練 50 代。

4.3 評估指標與基準模型

我們使用 Hit@k 作爲評估模型的指標，其中 $k \in \{1, 5, 10\}$，這個指標與 Top-k Accuracy 相同，它計算前 k 個模型找回的文件中，有出現正確相關文件的比例。

基準模型包含了經典的詞匹配模型 BM25 (Robertson et al., 2009) 與語意匹配的 DPR (Karpukhin et al., 2020) 模型，而在自迴歸的檢索模型上，還有 DSI 與使用額外的查詢生成 (Query Generation) 模型來加強的 DSI-QG 模型 (Zhuang et al., 2022)。其中，我們使用 Sentence-T5 來當作 DPR 中編碼器所使用的模型。在 DSI-QG 中，我們將每篇文件生成對應的五個查詢，再使用這些新生成的查詢與原訓練資料結合來訓練模型。

4.4 實驗結果

我們首先討論基準模型與我們提出的 DR-SCL 模型在整體 NQ10k 和 NQ100k 上的表現，結果如表 (3) 所示。使用語意匹配爲基礎的 DPR 模型在兩個語料集時，皆領先使用詞匹配爲基礎的 BM25 模型，代表著預訓練語言模型的加入，大幅的改善無法只使用詞語匹配來找到答案的問題。致使 DSI 模型在整體 NQ10k 與 NQ100k 的表現相較於 BM25 都來的差，可以發現 DSI 模型只要在遇到查詢的答案文件沒有在訓練時的檢索階段 (Retrieval Phase) 被訓練過 (No Overlap) 的情況下，表現都會非常差，這個現象也印證 DSI 模型在泛用能力上的低落，而若是遇到查詢的答案文件有在訓練時的檢索階段被訓練過 (Overlap)，DSI 模型就能領先 BM25，至於模型在 NQ10k 上檢索效果低於 BM25，我們推測是因爲相較於 NQ100k，NQ10k 有大部分的文件能用詞匹配的方式來達成，所以 DSI 的效果會略輸於 BM25。另外，DSI 在 NQ100k 的表現優於 NQ10k，是因爲在 NQ100k 的測試資料中有較多爲答案的文件也有出現在訓練資料中，如表2所示。我們提出的方法 DR-SCL($\beta = 0$) 在 NQ10k No Overlap 上，相較 DSI 進步了 12%，在 NQ100k No Overlap 上也進步了 5%，驗證了使用對比學習來增強原本 DSI 模型中查詢與文件之間的薄弱關係的有效性。再來是 DSI-QG 模型，它使用額外的查詢生成模型來產生更多與文件相關的查詢，這個方法簡單的解決了 DSI 在訓練檢索階段 (Retrieval Phase) 文件沒有被對應過的查詢，在 No Overlap 的表現上都有大幅的進步。DR-SCL+QG($\beta = 0$) 爲我們的方法加上一個額外的查詢生成模型，額外使用生成的模型再搭配使用對比學習方法加強查詢與文件的關係後，能比 DSI+QG 的方法效果再好一些。最後，當我們的方法 DR-

Encoder	Additional Pre-trained	Hit@1	Hit@10
T5	-	13.10	30.10
Sentence-T5	-	16.90	40.75
Sentence-T5	DPR Loss (6)	22.95	46.00
Sentence-T5	SCL Loss (7)	**24.00**	**48.30**

表 4. 爲使用不同的預訓練方法來訓練編碼器，對整體表現的影響。

Encoder	Additional Pre-trained	Hit@1	Hit@10
Sentence-T5	-	24.00	48.30
Sentence-T5	SCL Loss (7)	**25.80**	**52.20**

表 5. 爲使用不同模型編碼器來產生有語意結構的文件編號對整體表現的影響。

SCL+QG($\alpha = 0.7, \beta = \infty$) 在模型解碼文件編號時，如果犧牲一點運算速度參考模型編碼器提供的 ANN 分數，我們的提出的方法在 NQ10k Hit@1 時就能超越強大的 DPR 模型，讓自迴歸的檢索模型向實際應用又更邁進了一步。

4.5 消融研究 (Ablation Study)

4.5.1 Contrastive Pre-training

在這個小節中，我們將分析何種模型編碼器的預訓練方法對我們提出的模型會有最好的效果，結果如表4所示。直接使用沒有額外預訓練的 T5 模型，是其中效果最差的，因爲 T5 語言模型在設計時不像 BERT 一樣專注在語意特徵的的學習上，這導致了 T5 模型的編碼器沒辦法產生最佳的語意表示。我們嘗試使用額外以句子相似度訓練的 Sentence-T5 來直接初始化模型的編碼器，可以發現有著更強的語意表示對於自迴歸文件檢索模型是有幫助的。對比用與 DPR 相同的對比學習方式 (6)，使用監督式對比學習 (7) 來預訓練模型的編碼器可以取得最好的效果，這是因爲監督式對比學習有更多的正樣本與查詢和文件間有更豐富的互動。

4.5.2 Document IDs

在這個小節中，我們將探討使用哪一種編碼器來產生有語意結構的文件編號，會對我們的模型有最好的效果，結果如表5所示。可以發現有經過監督式對比學習訓練過的編碼器，在產生文件的語意向量時能有最好的效果，這是因爲經過訓練資料訓練過的編碼器能將不相關的文件推遠，相比直接使用沒看過訓練資料的 Sentence-T5 來的更好。

4.5.3 Alpha & Beta

在這個小節中，我們想了解模型在推理階段時超參數 α, β 的設置，結果如圖5所示。α 負責在產生第 m 個文件編號的字符時，控制 ANN

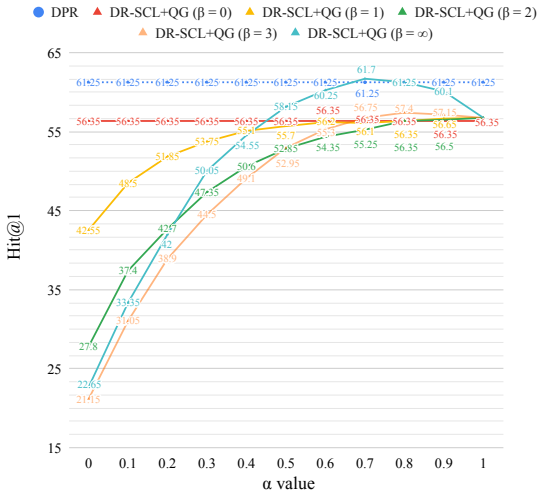

圖 5. 爲使用不同超參數 α, β 設定時，對整體表現的影響。

分數的比重，α 越小表示 ANN 分數所佔的比重越高。β 控制要使用 ANN 輔助產生幾個文件編號的字符，β 越大表示使用 ANN 輔助產生越多文件編號的字符。當 $\beta = 0$ 時，模型完全不使用 ANN 來輔助輸出; 而在 $\alpha = 0, \beta = \infty$ 時，模型的輸出即是 ANN 的輸出。最後在 $\alpha = 0.7, \beta = \infty$ 模型解碼器參考些許 ANN 的分數來預測文件編號時，模型會有最好的表現。

5 結論

在本研究中，我們提出了將監督式對比學習應用在自迴歸的檢索模型 DSI 上，來改善 DSI 在遇到新的查詢表述時，泛化能力不足的問題。並且，我們也提出了一個使用最近鄰居搜尋法 (ANN) 來輔助模型產生文件編號，讓模型可以在推理時，透過超參數的控制來平衡模型的精準度與速度。在公開的 Nature Question 語料集上，我們提出的方法與 DSI-QG 結合後，在 Hit@1 能超越強大的 DPR 模型，讓自迴歸的檢索模型向實際應用又邁進了一步。在未來的研究裡，我們希望不靠額外的查詢生成模型來輔助 DR-SCL，就能找到一個好的方式讓模型記憶文件資訊，且在查詢進入到模型後能順利的檢索出相關的文件，這將會是我們主要研究的方向。除此之外，因爲自迴歸的檢索模型可以完全端到端的訓練，它可以當作是一個大模型中具備檢索能力的元件，這是非常有潛力的。

References

David M Blei, Andrew Y Ng, and Michael I Jordan. 2003. Latent dirichlet allocation. *Journal of machine Learning research*, 3(Jan):993–1022.

Nicola De Cao, Gautier Izacard, Sebastian Riedel, and Fabio Petroni. 2021. Autoregressive entity retrieval. In *9th International Conference on Learning Representations, ICLR 2021, Virtual Event, Austria, May 3-7, 2021*. OpenReview.net.

Scott Deerwester, Susan T Dumais, George W Furnas, Thomas K Landauer, and Richard Harshman. 1990. Indexing by latent semantic analysis. *Journal of the American society for information science*, 41(6):391–407.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2019. BERT: pre-training of deep bidirectional transformers for language understanding. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, NAACL-HLT 2019, Minneapolis, MN, USA, June 2-7, 2019, Volume 1 (Long and Short Papers)*, pages 4171–4186. Association for Computational Linguistics.

Thomas Hofmann. 1999. Probabilistic latent semantic indexing. In *Proceedings of the 22nd annual international ACM SIGIR conference on Research and development in information retrieval*, pages 50–57.

Vladimir Karpukhin, Barlas Oguz, Sewon Min, Patrick S. H. Lewis, Ledell Wu, Sergey Edunov, Danqi Chen, and Wen-tau Yih. 2020. Dense passage retrieval for open-domain question answering. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing, EMNLP 2020, Online, November 16-20, 2020*, pages 6769–6781. Association for Computational Linguistics.

Omar Khattab and Matei Zaharia. 2020. Colbert: Efficient and effective passage search via contextualized late interaction over BERT. In *Proceedings of the 43rd International ACM SIGIR conference on research and development in Information Retrieval, SIGIR 2020, Virtual Event, China, July 25-30, 2020*, pages 39–48. ACM.

Prannay Khosla, Piotr Teterwak, Chen Wang, Aaron Sarna, Yonglong Tian, Phillip Isola, Aaron Maschinot, Ce Liu, and Dilip Krishnan. 2020. Supervised contrastive learning. *Advances in Neural Information Processing Systems*, 33:18661–18673.

Tom Kwiatkowski, Jennimaria Palomaki, Olivia Redfield, Michael Collins, Ankur Parikh, Chris Alberti, Danielle Epstein, Illia Polosukhin, Matthew Kelcey, Jacob Devlin, Kenton Lee, Kristina N. Toutanova, Llion Jones, Ming-Wei Chang, Andrew Dai, Jakob Uszkoreit, Quoc Le, and Slav Petrov. 2019. Natural questions: a benchmark for question answering research. *Transactions of the Association of Computational Linguistics*.

Mike Lewis, Yinhan Liu, Naman Goyal, Marjan Ghazvininejad, Abdelrahman Mohamed, Omer Levy, Veselin Stoyanov, and Luke Zettlemoyer. 2020. 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, ACL 2020, Online, July 5-10, 2020*, pages 7871–7880. Association for Computational Linguistics.

Patrick S. H. Lewis, Pontus Stenetorp, and Sebastian Riedel. 2021. Question and answer test-train overlap in open-domain question answering datasets. In *Proceedings of the 16th Conference of the European Chapter of the Association for Computational Linguistics: Main Volume, EACL 2021, Online, April 19 - 23, 2021*, pages 1000–1008. Association for Computational Linguistics.

Yinhan Liu, Myle Ott, Naman Goyal, Jingfei Du, Mandar Joshi, Danqi Chen, Omer Levy, Mike Lewis, Luke Zettlemoyer, and Veselin Stoyanov. 2019. Roberta: A robustly optimized bert pretraining approach. *arXiv preprint arXiv:1907.11692*.

Jianmo Ni, Gustavo Hernandez Ábrego, Noah Constant, Ji Ma, Keith B. Hall, Daniel Cer, and Yinfei Yang. 2022. Sentence-t5: Scalable sentence encoders from pre-trained text-to-text models. In *Findings of the Association for Computational Linguistics: ACL 2022, Dublin, Ireland, May 22-27, 2022*, pages 1864–1874. Association for Computational Linguistics.

Yixin Nie, Songhe Wang, and Mohit Bansal. 2019. Revealing the importance of semantic retrieval for machine reading at scale. 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 2019, Hong Kong, China, November 3-7, 2019*, pages 2553–2566. Association for Computational Linguistics.

Rodrigo Nogueira and Kyunghyun Cho. 2019. Passage re-ranking with bert. *arXiv preprint arXiv:1901.04085*.

Christos H Papadimitriou, Prabhakar Raghavan, Hisao Tamaki, and Santosh Vempala. 2000. Latent semantic indexing: A probabilistic analysis. *Journal of Computer and System Sciences*, 61(2):217–235.

Colin Raffel, Noam Shazeer, Adam Roberts, Katherine Lee, Sharan Narang, Michael Matena,

Yanqi Zhou, Wei Li, and Peter J. Liu. 2020. Exploring the limits of transfer learning with a unified text-to-text transformer. *J. Mach. Learn. Res.*, 21:140:1–140:67.

Stephen Robertson, Hugo Zaragoza, et al. 2009. The probabilistic relevance framework: Bm25 and beyond. *Foundations and Trends® in Information Retrieval*, 3(4):333–389.

Yi Tay, Vinh Q Tran, Mostafa Dehghani, Jianmo Ni, Dara Bahri, Harsh Mehta, Zhen Qin, Kai Hui, Zhe Zhao, Jai Gupta, et al. 2022. Transformer memory as a differentiable search index. *arXiv preprint arXiv:2202.06991*.

Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N. Gomez, Lukasz Kaiser, and Illia Polosukhin. 2017. Attention is all you need. In *Advances in Neural Information Processing Systems 30: Annual Conference on Neural Information Processing Systems 2017, December 4-9, 2017, Long Beach, CA, USA*, pages 5998–6008.

Thomas Wolf, Lysandre Debut, Victor Sanh, Julien Chaumond, Clement Delangue, Anthony Moi, Pierric Cistac, Tim Rault, Rémi Louf, Morgan Funtowicz, et al. 2019. Huggingface's transformers: State-of-the-art natural language processing. *arXiv preprint arXiv:1910.03771*.

Fengbin Zhu, Wenqiang Lei, Chao Wang, Jianming Zheng, Soujanya Poria, and Tat-Seng Chua. 2021. Retrieving and reading: A comprehensive survey on open-domain question answering. *arXiv preprint arXiv:2101.00774*.

Shengyao Zhuang, Houxing Ren, Linjun Shou, Jian Pei, Ming Gong, Guido Zuccon, and Daxin Jiang. 2022. Bridging the gap between indexing and retrieval for differentiable search index with query generation. *arXiv preprint arXiv:2206.10128*.