Vietnamese Legal Document Retrieval

Hà Hữu Phát 22521067 | **Trần Văn Thân** 22521322 | **Đỗ Quốc Thắng** 22521326 | **Lưu Đoàn Ngọc Phát** 22521070

Abstract

Truy xuất văn bản pháp luật là một nhiệm vụ quan trọng trong lĩnh vực xử lý ngôn ngữ tự nhiên, nhằm tìm ra các tài liệu pháp luật liên quan nhất từ một tập dữ liệu lớn dựa trên truy vấn của người dùng. Nhiệm vụ này đặt ra nhiều thách thức do tính phức tạp của ngôn ngữ pháp lý và quy mô khổng lồ của kho văn bản pháp luật. Trong đồ án này, nhóm chúng em đề xuất một cách tiếp cận truy xuất kết hợp, tận dụng tính hiệu quả của mô hình Bi-encoder và độ chính xác của mô hình Cross-encoder. Phương pháp này bắt đầu bằng việc mã hóa câu truy vấn và tập văn bản pháp luật thành các vector đặc trưng bằng Bi-encoder, cho phép truy xuất nhanh các tài liệu liên quan hàng đầu thông qua độ tương đồng cosine. Sau đó, các tài liệu được chọn lọc sẽ được xếp hạng lại bằng Cross-encoder, đánh giá mối quan hệ ngữ nghĩa chi tiết giữa truy vấn và từng tài liệu. Kết quả thực nghiệm cho thấy phương pháp này đã giải quyết được một số vấn đề phức tạp của bài toán, tuy nhiên phương pháp này vẫn cần cải thiện thêm để đạt được hiệu quả tối ưu.

1 Giới thiệu

Truy xuất thông tin là một lĩnh vực cốt lõi trong xử lý ngôn ngữ tự nhiên và đóng vai trò quan trọng trong các ứng dụng như tìm kiếm trên web, hệ thống hỏi đáp, và đặc biệt là trong lĩnh vực pháp lý. Với sự gia tăng nhanh chóng của các tài liệu pháp luật, bao gồm luật, quy định, nghị định, việc phát triển các hệ thống truy xuất hiệu quả và chính xác là điều cần thiết để hỗ trợ các chuyên gia pháp lý trong việc tra cứu thông tin và ra quyết định.

Tuy nhiên, nhiệm vụ truy xuất tài liệu pháp luật đặt ra nhiều thách thức đặc thù.

- Ngôn ngữ pháp luật thường rất phức tạp, bao gồm các thuật ngữ chuyên ngành, cấu trúc câu dài, và ngữ cảnh rộng.
- Quy mô khổng lồ của kho dữ liệu pháp lý yêu cầu các hệ thống truy xuất phải hoạt động nhanh chóng và hiệu quả.
- Để đảm bảo tuân thủ pháp lý, các kết quả truy xuất không chỉ cần liên quan mà còn phải có độ chính xác cao.

Trong đồ án này, nhóm chúng em đề xuất một hướng tiếp cận truy xuất kết hợp, sử dụng mô hình Bi-encoder và Cross-encoder để giải quyết các thách thức nêu trên. Mô hình Bi-encoder cho phép mã hóa nhanh các tài liệu pháp lý và truy vấn dưới dạng vector đặc trưng, giúp xác định nhanh các tài liệu liên quan hàng đầu. Trong khi đó, Cross-encoder đóng vai trò xếp hạng lại các tài liệu dựa trên đánh giá chi tiết ngữ nghĩa, đảm bảo các kết quả cuối cùng có chất lượng cao nhất. Chúng em thực hiện các thí nghiệm trên bộ dữ liệu pháp luật từ cuộc thi SoICT (1) và chứng minh tính hiệu quả của phương pháp này trong việc truy xuất các tài liệu liên quan.

2 Phương pháp thực hiện

2.1 Bi-encoder

Bi-encoder (hình 1) là một kiến trúc phổ biến trong xử lý ngôn ngữ tự nhiên dùng để học biểu diễn (embedding) của hai đoạn văn bản hoặc hai thực thể sao cho có thể so sánh độ tương đồng hoặc tính liên quan giữa chúng một cách hiệu quả.

Bi-encoder bao gồm hai bộ mã hóa (Encoder) riêng biệt, mỗi bộ mã hóa là một mạng neural như BERT... được sử dụng để mã hóa đầu vào (query và document) thành không gian đặc trưng. Sau đó, một độ đo tương đồng (cosine similarity, inner product) được sử dụng để tính sự tương quan giữa 2 vector biểu diễn query và document.

Trong đồ án này, chúng em sử dụng mô hình Bi-encoder **"bkai-foundation-models/vietnamese-bi-encoder"** (2), một mô hình được thiết kế tối ưu

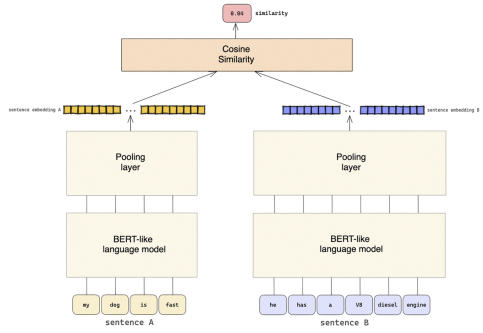

Figure 1: Kiến trúc Bi-Encoder.

cho tiếng Việt. Mô hình này được phát triển dựa trên kiến trúc Transformer và được huấn luyện trên dữ liệu lớn tiếng Việt để nắm bắt tốt các đặc trưng ngữ nghĩa và ngữ pháp của ngôn ngữ.

Ưu điểm:

- **Tính độc lập:** Vì truy vấn và tài liệu được mã hóa riêng biệt, các vector tài liệu có thể được tính toán trước và lưu trữ, giúp tăng tốc đáng kể trong quá trình truy xuất thời gian thực.
- **Khả năng mở rộng:** Dễ dàng áp dụng cho các tập dữ liệu lớn với hàng triệu tài liệu bằng cách sử dụng các thư viện tìm kiếm vector hiệu quả như FAISS.
- **Khả năng ứng dụng đa dạng:** Mô hình có thể được áp dụng cho các bài toán như tìm kiếm tài liệu, xếp hạng câu trả lời, hoặc các bài toán yêu cầu so sánh ngữ nghĩa.

Hạn chế:

- **Mất thông tin về tương tác cục bộ:** Vì bi-encoder mã hóa văn bản độc lập, nó không nắm bắt được các tương tác cục bộ giữa truy vấn và tài liệu.
- **Giới hạn trong biểu diễn vector:** Bi-encoder sử dụng không gian vector cố định để biểu diễn văn bản, điều này có thể hạn chế khả năng phân biệt các văn bản phức tạp hoặc có ngữ nghĩa tương tự.

2.2 Cross-Encoder

Cross-Encoder (hình 2) là một kiến trúc học sâu thường được sử dụng trong các bài toán xếp hạng tài liệu (document ranking) và đánh giá độ liên quan giữa cặp dữ liệu.

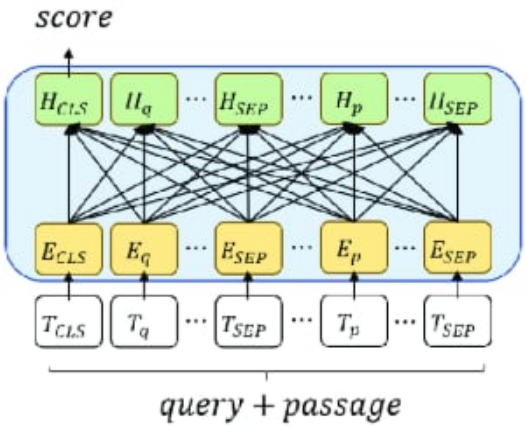

Figure 2: Kiến trúc Cross-Encoder.

Cross-Encoder xử lý query và document cùng lúc bằng cách nối chúng lại thành một chuỗi duy nhất, sau đó đưa qua một mạng neural (như BERT, RoBERTa,...). Mô hình sử dụng vector biểu diễn của token đặc biệt [CLS] để tính toán độ liên quan trực tiếp giữa query và document thông qua một tầng fully-connected.

Khác với Bi-encoder, Cross-encoder không mã hóa truy vấn và tài liệu độc lập. Thay vào đó, truy vấn và tài liệu được kết hợp thành một chuỗi đầu vào duy nhất, cho phép mô hình học mối quan hệ trực tiếp giữa các từ trong truy vấn và tài liệu. Điều này giúp Cross-encoder đạt độ chính xác cao hơn trong việc đánh giá độ liên quan, dù phải đánh đổi bằng chi phí tính toán cao hơn.

Ưu điểm:

- **Hiệu quả cao:** Cross-encoder xem xét mối liên kết toàn cục giữa truy vấn và tài liệu.
- **Học ngữ cảnh trực tiếp:** Mô hình có thể học được các mối quan hệ phức tạp mà Bi-encoder có thể bỏ sót.

Hạn chế:

- **Tốn tài nguyên:** Phải tính toán từ đầu cho mỗi cặp truy vấn-tài liệu, dẫn đến chi phí tính toán cao.
- **Không thể tiền tính toán:** Vì vector đặc trưng phụ thuộc vào cả truy vấn và tài liệu, không thể lưu trữ trước vector của tài liệu như Bi-encoder.

Dựa trên những ưu và nhược điểm trên, nhóm chúng em lựa chọn phương pháp sắp xếp lại 50 cặp query

và document liên quan nhất được chọn từ mô hình Bi-enocoder để tiết kiệm chi phí tính toán, sử dụng mô hình **itdainb/PhoRanker** (3).

2.3 Pipeline thực hiện

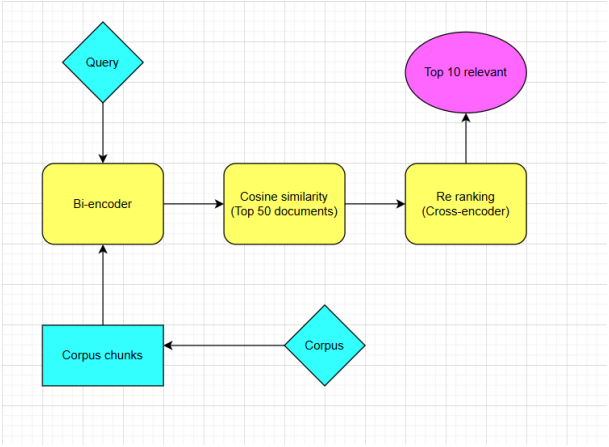

Figure 3: Pipeline thực hiện

Hình 3 minh họa pipeline thực hiện của bài toán.

- **Bước 1:** Bộ dữ liệu văn bản (corpus) được chia nhỏ thành các chunks theo phương pháp giới thiệu ở phần 3.3.
- **Bước 2:** Sử dụng mô hình Bi-encoder để tạo embedding (vector đặc trưng) cho query (truy vấn) và các corpus chunks.
- **Bước 3:** Thực hiện tính cosine similarity giữa embedding của query (q) và các embedding của các chunk (d) trong corpus.

$$cosine(q, d) = \frac{q \cdot d}{\|q\| \cdot \|d\|}$$

- **Bước 4:** Chọn 50 cặp $(query, document)$ có điểm tương đồng cao nhất ở bước 3.
- **Bước 5:** Các cặp $(query, document)$ được mã hóa đồng thời trong mô hình Cross-Encoder. Đầu vào của Cross-Encoder được biểu diễn dưới dạng:

$$x = [[CLS], query\,tokens, [SEP], doc\,tokens]$$

Vector đầu ra $h_{[CLS]}$ được lấy làm biểu diễn tổng hợp của cặp.

Điểm liên quan s được tính thông qua một tầng fully-connected:

$$s = \sigma(W \cdot h_{[CLS]} + b)$$

Trong đó:

- $h_{[CLS]}$: Vector biểu diễn đầu ra của token [CLS].
- W, b: Trọng số của tầng fully-connected.
- σ: hàm sigmoid

3 Thực nghiệm

3.1 Dataset

Trong bài toán Legal Document Retrieval (truy vấn văn bản pháp luật), chúng em lựa chọn bộ dữ liệu có cấu trúc từ các tài liệu pháp lý chuẩn mực, được xác thực từ các nguồn uy tín như hệ thống luật pháp quốc gia. Các văn bản pháp lý bao gồm các quy định, điều luật, và nghị định, được tổ chức và lưu trữ dưới dạng một văn bản. Tương tự như các bộ dữ liệu ngôn ngữ tự nhiên khác, các tài liệu pháp lý này thường có cấu trúc phức tạp và ngôn ngữ chuyên ngành, đòi hỏi các phương pháp xử lý ngôn ngữ tự nhiên đặc thù.

Chúng em sử dụng bộ dữ liệu được cung cấp từ cuộc thi BKAI-Vietnamese Legal Document Retrieval, bộ dữ liệu sẽ được phân chia thành các file chính:

- **corpus.csv** gồm các cột:
  - **text**: Một đoạn văn bản pháp luật bất kỳ (dạng string)
  - **cid**: Id của đoạn văn bản đó trong corpus (dạng int)
- **train.csv** gồm các cột:
  - **question**: Dạng văn bản của câu hỏi (dạng string)
  - **qid**: Mã id của câu hỏi (viết tắt của question_id, dạng string)
  - **context**: Các đoạn văn bản luật pháp liên quan (dạng list)
  - **cid**: Mã id của các đoạn văn bản pháp luật trong corpus có liên quan tới câu hỏi (viết tắt của context_id, dạng list)
- **public_test.csv** gồm các cột:
  - **question**: Dạng văn bản của câu hỏi (dạng string)
  - **qid**: Mã id của câu hỏi (viết tắt của question_id, dạng string)

Bộ dữ liệu sử dụng mang các đặc điểm quan trọng của ngôn ngữ pháp luật như:

- **Cách viết hoa, dấu chấm câu và các từ chuyên ngành**: Những yếu tố này cực kỳ quan trọng để đảm bảo tính chính xác của việc truy vấn và diễn giải.

- **Cấu trúc văn bản phức tạp**: Khác với các bộ dữ liệu ngôn ngữ tự nhiên như Penn Treebank hay Wikitext, ngữ nghĩa pháp luật đòi hỏi mô hình phải hiểu sâu hơn về cách sử dụng từ ngữ chuyên ngành và ngữ cảnh pháp lý.

Tuy nhiên, bộ dữ liệu tồn tại những sai sót, chẳng hạn như không nhất quán cid (context id) giữa hai file corpus và train, qid (question id) bị rỗng. Do đó, chúng em đã thực hiện điều chỉnh khoảng 250 mẫu dữ liệu (khoảng 0.1% bộ dữ liệu) để đảm bảo tính nhất quán của bộ dữ liệu.

3.2 Độ đo đánh giá

Để đánh giá kết quả truy vấn, chúng em sử dụng độ đo MRR@10 (Mean Reciprocal Rank) để đánh giá 10 kết quả đầu tiên từ hệ thống truy vấn. MRR@10 được tính như sau:

$$MRR = \frac{1}{N} \sum_{i=1}^{N} \frac{1}{rank_i}$$

Trong đó:

- N: Tổng số truy vấn (queries).
- $rank_i$: Hạng (rank) của kết quả đúng đầu tiên trong danh sách kết quả xếp hạng cho truy vấn i.

3.3 Vấn đề đối với văn bản pháp luật

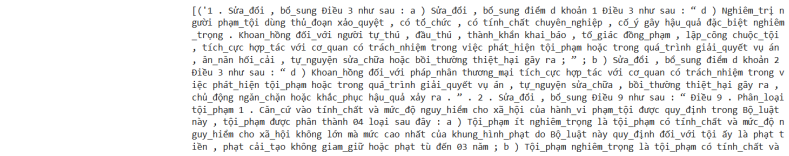
[('1 . Sửa_đổi , bổ_sung Điều 3 như sau : a) Sửa_đổi , bổ_sung điểm d khoản 1 Điều 3 như sau : " d) Nghiêm_trị n
gười phạm_tội dùng thủ_đoạn xảo_quyệt , có tổ_chức , có tính_chất chuyên_nghiệp , cố_ý gây hậu_quả đặc_biệt nghiêm
_trọng . Khoan_hồng đối_với người tự_thú , đầu_thú , thành_khẩn khai_báo , tố_giác đồng_phạm , lập_công chuộc_tội
, tích_cực hợp_tác với cơ_quan có trách_nhiệm trong việc phát_hiện tội_phạm hoặc trong quá_trình giải_quyết vụ án
, ăn_năn hối_cải , tự_nguyện sửa_chữa hoặc bồi_thường thiệt_hại gây ra ; " ; b) Sửa_đổi , bổ_sung điểm d khoản 2
Điều 3 như sau : " d) Khoan_hồng đối_với pháp_nhân thương_mại tích_cực hợp_tác với cơ_quan có trách_nhiệm trong v
iệc phát_hiện tội_phạm hoặc trong quá_trình giải_quyết vụ án , tự_nguyện sửa_chữa , bồi_thường thiệt_hại gây ra ,
chủ_động ngăn_chặn hoặc khắc_phục hậu_quả xảy ra . " . 2 . Sửa_đổi , bổ_sung Điều 9 như sau : " Điều 9 . Phân_loại
tội_phạm 1 . Căn_cứ vào tính_chất và mức_độ nguy_hiểm cho xã_hội của hành_vi phạm_tội được quy_định trong Bộ_luật
này , tội_phạm được phân thành 04 loại sau đây : a) Tội_phạm ít nghiêm_trọng là tội_phạm có tính_chất và mức_độ n
guy_hiểm cho xã_hội không lớn mà mức cao nhất của khung_hình_phạt do Bộ_luật này quy_định đối_với tội ấy là phạt t
iền , phạt cải_tạo không giam_giữ hoặc phạt tù đến 03 năm ; b) Tội_phạm nghiêm_trọng là tội_phạm có tính_chất và

Figure 4: Đoạn trích của văn bản dài nhất trong tập dữ liệu.

Các văn bản pháp luật thường có độ dài rất lớn, văn bản dài nhất bao gồm 261437 tokens (hình 4), lớn hơn rất nhiều so với khả năng xử lý của các mô hình Bi-encoder và Cross-encoder (256 tokens). Do đó, nhóm thực hiện chia văn bản thành các câu.

Câu dài nhất lúc này gồm 1722 tokens (đã giảm đi khoảng 150 lần so với văn bản gốc), nhóm nhận thấy có thể tách các câu này dựa trên các điều khoản

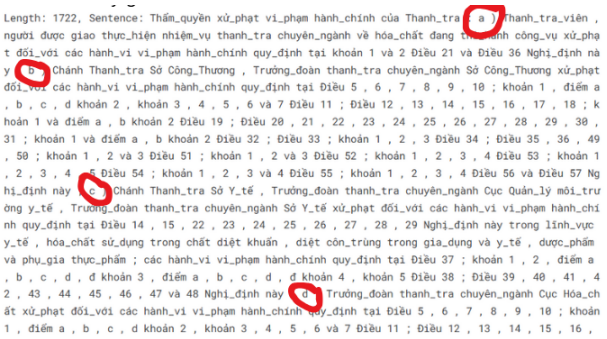
Length: 1722, Sentence: Thẩm_quyền xử_phạt vi_phạm hành_chính của Thanh_tra , a) Thanh_tra_viên ,
người được giao thực_hiện nhiệm_vụ thanh_tra chuyên_ngành về hóa_chất đang t hành công_vụ xử_pha
t đối_với các hành_vi vi_phạm hành_chính quy_định tại khoản 1 và 2 Điều 21 và Điều 36 Nghị_định nà
y , b) Chánh Thanh_tra Sở Công_Thương , Trưởng_đoàn thanh_tra chuyên_ngành Sở Công_Thương xử_phạt
đối_với các hành_vi vi_phạm hành_chính quy_định tại Điều 5 , 6 , 7 , 8 , 9 , 10 ; khoản 1 , điểm a
, b , c , d khoản 2 , khoản 3 , 4 , 5 , 6 và 7 Điều 11 ; Điều 12 , 13 , 14 , 15 , 16 , 17 , 18 ; k
hoản 1 và điểm a , b khoản 2 Điều 19 ; Điều 20 , 21 , 22 , 23 , 24 , 25 , 26 , 27 , 28 , 29 , 30 ,
31 ; khoản 1 và điểm a , b khoản 2 Điều 32 ; Điều 33 ; khoản 1 , 2 , 3 Điều 34 ; Điều 35 , 36 , 49
, 50 ; khoản 1 , 2 và 3 Điều 51 ; khoản 1 , 2 và 3 Điều 52 ; khoản 1 , 2 , 3 , 4 Điều 53 ; khoản 1
, 2 , 3 , 4 Điều 54 ; khoản 1 , 2 , 3 và 4 Điều 55 ; khoản 1 , 2 , 3 , 4 Điều 56 và Điều 57 Ng
hị_định này , c) Chánh Thanh_tra Sở Y_tế , Trưởng_đoàn thanh_tra chuyên_ngành Cục Quản_lý môi_trư
ờng y_tế , Trưởng_đoàn thanh_tra chuyên_ngành Sở Y_tế xử_phạt đối_với các hành_vi vi_phạm hành_chí
nh quy_định tại Điều 14 , 15 , 22 , 23 , 24 , 25 , 26 , 27 , 28 , 29 Nghị_định này trong lĩnh_vực
y_tế , hóa_chất sử_dụng trong chất diệt khuẩn , diệt côn_trùng trong gia_dụng và y_tế , dược_phẩm
và phụ_gia thực_phẩm ; các hành_vi vi_phạm hành_chính quy_định tại Điều 37 ; khoản 1 , 2 , điểm a
, b , c , d , đ khoản 3 , điểm a , b , c , d , đ khoản 4 , khoản 5 Điều 38 ; Điều 39 , 40 , 41 , 4
2 , 43 , 44 , 45 , 46 , 47 và 48 Nghị_định này , d) Trưởng_đoàn thanh_tra chuyên_ngành Cục Hóa_ch
ất xử_phạt đối_với các hành_vi vi_phạm hành_chính quy_định tại Điều 5 , 6 , 7 , 8 , 9 , 10 ; khoản
1 , điểm a , b , c , d khoản 2 , khoản 3 , 4 , 5 , 6 và 7 Điều 11 ; Điều 12 , 13 , 14 , 15 , 16 ,

Figure 5: Câu dài nhất sau khi chia văn bản thành các câu.

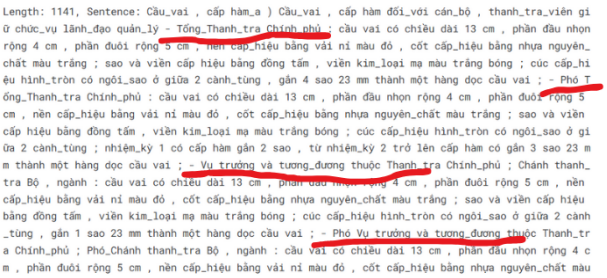
Length: 1141, Sentence: Cầu_vai , cấp hàm_a) Cầu_vai , cấp hàm đối_với cán_bộ , thanh_tra_viên gi
ữ chức_vụ lãnh_đạo quản_lý - Tổng_Thanh_tra Chính phủ ; cầu vai có chiều dài 13 cm , phần đầu nhọn
rộng 4 cm , phần đuôi rộng 5 cm , nền cấp_hiệu bằng vải nỉ màu đỏ , cốt cấp_hiệu bằng nhựa nguyên_
chất màu trắng ; sao và viền cấp hiệu bằng đồng tấm , viền kim_loại mạ màu trắng bóng ; cúc cấp_hi
ệu hình_tròn có ngôi_sao ở giữa 2 cành_tùng , gắn 4 sao 23 mm thành một hàng dọc cầu vai ; - Phó T
ổng_Thanh_tra Chính_phủ : cầu vai có chiều dài 13 cm , phần đầu nhọn rộng 4 cm , phần đuôi rộng 5
cm , nền cấp_hiệu bằng vải nỉ màu đỏ , cốt cấp_hiệu bằng nhựa nguyên_chất màu trắng ; sao và viền
cấp hiệu bằng đồng tấm , viền kim_loại mạ màu trắng bóng ; cúc cấp_hiệu hình_tròn có ngôi_sao ở gi
ữa 2 cành_tùng ; nhiệm_kỳ 1 có cấp hàm gắn 2 sao , từ nhiệm_kỳ 2 trở lên cấp hàm có gắn 3 sao 23 m
m thành một hàng dọc cầu vai ; - Vụ trưởng và tương_đương thuộc Thanh tra Chính_phủ ; Chánh thanh_
tra Bộ , ngành : cầu vai có chiều dài 13 cm , phần đầu nhọn rộng 4 cm , phần đuôi rộng 5 cm , nền
cấp_hiệu bằng vải nỉ màu đỏ , cốt cấp_hiệu bằng nhựa nguyên_chất màu trắng ; sao và viền cấp hiệu
bằng đồng tấm , viền kim_loại mạ màu trắng bóng ; cúc cấp_hiệu hình_tròn có ngôi_sao ở giữa 2 cành
_tùng , gắn 1 sao 23 mm thành một hàng dọc cầu vai ; - Phó Vụ trưởng và tương_đương thuộc Thanh_tr
a Chính_phủ ; Phó_Chánh thanh_tra Bộ , ngành : cầu vai có chiều dài 13 cm , phần đầu nhọn rộng 4 c
m , phần đuôi rộng 5 cm , nền cấp_hiệu bằng vải nỉ màu đỏ , cốt cấp_hiệu bằng nhựa nguyên_chất màu

Figure 6: Câu dài nhất sau khi chia câu thành các điều khoản nhỏ.

nhỏ (điều khoản a,b,c,.. được khoanh đỏ ở trên hình 5).

Câu dài nhất lúc này gồm 1141 tokens, nhóm nhận thấy có thể tách tiếp các câu này dựa trên các gạch đầu dòng (được gạch đỏ ở trên hình 6).

Cuối cùng, nhóm bỏ đi các câu có độ dài nhỏ (vô nghĩa), câu dài nhất lúc này gồm 558 tokens, tuy nhiên số lượng câu trong một văn bản tương đối lớn, với văn bản dài nhất bao gồm 2616 câu. Trung bình mỗi văn bản gồm 8 câu, mỗi câu trung bình gồm 24 tokens. Nhóm thực hiện biểu diễn văn bản bằng cách tính trung bình vector embedding giữa các câu ở bước Bi-encoder và tính trung bình score của các câu để biểu diễn mức độ liên quan của văn bản đối với câu truy vấn ở bước Cross-encoder.

Hình 7 minh họa cách tách một văn bản thành các câu theo thứ tự ở trên.

3.4 Kết quả

Table 1: Kết quả MRR@10 của các phương pháp

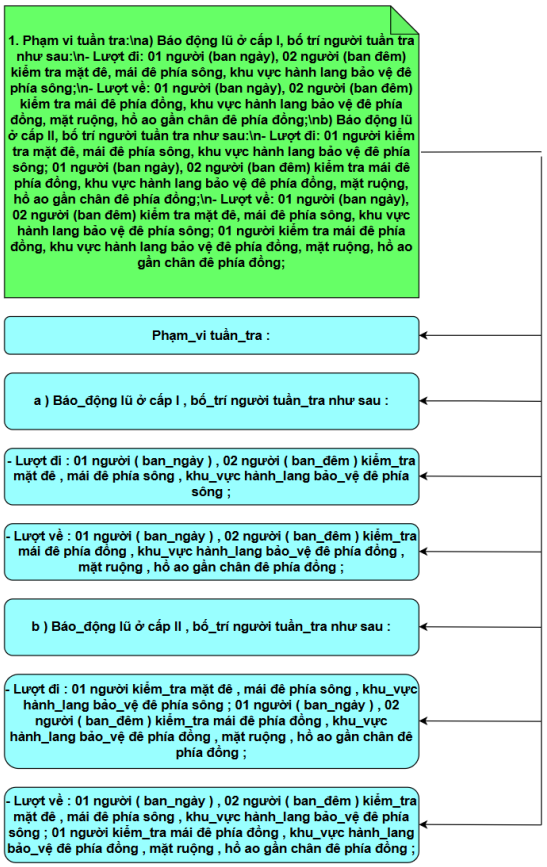

Figure 7: Biểu diễn văn bản thành các câu (văn bản gốc: màu xanh lá, các câu được tách ra: màu xanh dương).

Phương pháp	MRR@10
Bi-encoder	0.4647
Bi-encoder + Cross-encoder	0.546
Bi-encoder + Cross-encoder + Chunking	0.27

Bảng 1 biểu diễn kết quả của các phương pháp.

- **Chỉ sử dụng Bi-encoder:** Nếu chỉ sử dụng bi-encoder để chọn ra 10 văn bản liên quan nhất đối với mỗi câu truy vấn, kết quả độ đo MRR@10 là 0.4647
- **Sử dụng Bi-encoder kết hợp với Cross-encoder:** Nếu sử dụng bi-encoder để chọn ra 50 văn bản liên quan nhất đối với mỗi câu truy vấn và sử dụng Cross-encoder để sắp xếp lại và chọn ra 10 văn bản liên quan nhất thì chỉ số MRR@10 là 0.546, cho thấy sự hiệu quả của việc kết hợp hai phương pháp.
- **Sử dụng chunking:** Nếu sử dụng Bi-encoder và Cross-encoder kết hợp với phương pháp chunking giới thiệu ở phần 3.3, kết quả của độ đo là 0.27, kết quả tương đối thấp và không đạt được như kỳ vọng. Lý do và hướng giải quyết được đề cập ở phần 4.

4 Nhận xét

4.1 Vấn đề đối với việc biểu diễn văn bản bằng cách tính trung bình giữa các câu

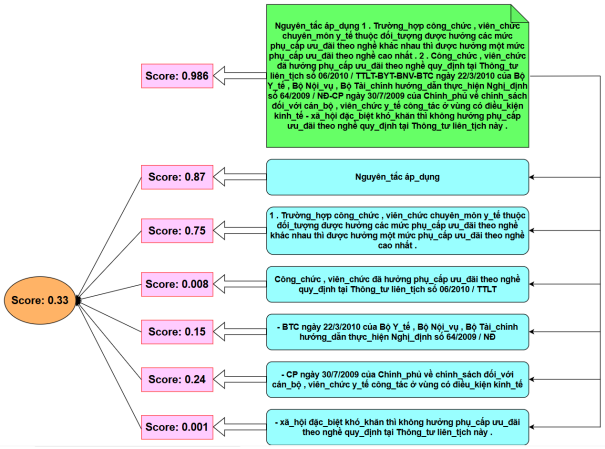

Figure 8: So sánh cross-encoder score giữa toàn bộ văn bản và từng câu của văn bản sau khi chunking.

Việc tách rời văn bản thành các câu sẽ không biểu diễn được mối liên quan giữa các câu, dẫn đến score của từng câu rời rạc sẽ không cao như score của một document (nếu document ngắn). Chẳng hạn như với câu query: **"Nguyên tắc áp dụng phụ cấp ưu đãi nghề y tế thế nào ?"**, do văn bản liên quan (màu xanh lá hình 8) tương đối ngắn, nằm trong khả năng xử lý của cả Bi-encoder và Cross-encoder nên độ tương đồng của toàn bộ văn bản đối với câu truy vấn tương đối cao (0.986) và nằm trong kết quả trả về. Ngược lại, nếu như tách văn bản đó thành các câu theo phương pháp ở trên (màu xanh dương), độ tương đồng của từng câu đối với câu truy vấn lại tương đối thấp dẫn đến score trung bình cũng thấp (0.33) nên hệ thống không trả về được (hình 8). Có thể thấy score của từng câu rời rạc sẽ không thể cao như score của toàn bộ văn bản được, và tất nhiên score trung bình cũng sẽ thấp hơn.

Bên cạnh đó, trong tập dữ liệu sử dụng, các phần liên quan đến câu truy vấn trong tập dữ liệu thường nằm ở đầu văn bản (những phần này nằm trong phạm vi xử lý của các mô hình Bi-encoder và Cross-encoder). Ví dụ với câu query: **"Hồ sơ đề nghị tự giải thể Quỹ giải thưởng tài năng nữ Việt Nam gồm những gì?** Mặc dù văn bản liên quan đến câu truy vấn tương đối dài (hình 9), sẽ bị cắt bớt (truncate) bởi Bi-encoder và Cross-encoder, tuy nhiên phần liên quan đến câu truy vấn lại nằm

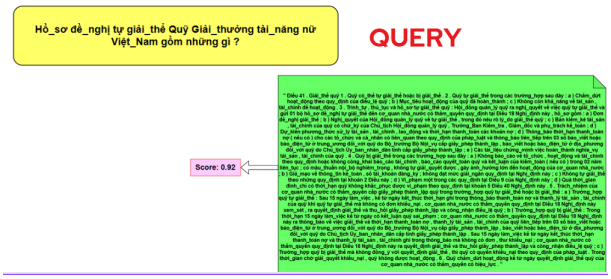

Figure 9: Văn bản liên quan đến câu query: **"Hồ sơ đề nghị tự giải thể Quỹ giải thưởng tài năng nữ Việt Nam gồm những gì?**

ở trong phạm vi xử lý của hai mô hình (được biểu diễn bởi những ô màu vàng ở hình 10) dẫn đến độ tương quan giữa văn bản và câu truy vấn vẫn rất cao (0.92). Trái lại, khi thực hiện chunking, mô hình có thể "nhìn thấy" toàn bộ câu văn trong văn bản, tuy nhiên cũng tương tự như ở trên, từng câu rời rạc sẽ không hiểu được bối cảnh toàn cục như ở toàn bộ văn bản, dẫn đến độ tương đồng trung bình thấp hơn (0.42) và không được trả về bởi hệ thống.

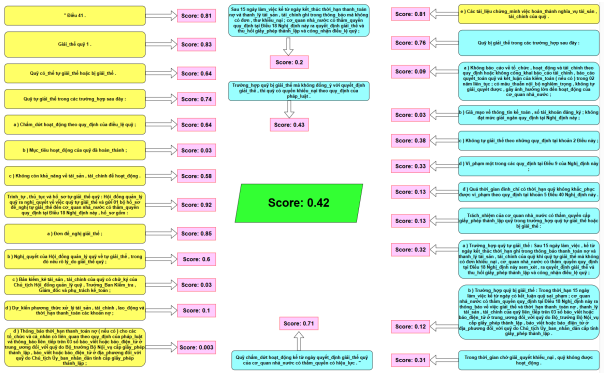

Figure 10: Văn bản (được chunking) liên quan đến câu query: **"Hồ sơ đề nghị tự giải thể Quỹ giải thưởng tài năng nữ Việt Nam gồm những gì?**, với những ô màu vàng là những câu nằm trong phạm vi xử lý của Bi-encoder và Cross-encoder.

Ngoài ra, trong một văn bản, đôi khi chỉ có một vài câu liên quan đến câu truy vấn, việc tính trung bình biểu diễn giữa các câu sẽ làm giảm tính liên quan giữa văn bản và câu truy vấn. Cũng với ví dụ ở hình 10, những câu liên quan đến văn bản (màu vàng) có score tương đối cao, tuy nhiên những câu còn lại (màu xanh dương) lại không liên quan đến câu truy vấn nên score rất thấp, dẫn đến score trung bình của toàn bộ văn bản cũng sẽ bị kéo xuống theo.

4.2 Ưu điểm của việc biểu diễn văn bản bằng cách tính trung bình giữa các câu

Phương pháp tiếp cận của nhóm sẽ đạt hiệu quả đối với những văn bản dài, nội dung câu trả lời nằm ở phần giữa hoặc cuối của văn bản. Ví dụ với câu query: **"Có phải công chứng hợp đồng chuyển nhượng quyền sử dụng đất khi được tặng cho trong thời kỳ hôn nhân hay không ?"**, văn bản liên quan đến câu truy vấn trên rất dài, và phần liên quan đến câu truy vấn nằm ngoài phạm vi xử lý của hai mô hình Bi-encoder và Cross-encoder (phần màu vàng là những câu nằm trong phạm vi xử lý của hai mô hình, còn phần màu đỏ là phần liên quan đến câu truy vấn, hình 12) dẫn đến score rất thấp (0.0004) và không được hệ thống trả về (hình 11). Trái lại, khi thực hiện chunking, việc "nhìn thấy" được toàn bộ văn bản, đặc biệt là câu văn ở trong ô màu đỏ hình 12 giúp cho mô hình nhận diện được sự tương đồng giữa văn bản và câu truy vấn, dẫn đến score cao hơn (0.09) và được trả về bởi hệ thống.

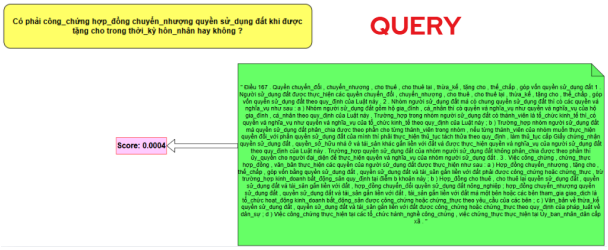

Figure 11: Văn bản liên quan đến câu query:**"Có phải công chứng hợp đồng chuyển nhượng quyền sử dụng đất khi được tặng cho trong thời kỳ hôn nhân hay không ?"**

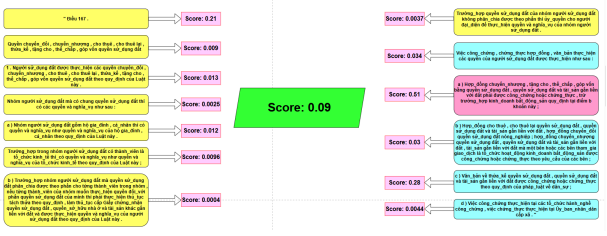

Figure 12: Văn bản (được chunking) liên quan đến câu query:**"Có phải công chứng hợp đồng chuyển nhượng quyền sử dụng đất khi được tặng cho trong thời kỳ hôn nhân hay không ?"**, với phần màu vàng là những câu nằm trong phạm vi xử lý của hai mô hình, còn phần màu đỏ là phần liên quan đến câu truy vấn.

4.3 Giải pháp

Có thể nhận thấy việc tính trung bình score giữa các câu sẽ làm cho những văn bản dài có score

thấp hơn những văn bản ngắn, do đó nhóm đề xuất sử dụng chunk có score cao nhất (nhóm chỉ thực hiện ở bước Cross-encoder, còn ở bước Bi-encoder vẫn thực hiện biểu diễn văn bản bằng cách tính embedding trung bình giữa các câu) đại diện cho văn bản. Với sự thay nhỏ trên, kết quả đã cải thiện đáng kể, từ 0.27 lên 0.46 (bảng 2).

Table 2: Kết quả MRR@10 của phương pháp cải thiện

Phương pháp	**MRR@10**
Bi-encoder (Mean) + Cross-encoder (Max) + Chunking	0.46

5 Kết luận và hướng phát triển

Có thể thấy, đối với tính chất của bộ dữ liệu mà nhóm sử dụng (phần lớn câu trả lời liên quan đến câu truy vấn nằm ở phần đầu của văn bản), phương pháp nhóm đề xuất chưa đạt được kết quả thật sự ấn tượng, tuy nhiên thông qua quá trình thực hiện đồ án và giải quyết bài toán bằng cách tiếp cận trên đã giúp nhóm tìm hiểu về phương pháp truy vấn kết hợp giữa Bi-encoder và Cross-encoder và phương pháp phân tích các văn bản trong bộ dữ liệu.

Bên cạnh đó, nhóm cũng đề xuất một số hướng phát triển cho bài toán:

- Thực hiện phương pháp chunking khác thay vì tách văn bản thành các câu ngắn như trên (khó nắm bắt được ngữ cảnh toàn cục của văn bản).
- Nhóm chỉ sử dụng mô hình pretrain mà không thực hiện tinh chỉnh các mô hình Bi-encoder và Cross-Encoder (do chi phí tính toán hạn chế), do đó có thể fine-tune mô hình để đạt kết quả cao hơn.
- Sử dụng hybrid retrieval kết hợp giữa sparse retrieval (sử dụng ElasticSearch) và dense retrieval (sử dụng Bi-encoder) để chọn lọc các văn bản liên quan tốt hơn (thay vì chỉ sử dụng Bi-encoder như hiện tại) trước khi re-rank bằng Cross-encoder.
- Ở bước Bi-encoder, thực hiện tính cosine similarity score đối với mỗi chunk của văn bản rồi chọn ra 50 văn bản có chunk có độ tương đồng nhất với câu truy vấn thay vì tính cosine similarity score cho cả văn bản (biểu diễn bằng cách tính trung bình các vector embedding của các chunk).

References

[1] SoICT Hackathon 2024 - Legal Document Retrieval https://aihub.ml/competitions/715

[2] BKAI bi-encoder. https://huggingface.co/bkai-foundation-models/vietnamese-bi-encoder

[3] PhoRanker. https://huggingface.co/itdainb/PhoRanker