ĐẠI HỌC QUỐC GIA TP. HCM
TRƯỜNG ĐẠI HỌC CÔNG NGHỆ THÔNG TIN

NGUYỄN VĂN KIỆT

NGHIÊN CỨU XÂY DỰNG MÔ HÌNH ĐỌC HIỂU TỰ ĐỘNG CHO VĂN BẢN TIẾNG VIỆT
(Machine Reading Comprehension for Vietnamese Texts)

Cán bộ hướng dẫn khoa học:

PGS. TS. NGUYỄN LƯU THÙY NGÂN

TS. NGUYỄN GIA TUẤN ANH

LUẬN ÁN TIẾN SĨ KHOA HỌC MÁY TÍNH

TP. HỒ CHÍ MINH – Năm 2024

Công trình được hoàn thành tại:
TRƯỜNG ĐẠI HỌC CÔNG NGHỆ THÔNG TIN, ĐHQG TP.HCM

Cán bộ hướng dẫn khoa học
1. PGS. TS. NGUYỄN LƯU THÙY NGÂN
2. TS. NGUYỄN GIA TUẤN ANH

Phản biện 1: .....
Phản biện 2: .....
Phản biện 3: .....
Phản biện độc lập 1: .....
Phản biện độc lập 2: .....
Luận án được bảo vệ trước Hội đồng chấm luận án họp tại: Trường Đại học Công nghệ thông tin, Đại học Quốc gia TP. Hồ Chí Minh

Vào lúc: … giờ … ngày … tháng … năm …

Có thể tìm hiểu luận án tại thư viện:
– Thư viện Quốc gia Việt Nam
– Thư viện Đại học Quốc gia Tp. Hồ Chí Minh
– Thư viện Trường Đại học Công nghệ Thông tin, Đại học Quốc gia TP. Hồ Chí Minh.

MỤC LỤC

DANH MỤC CÁC TỪ VIẾT TẮT

Từ viết tắt	Nội dung tiếng Anh	Nội dung tiếng Việt
Từ viết tắt từ tiếng Anh		
NLP	Natural Language Processing	Xử lý ngôn ngữ tự nhiên
AI	Artificial Intelligence	Trí tuệ nhân tạo
NLU	Natural Language Understanding	Hiểu ngôn ngữ tự nhiên
MRC	Machine Reading Comprehension	Đọc hiểu tự động
QA	Question Answering	Hỏi đáp tự động
SE	Search Engine	Công cụ tìm kiếm
VA	Virtual Assistant	Trợ lý ảo
QG	Question Generation	Tạo sinh câu hỏi
NLI	Natural Language Inference	Suy luận ngôn ngữ tự nhiên
VLSP	Vietnamese Language and Speech Processing	Hội thảo xử lý ngôn ngữ và giọng nói tiếng Việt
IE	Information Extraction	Rút trích thông tin
ML	Machine Learning	Học máy
EM	Exact Match	Độ đo chính xác
BERT	Bidirectional Encoder Representations from Transformers	Biểu diễn bộ mã hóa hai chiều từ Transformers
SAE	Supervised Answer Extractor	Bộ trích xuất câu trả lời
EE	Evidence Extractor	Bộ truy xuất minh chứng
LM	Language Model	Mô hình ngôn ngữ
POS	Part-Of-Speech	Nhãn từ loại
Từ viết tắt từ tiếng Việt		
TB		Trung bình
NCS		Nghiên cứu sinh
NNC		Nhà nghiên cứu
MHNN		Mô hình ngôn ngữ
HNQT		Hội nghị quốc tế
CTNC		Công trình nghiên cứu

DANH MỤC CÁC BẢNG

DANH MỤC CÁC HÌNH

DANH MỤC CÁC THUẬT TOÁN

CHƯƠNG 1: TỔNG QUAN

1.1. Động lực nghiên cứu

NCS tập trung nghiên cứu và xây dựng các bộ ngữ liệu phục vụ cho nghiên cứu đọc hiểu tự động cho văn bản tiếng Việt. Các mô hình MRC dựa trên kiến trúc MHNN, xây dựng các thành phần chính của các mô hình đọc hiểu và hỏi đáp tự động trên văn bản tiếng Việt. Áp dụng mô hình đọc hiểu có độ chính xác cao như một công nghệ nền tảng cốt lõi vào các ứng dụng hỗ trợ tìm kiếm thông tin như mô hình QA. Đọc hiểu tự động đã được ứng dụng vào một số ứng dụng thực tế. Đọc hiểu tự động có thể hỗ trợ các mô hình tìm kiếm thông tin ngày càng thông minh hơn. Đặc biệt, các mô hình MRC có thể thúc đẩy sự phát triển khả năng đọc hiểu văn bản của các trợ lý ảo. Luận án này thực hiện theo hai định hướng nghiên cứu chính: (1) xây dựng ngữ liệu để đánh giá các mô hình MRC trên văn bản tiếng Việt và (2) đề xuất các mô hình MRC trên ngữ liệu tiếng Việt.

1.2. Các đóng góp chính

Luận án đã có ba nội dung đóng góp chính:

- **Nội dung #1 – Xây dựng ngữ liệu và đánh giá các mô hình đọc hiểu trên ngữ liệu tiếng Việt**: Trong Nội dung #1, luận án tập trung vào xây dựng các bộ ngữ liệu cho tiếng Việt (là một ngôn ngữ có ít các ngữ liệu cho việc phát triển và đánh giá các thuật toán học máy trong AI và NLP). Các bộ ngữ liệu đã được trình bày trong **Chương 3** và được công bố tại các tạp chí và hội nghị với các công trình: **[CT1]**, **[CT4]**, **[CT5]** và [**CT6**].
- **Nội dung #2 – Đề xuất mô hình đọc hiểu tiếng Việt tích hợp MHNN với truy xuất minh chứng**: Trong Nội dung #2, kế thừa từ các kết quả thử nghiệm đầu tiên đã đạt được trên các bộ ngữ liệu trong **Nội dung #1**, luận án xây dựng, thiết kế và triển khai mô hình MRC dựa trên những phân tích đặc điểm tiếng Việt và các MHNN dựa trên kiến trúc Transformer, với mô hình được đề xuất là ViReader. Các khám phá này đã được trình bày trong Chương 4, và một phần trong **Chương 5**. Các đóng góp nghiên cứu về ViReader được công bố tại các tạp chí và hội nghị với các công trình: **[CT2]** và **[CT3]**.
- **Nội dung #3**: **Xây dựng phương pháp hỏi đáp tiếng Việt tích hợp đọc hiểu tự động**: Trong **Nội dung #3**, kế thừa từ các kết quả thử nghiệm đầu tiên đã đạt được trên các bộ ngữ liệu trong **Nội dung #1**, luận án xây dựng, thiết kế và triển khai mô hình QA dựa trên những đóng góp nghiên cứu của mô hình đọc hiểu ViReader (trong Nội dung #2) để đề xuất các mô hình QA trên ngữ liệu tiếng Việt: **XLMRQA** và **ViQAS**, được công bố tại các tạp chí và hội nghị với các công trình: **[CT3]** và [**CT7**]**.**

1.3. Mục đích, đối tượng và phạm vi nghiên cứu

Mục đích: Để có thể nghiên cứu và triển khai mô hình MRC trên ngữ liệu tiếng Việt. **Đối tượng**: NCS thực hiện bài toán đọc hiểu tự động trên ngữ liệu tiếng Việt. **Phạm vi**: Nghiên cứu này được giới hạn trên đọc hiểu tự động có câu trả lời được

rút trích trực tiếp từ văn bản tiếng Việt trên cả miền mở (các bài viết trên Wikipedia tiếng Việt) và miền đóng (các tin tức về sức khỏe).

1.4. Ý nghĩa khoa học và thực tiễn

- **Nghiên cứu và đề xuất các bộ ngữ liệu cho nghiên cứu đọc hiểu trên ngữ liệu tiếng Việt**: NCS đã cung cấp cho cộng đồng sử dụng một bộ ngữ liệu đầy thách thức với nhiều nhóm tham gia đến từ các trường đại học và các doanh nghiệp uy tín. Những thách thức này đã thúc đẩy các nghiên cứu các bộ ngữ liệu và mô hình trong đọc hiểu và hỏi đáp tự động trên ngữ liệu văn bản tiếng Việt.
- **Nghiên cứu và đề xuất các phương pháp đọc hiểu và hỏi đáp tự động trên ngữ liệu tiếng Việt**: NCS đã đề xuất ViReader, một phương pháp đọc hiểu tự động tiếng Việt tích hợp MHNN và truy xuất minh chứng. Tiếp theo, NCS đã đề xuất **XLMRQA** và **ViQAS**, một mô hình QA đầu tiên dựa trên các mô hình đọc hiểu tự động trên ngữ liệu tiếng Việt.
- **Các đóng góp nghiên cứu của nghiên cứu sinh có thể thúc đẩy sự phát triển nhiều nhiệm vụ nghiên cứu khác của hiểu ngôn ngữ tự nhiên tiếng Việt**: đọc hiểu tự động, hỏi đáp trong văn bản, hỏi đáp trực quan và tạo sinh câu hỏi – câu trả lời. Những kết quả khả quan có thể ứng dụng vào các ứng dụng thực tế, ví dụ như các hệ thống hỏi đáp trong văn bản luật hoặc trong văn bản sức khỏe.

1.5. Bố cục luận án

Luận án được tổ chức thành 06 Chương, các công trình khoa học công bố và tài liệu tham khảo. Các đóng góp chính được được tổ chức trong các chương chính: **Chương 3**, **Chương 4** và **Chương 5**.

CHƯƠNG 2: CƠ SỞ LÝ THUYẾT VÀ CÁC CÔNG TRÌNH LIÊN QUAN

2.1. Lịch sử đọc hiểu tự động

Xử lý ngôn ngữ tự nhiên (Natural Language Processing – NLP) đã chứng kiến lịch sử hình thành và phát triển lâu dài qua gần năm thập kỷ của đọc hiểu tự động. Đọc hiểu tự động (Machine Reading Comprehension - MRC) là phương pháp để đánh giá mức độ hiểu văn bản của máy tính, thuộc lĩnh vực hiểu ngôn ngữ tự nhiên (Natural Language Understanding - NLU), gắn liền với sự phát triển của hỏi đáp tự động (Question Answering), rút trích thông tin (Information Extraction) và học máy (Machine Learning). Để hiểu ngôn ngữ tự nhiên, các nhà nghiên cứu (NNC) xử lý ngôn ngữ tự nhiên đã thực hiện và đánh giá nhiều nhiệm vụ nghiên cứu trong thời gian qua: (1) Những vấn đề cơ bản nền tảng trong xử lý ngôn ngữ bao gồm việc tách từ, gán nhãn từ loại, phân tích cú pháp, gán nhãn thực thể và MHNN; và (2) xây dựng các ứng dụng NLP (phân loại văn bản, phân tích cảm xúc, đọc hiểu tự động và hỏi đáp tự động). Để đánh giá mức độ hiểu một văn bản sâu hơn, đọc hiểu tự động yêu cầu máy tính phải hiểu một bài đọc (văn bản) và dự đoán câu trả lời cho các câu hỏi liên quan đến bài đọc đó. Đọc hiểu tự động là một bài toán được cộng đồng nghiên cứu NLP được quan tâm với những nguyên nhân chính sau: (1) các bộ ngữ liệu chất lượng và kích thước lớn được công bố cho đánh giá và phát triển các mô hình đọc hiểu tự động dựa trên học máy, đặc biệt trên các ngôn ngữ giàu tài nguyên nghiên cứu như tiếng Anh và (2) sự phát triển của các mô hình đọc hiểu dựa trên kiến trúc nơ-ron và MHNN cùng với khả năng tính toán của máy tính.

2.2. Định nghĩa đọc hiểu tự động

Định nghĩa 1: Bài toán đọc hiểu tự động có thể mô hình hoá dựa trên học máy có giám sát: cho một tập hợp các mẫu ngữ liệu huấn luyện $\{(D_i, Q_i, A_i)\}_{i=1}^{N}$ và mục đích là xây dựng một hàm dự đoán f nhận một đầu vào là một văn bản D_i và một câu hỏi Q_i và trả về đầu ra là một câu trả lời A_i. Bài toán đọc hiểu tự động được mô tả như sau:

Đầu vào (Input):

- Một câu hỏi Q_i;
- Một văn bản D_i;
- Một tập huấn luyện n bộ ba câu hỏi-văn bản-câu trả lời được tạo sẵn (D_1, Q_1, A_1), (D_2, Q_2, A_2), ..., (D_N, Q_N, A_N).

Đầu ra (Output):

- Một mô hình đọc hiểu f đã được huấn luyện:

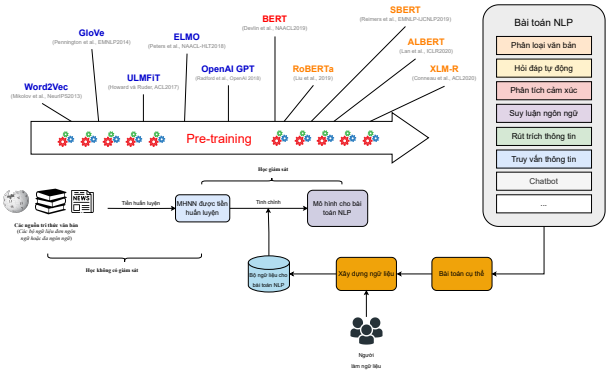
$$f: (D_i, Q_i) \rightarrow A_i \quad (2.2)$$

Định nghĩa 2: Bộ dữ liệu đọc hiểu tự động được định nghĩa là một tập hợp N bộ ba $\{(D_i, Q_i, A_i)\}_{i=1}^{N}$. Trong đọc hiểu tự động, câu trả lời A_i có thể có các dạng

rất khác nhau. Dựa vào loại câu trả lời, các bài toán đọc hiểu có thể chia thành bốn dạng khác nhau như sau: đọc hiểu với điền vào chỗ trống (Cloze-based MRC), đọc hiểu trắc nghiệm với nhiều lựa chọn (Multiple-choice MRC), đọc hiểu với câu trả lời được rút trích trực tiếp từ văn bản (Span-based MRC) và đọc hiểu với câu trả lời tự do (Free form-based MRC).

2.3. Phương pháp đọc hiểu tự động

Theo thời gian phát triển của đọc hiểu tự động, các phương pháp đọc hiểu tự động được phân thành bốn phương pháp chính: mô hình đơn giản dựa trên các quy tắc, học máy dựa trên đặc trưng, các phương pháp dựa trên mạng nơ-ron truyền thống và các phương pháp đọc hiểu tích hợp MHNN (dựa theo học chuyển tiếp). Trong luận án này, NCS mong muốn tận dụng hướng tiếp cận học chuyển tiếp dựa trên MHNN trong thiết kế mô hình đề xuất để cải thiện hiệu suất bài toán đọc hiểu tự động tiếng Việt. Học chuyển tiếp là một phương pháp học máy phù hợp với tiếng Việt – được biết là ngôn ngữ ít tài nguyên nghiên cứu. Học chuyển tiếp yêu cầu ít ngữ liệu được gán nhãn hơn so với các phương pháp dựa trên học sâu truyền thống. Điểm đáng chú ý nhất trong các công trình gần đây về học chuyển tiếp trong NLP nằm ở việc sử dụng các biểu diễn ngôn ngữ được huấn luyện sẵn trên lượng lớn ngữ liệu chưa được gán nhãn như BERT. Do đó, phương pháp BERT có thể được sử dụng trong các bài toán trong xử lý ngôn ngữ trên các ngôn ngữ ít tài nguyên, đặc biệt, liên quan đến xây dựng mô hình đọc hiểu tự động cho văn bản tiếng Việt.



Hình 2.1. Phương pháp tiếp cận các bài toán NLP theo học chuyển tiếp được sử dụng cho các ngôn ngữ ít tài nguyên.

2.4. Ngữ liệu đọc hiểu tự động

NCS tiến hành khảo sát những bộ ngữ liệu nổi tiếng trên thế giới và ảnh hưởng đến việc xây dựng ngữ liệu đọc hiểu tự động. MCTest là một bộ ngữ liệu đầu tiên cho đánh giá đọc hiểu tự động sử dụng học máy giám sát với dạng đọc hiểu trắc

nghiệm. MCTest có hạn chế đáng kể do kích thước nhỏ nên ngay sau đó, RACE được công bố với kích thước lớn để phù hợp cho việc huấn luyện các mô hình học sâu. Để làm phức tạp các bài toán đọc hiểu tự động và ngữ liệu đủ lớn cho các mô hình học sâu, một loạt ngữ liệu được tạo ra trên nhiều ngôn ngữ khác nhau như SQuAD, NewsQA, CMRC, KorQuAD, JaQuAD, FQuAD và SberQuAD.

2.5. Thông số đánh giá

Để đánh giá mức độ đọc hiểu của các mô hình, chúng ta cần so sánh câu trả lời do mô hình dự đoán với câu trả lời mong đợi. Hai thông số đánh giá EM và F_1 được sử dụng để đo hiệu suất của các mô hình MRC.

2.6. Hỏi đáp dựa trên đọc hiểu tự động

Đọc hiểu tự động (Machine Reading Comprehension) và hỏi đáp tự động (Question Answering) tưởng chừng là hai bài toán độc lập nhưng chúng có mối quan hệ gần gũi với nhau. Đọc hiểu tự động là một trường hợp đặc biệt của hỏi đáp tự động vì mô hình hỏi đáp tìm câu trả lời cho câu hỏi trên một văn bản thay vì trên nhiều văn bản. Mô hình hỏi đáp và đọc hiểu tự động có thể sử dụng chung mô hình hoá bài toán, phương pháp tiếp cận và phương pháp đánh giá. Những mô hình đọc hiểu và hỏi đáp đầu tiên trên những bộ ngữ liệu có kích thước nhỏ và miền đóng của một lĩnh vực nào đó. Những mô hình QA này chủ yếu được phát triển tập trung trên ngôn ngữ tiếng Anh. Phương pháp chủ yếu được thực hiện trong các mô hình QA này là dựa trên các quy tắc và cũng như thiếu đánh giá tự động nghiêm ngặt.

TREC đã tổ chức một chuỗi hội thảo liên tục tập trung vào các lĩnh vực nghiên cứu truy xuất thông tin khác nhau. Trong đó, hỏi đáp tự động (Question Answering) là một bài toán quan trọng được cộng đồng nghiên cứu quan tâm tại Hội thảo TREC qua nhiều năm. Các mô hình QA thành công tại Hội thảo TREC tương đối phức tạp bao gồm rất nhiều thành phần, trong đó có 03 thành phần chính: xử lý câu hỏi, xử lý văn bản và xử lý câu trả lời. Mô hình này có nhiều tác động đến các nghiên cứu về mô hình QA cho đến khi các mô hình đọc hiểu có giám sát xuất hiện. Trong hơn năm năm qua, đọc hiểu tự động như một thành phần không thể thiếu đối với mô hình QA hiện đại. Mô hình hỏi đáp DrQA bao gồm hai thành phần chính: mô hình truy xuất văn bản (Document Retriever) và mô hình đọc hiểu (Document Reader), khác nhiều so với mô hình QA truyền thống phức tạp trước đây gồm rất nhiều thành phần (xử lý câu hỏi, xử lý văn bản và xử lý câu trả lời) và mỗi thành phần có rất nhiều xử lý phức tạp. Đối với ngôn ngữ ít tài nguyên như tiếng Việt, các mô hình và các bộ ngữ liệu cần được quan tâm và nghiên cứu nghiêm túc hơn.

Định nghĩa 3: Cho một câu hỏi Q, mục tiêu của mô hình hỏi đáp là tìm kiếm văn bản D có liên quan nhất từ một tập hợp các văn bản ứng viên C bằng cách sử dụng mô hình truy xuất văn bản, và sau đó, trích xuất câu trả lời đúng A từ đoạn văn đã chọn từ mô hình đọc hiểu tự động. Hỏi đáp dựa trên đọc hiểu tự động được mô hình hóa như sau:

- **Đầu vào**: một tập hợp các văn bản $\mathcal{D} = \{D_1, D_2, \dots, D_N\}$ và một câu hỏi Q;

- **Đầu ra**: một câu trả lời $A = Reader(Q, Retriever(Q, \mathcal{D}))$;

Cụ thể, mô hình hỏi đáp dựa trên đọc hiểu được biểu diễn bởi hai mô-đun chính:

Mô-đun truy xuất văn bản (Retriever): Thành phần truy xuất văn bản chọn một tập con các đoạn văn $\mathcal{D}_i'$ từ tập đoạn văn ứng viên lớn $\mathcal{D}$ dựa trên sự liên quan đến câu hỏi Q. Hàm truy xuất văn bản có thể được tính theo Công thức (2.7).

$$\mathcal{D}_i' = Retriever(Q, \mathcal{D}, K) \quad (2.7)$$

Với K là số văn bản cần truy xuất và xác định trước.

Mô-đun đọc hiểu văn bản (Reader): Thành phần đọc hiểu nhận vào một câu hỏi Q và một văn bản đã truy xuất $\mathcal{D}_i'$ và trích xuất câu trả lời A_i. Hàm đọc hiểu tự động nhận vào một câu hỏi Q có thể được tính theo Công thức (2.8).

$$A_i = Reader(Q, \mathcal{D}_i'); \quad (2.8)$$

Trong trường hợp K>1, mô hình hỏi đáp có thêm một mô-đun xếp hạng K câu trả lời ứng cử theo Công thức (2.9).

$$A = argmax_{i=1}^{K} \text{Score}(A_i) \quad (2.9)$$

Đánh giá của mô hình hỏi đáp thường liên quan đến cả hiệu suất truy xuất (bộ truy xuất chọn các đoạn văn có liên quan) và hiệu suất đọc hiểu (mô hình đọc hiểu trích xuất câu trả lời từ các đoạn văn đã chọn) trong ngữ cảnh của một câu hỏi cụ thể. Trong luận án, NCS sẽ trình bày và thảo luận về các nghiên cứu liên quan đến mô hình QA và cũng chứng minh rằng đọc hiểu tự động hữu ích trong việc đề xuất mô hình QA cho ngữ liệu tiếng Việt (xem Chương 5).

2.7. Thách thức trong đọc hiểu và hỏi đáp tự động tiếng Việt

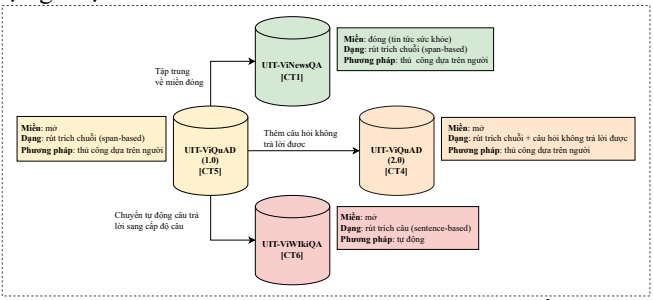
Qua quá trình khảo sát, NCS nhận thấy đọc hiểu và hỏi đáp tự động cho văn bản tiếng Việt có ba thách thức chính sau:

- Thách thức #1: Tiếng Việt là ngôn ngữ ít ngữ liệu cho những nghiên cứu NLP.
- Thách thức #2: Tiếng Việt chưa có nghiên cứu nhiều về các mô hình đọc hiểu tự động.
- Thách thức #3: Tiếng Việt chưa có nhiều nghiên cứu về hệ thống hỏi đáp, đặc biệt tích hợp mô hình MRC vào hệ thống QA trên văn bản tiếng Việt.

CHƯƠNG 3: XÂY DỰNG NGỮ LIỆU VÀ ĐÁNH GIÁ ĐỌC HIỂU TỰ ĐỘNG TRÊN VĂN BẢN TIẾNG VIỆT

Tóm tắt

Khoảng hơn 97 triệu người sử dụng tiếng Việt để giao tiếp hằng ngày trên thế giới. Mặc dù tiếng Anh và tiếng Trung chứng kiến sự phát triển nhanh chóng của các bộ ngữ liệu và các mô hình trong MRC, nhưng có rất ít nghiên cứu về MRC trong tiếng Việt, MRC giúp máy tính hiểu một văn bản và trả lời các câu hỏi liên quan đến văn bản đó. Do thiếu bộ ngữ liệu chuẩn trong tiếng Việt, NCS đã xây dựng và công bố bốn bộ ngữ liệu MRC trong văn bản tiếng Việt từ miền mở đến miền đóng: UIT-ViQuAD (phiên bản 1.0 [CT5] và phiên bản 2.0 [CT4]), UIT-ViNewsQA [CT1] và UIT-ViWikiQA [CT6] có kích thước từ 22K đến gần 36K cặp câu hỏi – câu trả lời. Hình 3.1 trình bày tổng quan về các đóng góp về xây dựng các bộ ngữ liệu của NCS.



Hình 3.1. Đóng góp các bộ ngữ liệu cho MRC tiếng Việt.

3.1. Giới thiệu và động lực xây dựng các bộ ngữ liệu

Đầu tiên, NCS xây dựng UIT-ViQuAD, một bộ ngữ liệu mới cho ngôn ngữ ít tài nguyên như tiếng Việt để đánh giá các mô hình đọc hiểu và hỏi đáp. Bộ ngữ liệu này bao gồm hơn 23.000 câu hỏi được tạo bởi người dựa trên 5.109 bài đọc từ 174 bài báo tiếng Việt trên Wikipedia (như một nguồn văn bản chứa nhiều tri thức). Đặc biệt, NCS đề xuất một quy trình tạo bộ ngữ liệu mới để đánh giá các mô hình MRC trong văn bản tiếng Việt. Các phân tích chuyên sâu trên bộ ngữ liệu cho thấy rằng bộ ngữ liệu của NCS yêu cầu các khả năng suy luận đơn giản như so khớp từ (Word Matching) và các yêu cầu suy luận phức tạp như suy luận trên một câu hoặc nhiều câu. Thêm vào đó, NCS tiến hành các thử nghiệm về các phương pháp MRC tiên tiến trên tiếng Anh và tiếng Trung như những mô hình đọc hiểu cơ sở đầu tiên trên UIT-ViQuAD, sẽ được so với hiệu suất của các mô hình đọc hiểu khác trong tương lai. NCS cũng ước tính hiệu suất của người trên bộ ngữ liệu và được so sánh với kết quả thử nghiệm của các mô hình đọc hiểu dựa trên học máy và MHNN.

Qua đó, sự khác biệt đáng kể giữa hiệu suất của người và mô hình tốt nhất trên bộ ngữ liệu cho thấy rằng các cải tiến MRC có thể được khám phá trên UIT-ViQuAD thông qua những nghiên cứu trong tương lai. Bộ ngữ liệu UIT-ViQuAD được sử dụng miễn phí nhằm khuyến khích cộng đồng nghiên cứu giải quyết những thách thức trong MRC tiếng Việt.

Lấy cảm hứng từ bộ ngữ liệu UIT-ViQuAD [CT5] và đa dạng MRC trong tiếng Việt, NCS cũng mở rộng nghiên cứu và tạo thêm 03 bộ ngữ liệu khác cho nghiên cứu đọc hiểu tiếng Việt ở nhiều khía cạnh khác nhau: UIT-ViNewsQA [CT1] cho miền văn bản tin tức sức khỏe thay vì miền mở Wikipedia tiếng Việt như UIT-ViQuAD; UIT-ViWikiQA [CT6] cho câu trả lời cấp độ câu thay vì câu trả lời cấp độ từ hoặc cụm từ như UIT-ViQuAD; và UIT-ViQuAD phiên bản 2.0 [CT4] mở rộng thêm những câu hỏi không có câu trả lời.

3.2. Bộ ngữ liệu đọc hiểu tự động cho văn bản Wikipedia tiếng Việt

Bài đọc: "Nước biển có độ mặn không đồng đều trên toàn thế giới mặc dù phần lớn có độ mặn nằm trong khoảng từ **3,1%** tới 3,8%. Khi sự pha trộn với nước ngọt đổ ra từ các con sông hay gần các sông băng đang tan chảy thì nước biển nhạt hơn một cách đáng kể. Nước biển nhạt nhất có tại **vịnh Phần Lan**, một phần của biển Baltic."

Câu hỏi 1: "Độ mặn thấp nhất của nước biển là bao nhiêu?"

Câu trả lời: "**3.1%**"

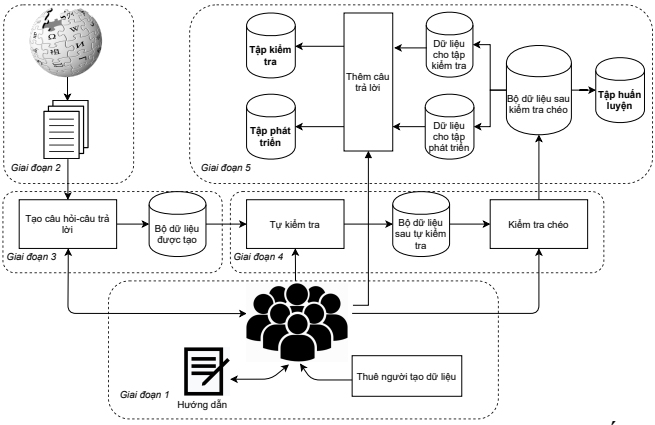
Câu hỏi 2: "Nước biển ở đâu có hàm lượng muối thấp nhất?"

Câu trả lời: "**Vịnh Phần Lan**"

Hình 3.2. Minh hoạ câu hỏi cho đọc hiểu trên Wikipedia tiếng Việt.

Trong nghiên cứu này, NCS chọn các văn bản tiếng Việt trên Wikipedia - nguồn tri thức mở không ngừng phát triển có thể tận dụng để phát triển các nghiên cứu thông minh tự động. Để hiểu rõ hơn nghiên cứu này, Hình 3.2 minh hoạ cho một số ví dụ đọc hiểu trên văn bản Wikipedia tiếng Việt. Trong nghiên cứu [CT5], NCS có ba đóng góp chính được mô tả như sau: (1) NCS tạo bộ ngữ liệu để đánh giá MRC tiếng Việt: UIT-ViQuAD bao gồm 23.074 câu hỏi do người tạo dựa trên 5.109 văn bản từ 174 bài báo Wikipedia tiếng Việt. (2) Để hiểu sâu hơn về bộ ngữ liệu này, NCS phân tích bộ ngữ liệu dựa trên độ dài (độ dài câu hỏi, độ dài câu trả lời và độ dài bài đọc) và phân tích bộ ngữ liệu dựa trên các khía cạnh ngôn ngữ khác nhau như loại câu hỏi, loại câu trả lời và loại suy luận. (3) Để có cái nhìn sâu sắc đầu tiên về những đánh giá trên bộ ngữ liệu UIT-ViQuAD, NCS tiến hành thử nghiệm với các mô hình MRC tiên tiến trên tiếng Anh và tiếng Trung. Sau đó, NCS so sánh hiệu suất giữa các mô hình và người dựa trên các khía cạnh ngôn ngữ khác nhau. Những phân tích chuyên sâu này cung cấp nhiều hiểu biết sâu sắc về MRC tiếng Việt.

Cho đến năm 2020 (thời điểm NCS bắt đầu thực hiện luận án), chưa có bất kỳ bộ ngữ liệu nào về văn bản Wikipedia tiếng Việt cho nghiên cứu MRC có câu trả lời được rút trích trực tiếp từ văn bản (span-based MRC). Như đã đề cập ở trên, một bộ ngữ liệu chuẩn cho bài toán MRC và có thể khuyến khích các NNC khám phá các mô hình MRC tốt nhất trong tiếng Việt. Vì vậy, đây là động lực chính của NCS để tạo ra bộ ngữ liệu mới cho MRC tiếng Việt.



Hình 3.3. Quy trình xây dựng bộ ngữ liệu MRC Wikipedia tiếng Việt.

Quy trình xây dựng ngữ liệu

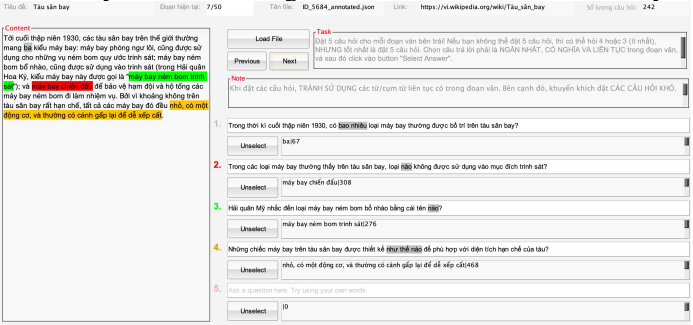
NCS giới thiệu quy trình tạo bộ ngữ liệu MRC tiếng Việt (Hình 3.3). Đặc biệt, NCS xây dựng bộ ngữ liệu UIT-ViQuAD thông qua năm giai đoạn bao gồm tuyển người làm ngữ liệu, thu thập ngữ liệu, tạo các mẫu ngữ liệu {câu hỏi, câu trả lời}, kiểm tra và thu thập thêm câu trả lời. NCS mô tả các giai đoạn chi tiết như sau:

Giai đoạn 1 - Tuyển người làm ngữ liệu: Chất lượng của bộ ngữ liệu phụ thuộc vào những người làm ngữ liệu và quy trình tạo ngữ liệu. NCS trình bày việc tuyển dụng những người làm ngữ liệu để tạo bộ ngữ liệu mới theo một quy trình nghiêm ngặt.

Giai đoạn 2 – Thu thập ngữ liệu: Tương tự như bộ ngữ liệu nổi tiếng SQuAD, NCS cũng sử dụng phương pháp lấy những bài báo uy tín trên Wikipedia của Dự án Nayuki để lấy một bộ ngữ liệu gồm 5.000 bài báo tiếng Việt được xếp hạng đầu tiên, từ đó NCS chọn ngẫu nhiên một số bài báo để tạo ngữ liệu MRC tiếng Việt. Mỗi bài đọc tương ứng với một đoạn trong một bài báo.

Giai đoạn 3 – Tạo các mẫu {câu hỏi, câu trả lời}: Người làm ngữ liệu đọc hiểu trên từng bài đọc và sau đó tạo các mẫu ngữ liệu {câu hỏi, câu trả lời} trên công cụ (xem Hình 3.4). Trong quá trình tạo các mẫu ngữ liệu {câu hỏi, câu trả lời}, người làm ngữ liệu tuân theo các quy tắc: (1) Người làm ngữ liệu được yêu

cầu tạo ít nhất ba cặp câu hỏi – câu trả lời trên mỗi bài đọc. (2) Người làm ngữ liệu được khuyến khích đặt câu hỏi bằng ngôn ngữ của họ. (3) Câu trả lời là các chuỗi văn bản liên tục trong bài đọc được trích xuất để trả lời các câu hỏi. (4) Người làm ngữ liệu được khuyến khích đa dạng hóa các câu hỏi, câu trả lời và suy luận.



Hình 3.4. Công cụ tạo ngữ liệu đọc hiểu trên văn bản tiếng Việt.

Giai đoạn 4 - Kiểm tra câu hỏi và câu trả lời: Trong giai đoạn này, người làm ngữ liệu thực hiện hai bước khác nhau để kiểm tra những lỗi thường xảy ra trong các mẫu ngữ liệu {câu hỏi, câu trả lời} bao gồm quá trình tự kiểm tra và quá trình kiểm tra chéo. Các lỗi được phân thành năm loại khác nhau: câu hỏi không rõ ràng, lỗi chính tả, câu trả lời không chính xác, thiếu hoặc thừa thông tin trong câu trả lời và câu trả lời sai vị trí trong văn bản. Hai giai đoạn hỗ trợ kiểm tra lỗi: tự kiểm tra và kiểm tra chéo.

Giai đoạn 5 - Thu thập thêm câu trả lời: Để nâng cao việc đánh giá trên bộ ngữ liệu, đối với tập phát triển và tập kiểm tra của bộ ngữ liệu, mỗi câu hỏi được thêm ba câu trả lời khác khác nhau. Trong giai đoạn này, những người làm ngữ liệu không thể nhìn thấy câu trả lời của nhau và họ được khuyến khích đưa ra các câu trả lời đa dạng.

Phân tích bộ ngữ liệu

Thống kê tổng quan

Số liệu thống kê của các tập huấn luyện (Train), tập phát triển (Dev) và tập kiểm tra (Test) trong bộ ngữ liệu đề xuất được mô tả chi tiết trong Bảng 3.1. Số lượng câu hỏi của UIT-ViQuAD là 23.074 câu hỏi. Trong Bảng 3.1, số lượng bài báo và bài đọc, độ dài trung bình của câu hỏi và câu trả lời cũng như kích thước từ vựng được trình bày.

Bảng 3.1. Thống kê tổng quan về bộ ngữ liệu đọc hiểu Wikipedia tiếng Việt.

Đặc điểm	Train	Dev	Test	Toàn bộ
Số lượng bài báo	138	18	18	174
Số lượng bài đọc	4.101	515	493	5.109
Số lượng câu hỏi	18.579	2.285	2.210	23.074
Độ dài TB của bài đọc	153,9	147,9	155,0	153,4

Độ dài TB của câu hỏi	12,2	11,9	12,2	12,2
Độ dài TB của câu trả lời	8,1	8,4	8,9	8,2
Kích thước từ vựng	36.174	9.184	9.792	41.773

Phân tích theo độ dài

NCS phân tích dựa trên thống kê của bộ ngữ liệu theo ba loại độ dài bao gồm độ dài câu hỏi (xem Bảng 3.2), độ dài câu trả lời (xem Bảng 3.2) và độ dài bài đọc (xem Bảng 3.3). Các câu hỏi từ 11-15 từ của bộ ngữ liệu chiếm tỷ lệ cao là 45,29%. Các câu trả lời hầu hết có độ dài từ 1 đến 10 từ, chiếm 73,68%. Độ dài của các bài đọc phần lớn từ 101 đến 200 từ với 73,13%.

Bảng 3.2. Thống kê phân bố bộ ngữ liệu theo độ dài của câu hỏi và câu trả lời.

Độ dài	Câu hỏi				Câu trả lời			
	Train	Dev	Test	Toàn bộ	Train	Dev	Test	Toàn bộ
1 – 5	1,03	1,44	0,95	1,06	**54,12**	**50,63**	**52,26**	**53,60**
6 – 10	**35,99**	**38,38**	**33,21**	**35,96**	19,95	22,14	19,10	20,08
11 – 15	**44,97**	**44,29**	**49,05**	**45,29**	10,86	10,81	10,81	10,85
16 – 20	15,01	13,61	14,07	14,78	6,28	7,48	6,83	6,45
>20	3,00	2,28	2,71	2,90	8,80	8,93	11,00	9,02

Bảng 3.3. Thống kê phân bố bộ ngữ liệu theo độ dài bài đọc.

Độ dài	Train	Dev	Test	Toàn bộ
<101	11,41	10,10	11,16	11,25
101 – 150	47,50	53,59	45,44	**47,92**
151 – 200	24,99	23,69	28,60	**25,21**
201 – 250	9,41	8,93	9,94	9,41
251 – 300	4,02	2,52	1,83	3,66
>300	2,66	1,17	3,04	2,54

Phân tích theo khía cạnh ngôn ngữ

NCS phân tích trên tập phát triển (Dev) theo các loại khác nhau như loại câu hỏi, loại câu trả lời và loại suy luận. NCS cũng chia các câu hỏi thành 07 loại khác nhau: Who (Ai), What (Cái gì), When (Khi nào), Where (Ở đâu), Why (Tại sao), How (Như thế nào) và Others (những loại câu hỏi khác). Câu hỏi What chiếm tỷ trọng lớn nhất 49,97%.

NCS phân loại các câu trả lời dựa trên các loại ngôn ngữ của chúng như thời gian (N1), số khác (N2), người (E1), địa điểm (E2), thực thể khác (E3), cụm danh từ (P1), cụm động từ (P2), cụm tính từ (P3), cụm giới từ (P4), mệnh đề (P5) và các cụm từ khác (O). Các cụm danh từ thông dụng chiếm tỷ lệ lớn nhất trên UIT-ViQuAD, con số này tương tự với thống kê của SQuAD và NewsQA. Ngoài ra, các cụm động từ (P2) và các thực thể khác (E3) xếp hạng phần trăm thứ hai và thứ ba trong bộ ngữ liệu UIT-ViQuAD.

Để khám phá mức độ khó của suy luận trong câu hỏi, NCS tiến hành thực hiện quá trình gán nhãn cho các cấp độ suy luận khác nhau của câu hỏi. Năm loại suy luận khác nhau với thứ tự độ khó tăng dần xuất hiện trong bộ ngữ liệu: so khớp từ (WM), diễn giải lại (PP), suy luận trên một câu (SSR), suy luận trên nhiều câu (MSR) và mơ hồ hoặc không đủ thông tin để dự đoán câu trả lời (AoI).

3.3. Bộ ngữ liệu đọc hiểu tự động cho văn bản tin tức sức khỏe tiếng Việt

UIT-ViQuAD là bộ ngữ liệu đầu tiên cho đánh giá các mô hình MRC miền mở trong văn bản Wikipedia tiếng Việt. Để đa dạng ngữ liệu tiếng Việt cho đánh giá các mô hình MRC, NCS tiến hành xây dựng một bộ ngữ liệu tiếng Việt quy mô lớn mới dựa trên các bài báo tin tức sức khỏe để đánh giá các mô hình MRC.

Quy trình xây dựng ngữ liệu

Quy trình xây dựng bộ ngữ liệu UIT-ViNewsQA trải qua sáu bước như sau:

Bước 1 - Tuyển người làm ngữ liệu: Người làm ngữ liệu được tuyển để tạo các mẫu ngữ liệu {câu hỏi, câu trả lời} dựa trên quy trình ba giai đoạn nghiêm ngặt được mô tả như sau: (1) Sinh viên đang học đại học phải có kiến thức tổng quát tốt và thích đọc các bài báo trực tuyến về lĩnh vực sức khỏe đăng ký trở thành người làm ngữ liệu để tạo các mẫu {câu hỏi, câu trả lời} cho nhiệm vụ MRC, (2) Người làm ngữ liệu được chọn phải vượt qua bài kiểm tra đọc hiểu và (3) Người làm ngữ liệu chính thức được tập huấn cẩn thận để làm quen với các hướng dẫn làm ngữ liệu với 200 mẫu {câu hỏi, câu trả lời} trước khi xây dựng bộ ngữ liệu.

Bước 2 - Hướng dẫn xây dựng ngữ liệu: Những người làm ngữ liệu đọc và hiểu từng bài báo và sau đó tạo các câu hỏi và lựa chọn các chuỗi trong văn bản để làm câu trả lời cho các câu hỏi.

Bước 3 - Chuẩn bị ngữ liệu: Các bài báo tin tức liên quan đến lĩnh vực sức khỏe được thu thập từ trang báo uy tín VnExpress. NCS chọn nguồn này vì đây là một trong những trang báo điện tử tiếng Việt được nhiều người đọc nhất và ngôn ngữ sử dụng trong các bài viết dễ hiểu đối với độc giả phổ thông, giúp là hữu ích cho các ứng dụng thực tế. Tất cả hình ảnh, số liệu và bảng đều bị loại bỏ khỏi các bài viết này và các bài viết ngắn hơn 300 ký tự hoặc những bài viết chứa nhiều ký tự đặc biệt hoặc biểu tượng sẽ bị xóa.

Bước 4 - Tạo mẫu ngữ liệu {câu hỏi, câu trả lời}. Theo những hướng dẫn tạo ngữ liệu, người làm ngữ liệu phải tạo các mẫu ngữ liệu {câu hỏi, câu trả lời} cho mỗi bài đọc. Người làm ngữ liệu đọc hiểu văn bản, sau đó nhập câu hỏi và chọn câu trả lời được trích xuất trực tiếp từ bài viết và cho phép người làm ngữ liệu lưu nội dung bài đọc, câu hỏi và câu trả lời trên tập tin *.json.

Bước 5 - Phân tích lỗi và chỉnh sửa: Người làm ngữ liệu khó tránh khỏi sai sót khi tự tạo câu hỏi và chọn câu trả lời từ bài đọc. Để nâng cao chất lượng của bộ ngữ liệu, quy trình kiểm tra được thực hiện để giảm thiểu các lỗi này. Để phân tích các loại lỗi có thể mắc phải trong việc tạo các mẫu ngữ liệu {câu hỏi, câu trả lời}, 1.183 mẫu câu hỏi – câu trả lời (chiếm trên 5% bộ ngữ liệu) được lấy ngẫu nhiên

để kiểm tra lỗi và tìm ra 335 mẫu ngữ liệu câu hỏi – câu trả lời có lỗi. Căn cứ vào câu hỏi hoặc câu trả lời, NCS chia các lỗi này thành 5 loại khác nhau như câu hỏi sai chính tả (Lỗi loại 1), trả lời sai (Lỗi loại 2), trả lời thiếu hoặc thừa thông tin (Lỗi loại 3), câu hỏi không rõ ràng (Lỗi loại 4) và câu trả lời sai vị trí câu trả lời trong văn bản (Lỗi loại 5).

Bước 6 - Thu thập thêm câu trả lời: Để ước tính hiệu suất của người và để tăng cường đánh giá thử nghiệm đa dạng câu trả lời, ba câu trả lời nữa được thêm cho mỗi câu hỏi. Mỗi người làm ngữ liệu xem các câu hỏi với một bài đọc tương ứng, không biết câu trả lời đầu tiên và chọn một chuỗi trong bài đọc để trả lời câu hỏi. Những người làm ngữ liệu tuân thủ các quy tắc và tránh mắc các lỗi được đề cập trước đó.

Phân tích bộ ngữ liệu

Thống kê tổng quan

Bảng 3.4. Tổng quan về bộ ngữ liệu đọc hiểu tin tức sức khỏe tiếng Việt.

Đặc điểm	Train	Dev	Test	Toàn bộ
Số lượng bài báo	3.517	500	399	4.416
Số lượng câu hỏi	17.568	2.497	1.992	22.057
Độ dài TB của bài báo	342,9	323,9	360,4	342,4
Độ dài TB của câu hỏi	10,6	10,8	10,3	10,6
Độ dài TB của câu trả lời	10,7	10,3	10,9	10,7
Kích thước từ vựng	29.111	10.765	10.020	32.749

Phân tích dựa trên từ vựng

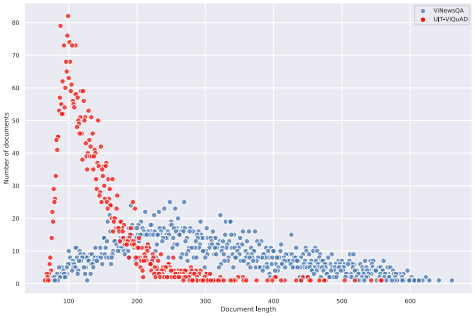
Để hiểu lĩnh vực sức khỏe, tần số xuất hiện các từ cho các bài báo, câu hỏi và câu trả lời trong bộ ngữ liệu được tiến hành thực hiện. Những từ này thuộc lĩnh vực sức khỏe, đây cũng là một đặc điểm nổi bật của bộ ngữ liệu. Phần lớn các từ giống nhau về chủ đề sức khỏe đều xuất hiện trong các bài đọc, câu hỏi và câu trả lời. Cụ thể, sự phân bố của các từ phổ biến khá giống nhau trong các tập huấn luyện, tập phát triển và tập kiểm tra. Mười từ phổ biến trên UIT-ViNewsQA và UIT-ViQuAD rất khác nhau vì những từ có tần suất xuất hiện cao này trên bộ ngữ liệu miền mở UIT-ViQuAD thuộc về nhiều lĩnh vực khác nhau như lịch sử, địa lý, kinh tế và chính trị.

Phân tích dựa trên độ dài

Câu hỏi có 8–9 từ chiếm tỷ lệ cao nhất với 25,67% trên bộ ngữ liệu UIT-ViNewsQA. Hầu hết các câu hỏi trong bộ ngữ liệu đều có độ dài từ 6 đến 13 từ, chiếm khoảng 80% bộ ngữ liệu. Câu hỏi rất ngắn (4-5 từ) và câu hỏi dài (>=18 từ) chiếm tỷ lệ thấp lần lượt là 2,68% và 3,91%.

Tỷ lệ phần trăm lớn nhất (14,73%) bao gồm các câu trả lời có độ dài từ 3–4 từ. Hầu hết các câu trả lời (gần 60%) có độ dài từ 1–10 từ. Các câu trả lời dài hơn (hơn 10 từ) chiếm tỷ lệ thấp trong bộ ngữ liệu UIT-ViNewsQA. Do cùng loại đọc hiểu tự động, hai bộ ngữ liệu: UIT-ViNewsQA và UIT-ViQuAD có phân bố độ dài câu trả lời khá tương tự nhau.

Độ dài trung bình của các bài đọc trên UIT-ViNewsQA dài hơn đáng kể so với trên UIT-ViQuAD. Mỗi bài đọc của UIT-ViQuAD là một đoạn văn, trong khi mỗi bài đọc của UIT-ViNewsQA là một bài báo tin tức về sức khỏe. Hình 3.5 cho thấy sự phân bố độ dài bài đọc khác nhau của hai bộ ngữ liệu tiếng Việt.



Hình 3.5. Phân bố độ dài bài đọc trong văn bản Wikipedia và tin tức sức khỏe tiếng Việt.

Phân tích dựa trên loại câu hỏi – câu trả lời

Các câu hỏi tiếng Việt được chia thành bảy loại câu hỏi khác nhau: Who (Ai), What (Cái gì), When (Khi nào), Where (Ở đâu), Why (Tại sao), How (Như thế nào) và các loại câu hỏi khác, tương tự như trên UIT-ViQuAD và CMRC. Cụ thể, câu hỏi What chiếm tỷ trọng lớn nhất với 54,35%. So với bộ ngữ liệu SQuAD và UIT-ViQuAD, tỷ lệ câu hỏi What trong bộ ngữ liệu UIT-ViNewsQA gần như tương đương. Bộ ngữ liệu UIT-ViNewsQA yêu cầu các câu hỏi khó và phức tạp để trả lời như câu hỏi How (Như thế nào) và Why (Tại sao). Các loại câu hỏi How và Why đứng thứ hai và thứ ba với tỷ lệ tương ứng là 13,46% và 12,17%.

Câu trả lời được chia thành 11 loại khác nhau bao gồm số (thời gian, số khác), thực thể (người, địa điểm, thực thể khác), cụm từ (cụm danh từ, cụm tính từ, cụm động từ, cụm giới từ, mệnh đề) và các loại khác. Thứ tự ưu tiên của tạo ngữ liệu là số, thực thể, cụm từ và các loại khác. Trong khi cụm động từ chiếm tỷ lệ cao nhất với 34,84% thì cụm giới từ lại chiếm tỷ lệ thấp nhất với 0,8%.

3.4. Bộ ngữ liệu đọc hiểu tự động cấp độ câu cho văn bản tiếng Việt

Tiếp tục đóng góp các bộ ngữ liệu đọc hiểu cho văn bản tiếng Việt: UIT-ViQuAD và UIT-ViNewsQA, NCS tiếp tục tạo UIT-ViWikiQA, một bộ ngữ liệu mới cho MRC dựa trên trích xuất câu cho đánh giá các mô hình đọc hiểu và hỏi đáp tự động. Câu trả lời cấp độ câu sẽ giúp người đọc hiểu và biết được nhiều thông tin liên quan đến câu hỏi hơn so với câu trả lời chỉ là từ hoặc cụm từ. Với mong muốn giảm độ phức tạp để xác định câu trả lời trong văn bản và đa dạng loại MRC trong tiếng Việt, một bộ ngữ liệu mới được đề xuất để đánh giá các mô hình MRC.

Quy trình xây dựng ngữ liệu

Tận dụng bộ ngữ liệu có sẵn UIT-ViQuAD, một bộ ngữ liệu mới được tạo ra dựa trên phương pháp tiếp cận tự động dựa trên những mẫu ngữ liệu có sẵn. Thuật toán 3.1 mô tả quá trình chuyển đổi tự động tự câu trả lời cấp độ chuỗi (span-based MRC) sang cấp độ câu (Sentence-based MRC), tạo ra bộ ngữ liệu mới với tên là: UIT-ViWikiQA [CT6].

Thuật toán 3.1. *Quá trình chuyển đổi tự động tự câu trả lời cấp độ chuỗi sang cấp độ câu.*

Đầu vào:

- Một văn bản D.
- Câu hỏi Q và câu trả lời A.
- Vị trí bắt đầu Istart[i] của câu trả lời A.

Đầu ra: Trả lại câu chứa A (ST[i]) và vị trí bắt đầu Ostart[i]

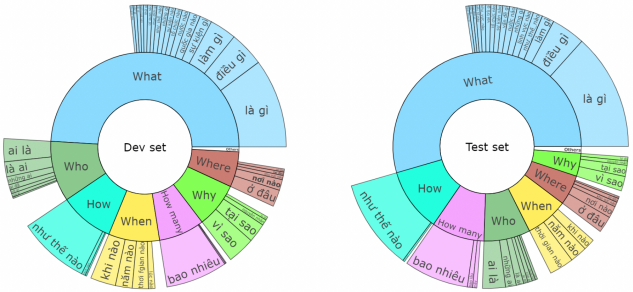
Function Chuyển một mẫu ngữ liệu đọc hiểu cấp độ span sang cấp độ câu($D, Q, A,$ Istart[i])
    Khởi tạo: Sentences ← Tách văn bản D thành danh sách các câu.
    Khởi tạo: Start ← 0
    Khởi tạo: End ← -1
    For i ∈ range(0, len($Sentences$) − 1)
        Cập nhật: Start ← End + 1
        Cập nhật: Start ← Start + len(Sentences[i])
        if Start ≤ Istart[i] ≤ End then
            break
        end if
    End for
    Cập nhật: ST[i] ← S[i]
    Cập nhật: Ostart[i] ← D.find(ST[i])
    Return ST[i], Ostart[i]
End function

Tận dụng bộ ngữ liệu UIT-ViQuAD [CT5] đã nghiên cứu trước đó, một mẫu bao gồm ngữ cảnh, câu hỏi, câu trả lời, bắt đầu câu trả lời và mã định danh id. Thuật toán 3.1 cập nhật câu trả lời (chuyển từ cấp độ chuỗi sang cấp độ câu) và bắt đầu câu trả lời sử dụng bài đọc và bắt đầu câu trả lời (AS) của các mẫu ngữ liệu gốc của UIT-ViQuAD. Tiếp theo, bài đọc (D) được tách và chuyển thành thành danh sách các câu S = (S_1, S_2, S_3, …, S_n) thì S_i ∈ S là câu trả lời của câu hỏi trong bộ ngữ liệu sao cho Start(S_i) ≤ AS < End(S_i), trong đó Start(S_i) và End(S_i) lần lượt là vị trí bắt đầu và kết thúc của S_i trong bài đọc.

Phân tích bộ ngữ liệu

Do bộ ngữ liệu UIT-ViWikiQA được chuyển đổi từ bộ ngữ liệu UIT-ViQuAD, bộ ngữ liệu UIT-ViWikiQA thừa hưởng các đặc điểm của bộ ngữ liệu UIT-

ViQuAD. Để hiểu rõ đặc điểm của các loại câu hỏi tiếng Việt hơn, NCS tiến hành phân tích sâu hơn trên các loại câu hỏi. Trong tiếng Việt, các từ để hỏi cho mỗi loại câu hỏi rất đa dạng, như trong Hình 3.6. Trong câu hỏi What, các từ nghi vấn như "là gì", "cái gì", "điều gì", "làm gì" đều có nghĩa là "cái gì", trong đó "là gì" chiếm tỷ lệ cao nhất với 20,69% ở câu hỏi What trong tập phát triển và 24,42% trong tập kiểm tra. Tương tự như câu hỏi What, hiện tượng đa dạng từ hỏi cũng xuất hiện ở các kiểu câu hỏi: Who, When, Why và Where. Cụ thể, đối với kiểu câu hỏi When của tiếng Việt, các từ để hỏi như "khi nào", "năm nào" hay "thời nào" có nghĩa tương tự như từ để hỏi "khi nào". Tuy nhiên, ở dạng câu hỏi Như thế nào (How) và Bao nhiêu (How many) ít đa dạng từ hỏi hơn. Ví dụ, từ hỏi "như thế nào" chiếm 87,79% câu hỏi How trong tập phát triển và 95,02% trong tập kiểm tra. Trong bộ ngữ liệu đọc hiểu Wikipedia tiếng Việt, các mô hình có thể gặp phải thách thức không chỉ ở các câu hỏi mang tính suy luận cao như Why (Tại sao) và How (Như thế nào) mà còn ở sự đa dạng của các từ hỏi trong mỗi loại câu hỏi. Việc đa dạng từ hỏi cũng đặt ra nhiều thách thức cho các mô hình đọc hiểu và hỏi đáp tự động tiếng Việt, đã được NCS đề xuất giải pháp xử lý trong Chương 5.



Hình 3.6. Phân bố các loại câu hỏi và từ hỏi trên tập phát triển và tập kiểm tra của bộ ngữ liệu đọc hiểu Wikipedia tiếng Việt.

3.5. Mở rộng bộ ngữ liệu đọc hiểu tự động tiếng Việt với câu hỏi không trả lời được

Các bộ ngữ liệu hiện có của tiếng Việt cho MRC tiếng Việt [**CT1**], **[CT5]**, [**CT6**] chỉ tập trung vào các câu hỏi có thể trả lời được. Tuy nhiên, thực tế có những câu hỏi không có câu trả lời chính xác được nêu trong văn bản đã cho trước. Để giải quyết điểm yếu của những bộ ngữ liệu về MRC tiếng Việt, NCS đã đề xuất một bộ ngữ liệu chuẩn có tên UIT-ViQuAD 2.0 để đánh giá các mô hình MRC và mô hình QA tiếng Việt, tập trung vào hai dạng câu hỏi trả lời được và câu hỏi không trả lời được. NCS đã đề xuất UIT-ViQuAD 2.0 như bộ ngữ liệu chuẩn cho cộng đồng nghiên cứu đánh giá bài toán MRC tiếng Việt tại VLSP 2021. Bài toán nghiên cứu này thu hút 77 nhóm tham gia từ 34 trường đại học và các doanh nghiệp nghiên cứu AI. Trong nội dung này, NCS trình bày về cuộc thi, giới thiệu về các

phương pháp đã được những nhóm tham gia thử thách đạt được những kết quả thử nghiệm tích cực. Hiệu suất cao nhất là 77,24% cho F_1 và 67,43% cho EM trên tập kiểm tra. Các mô hình MRC tiếng Việt do ba nhóm xếp hạng tốp đầu đề xuất phương pháp đều dựa trên XLM-R, một MHNN tiên tiến được huấn luyện sẵn dựa trên kiến trúc Transformer. Bộ ngữ liệu UIT-ViQuAD 2.0 thúc đẩy các NNC khám phá sâu hơn về MRC tiếng Việt và các bài toán nghiên cứu liên quan như trả lời câu hỏi, tạo sinh câu hỏi và suy luận ngôn ngữ tự nhiên. Toàn bộ nghiên cứu này được NCS trình bày trong công trình khoa học và công bố tại một tạp chí uy tín: [CT4].

3.6. Những đánh giá đầu tiên trên các mô hình đọc hiểu tự động tiếng Việt

Để định hướng các nghiên cứu mô hình MRC trong tương lai, NCS tiến hành thử nghiệm và đánh giá các mô hình cơ sở đầu tiên để trả lời các câu hỏi nghiên cứu sau: (1) Liệu rằng các mô hình đọc hiểu tốt trên các ngôn ngữ giàu tài nguyên: tiếng Anh và tiếng Trung có hoạt động tốt cho MRC tiếng Việt không? (2) Liệu rằng các yếu tố như các dạng MRC trích xuất và miền dữ liệu có tác động đến khả năng đọc hiểu của các mô hình cơ sở không? (3) Liệu rằng độ dài các văn bản có tác động đến khả năng đọc hiểu của các mô hình cơ sở không?

Đọc hiểu tự động trong văn bản Wikipedia tiếng Việt

NCS tiến hành thử nghiệm với các mô hình MRC tiên tiến như các mô hình dựa trên mạng nơ-ron truyền thống: DrQA, QANet và các MHNN dựa trên kiến trúc Transformer: mBERT, XLM-R_{Base} và XLM-R_{Large} để đánh giá khả năng MRC của máy tính. Để đo độ khó của bộ ngữ liệu đề xuất, NCS cũng ước tính hiệu suất của người trong bài toán MRC tiếng Việt. Tương tự như đánh giá trên bộ ngữ liệu tiếng Anh và tiếng Trung, NCS sử dụng hai thông số đánh giá là độ chính xác (EM) và độ đo F_1 để đánh giá hiệu suất của các mô hình MRC trên bộ ngữ liệu MRC trên văn bản Wikipedia tiếng Việt.

Bảng 3.5. Hiệu suất mô hình đọc hiểu trên Wikipedia tiếng Việt.

Mô hình	EM		F_1	
	Dev	Test	Dev	Test
DrQA	43,98	40,91	65,09	63,44
QANet	39,66	46,05	63,82	68,06
mBERT	62,20	59,28	80,77	80,00
XLM-R_{Base}	63,87	63,00	81,90	81,95
XLM-R_{Large}	**69,18**	**68,98**	**87,14**	**87,02**
Người	85,65	85,59	95,19	94,69

Bảng 3.5 trình bày hiệu suất của các mô hình đọc hiểu tiếng Việt cùng với hiệu suất của người trên tập phát triển và tập kiểm tra của bộ ngữ liệu UIT-ViQuAD. Trên hai độ đo: EM và F_1, XLM-R_{Large} vượt trội hơn đáng kể so với các mô hình đọc hiểu khác nhưng vẫn thấp hơn hiệu suất của người. Trong bài kiểm tra, mô hình dự đoán câu trả lời với F_1 là 87,02%. Tuy nhiên, độ chính xác (EM) của mô

hình XLM-R_{Large} đạt 68,98%, thấp hơn đáng kể so với F_1, chứng tỏ rằng việc xác định chính xác vị trí câu trả lời là rất khó (vì độ chính xác EM của một câu hỏi được tính là 0 nếu câu trả lời dự đoán có một ký tự không khớp).

Để kiểm tra xem các mô hình MRC hoạt động tốt như thế nào trên văn bản Wikipedia tiếng Việt, NCS phân tích hiệu suất của mô hình học máy theo độ đo F_1. Bảng 3.6 trình bày các phân tích theo độ dài trên kết quả thử nghiệm của các mô hình đọc hiểu trên tập phát triển. Nhìn chung, hiệu suất của các MHNN dựa trên kiến trúc Transformer: mBERT và XLM-R tốt hơn so với các mô hình dựa trên mạng nơ-ron truyền thống: QANet và DrQA Tuy nhiên, hiệu suất của tất cả các thuật toán học máy đều thấp hơn người ở các loại độ dài khác nhau. Qua phân tích trên độ dài bài đọc khác nhau cho thấy sự dao động hiệu suất của hầu hết các mô hình MRC, hoạt động tốt đối với các bài đọc ngắn (<100 từ). Dựa trên hiệu suất trung bình của các mô hình học máy, NCS nhận thấy các bài đọc có độ dài từ >101 đạt kết quả thấp hơn so với những bài đoạn nhỏ hơn 100 từ. Như vậy, các bài đọc dài hơn có khả năng chứa những thông gây nhiễu làm giảm hiệu suất của các mô hình học máy, nhưng đối với người thì hiệu suất ổn định trên mọi độ dài khác nhau của bài đọc.

Bảng 3.6. Hiệu suất của các mô hình theo độ dài văn bản trên bộ ngữ liệu đọc hiểu Wikipedia tiếng Việt.

Độ dài	Tỷ lệ	DrQA	QANet	BERT	XLM-R_{Base}	XLM-R_{Large}	TB	Người
<101	11,25	70,92	73,86	83,80	84,19	88,33	80,22	95,99
101-150	47,92	65,70	67,68	80,50	82,24	87,09	**76,64**	95,25
151-200	25,21	61,81	66,16	79,70	80,03	86,03	**74,75**	95,53
201-250	9,41	62,81	68,07	81,07	79,47	86,96	**75,68**	95,65
>250	6,20	67,85	69,76	80,94	88,12	89,62	**79,26**	94,79

Như vậy, với thử nghiệm các mô hình trên bộ ngữ liệu đọc hiểu Wikipedia tiếng Việt, NCS thấy rằng: MHNN đạt kết quả tốt hơn so với mô hình dựa trên mạng nơ-ron truyền thống, và mô hình XLM-R đạt kết quả tốt nhất. Bên cạnh đó, các văn bản tiếng Việt dài hơn làm giảm hiệu suất của các mô hình đọc hiểu tự động.

Đọc hiểu tự động trên văn bản tin tức sức khỏe tiếng Việt

Bảng 3.7 so sánh hiệu suất của các thuật toán học máy và người trên bộ ngữ liệu UIT-ViNewsQA với các mô hình mạng nơ-ron truyền thống: DrQA, QANet và các MHNN dựa trên kiến trúc Transformer: mBERT và ALBERT. Các mô hình dựa trên học chuyển tiếp sử dụng hai MHNN là BERT và ALBERT vượt trội đáng kể so với các mô hình dựa trên mạng nơ-ron truyền thống (DrQA và QANet), nhưng không thể vượt qua hiệu suất của người. Mô hình tốt nhất ALBERT đạt EM là 65,26% và F_1 là 84,89%. Đặc biệt, hiệu suất (EM và F_1) của ALBERT tốt hơn DrQA, với 19,43% EM và 10,80% F_1. ALBERT cũng hoạt động tốt hơn mô hình

QANet với mức chênh lệch là 8,55% (theo EM) và 5,10% (theo F_1). Trong khi đó, hiệu suất khác nhau giữa người và mô hình tốt nhất ALBERT (tương ứng là 14,53% theo EM và 10,90% theo F_1) là rất đáng kể, qua đó cho thấy các phương pháp đọc hiểu dựa MHNN là triển vọng và cần được xem xét trong nghiên cứu thêm trong tương lai. Trên cùng các phương pháp DrQA, QANet và mBERT, các mô hình MRC trên miền văn tin tức sức khỏe tiếng Việt (xem Bảng Bảng 3.7) đạt kết quả tốt hơn trên văn bản miền mở Wikipedia tiếng Việt (xem Bảng 3.5).

Bảng 3.7. Hiệu suất các mô hình MRC trên tin tức sức khỏe tiếng Việt.

Mô hình	EM		F_1	
	Dev	Test	Dev	Test
DrQA	49.26	45.83	74.03	74.09
QANet	57.80	56.71	78.39	79.79
mBERT	64.56	63.81	81.47	83.19
ALBERT	64.24	**65.26**	83.52	**84.89**
Người	75.19	79.79	92.77	95.79

NCS phân tích hiệu suất của các mô hình đọc hiểu trên các độ dài bài đọc khác nhau trên bộ ngữ liệu UIT-ViNewsQA. Các kết quả chi tiết được trình bày trong Bảng 3.8. Nhìn chung, các bài báo ngắn thu được kết quả chính xác hơn so với các bài báo dài hơn. Tất cả các mô hình có xu hướng hoạt động tốt hơn với các bài báo ngắn (<201 từ) vì các bài báo dài hơn có mức độ gây nhiễu cao hơn và làm giảm hiệu suất của các mô hình MRC. Do đó, dự đoán câu trả lời đúng trên các bài báo dài hơn là một thách thức đối với mô hình MRC, điều này có thể truyền cảm hứng cho một ý tưởng loại bỏ thông tin gây nhiễu bằng cách truy xuất minh chứng liên quan trước khi trích xuất câu trả lời đúng trong văn bản.

Bảng 3.8. Hiệu suất mô hình theo độ dài văn bản trên bộ ngữ liệu đọc hiểu tin tức sức khỏe tiếng Việt.

Độ dài	Tỷ lệ	DrQA	QANet	BERT	ALBERT	TB	Người
<101	0,34	83,49	89,85	98,00	82,44	88,45	97,06
101-200	15,51	78,07	81,45	85,79	87,21	83,13	94,93
201-300	29,12	74,25	79,22	82,63	84,57	80,17	93,20
301-400	23,39	71,83	76,58	79,26	82,37	**77,51**	92,87
401-500	16,15	72,17	74,27	78,55	79,36	**76,09**	90,49
501-600	10,17	74,01	80,53	81,33	82,94	**79,70**	92,37
>600	5,32	72,49	76,76	75,41	81,12	**76,45**	91,94

Như vậy, các mô hình MRC trên miền đóng – văn tin tức sức khỏe tiếng Việt đạt kết quả tốt hơn trên văn bản miền mở Wikipedia tiếng Việt. Với thử nghiệm các mô hình trên bộ ngữ liệu đọc hiểu Wikipedia tiếng Việt, NCS tiếp tục thấy rằng: MHNN đạt kết quả tốt hơn so với mô hình dựa trên mạng nơ-ron truyền

thống. Bên cạnh đó, các văn bản tiếng Việt dài hơn vẫn tiếp tục làm giảm hiệu suất của các mô hình đọc hiệu tự động.

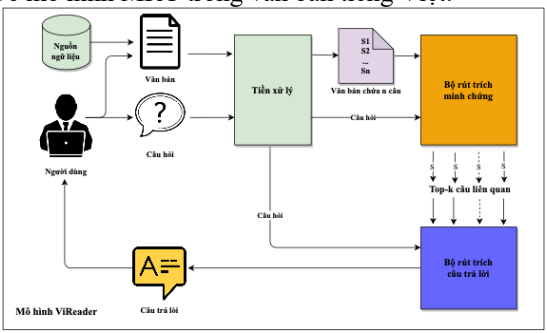
Đọc hiểu tự động cấp độ câu trên văn bản Wikipedia tiếng Việt

MRC cấp độ câu tiếng Việt có ba hướng tiếp cận chính: xếp hạng (Word Count và BM25), phân lớp (maLSTM và BiGRU) và đặc biệt, mô hình MRC (QANet, mBERT, PhoBERT và XLM-R). Các mô hình theo hướng tiếp cận dựa trên phân loại đạt hiệu suất thấp nhất và hiệu suất cao nhất thuộc về các mô hình theo hướng tiếp cận dựa trên MRC. Mô hình tốt nhất (XLM-R_{Large}) đạt 88,77% (theo F_1) và 85,87% (theo EM) trên tập kiểm tra. Hai nhóm phương pháp dựa trên phân loại và xếp hạng đạt kết quả thấp hơn so với nhóm phương pháp dựa trên MRC. So cùng phương pháp QANet và XLM-R, các mô hình MRC trích xuất trên cấp độ câu đạt kết quả tốt hơn trên cấp độ chuỗi.

CHƯƠNG 4: MÔ HÌNH ĐỌC HIỂU TỰ ĐỘNG TÍCH HỢP RÚT TRÍCH MINH CHỨNG TRÊN VĂN BẢN TIẾNG VIỆT

4.1. Mô hình đọc hiểu tự động trong văn bản tiếng Việt

NCS đề xuất một mô hình MRC có tên là ViReader (viết tắt của Vietnamese Reader) bao gồm hai thành phần chính: mô hình truy xuất câu minh chứng (STR) và mô hình rút trích câu trả lời (Answer Extractor). Cho một câu hỏi dưới dạng ngôn ngữ tự nhiên và một văn bản, mô hình truy xuất câu minh chứng sẽ truy xuất các câu ứng viên có khả năng chứa câu trả lời dựa trên xếp hạng các câu trong văn bản bằng cách ước tính mức độ liên quan ngữ nghĩa giữa câu hỏi và mỗi câu trong văn bản. Sau đó, mô hình trích xuất câu trả lời sẽ đọc và hiểu một văn bản có chứa k câu được xếp hạng cao nhất để trích xuất các câu trả lời có thể. Hình 4.1 trình bày tổng quan về mô hình MRC trong văn bản tiếng Việt.



Hình 4.1. Tổng quan về kiến trúc của mô hình đọc hiểu ViReader.

Tiền xử lý

Mô-đun tiền xử lý nhận vào các văn bản ngôn ngữ tự nhiên chưa qua xử lý (bao gồm cả câu hỏi và văn bản), đưa văn bản về chữ viết thường, xóa các ký tự đặc biệt (ví dụ: dòng mới và khoảng trắng thừa) và tách tiếng cho các câu hỏi/văn bản. Sau đó, mô hình truy xuất câu minh chứng và mô hình trích xuất câu trả lời được sử dụng các câu hỏi và văn bản đã được tiền xử lý trước.

Thuật toán 4.1. *Mã giả cho mô hình truy xuất câu minh chứng để trích xuất k câu có liên quan nhất dựa trên câu hỏi Q và văn bản D.*

Đầu vào:

- Một văn bản D.
- Một câu hỏi Q.
- Một hằng số K: số câu liên quan đến câu hỏi được rút trích từ văn bản D

Đầu ra: Trả về danh sách L gồm top *k* câu.

1: **Function** *Trích xuất top k câu liên quan nhất đến* $Q(D, Q, K)$

2: **Khởi tạo**: Scores = [] *# tập lưu trữ các điểm số của các câu liên quan đến câu hỏi.*

3: **Khởi tạo**: SL = Document_Segmentation(D) *# tách văn bản D thành danh sách các câu.*

Khởi tạo: L = [] # *Danh sách L lưu trữ top k câu liên quan.*

4: **Khởi tạo**: $q = Sentence\_Transformer(Q)$

5: **For** i ∈ range$(0, \text{len}(SL) - 1)$

6: **Khởi tạo**: s $= Sentence\_Transformer(SL[i])$

7: **Cập nhật**: $Scores[i] =$ **semntic_similarity**(q, s)

8: **i = i + 1**

10: **End for**

11: **Cập nhật:** Scores = Sorting (Scores) # *sắp xếp danh sách Score giảm dần*

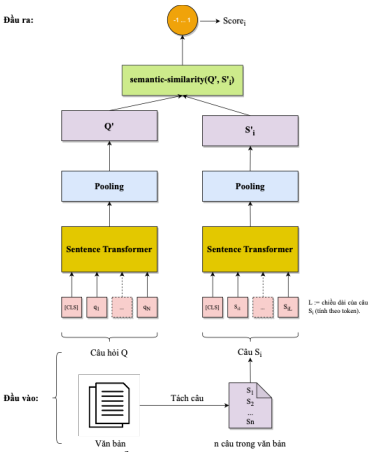
12: **Cập nhật**: L = extract(K, SL, Scores) # *trích xuất top k câu liên quan (điểm số cao nhất của list Scores) từ danh sách SL.*

13: **Return** L

14: **End function**

Mô hình truy xuất câu minh chứng

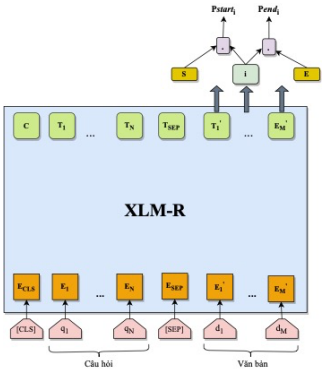
Thuật toán 4.1 mô tả quy trình của mô hình truy xuất câu minh chứng trả về *k* câu phù hợp nhất từ văn bản để đưa vào mô hình trích xuất câu trả lời trong mô hình đọc hiểu ViReader. NCS áp dụng các thuật toán truy xuất thông tin cho mô hình truy xuất câu minh chứng. Nó cũng ước tính điểm dựa trên sự tương đồng về ngữ nghĩa để cho biết mức độ liên quan của câu S_i trong văn bản D với câu hỏi Q. Lấy cảm hứng từ Sentence-BERT và các biến thể của Sentence-BERT, NCS đã sử dụng Sentence Transformer cho mô hình truy xuất câu minh chứng vì NCS tập trung vào truy xuất cấp độ câu. NCS đặt tên phương pháp này là STR (**S**entence **T**ransformer **R**etriever). Hình 4.2 cho thấy quá trình ước tính điểm dựa trên sự tương đồng về ngữ nghĩa giữa câu hỏi Q và câu S_i trong văn bản D. Mô hình truy xuất dựa trên Sentence Transformer để trích xuất một biểu diễn của câu có độ dài cố định với kiến trúc mạng nơ-ron Siamese để tạo ra các biểu diễn câu có ý nghĩa về mặt ngữ nghĩa có thể được sử dụng trong các mô hình học không giám sát (Unsupervised Learning) dựa trên sự tương đồng về ngữ nghĩa.



Hình 4.2. Quá trình ước lượng điểm dựa trên sự tương đồng về ngữ nghĩa giữa câu hỏi Q và câu Si trong văn bản D.

Mô hình trích xuất câu trả lời

Sau khi lấy được văn bản chứa *k* câu có liên quan nhất từ mô hình truy xuất câu minh chứng, mô hình trích xuất câu trả lời xác định một chuỗi các từ (span) như một câu trả lời cho câu hỏi cho trước. Mô hình trích xuất câu trả lời đọc câu hỏi Q bao gồm N token $\{q_1, q_2, ..., q_N\}$ và văn bản đã truy xuất D gồm có M token $\{d_1, d_2, ..., d_M\}$ từ tập hợp của các câu đã truy xuất.



Hình 4.3. Thành phần rút trích câu trả lời của mô hình đọc hiểu ViReader là được xây dựng dựa trên mô hình đọc hiểu văn bản tự động XLM-R

Mô hình trích xuất câu trả lời được mô hình hóa thành hai xác suất liên quan đến câu trả lời có thể được tính cho mỗi token p_i trong văn bản truy xuất câu D, đó là

$Pstart_i$, trong đó d_i là vị trí bắt đầu câu trả lời trích xuất trực tiếp từ văn bản và $Pend_i$, trong đó d_i là vị trí kết thúc của câu trả lời.

Lấy cảm hứng từ sự thành công của nhiều CTNC với kiến trúc Transformer như BERT và XLM-R, NCS triển khai phương pháp học chuyển tiếp bằng MHNN được huấn luyện sẵn XLM-R_{Large} đã được huấn luyện về các ngôn ngữ khác nhau (bao gồm cả tiếng Việt) để nâng cao mô hình trích xuất câu trả lời trên các bộ ngữ liệu MRC đã công bố. Các mô hình MRC dựa trên các biểu diễn token được ngữ cảnh hóa (Contextualized) và kiến trúc Transformer đạt được hiệu suất tốt hơn các mô hình khác. Hình 4.3 mô tả mô hình trích xuất câu trả lời của mô hình đọc hiểu ViReader. XLM-R biểu diễn cho hai chuỗi đầu vào (câu hỏi và văn bản) là một chuỗi các token được phân tách bằng mã [SEP]. MHNN được huấn luyện sẵn tạo một biểu diễn T_i' cho mỗi token i' của văn bản T. Đối với MRC có câu trả lời được rút trích trực tiếp từ văn bản, NCS biểu thị câu hỏi là chuỗi đầu tiên và văn bản là chuỗi thứ hai. NCS cũng cần thêm một số cấu trúc vào đầu ra được huấn luyện trong giai đoạn tinh chỉnh. Mô hình này thêm hai biểu diễn theo ngữ cảnh: một biểu diễn vị trí bắt đầu câu trả lời S và một biểu diễn vị trí kết thúc câu trả lời E. Để có xác suất bắt đầu câu trả lời cho mỗi token đầu ra T_i', NCS tính tích vô hướng của S và T_i', sau đó chuẩn hóa trên tất cả các token T_i' trong văn bản:

$$Pstart_i = \frac{e^{S.T_i'}}{\sum_j e^{S.T_j'}} \qquad (4.1)$$

Tương tự như ước tính xác suất bắt đầu câu trả lời, xác suất vị trí kết thúc câu trả lời cũng được tính theo:

$$Pend_i = \frac{e^{E.T_i'}}{\sum_j e^{E.T_j'}} \qquad (4.2)$$

Do đó, điểm của một câu trả lời ứng viên là một chuỗi liên tục từ vị trí i đến vị trí j được Công thức (4.3).

$$Scorei,j = S.Ti' + E.Tj' \qquad (4.3)$$

Cuối cùng, khoảng câu trả lời có điểm cao nhất, với được chọn, được dự đoán bởi mô hình trích xuất câu trả lời. Hàm mục tiêu huấn luyện là tổng log-likelihoods

của các chỉ số bắt đầu và kết thúc của câu trả lời mong đợi cho mỗi mẫu ngữ liệu được huấn luyện. Trong chương này, NCS hướng tới việc tinh chỉnh mô hình đọc hiểu ViReader trên ngữ liệu MRC tiếng Việt.

4.2. Kết quả

Kết quả thử nghiệm của mô hình truy xuất các câu minh chứng liên quan đến câu hỏi

Đầu tiên, NCS xác định không gian tìm kiếm câu trả lời là tập hợp tất cả các câu trong mỗi văn bản (bài đọc) trong bộ ngữ liệu UIT-ViQuAD dựa trên Wikipedia tiếng Việt để đánh giá mô hình truy xuất câu minh chứng. Câu có chứa câu trả lời sau đó được đánh dấu là câu thích hợp cho mỗi câu hỏi trong mẫu ngữ liệu {câu hỏi, câu trả lời}. Độ chính xác Accuracy@K (Acc@K) được sử dụng để đo Hiệu suất của các mô hình truy xuất câu minh chứng và được định nghĩa là tỷ lệ phần trăm câu hỏi có câu trả lời xuất hiện ở một trong *k* câu được truy xuất. Với Q = {q_1, q_2, ...,q_n} làm tập hợp câu hỏi và S dưới dạng tập hợp các câu trong văn bản, a_q là câu trả lời của câu hỏi q trong Q, s_q trong S là câu chứa a_q và $S^*_K(q) \subseteq S$ là tập hợp các câu có liên quan nhất của K được dự đoán bởi mô hình truy xuất câu minh chứng. Công thức (4.5) cho biết cách tính Acc@K:

$$\mathbf{Acc@K} = \frac{1}{|Q|} \sum_{1}^{n} \begin{cases} 1 & \text{if } s_q \in S^*_K(q), \\ 0 & \text{Otherwise} \end{cases} \qquad (4.5)$$

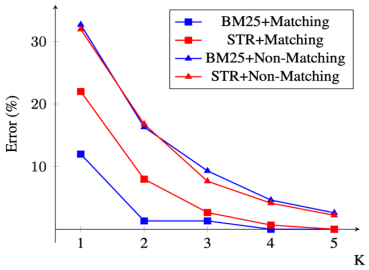
Để chứng minh mô hình truy xuất câu đề xuất hiệu quả, NCS đã so sánh các mô hình truy xuất câu minh chứng khác. Mô hình truy xuất câu minh chứng dựa trên Transformer được so với ba mô hình truy xuất cơ sở (baselines): TF-IDF, TextRank và BM25. Phương thức TF-IDF đóng vai trò là mô hình cơ sở của mô hình truy xuất câu minh chứng dựa trên token cung cấp so sánh trực tiếp với phương pháp dựa trên Transformer. Mặt khác, TextRank và BM25 đại diện cho các truy xuất câu hiệu quả mà không sử dụng các MHNN.

Bảng 4.1. Hiệu suất Acc@K (%) của các mô hình truy xuất câu minh chứng.

Acc@K	1	2	3	4	5	6	7	8	9	10	11	12
TF-IDF	34,84	52,35	69,59	82,76	90,45	94,48	96,56	98,01	98,46	98,87	99,28	99,5
TextRank	57,82	76,88	87,74	93,44	96,83	98,46	99,00	99,32	99,41	99,59	99,73	99,77
BM25	**68,6**	**84,57**	91,09	95,61	97,47	98,78	99,19	99,32	99,5	99,77	99,82	99,91
STR	**68,6**	83,76	**92,13**	**95,97**	**97,87**	**98,96**	**99,37**	**99,55**	**99,64**	**99,82**	**99,91**	**100**

Bảng 4.1 trình bày độ chính xác Acc@K của các mô hình truy xuất câu minh chứng. Các độ chính xác cuối cùng được đo trên tập kiểm tra của bộ ngữ liệu UIT-ViQuAD. Các thử nghiệm chứng minh khả năng xếp hạng câu dựa trên Transformer hoạt động tốt hơn các thuật toán xếp hạng TF-IDF, TextRank và BM25 gần như ở tất cả giá trị K. Khả năng bắt ngữ cảnh và ngữ nghĩa của các MHNN dựa trên kiến trúc Transformer tốt hơn so với các thuật toán dựa trên túi từ (bag of words) như BM25. Hình 4.4 cho thấy lỗi của mô hình truy xuất câu minh chứng hiệu quả (BM25 và STR) đối với các câu dạng so khớp (matching) và không

khớp (non-matching). Với K<3, độ lỗi của BM25 thấp hơn đáng kể so với STR về câu hỏi so khớp (matching), chứng tỏa BM25 xử lý tốt đối với những câu hỏi dạng so khớp (matching) và STR bắt ngữ cảnh tốt hơn đối với những dạng câu hỏi không so khớp (non-matching).



Hình 4.4. Lỗi (Error) của mô hình truy xuất câu minh chứng đối với các câu hỏi so khớp và không so khớp.

Hiệu suất các mô hình đọc hiểu tự động

Để đánh giá khả năng của mô hình đọc hiểu đề xuất, mô hình trích xuất câu trả lời được cung cấp một câu hỏi và một văn bản để dự đoán câu trả lời cho câu hỏi này. Câu trả lời dự đoán (chuỗi các từ liên tục) sau đó được so với câu trả lời mong đợi. Trong tất cả các thử nghiệm, NCS sử dụng tuân theo các tập ngữ liệu huấn luyện, tập ngữ liệu phát triển và tập ngữ liệu kiểm tra như được phân chia trong bộ ngữ liệu UIT-ViQuAD **[CT5]**. Ngoài ra, NCS sử dụng hai thông số đánh giá, độ chính xác EM và độ đo F1, để đo hiệu suất của các mô hình MRC. Cụ thể, trong nghiên cứu này, hai độ đo đánh giá được tính như sau:

Độ chính xác (EM) là thông số đánh giá đầu tiên để đánh giá mô hình đọc hiểu dựa trên trích xuất chuỗi (span-based). Tỷ lệ các câu trả lời dự đoán khớp chính xác với các câu trả lời mong đợi được xác định bởi độ chính xác EM.

Bảng 4.2. Hiệu suất của mô hình đọc hiểu văn bản tiếng Việt.

Mô hình	EM	F1
DrQA	40,91	63.44
QANet	46,05	68,06
BERT	59,28	80,00
XLM-R_{Base}	63,00	81,95
XLM-R_{Large}	68,98	87,02
ViReader	70,83	89,54

NCS so sánh mô hình đề xuất với các mô hình cơ sở khác nhau sử dụng mạng nơ-ron truyền thống (DrQA Reader và QANet) và các mô hình dựa trên kiến trúc Transformer (BERT và XLM-R) về mặt hiệu suất. Bảng 4.2 hiển thị kết quả độ chính xác (EM) và độ đo F1 của các thử nghiệm. Mô hình đề xuất ViReader hoạt động tốt hơn các MHNN như BERT và XLM-R. So với các mô hình đọc hiểu khác

dựa trên mạng nơ-ron học sâu và kiến trúc Transformer, các kết quả đạt được những cải tiến đáng kể trong khoảng từ 1,85% đến 29,92% theo độ chính xác EM và từ 2,52% đến 26,10% theo độ đo F1.

NCS mong muốn ViReader có khả năng được tinh chỉnh và hiệu quả trên hai bộ ngữ liệu khác: UIT-ViNewsQA **[CT1]** và BiPaR. Trái ngược với UIT-ViQuAD, UIT-ViNewsQA dành riêng cho miền đóng (dành cho tin tức trực tuyến thuộc miền chăm sóc sức khỏe) để đánh giá MRC và chứa các văn bản dài hơn nhiều **[CT1]**. BiPaR là bộ ngữ liệu song ngữ về tiểu thuyết, có thể để đánh giá MRC trên các ngôn ngữ khác như tiếng Anh và tiếng Trung. Chương 3 đã trình bày các đặc điểm của hai bộ ngữ liệu so với bộ ngữ liệu UIT-ViQuAD. Để xác định xem mô hình đọc hiểu đề xuất ViReader có hoạt động tốt trên hai bộ ngữ liệu chuẩn khác: UIT-ViNewsQA và BiPaR hay không.

Bảng 4.3. Hiệu suất của mô hình MRC.

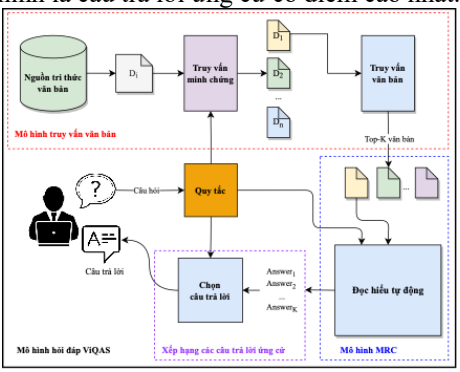
Mô hình	Tiếng Việt				Ngôn ngữ khác			
	UIT-ViQuAD		UIT-ViNewsQA		BiPaR			
					Tiếng Anh		Tiếng Trung	
	EM	**F1**	**EM**	**F1**	**EM**	**F1**	**EM**	**F1**
DrQA Reader	40,91	63,44	45,83	74,09	27,00	39,29	37,40	53,11
QANet	46,05	68,06	56,71	79,79	-	-	-	-
$BERT_{Base}$	59,28	80,00	63,81	83,19	41,40	55,03	48,87	64,09
$BERT_{Large}$	-	-	-	-	42,53	56,48	-	-
ALBERT	-	-	65,26	84,89	-	-	-	-
$XLM\text{-}R_{Base}$	63,00	81,95	-	-	-	-	-	-
$XLM\text{-}R_{Large}$	68,98	87,02	-	-	-	-	-	-
ViReader	**70,83**	**89,54**	**72,94**	**89,91**	**49,40**	**62,68**	**52,87**	**70,22**
So với $BERT_{Base}$	11,55↑	9,54↑	9,13↑	6,72↑	8,00↑	7,65↑	4,00↑	6,13↑
So với MHCS tốt nhất	1,85↑	2,52↑	7,68↑	7,68↑	6,87↑	6,20↑	4,00↑	6,13↑

Bảng 4.3 hiển thị so sánh hiệu suất của các mô hình MRC trên bộ ngữ liệu UIT-ViNewsQA, BiPaR và UIT-ViQuAD. Các kết quả thử nghiệm cho thấy mô hình ViReader cũng đạt kết quả vượt trội so với các mô hình khác trên cả tiếng Việt, tiếng Anh và tiếng Trung.

CHƯƠNG 5: MÔ HÌNH HỎI ĐÁP TIẾNG VIỆT TÍCH HỢP ĐỌC HIỂU TỰ ĐỘNG

5.1. Mô hình hỏi đáp tiếng Việt tích hợp đọc hiểu tự động

Trong chương này, NCS mô tả một kiến trúc mô hình QA mới cho tiếng Việt dựa trên học chuyển tiếp nhiều giai đoạn với các MHNN, bao gồm các quy tắc tiền xử lý, mô hình truy xuất văn bản, mô hình MRC và xếp hạng các câu trả lời ứng cử. Trong các thành phần này, hai giai đoạn cốt lõi là mô hình truy xuất văn bản và mô hình đọc hiểu văn bản với các MHNN dựa trên BERTology. Mô hình QA được mô tả tổng thể trong Hình 5.1. Thứ nhất, một câu hỏi đầu vào được chuẩn hóa bằng các quy tắc tiền xử lý trước khi đưa vào các thành phần khác của mô hình. Mô hình truy xuất văn bản tìm các văn bản chứa câu trả lời ứng cử bằng cách xếp hạng một nhóm văn bản dựa trên mức độ liên quan giữa câu hỏi và văn bản. Sau đó, mô hình đọc hiểu văn bản sẽ đọc các văn bản được xếp hạng cao nhất để xác định các câu trả lời chính xác. Mỗi câu trả lời ứng cử có điểm được tính bởi xếp hạng lại câu trả lời. Đầu ra của mô hình là câu trả lời ứng cử có điểm cao nhất.



Hình 5.1. Tổng quan về mô hình hỏi đáp ViQAS.

5.3.1. Các quy tắc tiền xử lý

Trong công trình [**CT6**], các cụm từ để hỏi đối với mỗi dạng câu hỏi tiếng Việt rất đa dạng. Ví dụ, các cụm từ câu hỏi trong câu hỏi What là "là gì", "cái nào", "điều gì", "điều nào" và các cụm từ tương tự. Tuy nhiên, chỉ có một cụm để hỏi chiếm tỷ lệ đáng kể nhất cho mỗi dạng câu hỏi. Họ cũng cho thấy các cụm từ hỏi có tác động đến hiệu suất của các mô hình. NCS đề xuất một số quy tắc để xử lý năm loại câu hỏi (When, Where, Who, What và Why) nhằm cải thiện kết quả cho toàn mô hình. **Thuật toán 5.1** trình bày cách quy tắc xử lý các loại câu hỏi như When, Where, Who, Why, và What, trước khi được đưa vào các thành phần của mô hình như mô hình truy xuất văn bản và mô hình đọc hiểu. NCS không thiết kế các quy tắc cho How (như thế nào) và How many (về số lượng) vì các từ hỏi ít đa dạng hơn và rõ ràng hơn.

Thuật toán 5.1. *Tiền xử lý câu hỏi trước khi đưa vào các thành phần còn lại của mô hình hỏi đáp ViQAS.*

Đầu vào (Input):

- Một câu hỏi Q dưới dạng ngôn ngữ tự nhiên.
- Danh sách chứa các từ hỏi về thời gian: WhenL = {thời gian nào, lúc nào, thời điểm nào, năm bao nhiêu, bao giờ, trong thời gian nào, **khi nào**}.
- Danh sách chứa các từ hỏi về nơi chốn: WhereL = {ở đâu, tại đâu, địa điểm nào, vị trí nào, quốc gia nào, nước nào, **nơi nào**}.
- Danh sách chứa các từ hỏi về người: WhoL = {người nào, **ai**}.
- Danh sách chứa các từ hỏi về lý do: WhyL = {lý do gì, lí do gì, lí do nào, lý do nào, nguyên nhân gì, nguyên nhân do đâu, nhờ đâu, tại sao, vì sao, do đâu mà, **nguyên nhân nào**}.
- Danh sách chứa các từ hỏi về sự vật, sự việc, sự kiện và hiện tượng: WhatL = \{nouns + nào/gì\} \{WhenL, WhereL, WhoL và WhyL}.
- Lưu ý rằng: Các từ hỏi đại diện được in đậm trong các danh sách trên.

Đầu ra (Output): Trả lại câu hỏi đã xử lý.

Function Xử lý câu hỏi(Q, WhenL, WhereL, WhoL, WhyL, WhatL)

    Khởi tạo: SQP ← null *# biến lưu trữ từ hỏi tìm được trong câu hỏi Q*

    If: Q.chứa(P) = True và P ∈ WhenL:

        Cập nhật: SQP ← Một từ hỏi đại diện cho câu hỏi về thời gian.

    If: Q. chứa (P) = True và P ∈ WhereL:

        Cập nhật: SQP ← Một từ hỏi đại diện cho câu hỏi về nơi chốn.

    If: Q.chứa (P) = True và P ∈ WhoL:

        Cập nhật: SQP ← Một từ hỏi đại diện cho câu hỏi về người.

    If: Q.chứa (P) = True và P ∈ WhyL:

        Cập nhật: SQP ← Một từ hỏi đại diện cho câu hỏi về lý do.

    If: Q.chứa(P) = True và P ∈ WhatL:

        Cập nhật: SQP ← Một từ hỏi đại diện cho câu hỏi về sự vật, sự việc, sự kiện và hiện tượng.

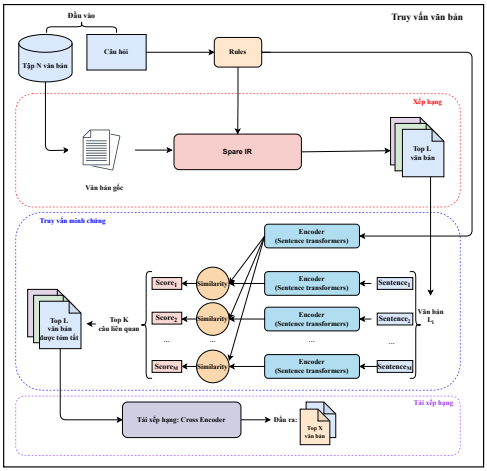
    Q ← Q.thay_thế(P, SQP) *# thay thế từ SQP cho từ P trong câu hỏi Q*

    Return Q

    End Function

Dựa trên đặc điểm của từng bộ ngữ liệu và ngôn ngữ của bộ ngữ liệu, các quy tắc có thể dễ dàng điều chỉnh để tối ưu hóa nhằm đạt kết quả tốt nhất có thể.

5.3.2. Mô hình truy xuất văn bản (ViDR)



Hình 5.2. Mô hình truy xuất văn bản ViDR của ViQAS.

NCS đề xuất một mô hình truy xuất văn bản dựa trên kiến trúc Sentence Transformer hiệu quả để trước tiên giới hạn tập văn bản liên quan đến truy vấn hoặc câu hỏi, các văn bản này có thể chứa câu trả lời. Hình 5.2 trình bày phương pháp truy xuất văn bản được đề xuất của NCS. Mô hình này bao gồm tiền truy xuất văn bản, truy xuất minh chứng và tái xếp hạng văn bản. NCS mong muốn tận dụng sức mạnh của phương pháp dựa trên vector thưa cho tiền truy vấn văn bản (Pre-Retriever) và phương pháp dựa trên vector đặc cho tái xếp hạng (Re-ranker). NCS sử dụng BERT cho Re-ranker vì khả năng nắm bắt ngữ cảnh tốt trong văn bản. Tuy nhiên, BERT có các văn bản được mã hóa với nhiều nhất chỉ có 512 tokens. Do đó, NCS tích hợp mô hình truy xuất minh chứng để tìm các câu có liên quan và giảm kích thước của mỗi văn bản để tăng hiệu quả của tái xếp hạng (Re-ranker) dựa trên BERT.

5.3.2.1. Tiền truy xuất văn bản (Pre-Retriever)

Đầu vào của mô hình của NCS tại mô-đun tiền truy vấn văn bản là một câu hỏi Q và một tập văn bản $D = \{d_1; d_2; d_3; \ldots; d_N\}$. Mô-đun tiền truy xuất văn bản sử dụng câu hỏi Q để truy xuất các văn bản có nội dung liên quan đến câu hỏi để tìm một tập hợp con của D. NCS cài đặt hai phương pháp truy xuất thông tin spare phổ biến nhưng hiệu quả bao gồm TF-IDF và BM25 Plus cho tiền truy xuất văn bản. Đặc biệt, NCS tính điểm cho mỗi văn bản trong D thể hiện sự tương đồng của văn bản đó với câu hỏi. Điểm này là sự kết hợp giữa BM25 Plus và TF-IDF với điểm $Score_{BM25Plus}$ là điểm tương đồng ngữ nghĩa giữa câu hỏi và văn bản với mô hình BM25 Plus và điểm tương đồng ngữ nghĩa $Score_{TF\text{-}IDF}$ đối với mô hình TF-IDF. Hệ số alpha (α) là một siêu tham số được tìm thấy bằng cách tinh chỉnh dựa trên

độ chính xác cho bộ ngữ liệu phát triển. Sau đó, NCS chọn ra L văn bản có số điểm cao nhất. Kết quả là, một số lượng lớn các văn bản không liên quan được loại bỏ, dẫn đến cải thiện hiệu quả của các mô-đun tiếp theo.

$$score = \alpha.Score_{BM25Plus} + (1-\alpha).Score_{TF\text{-}IDF} \quad (5.1)$$

Với $\alpha \in [0, 1]$ là một siêu tham số.

5.3.2.2. Mô hình truy vấn minh chứng (Evidence Extractor)

Kế thừa sự thành công của công trình **[CT2]**, truy vấn minh chứng giúp nâng cao hiệu quả của MRC tiếng Việt, NCS tiếp tục tích hợp truy xuất minh chứng vào truy vấn văn bản ViDR, mô hình đọc hiểu ViReader+ và mô hình hỏi đáp ViQAS. Quá trình này bao gồm ba giai đoạn chính như: tạo bộ ngữ liệu, tinh chỉnh mô hình và áp dụng mô hình mới được tinh chỉnh cho mô hình truy xuất minh chứng để trích xuất các câu minh chứng nâng cao hiệu quả cho mô hình truy xuất văn bản và mô hình MRC.

Thuật toán 5.2. *Chuyển các mẫu trong bộ ngữ liệu MRC sang các cặp tương đồng giữa câu hỏi – câu chứa câu trả lời trong bộ ngữ liệu mới*

Đầu vào (Input):

- Một câu hỏi Q.
- Một văn bản D.
- (Q, D) ∈ một tập ngữ liệu đọc hiểu tự động.

Đầu ra (Output): Trả về tối đa K mẫu ngữ liệu câu hỏi - câu có khả năng chứa câu trả lời và điểm tương đồng.

Function Chuyển mẫu ngữ liệu đọc hiểu sang mẫu ngữ liệu tương đồng câu hỏi – câu có khả năng chứa câu trả lời (Q, D, K)

Khởi tạo: Scores = [] *# tập lưu trữ các điểm số của các câu liên quan đến câu hỏi.*

Khởi tạo: SL = Document_Segmentation(D) *# tách văn bản D thành danh sách các câu.*

Khởi tạo: L = [] *# Danh sách lưu trữ top k câu liên quan.*

Khởi tạo: ScoreL = [] # Danh sách lưu trữ các điểm số của các mẫu ngữ liệu câu hỏi – câu có khả năng chứa câu trả lời.

Khởi tạo: $q = Sentence\_Transformer(Q)$

For i ∈ range$(0, \text{len}(SL) - 1)$

    Khởi tạo: s $= Sentence\_Transformer(SL[i])$

    Cập nhật: $Scores[i] =$ **semntic_similarity**(q, s)

    i = i + 1

End for

Cập nhật: Scores = Sorting (Scores) *# sắp xếp danh sách Score giảm dần*

Cập nhật: L = extract(K, SL, Scores) *# trích xuất top k câu liên quan (điểm số cao nhất của list Scores) từ danh sách SL.*

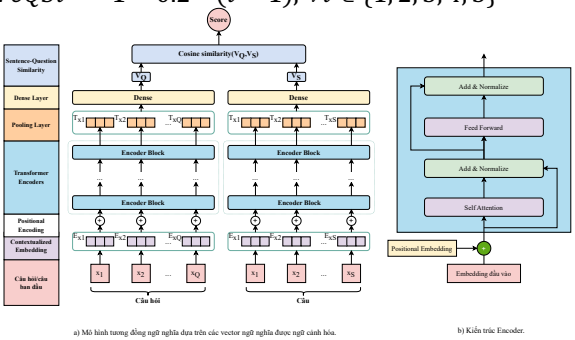
Cập nhật: ScoreL = ScoreQSi(L) # tính điểm của mỗi mẫu ngữ liệu câu hỏi – câu trả lời dựa trên Công thức (5.2).

Return L, ScoreL

16: **End function**

Ngữ liệu huấn luyện: Đầu tiên, NCS tạo một bộ ngữ liệu để huấn luyện mô hình truy xuất minh chứng từ bộ ngữ liệu MRC tiếng Việt với **Thuật toán 5.2**. Lấy cảm hứng từ phương pháp STR của mô hình đọc hiểu ViReader **[CT2]**, NCS ước tính độ tương đồng về ngữ nghĩa của câu hỏi và các câu trong văn bản. Đối với mỗi mẫu ngữ liệu câu hỏi-văn bản trong bộ ngữ liệu MRC, NCS lấy năm câu có độ tương đồng đáng kể nhất dựa trên STR và sắp xếp chúng theo thứ tự giảm dần về độ tương đồng ngữ nghĩa. NCS chọn tất cả các câu của văn bản có ít hơn năm câu. Sau đó, mỗi mẫu câu hỏi – câu trả lời được gán cho một số điểm thể hiện mức độ liên quan của mỗi câu đối với câu hỏi. Điểm có giá trị từ 0.2 đến 1.0, với điểm của mẫu ngữ liệu câu hỏi – câu thứ i có điểm được tính theo Công thức **(**5.2**)**.

$$ScoreQSi = 1 - 0.2 * (i - 1), \forall i \in \{1, 2, 3, 4, 5\} \quad (5.2)$$



a) Mô hình tương đồng ngữ nghĩa dựa trên các vector ngữ nghĩa được ngữ cảnh hóa. b) Kiến trúc Encoder.

Hình 5.3. Mô hình dựa trên Transformer cấp độ câu cho bài toán ước tính độ tương đồng giữa câu trả lời và câu hỏi về ngữ nghĩa.

Tinh chỉnh mô hình: NCS tinh chỉnh mô hình truy xuất minh chứng để tính toán độ tương đồng ngữ nghĩa của câu hỏi và câu trong văn bản với kiến trúc Sentence Transformer, được mô tả như trong Hình 5.3. **Thuật toán 5.3** mô tả quá trình xác định kiến trúc mô hình và tinh chỉnh mô hình. Mô hình này được tinh chỉnh MHNN được huấn luyện sẵn trên tiếng Việt, là MHNN được huấn luyện sẵn đơn ngôn ngữ dành cho tiếng Việt và hiệu quả cũng được chứng minh trong nhiều bài toán. Để xây dựng một câu được mã hóa vector với kích thước cố định, NCS thêm các hàm pooling và dense vào đầu ra của MHNN. Đầu vào cho mô hình là một mẫu ngữ liệu câu hỏi – câu trong văn bản và đầu ra là một điểm tương đồng ngữ nghĩa của chúng. Công thức độ tương đồng ngữ nghĩa giữa các câu được mã hóa vector u và vector v được hình thành từ mẫu ngữ liệu câu hỏi - câu được sử dụng để xác định độ tương đồng ngữ nghĩa. Hàm mất mát dựa trên sai số toàn phương trung bình (Mean Squared Error) là hàm mục tiêu cho mô hình. Vì phương pháp của NCS cần mã hóa các các vector ở cấp độ câu, nên NCS sử dụng quy trình tinh chỉnh mô hình của BERT cho cấp độ câu (SBERT). Lấy cảm hứng từ SBERT, mô hình tinh chỉnh

có khả năng mã hóa các câu và tạo các vector mã hóa để truy xuất minh chứng chính xác hơn.

Thuật toán 5.3. *Huấn luyện bài toán tương đầu giữa câu hỏi - câu có khả năng chứa câu trả lời.*

Đầu vào (Input):

- Một tập huấn luyện TS và tập phát triển DS của bài toán tương đồng câu hỏi – câu có khả năng chứa câu trả lời.
- Chiều dài chuỗi đầu vào Max.

Đầu ra (Output): Trả về mô hình phù hợp với bộ ngữ liệu.

1: **Function** Huấn luyện bài toán tương đồng giữa câu hỏi – câu có khả năng chứa câu trả lời (TS, DS, Max)

2: **TS** ← Load_Training_Data() *# lưu trữ ngữ liệu huấn luyện vào danh sách TS.*

3: **DS** ← Load_Development_Data() *# lưu trữ ngữ liệu huấn luyện vào danh sách DS.*

4: **WEL** ← Load_Pre-trained_Language_Model() *# lưu trữ MHNN được huấn luyện sẵn với chiều dài chuỗi đầu vào Max.*

5: **PL** ← Add_Pooling_Layer(số_chiều_word_embedding) # thêm lớp pooling (mean) với số chiều word embedding.

6: **DM**← $Put(connected\ dense\ layer,\ pooling\ layer,\ số\ chiều\_word\_embedding, Tanh)$ #Đặt lớp connected dense layer lên trên lớp gộp với chức năng kích hoạt Tanh và số chiều word embedding.

7: **M** ← Sentence_Transformer(WEL, PL, DM)

8: **F** ← Cosin_Similarity_Loss() #Sử dụng hàm loss Cosin similarity.

9: **M** ← Training(M, F, TS, DS) # Huấn luyện mô hình M với hàm mất mát F trên tập huấn luyện TS và điều chỉnh mô hình M trên tập phát triển DS

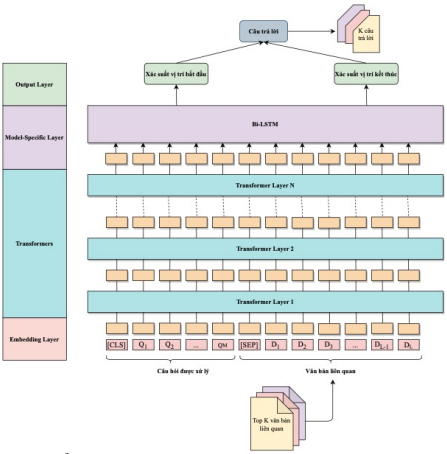
10: **Return M**

Lấy cảm hứng từ phương pháp STR của mô hình đọc hiểu ViReader **[CT2]**, NCS sử dụng mô hình Sentence Transformer để trích xuất các câu liên quan đến câu hỏi nhằm tăng hiệu suất của mô-đun tái xếp hạng. Thay vì cách tiếp cận không giám sát với mô hình được huấn luyện sẵn đa ngôn ngữ như STR của ViReader, NCS huấn luyện cách tiếp cận có giám sát cho mô hình Sentence Transformer và sử dụng nó để trích xuất câu minh chứng. NCS chia văn bản thành các câu riêng lẻ cho từng cặp văn bản-câu hỏi, sau đó chúng được biểu diễn dựa vào mô hình Sentence Transformer và tạo vector đặc trưng cho mỗi câu. Sau đó, mô hình ViDR chọn *k* câu của văn bản có độ tương đồng cao nhất với vector biểu diễn của câu hỏi, sử dụng Công thức **(**5.3**)**. NCS kết hợp *k* câu đã truy xuất để tạo thành một văn bản mới và một số câu có nội dung gây nhiễu sẽ bị loại bỏ. Văn bản mới ngắn hơn nhưng vẫn chứa đủ nội dung để trả lời câu hỏi và giúp mô hình tập trung vào những phần thiết yếu hơn của văn bản.

$$\cos(\mathbf{u}, \mathbf{v}) = \frac{\mathbf{uv}}{|\mathbf{u}||\mathbf{v}|} = \frac{\sum_{i=1}^{n} \mathbf{u}_i \mathbf{v}_i}{\sqrt{\sum_{i=1}^{n} (\mathbf{u}_i)^2} \sqrt{\sum_{i=1}^{n} (\mathbf{v}_i)^2}} \quad (5.3)$$

5.3.3.3. Xếp hạng lại các văn bản (Text Re-Ranker)

Trong thành phần xếp hạng lại, NCS tìm thấy một tập hợp các văn bản X (X ≤ L ≤ N) từ tập hợp các văn bản L được tổng hợp bởi thành phần truy vấn minh chứng. NCS đưa ra phương pháp tiếp cận dựa trên bộ mã hóa chéo (cross-encoder) để tìm các văn bản có độ tương đồng hiệu quả với câu hỏi. NCS hy vọng khả năng chứa câu trả lời cho các văn bản cũng cao. Dựa trên SBERT và các biến thể của SBERT, NCS sử dụng cross-encoder và mô hình được huấn luyện sẵn ở cấp văn bản để truy xuất văn bản trong mô hình QA. Mô hình xếp hạng lại mã hóa câu hỏi và mỗi văn bản để ước tính độ tương đồng ngữ nghĩa giữa mỗi văn bản với câu hỏi. NCS trích xuất X văn bản có độ tương đồng ngữ nghĩa cao nhất và mong muốn câu trả lời nằm trong các văn bản đã chọn. Thông qua mô hình Re-ranker, nhiều văn bản có nội dung không liên quan liên tục được loại bỏ nhằm thu hẹp phạm vi tìm kiếm của mô hình nhưng vẫn đảm bảo khả năng tìm được câu trả lời cho câu hỏi. Để làm được như vậy, điểm tương đồng ngữ nghĩa giữa câu hỏi và mỗi văn bản được tính. Sau đó, X văn bản xếp hạng cao nhất (dựa trên điểm tương tự cao nhất) được trả về.



Hình 5.4. Mô hình đọc hiểu văn bản.

5.3.3. Mô hình đọc hiểu văn bản (Text Reader)

NCS đề xuất một mô hình đọc hiểu bao gồm hai thành phần: mô-đun tiền xử lý và mô hình đọc hiểu văn bản. Đầu vào bao gồm một câu hỏi và văn bản *k* câu bao gồm *k* câu có liên quan nhất với nội dung của câu hỏi. Sau đó, mô hình trích xuất câu trả lời đọc một văn bản truy xuất minh chứng bao gồm các câu được xếp hạng cao nhất để trích xuất các câu trả lời có thể. Văn bản truy xuất minh chứng được kế thừa từ giai đoạn truy xuất minh chứng của truy xuất văn bản. Tổng quan kiến

trúc của mô hình đọc hiểu được thể hiện trong Hình 5.4. Kiến trúc mô hình và cách huấn luyện mô hình đọc hiểu được mô tả như sau.

5.3.3.1. Kiến trúc mô hình

Trong mô-đun tiền xử lý, NCS xử lý trước câu hỏi và văn bản trước khi trích xuất các câu trả lời. Đối với câu hỏi, NCS áp dụng tất cả các quy tắc được mô tả trong phần quy tắc tiền xử lý. Đối với văn bản, NCS tóm tắt chúng bằng cách trích ra các câu chứa câu trả lời ứng cử dựa trên độ tương đồng giữa câu hỏi và mỗi câu trong văn bản. NCS sử dụng mô hình Sentence Transformer đã huấn luyện sẵn để vector hóa các câu và câu hỏi nhằm tạo vector đặc trưng của chúng. Tính tương đồng ngữ nghĩa của chúng được tính toán để tìm k-câu phù hợp nhất với câu hỏi. Tuy nhiên, nếu vận hành mô hình hỏi đáp ViQAS đầy đủ, giai đoạn này bị bỏ qua vì nó đã được thực hiện ở mô-đun truy xuất văn bản.

Các phương pháp tiếp cận dựa trên học chuyển tiếp được thực hiện trên mô hình đề xuất của NCS. Đặc biệt, MHNN đa ngôn ngữ XLM-R được sử dụng cho mô hình trích xuất câu trả lời này. Sức mạnh của biểu diễn từ theo ngữ cảnh cho phép các mô hình MRC dựa trên biểu diễn theo ngữ cảnh và kiến trúc Transformer để đạt được hiệu suất thậm chí cao hơn các mô hình khác. Hình 5.4 cho thấy thành phần MRC của mô hình hỏi đáp ViQAS. Kiến trúc BERTology chung biểu diễn cho hai chuỗi đầu vào dưới dạng một chuỗi token từng đoạn được phân tách bằng mã [SEP]. Mô hình được huấn luyện sẵn tạo ra một biểu diễn token đầu ra cho mọi token trong văn bản. Đối với khả năng MRC có câu trả lời được rút trích trực tiếp từ văn bản, câu hỏi là chuỗi trình tự đầu tiên và văn bản là chuỗi trình tự thứ hai được mã hóa. Chúng ta cũng cần thêm một số cấu trúc vào đầu ra được huấn luyện trong giai đoạn tinh chỉnh.

NCS thêm một lớp BiLSTM vào sau lớp sau cùng của XLM-R để xử lý vector biểu diễn chuyển tiếp (forward) $\overrightarrow{T_i'}$ và vector biểu diễn ngược (backward) $\overleftarrow{T_i'}$ tại mỗi token thứ i, thu được vector biểu diễn theo ngữ cảnh chuyển tiếp (forward) $\overrightarrow{L_i'}$ và vector biểu diễn từ theo ngữ cảnh ngược (backward) $\overleftarrow{L_i'}$. Tại mỗi token, vector embeding ngữ cảnh cuối cùng thu được bằng cách kết hợp với $\overrightarrow{L_i'}$ và $\overleftarrow{L_i'}$.

$$\overrightarrow{L_i'} = LSTM(\overrightarrow{T_i'}) \tag{5.4}$$

$$\overleftarrow{L_i'} = LSTM(\overleftarrow{T_i'}) \tag{5.5}$$

$$L_i' = \overrightarrow{L_i'} + \overleftarrow{L_i'} \tag{5.6}$$

Hai vector biểu diễn ngữ cảnh hóa thu được là một vector biểu diễn của vị trí bắt đầu câu trả lời (S) và một vector biểu diễn của vị trí kết thúc của câu trả lời (E) tương ứng với bộ phận tử thứ nhất và thứ hai của vector ngữ cảnh cuối cùng L_i' của mỗi token thứ i trong văn bản T. Xác suất vị trí bắt đầu và xác suất vị trí kết thúc

của câu trả lời được xác định thông qua các hàm softmax tương ứng theo Công thức (5.7) và Công thức (5.8):

$$Pstart_i = \frac{exp(S.L_i')}{\sum_j exp(S.L_j')} \quad (5.7)$$

$$Pend_k = \frac{exp(E.L_k')}{\sum_j \exp(E.L_j')} \quad (5.8)$$

Cross-entropy được sử dụng làm hàm mất mát để tính toán tổn thất vị trí bắt đầu và vị trí kết thúc ($loss_{start}$ và $loss_{end}$) cho mô hình đọc hiểu. Ystart và Yend là các đầu ra thực tế với các giá trị phi xác suất. Công thức (5.9) và Công thức (5.10) lần lượt ước tính các hàm tổn thất cho $loss_{start}$ và $loss_{end}$, sau đó lấy trung bình chung của chúng để tạo trung bình điểm mất mát tổng thể trên N mẫu huấn luyện theo Công thức (5.11).

$$loss_{start} = -\frac{1}{N}\sum_{i=1}^{N}\Big[Ystart_i * log(Pstart_i) + (1 - Ystart_i) * log(1 - Pstart_i)\Big] \quad (5.9)$$

$$loss_{end} = -\frac{1}{N}\sum_{i=1}^{N}\Big[Yend_i * log(Pend_i) + (1 - Yend_i) * log(1 - Pend_i)\Big] \quad (5.10)$$

$$loss_{average} = \frac{loss_{start} + loss_{end}}{2} \quad (5.11)$$

5.3.3.2. Huấn luyện mô hình đọc hiểu

Trước khi huấn luyện, NCS xử lý trước bộ ngữ liệu đọc hiểu của máy để huấn luyện cho các mô hình đọc hiểu ViReader+. Các tập huấn luyện và tập phát triển được tải (load) lên và áp dụng các quy tắc xử lý cho tất cả các câu hỏi cho cả hai tập ngữ liệu. Giai đoạn tiền xử lý làm thay đổi văn bản và chỉ số bắt đầu của câu trả lời trong văn bản mới sẽ được cập nhật theo **Thuật toán 5.4.** Đầu tiên, văn bản được đưa về một danh sách các câu và load mô hình Sentence Transformer. Mô hình này được huấn luyện và trình bày trong phần truy xuất văn bản. Phương pháp trích xuất các câu minh chứng để trích xuất *k* câu của văn bản. Đối với các văn bản có ít hơn *k* câu, văn bản không thay đổi. Vị trí bắt đầu câu trả lời được cập nhật trong trường hợp văn bản thay đổi. Việc cập nhật được thực hiện dựa trên tìm kiếm câu chứa câu trả lời và vị trí bắt đầu câu trả lời.

Thuật toán 5.4. *Truy vấn k câu từ văn bản D liên quan đến câu hỏi Q và cập nhật chỉ số bắt đầu cho ngữ liệu.*

Đầu vào (Input):

- Cho một câu hỏi Q, một văn bản D, câu trả lời A và vị trí bắt đầu câu trả lời A trong văn bản D là AS.

Đầu ra (Output): Trả về văn bản mới ND gồm k câu liên quan đến câu hỏi Q và vị trí bắt đầu mới của câu trả lời New_AS trong văn bản ND.

Function Trích xuất k câu liên quan và cập nhật chỉ số bắt đầu của câu trả lời (D, Q, A, AS)

SL ← Khởi tạo danh sách chứa các câu (D) *# lưu trữ danh sách các câu trong văn bản D vào danh sách SL.*

S ← Tìm câu chứa câu trả lời (D, A, AS) *# tìm câu chứa câu trả lời A trong văn bản D dựa trên vị trí bắt đầu AS.*

Start ← Tìm câu chứa câu trả lời (A, S) *# tìm vị trí bắt đầu câu trả lời A trong câu S.*

ST ← Load_Pre-trained_Language_Model() *# lưu trữ MHNN được huấn luyện sẵn.*

Question_embedding ← ST(Q)

Sentence_embeddings ← ST(SL)

Scores ← cosine_similarity(Question_embedding, Sentence_embeddings)

Scores ← Ranking(Scores) # xếp hạng theo thứ tự giảm dần

ND ← Chọn top k câu và kết hợp các câu thành văn bản mới

New_AS ← ND.find(S) + Start

Return ND, New_AS

Sau khi truy xuất các câu minh chứng cho văn bản trong bô ngữ liệu, NCS xử lý trước ngữ liệu và huấn luyện mô hình trích xuất câu trả lời thông qua **Thuật toán 5.5**. Đối với bộ ngữ liệu, các câu hỏi phải tuân theo các quy tắc tiền xử lý đã được giới thiệu trong Mục 5.3.1. NCS xác định cấu trúc mô hình như trong Hình 5.4. Bộ tokenizer được sử dụng để tạo đặc trưng ngữ liệu dựa trên đầu vào bao gồm văn bản và câu hỏi. Hàm Cross-Entropy được sử dụng như một hàm mất mát trong quá trình huấn luyện mô hình. Sau đó, NCS huấn luyện mô hình dựa trên đặc điểm ngữ liệu của tập huấn luyện và hàm mất mát. Mô hình sau đó được đánh giá trên tập phát triển. Cuối cùng, thuật toán huấn luyện mô hình (**Thuật toán 5.5**) trả về một mô hình trích xuất các câu trả lời dựa trên các cặp văn bản-câu hỏi.

Thuật toán 5.5. *Tiền xử lý và huấn luyện mô hình rút trích câu trả lời.*

Đầu vào (Input):

- Một tập huấn luyện TS và tập phát triển DS của bài toán rút trích câu trả lời.

Đầu ra (Output): Trả về mô hình phù hợp với bộ ngữ liệu.

Giai đoạn 1: Tiền xử lý dữ liệu

Function Tiền xử lý trước khi thực hiện mô hình

TS ← Load_Training_Data() # lưu trữ ngữ liệu của tập huấn luyện vào danh sách TS.

DS ← Load_Development_Data() # lưu trữ ngữ liệu của tập phát triển vào danh sách DS.

TS ← Thực hiện tiền xử lý với Thuật toán 5.1 trên các câu hỏi của TS.

5: DS ← Thực hiện tiền xử lý với Thuật toán 5.1 trên các câu hỏi của DS.

6: TS ← Cập nhật văn bản và vị trí câu trả lời của mỗi ngữ liệu huấn luyện trong tập TS theo **Thuật toán 5.4**

7: DS ← Cập nhật văn bản và vị trí câu trả lời của mỗi ngữ liệu huấn luyện trong tập DS theo **Thuật toán 5.4**

Giai đoạn 2: Huấn luyện mô hình

8: **Function** Huấn luyện mô hình rút trích câu trả lời

9: **M** ← Định nghĩa kiến trúc và load mô hình lên

10: **T** ← Load bộ tokenizer

11: **TF** ← T(TS) # Khởi tạo đặc trưng token cho tập huấn luyện

12: **DF** ← T(DS) # Khởi tạo đặc trưng token cho tập phát triển

13: **F** ← Cross Entropy() #Khởi tạo hàm mất mát

14: **M** ← M(TF, DF, F) #Huấn luyện mô hình M với hàm mất mát F trên tập đặc trưng ngữ liệu huấn luyện TF và điều chỉnh mô hình trên tập đặc trưng ngữ liệu phát triển DF

15: **Return M**

5.3.4. Xếp hạng các câu trả lời ứng cử (Answer Re-ranker)

Trong xếp hạng lại của quá trình truy xuất văn bản, NCS sử dụng độ tương đồng ngữ nghĩa, điểm ($S_{Retriever}$) được gán cho mỗi văn bản trong tập hợp các văn bản X. Mô hình đọc hiểu dự đoán chuỗi văn bản liên tục tối ưu trong mọi trường hợp với một điểm số (S_{Reader}). Phương pháp của NCS sử dụng thuật toán nội suy tuyến tính để kết hợp điểm của mô hình truy xuất văn bản với điểm của mô hình đọc hiểu, theo Công thức (5.12):

$$S_{answer} = \beta * S_{Reader} + (1 - \beta) * S_{Retriever} \quad (5.12)$$

trong đó $\beta \in [0, 1]$ là một siêu tham số. NCS điều chỉnh β với 2.000 mẫu ngữ liệu câu hỏi – câu trả lời được chọn ngẫu nhiên từ mỗi bộ ngữ liệu.

5.2. Kết quả

Các kết quả thử nghiệm

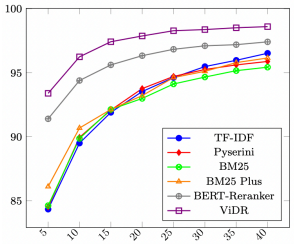
Các đánh giá thử nghiệm của mô hình truy xuất văn bản và mô hình đọc hiểu được trình bày riêng lẻ trong phần này, tiếp theo là các đánh giá về sự kết hợp của chúng, cũng như phương pháp hỏi đáp đề xuất của NCS, dành cho các mô hình QA trên ngữ liệu tiếng Việt.

Kết quả trên truy xuất văn bản

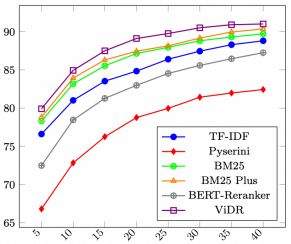
Bảng 5.1. Các kết quả trên các mô hình truy xuất văn bản tiếng Việt.

Các bộ ngữ liệu	Trên tin tức	Trên văn bản Wikipedia
TF-IDF_	84,84	89,50
Pyserini_	78,77	89,95
BM25_	87,15	89,86

BM25 Plus_	87,45	90,68
BERT-Ranker	72,98	94,39
ViDR	**89,11**	**96,24**



a) Ngữ liệu Wikipedia tiếng Việt



b) Văn bản tin tức

Hình 5.5. Hiệu quả mô hình theo số lượng văn bản truy xuất.

Trên tất cả các bộ ngữ liệu tiếng Việt, phương pháp đề xuất của NCS (ViDR) vượt trội hơn so với TF-IDF, Pyserini, BM25, BM25 Plus và BERT-Ranker. Mô hình đề xuất của NCS (ViDR) đạt được hiệu suất tốt hơn các mô hình khác. Khi *k* lớn hơn, mô hình đề xuất của NCS hiệu quả hơn. Bởi vì các văn bản trên tin tức dài hơn trên Wikipedia, hiệu suất các mô hình rút trích thông tin rõ ràng là hiệu quả hơn trên Wikipedia. BERT-Reranker không thể hiệu quả hơn BM25 Plus trên các văn bản dài (các bài báo tin tức) vì BERT-Ranker chỉ mã hóa tốt các văn bản có độ dài 512 token. Tuy nhiên, mô hình đề xuất của NCS vẫn hiệu quả trên các văn bản dài khi so với BM25 Plus.

Kết quả trên đọc hiểu tự động

Bảng 5.2. Các kết quả trên các mô hình MRC tiếng Việt.

Mô hình	UIT-ViNewsQA		UIT-ViQuAD		UIT-ViWikiQA	
	EM	F1	EM	F1	EM	F1
WikiBERT_	62,30	82,85	56,11	75,91	82,88	87,16
BERT_	63,81	83,19	59,28	80,00	84,74	88,57
ALBERT_	65,26	84,89	57,87	78,58	76,15	82,24
PhoBERT$_{Large}$_	70,98	88,89	66,17	84,27	87,09	90,46
XLM-R$_{Large}$_	71,49	89,33	68,98	87,02	85,87	88,77
ViReader_	72,94	89,91	70,83	89,54	89,41	91,70
ViReader+	**76,46**	**91,84**	**72,31**	**89,92**	**92,26**	**94,26**

Tiếp theo, NCS đánh giá thành phần đọc hiểu văn bản trên ba bộ ngữ liệu chuẩn: UIT-ViQUAD **[CT5]**, UIT-ViWikiQA [**CT6**] và UIT-ViNewsQA **[CT1]**. NCS so sánh cách tiếp cận được đề xuất của NCS với ViReader **[CT2]** và các mô hình hiệu quả khác sử dụng các MHNN như WikiBERT, BERT, ALBERT, PhoBERT và XLM-R. Bảng 5.2 cho thấy kết quả thử nghiệm của NCS trên các mô hình MRC. Mô hình đọc hiểu văn bản được đề xuất của NCS hiệu quả hơn các mô hình đọc

hiểu dựa trên các MHNN hiệu quả như WikiBERT, BERT, ALBERT, PhoBERT, XLM-R và mô hình đặc trưng cho tiếng Việt như ViReader.

Bảng 5.3. Hiệu suất các mô hình hỏi đáp tiếng Việt.

Phương pháp	ViNewsQA		ViQuAD		ViWikiQA	
	EM	**F1**	**EM**	**F1**	**EM**	**F1**
DrQA	28,92	47,76	18,42	37,86	35,84	55,92
BERTserini	32,63	48,84	39,46	58,30	53,93	62,57
XLMRQA	38,40	51,76	51,94	64,99	56,29	64,30
ORQA	36,98	52,72	41,40	55,84	62,40	69,86
COBERT	32,08	49,18	33,76	54,70	60,90	70,44
NeuralQA	34,84	48,79	42,35	58,02	55,79	64,65
BERTBM25	39,66	57,14	44,48	60,33	58,73	66,42
ViQAS	**50,40**	**64,12**	**56,47**	**70,82**	**68,55**	**75,16**

Kết quả trên mô hình hỏi đáp

NCS trình bày các kết quả trên các tập kiểm tra của các bộ ngữ liệu này. Như các số liệu trong Bảng 5.3, NCS so sánh hiệu quả của mô hình hỏi đáp ViQAS của NCS với các phương pháp tiên tiến trước đây (DrQA, BERTserini, XLMRQA [**CT7**], ORQA, COBERT, NeuralQA và BERTBM25) và các mô hình đã triển khai khác bởi NCS . Mô hình hỏi đáp ViQAS vượt trội hơn tất cả các phương pháp tiếp cận tiên tiến với một sự cải thiện đáng kể.

CHƯƠNG 6: KẾT LUẬN VÀ HƯỚNG PHÁT TRIỂN

6.1. Kết luận

Luận án đã có ba đóng góp chính sau:

- **Đóng góp #1: Xây dựng ngữ liệu và đánh giá các mô hình đọc hiểu trên ngữ liệu tiếng Việt**. Các đóng góp nghiên cứu về các bộ ngữ liệu được công bố tại các tạp chí và hội nghị: **[CT1]**, **[CT4]**, **[CT5]**, và [**CT6**].
- **Đóng góp #2: Đề xuất mô hình MRC tích hợp truy vấn minh chứng trên ngữ liệu tiếng Việt**. Các đóng góp nghiên cứu thử nghiệm về ViReader được công bố tại tạp chí quốc tế: **[CT2]** (xem chi tiết trong **Chương 4)** và đánh giá mở rộng mô hình đọc hiểu **ViReader+** trên mô hình hỏi đáp ViQAS cho tiếng Việt được công bố tại tạp chí quốc tế: **[CT3]**.
- **Đóng góp #3: Đề xuất mô hình hỏi đáp tích hợp MRC trên ngữ liệu tiếng Việt**. Luận án đã đề xuất thiết kế và xây dựng các mô hình QA đạt hiệu quả cao trên ngữ liệu tiếng Việt: **XLMRQA** và **ViQAS.** Các đóng góp nghiên cứu các mô hình QA trên ngữ liệu tiếng Việt được công bố tại các tạp chí và hội nghị: **[CT3]**, [**CT7**].

6.2. Các hạn chế và các hướng phát triển

Về ngữ liệu cho đánh giá các mô hình MRC: Một số xu hướng cần chú ý trong việc xây dựng ngữ liệu cho đánh giá MRC trên ngữ liệu tiếng Việt như sau: nâng cao chất lượng và độ phức tạp của ngữ liệu, mở rộng các miền ngữ liệu, hướng đến những dạng MRC khác nhau và hướng đến chắt lọc ngữ liệu cho MRC.

Về mô hình đọc hiểu và hỏi đáp tự động: Một số hạn chế vẫn chưa giải quyết được, vì vậy đọc hiểu và hỏi đáp tự động tiếng Việt có thể khám phá theo những hướng sau: các phương pháp MRC dựa các MHNN tạo sinh: BART, T5 và OpenGPT; tận dụng các đặc trưng ngôn ngữ và tri thức ngoài để nâng cao hiệu suất; Các mạng nơ-ron dựa trên đồ thị GNN (Graph Neural Networks) và hậu xử lý các câu trả lời là rất quan trọng để tạo nên mô hình đọc hiểu/hỏi đáp trở nên tự nhiên hơn.

Về mở rộng ứng dụng của các mô hình MRC: Ngoài ứng dụng MRC vào thiết kế, xây dựng và đánh giá các mô hình QA, các mô hình MRC được tích hợp cho nhiều ứng dụng khác như Chatbot hoặc các trợ lý ảo.

CÔNG BỐ KHOA HỌC

Trong thời gian hoàn thành luận án, NCS đã công bố 07 bài báo khoa học, trong đó: 04 bài báo đăng tại các tạp chí uy tín và 03 bài báo đăng tại các hội nghị quốc tế uy tín:

[CT1] Kiet Van Nguyen; Tin Van Huynh; Duc-Vu Nguyen; Anh Gia-Tuan Nguyen; Ngan Luu-Thuy Nguyen; "***New Vietnamese Corpus for Machine Reading Comprehension of Health News Articles"***. TALLIP. 2022.

[CT2] Kiet Van Nguyen; Nhat Duy Nguyen; Phong Nguyen-Thuan Do; Anh Gia-Tuan Nguyen; Ngan Luu-Thuy Nguyen; "***ViReader: A Wikipedia-based Vietnamese Reading Comprehension System using Transfer Learning"***. JIFS. 2021.

[CT3] Kiet Van Nguyen; Phong Nguyen-Thuan Do; Nhat Duy Nguyen; Anh Gia-Tuan Nguyen; Ngan Luu-Thuy Nguyen; "***Multi-Stage Transfer Learning with BERTology-Based Language Models for Question Answering System in Vietnamese"***. IJMLC. 2023.

[CT4] Kiet Van Nguyen; Son Quoc Tran; Luan Thanh Nguyen; Tin Van Huynh; Son T. Luu; Ngan Luu-Thuy Nguyen*; "**VLSP 2021 - ViMRC Challenge: Vietnamese Machine Reading Comprehension"***. JSCSCE. 2022.

Danh sách các bài báo HNQT uy tín:

[CT5] Kiet Van Nguyen; Duc-Vu Nguyen; Anh Gia-Tuan Nguyen; Ngan Luu-Thuy Nguyen. "***A Vietnamese Dataset for Evaluating Machine Reading Comprehension"***. COLING 2020.

[**CT6**] Phong Nguyen-Thuan Do; Nhat Duy Nguyen; Tin Van Huynh, **Kiet Van Nguyen (corresponding author)**; Anh Gia-Tuan Nguyen, and Ngan Luu-Thuy Nguyen. "***Sentence Extraction-Based Machine Reading Comprehension for Vietnamese"***. KSEM 2021.

[**CT7**] **Kiet Van Nguyen**; Phong Nguyen-Thuan Do; Nhat Duy Nguyen; Tin Van Huynh; Anh Gia-Tuan Nguyen; Ngan Luu-Thuy Nguyen; "***XLMRQA: Open-Domain Question Answering on Vietnamese Wikipedia-based Textual Knowledge Source"***. ACIIDS 2021.