AN EFFECTIVE METHOD COMBINING DEEP LEARNING MODELS AND REINFORCEMENT LEARNING TECHNOLOGY FOR EXTRACTIVE TEXT SUMMARIZATION

Luu Minh Tuan[1,2], Le Thanh Huong[1*], Hoang Minh Tan[1]
[1]Hanoi University of Science and Technology, [2]National Economics University

ARTICLE INFO

Received: 13/7/2021
Revised: 12/8/2021
Published: 12/8/2021

KEYWORDS

Text summarization
Reinforcement learning
BERT model
CNN
GRU

ABSTRACT

Automatic text summarization is an important problem in natural language processing. Text summarization extracts the most important information from one or many source texts to generate a brief, concise summary that still retains main ideas, correct grammar and ensures the coherence of the text. With the application of machine learning techniques as well as deep learning models in automatic text summarization models gave summaries that were closely resemble human reference summaries. In this paper, we propose an effective extractive text summarization method by combining the deep learning models, the reinforcement learning technique and MMR method to generate the summary. Our proposed method is experimented on CNN dataset (English) and Baomoi dataset (Vietnamese) giving F1-score accuracy results with Rouge-1, Rouge-2, Rouge-L are 31.36%, 12.84%, 28.33% and 51.95%, 24.38%, 37.56%, respectively. The experimental results show that our proposed summarization method has achieved good results for English and Vietnamese text summarization.

MỘT PHƯƠNG PHÁP KẾT HỢP CÁC MÔ HÌNH HỌC SÂU VÀ KỸ THUẬT HỌC TĂNG CƯỜNG HIỆU QUẢ CHO TÓM TẮT VĂN BẢN HƯỚNG TRÍCH RÚT

Lưu Minh Tuấn[1,2], Lê Thanh Hương[1*], Hoàng Minh Tân[1]
[1]Trường Đại học Bách khoa Hà Nội, [2]Trường Đại học Kinh tế Quốc dân

THÔNG TIN BÀI BÁO

Ngày nhận bài: 13/7/2021
Ngày hoàn thiện: 12/8/2021
Ngày đăng: 12/8/2021

TỪ KHÓA

Tóm tắt văn bản
Học tăng cường
Mô hình BERT
Mạng CNN
Mạng GRU

TÓM TẮT

Tóm tắt văn bản tự động là bài toán quan trọng trong xử lý ngôn ngữ tự nhiên. Tóm tắt văn bản trích rút các thông tin quan trọng nhất từ một hoặc nhiều văn bản nguồn để tạo ra một văn bản tóm tắt ngắn gọn, súc tích nhưng vẫn giữ được các ý chính, đúng ngữ pháp và đảm bảo được tính mạch lạc của văn bản. Với việc áp dụng các kỹ thuật học máy cũng như các mô hình học sâu trong các mô hình tóm tắt văn bản tự động đã cho các bản tóm tắt gần giống với các bản tóm tắt tham chiếu của con người. Trong bài báo này, chúng tôi đề xuất một phương pháp tóm tắt văn bản hướng trích rút hiệu quả sử dụng kết hợp các mô hình học sâu, kỹ thuật học tăng cường và phương pháp MMR để sinh bản tóm tắt. Phương pháp đề xuất của chúng tôi được thử nghiệm trên các bộ dữ liệu CNN (tiếng Anh) và Baomoi (tiếng Việt) cho các kết quả độ chính xác F1-score với Rouge-1, Rouge-2, Rouge-L là 31,36%, 12,84%, 28,33% và 51,95%, 24,38%, 37,56% tương ứng. Các kết quả thử nghiệm cho thấy phương pháp tóm tắt đề xuất của chúng tôi đã đạt các kết quả tốt cho tóm tắt văn bản tiếng Anh và tiếng Việt.

DOI: https://doi.org/10.34238/tnu-jst.4747

[*] Corresponding author. *Email: huonglt@soict.hust.edu.vn*

1. Giới thiệu

Tóm tắt văn bản giúp chúng ta lựa chọn được những thông tin hữu ích, giảm thiểu không gian lưu trữ và thời gian xử lý. Có hai hướng tiếp cận tóm tắt văn bản phổ biến là tóm tắt hướng trích rút thường lựa chọn các câu từ văn bản nguồn, trong khi đó tóm tắt hướng tóm lược thực hiện lựa chọn các từ, các cụm từ trong văn bản nguồn hoặc có thể tạo ra các từ mới, các cụm từ mới để sinh ra bản tóm tắt. Các phương pháp tóm tắt hướng trích rút giai đoạn đầu thường sử dụng kỹ thuật cho điểm câu để lựa chọn tốp các câu có điểm cao nhất đưa vào bản tóm tắt như LEAD [1], LexRank [2], TextRank [3]. Các phương pháp này thường kết hợp với kỹ thuật điều chỉnh trọng số ở mức từ, đây là một trong các yếu tố ảnh hưởng đến chất lượng của bản tóm tắt đầu ra. Gần đây, các kỹ thuật học máy, học sâu được sử dụng để phát triển các hệ thống tóm tắt văn bản hiệu quả như phương pháp độ liên quan cận biên tối đa (MMR) [4] loại bỏ các thông tin dư thừa trong bản tóm tắt. Hệ thống [5] thực hiện trích rút câu sử dụng mạng CNN để sinh bản tóm tắt. Hệ thống [6] coi nhiệm vụ tóm tắt văn bản hướng trích rút là nhiệm vụ gán nhãn câu dựa trên xác suất được chọn của các câu. Hệ thống [7] sử dụng mô hình mạng nơron khép kín (end-to-end) để lựa chọn câu đưa vào bản tóm tắt. Hệ thống [8] coi nhiệm vụ tóm tắt hướng trích rút là bài toán phân loại văn bản và tính toán xác suất được chọn của các câu để sinh bản tóm tắt. Trong khi đó, hệ thống MATCHSUM [9] coi nhiệm vụ tóm tắt hướng trích rút là bài toán so khớp ngữ nghĩa văn bản để sinh bản tóm tắt thay vì trích rút các câu riêng lẻ, nhưng hệ thống này yêu cầu tài nguyên huấn luyện cho mô hình lớn. Bên cạnh đó, các kỹ thuật học tăng cường cũng đã chứng minh được tính hiệu quả trong các hệ thống tóm tắt văn bản. Hệ thống [10] sử dụng điểm ROUGE như một phần của hàm điểm thưởng, kỹ thuật học tăng cường Q-Learning được sử dụng trong [11]. Hệ thống [12] kết hợp kỹ thuật học tăng cường với các kỹ thuật học sâu để xây dựng hệ thống tóm tắt hướng trích rút. Các kỹ thuật học máy và học sâu cũng được sử dụng trong các nghiên cứu về tóm tắt văn bản tiếng Việt như trong [13], [14]. Nghiên cứu trong [13] trích rút câu đưa vào bản tóm tắt sử dụng thuật toán di truyền, trong khi đó hệ thống [14] xây dựng mô hình seq2seq với cơ chế chú ý để sinh bản tóm tắt đầu ra. Nhìn chung, các phương pháp tóm tắt trên chưa quan tâm nhiều đến biểu diễn ngữ cảnh và ngữ nghĩa của từ trong văn bản đầu vào.

Trong các hệ thống tóm tắt, vấn đề mã hóa văn bản đầu vào có vai trò quan trọng quyết định chất lượng của bản tóm tắt nên một số nghiên cứu đã sử dụng các mô hình mã hóa từ được huấn luyện trước như mô hình word2vec [15], GloVe [16], nhưng các mô hình này không biểu diễn được ngôn ngữ theo ngữ cảnh. Gần đây, mô hình BERT (Bidirectional Encoder Representations from Transformers) huấn luyện trước [17] được phát triển để biểu diễn ngôn ngữ theo ngữ cảnh hai chiều đã tạo ra các mô hình hiệu quả cho bài toán tóm tắt văn bản.

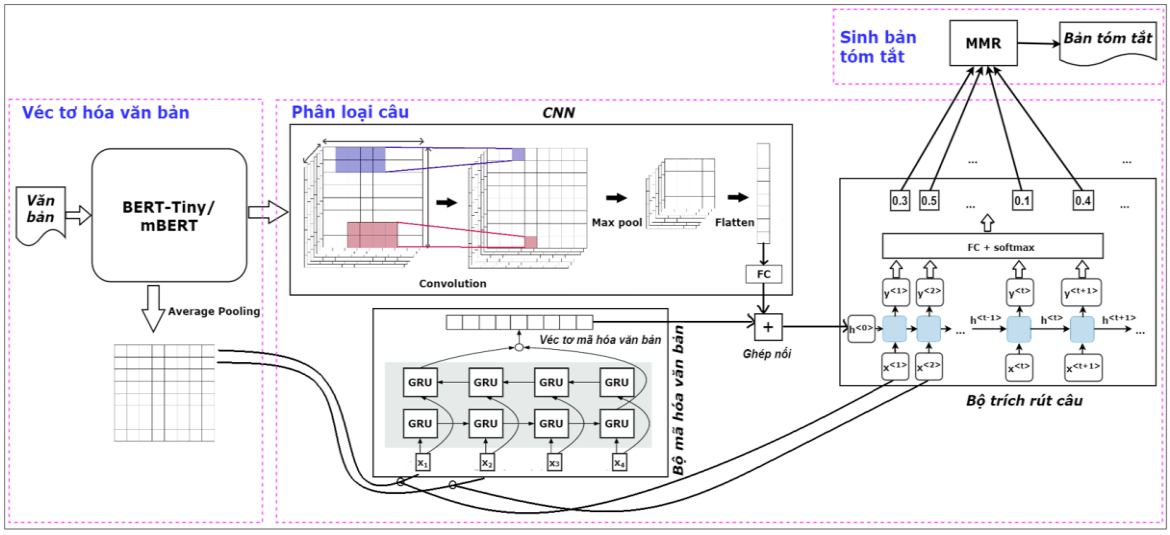
Trong bài báo này, chúng tôi sử dụng hai mô hình của mô hình BERT huấn luyện trước (pretrained BERT), đó là BERT thu gọn (BERT-Tiny) [18], BERT đa ngôn ngữ (mBERT) [19] để mã hóa văn bản tiếng Anh, tiếng Việt tương ứng. Mô hình phân loại câu được xây dựng sử dụng mạng nơron tích chập (CNN), mô hình chuỗi sang chuỗi (seq2seq) với *bộ mã hóa văn bản* sử dụng mạng GRU hai chiều (biGRU) và *bộ trích rút câu* sử dụng mạng GRU một chiều. Bộ trích rút câu được huấn luyện sử dụng kỹ thuật học tăng cường Deep Q-Learning (DeepQL) [20] để tăng hiệu quả cho mô hình tính xác suất được chọn của các câu. Cuối cùng, phương pháp MMR được sử dụng để loại bỏ thông tin dư thừa cho bản tóm tắt. Phương pháp tóm tắt đề xuất được thử nghiệm trên bộ dữ liệu CNN, Baomoi cho tóm tắt tiếng Anh, tiếng Việt tương ứng. Độ đo ROUGE tiêu chuẩn [21] gồm điểm F1-Score trên Rouge-1, Rouge-2 và Rouge-L được sử dụng để đánh giá hiệu quả của các hệ thống tóm tắt trong bài báo. Kết quả thử nghiệm cho thấy phương pháp đề xuất đạt kết quả tốt hơn các hệ thống hiện đại khác trên cùng bộ dữ liệu thử nghiệm.

Phần còn lại của bài báo được bố cục như sau: Phần 2 trình bày phương pháp tóm tắt đề xuất của chúng tôi. Phần 3 trình bày các kết quả thử nghiệm và đánh giá phương pháp đề xuất. Cuối cùng, phần 4 là kết luận và đề xuất hướng phát triển cho nghiên cứu trong tương lai.

2. Phương pháp đề xuất

2.1. Mô hình tóm tắt văn bản đề xuất

Mô hình tóm tắt văn bản đề xuất của chúng tôi gồm 03 mô đun chính: Véctơ hóa văn bản, phân loại câu và sinh bản tóm tắt, được biểu diễn trong Hình 1.



Hình 1. *Mô hình tóm tắt văn bản đề xuất*

2.1.1. Véctơ hóa văn bản

Văn bản đầu vào được xử lý tách câu và lấy 64 câu đầu tiên để biểu diễn cho văn bản. Sau đó, lấy 128 từ đầu tiên để biểu diễn cho mỗi câu (đệm "0" nếu cần). Các câu này được mã hóa sử dụng các mô hình BERT-Tiny (với 2 lớp, 128 chiều, 4 triệu tham số), mBERT (với 12 lớp, 768 chiều, 110 triệu tham số) để thu được các *véctơ mã hóa từ* 128 chiều, 768 chiều cho tiếng Anh, tiếng Việt tương ứng. Các véctơ này được sử dụng làm đầu vào cho mạng CNN để trích rút các đặc trưng văn bản, đồng thời các véctơ mã hóa từ của mỗi câu được xử lý bởi phép toán Average Pooling để sinh ra *véctơ mã hóa câu* 128 chiều, 768 chiều tương ứng, được sử dụng làm đầu vào cho *bộ mã hóa văn bản* và *bộ trích rút câu* trong mô hình seq2seq của mô đun phân loại câu.

2.1.2. Phân loại câu

Chúng tôi coi bài toán tóm tắt văn bản như nhiệm vụ phân loại văn bản. Mục đích của mô đun là tính xác suất được chọn của các câu đưa vào bản tóm tắt. Để thực hiện nhiệm vụ này, mô đun phân loại câu được xây dựng gồm các thành phần chính sau đây.

(a) Mạng CNN: Kiến trúc mạng CNN [22] được sử dụng và hiệu chỉnh cho mô hình đề xuất. Kiến trúc mạng CNN đề xuất gồm 2 lớp tích chập (Convolution) (lớp thứ nhất có 64 bộ lọc, lớp thứ hai có 16 bộ lọc) với *Kernel* kích thước 4x4. Sau mỗi lớp *Convolution* đều có một lớp *Max Pool* để giảm số lượng tham số cho mô hình. Để sinh *đặc trưng* cho xâu đầu vào, chúng tôi sử dụng một cửa sổ trượt trên một phần của câu và trên một vài câu cạnh nhau (được minh họa trong Hình 1). Sau khi trượt trên toàn bộ văn bản sẽ sinh ra một *bản đồ đặc trưng* (feature map). Sau đó, các *feature map* được áp dụng phép toán *Max pool* để giảm chiều, làm phẳng (Flatten), rồi đưa qua lớp mạng nơron kết nối đầy đủ (FC) không có hàm kích hoạt (xem như phép chiếu để giảm chiều) nhận đầu vào là véctơ 256 chiều, 1.024 chiều để thu được một véctơ mã hóa văn bản 64 chiều, 256 chiều cho tiếng Anh, tiếng Việt tương ứng.

(b) Mô hình seq2seq: Mô hình seq2seq [23] gồm bộ mã hóa và bộ giải mã. Kiến trúc mô hình seq2seq của chúng tôi được xây dựng gồm *bộ mã hóa văn bản* và *bộ trích rút câu*. Cả hai thành phần này đều nhận đầu vào là tập gồm *H* véctơ câu (với *H* là số lượng câu lớn nhất của văn bản).

- *Bộ mã hóa văn bản:* Chúng tôi sử dụng mạng biGRU [24] có 256 trạng thái ẩn (bằng 2*128 trạng thái ẩn) cho cả tiếng Anh và tiếng Việt. Đầu vào tại mỗi bước *t* là một véctơ câu 128 chiều, 768 chiều tương ứng cho tiếng Anh, tiếng Việt biểu diễn cho câu x_t. Sau *H* bước thu được 2 véctơ trạng thái nhớ tương ứng của 2 lớp GRU theo chiều tiến và GRU theo chiều lùi (mỗi véctơ có 128 chiều) mã hóa cho văn bản đầu vào. Hai véctơ này được ghép nối với véctơ đầu ra của mạng CNN bởi phép toán *"ghép nối"* (ký hiệu $\boxed{+}$) để thu được véctơ có 320 chiều, 512 chiều cho tiếng Anh, tiếng Việt tương ứng, được sử dụng làm véctơ trạng thái nhớ đầu vào cho bộ trích rút câu để tính xác suất lựa chọn của các câu.
- *Bộ trích rút câu:* Mạng GRU được sử dụng gồm 320 trạng thái ẩn, 512 trạng thái ẩn cho tiếng Anh, tiếng Việt tương ứng, số trạng thái ẩn bằng số chiều của véctơ mã hóa câu sau phép toán ghép nối. Ở mỗi bước *i*, câu đầu vào $x^{<i>}$ được đệm với "0" nếu cần để đảm bảo độ dài câu bằng số trạng thái ẩn của mạng GRU, đầu ra $y^{<i>}$ tương ứng được đưa qua lớp FC (với hàm kích hoạt **softmax)** nhận đầu vào là véctơ 320 chiều, 512 chiều cho tiếng Anh, tiếng Việt tương ứng và đầu ra là véctơ 2 chiều chứa xác suất được chọn của các câu.

2.1.3. Sinh bản tóm tắt

Xác suất được chọn của các câu từ bộ trích rút câu được sắp xếp theo thứ tự giảm dần. Các câu có xác suất cao sẽ được chọn đưa vào tóm tắt cho đến khi đạt độ dài giới hạn của bản tóm tắt. Phương pháp MMR dùng trong tìm kiếm thông tin [4] được định nghĩa lại để áp dụng cho bài toán tóm tắt văn bản nhằm loại bỏ thông tin dư thừa dựa trên độ tương đồng giữa câu đang xét và các câu đã có trong bản tóm tắt. Công thức tính MMR như sau:

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

Với: *C* là tập các câu ứng cử viên để chọn đưa vào bản tóm tắt, *S* là tập các câu đã có trong bản tóm tắt, *Q* là một câu trong tập *C*, D_i, D_j tương ứng là câu đang xét, câu đã có trong bản tóm tắt, λ là siêu tham số ($\lambda \in [0;1]$), Sim_1, Sim_2 là độ tương đồng giữa hai câu *u* và *v* tính theo công thức:

$$Sim_1(u,v) = Sim_2(u,v) = \frac{\sum_{w \in v} tf_{w,u} tf_{w,v} (idf_w)^2}{\sqrt{\sum_{w \in u} (tf_{w,u} idf_w)^2}} \quad (2)$$

Với: $tf_{w,u}$ là tần suất thuật ngữ của từ *w* trong câu *u*; idf_w là độ quan trọng của từ *w*.

2.2. Huấn luyện mô hình với kỹ thuật học tăng cường

Trước hết, mô hình phân loại câu được huấn luyện để trạng thái ẩn đầu vào có đầy đủ các thông tin cần thiết của mô hình. Sau đó, *bộ trích rút câu* được huấn luyện tiếp sử dụng kỹ thuật học tăng cường Deep Q-Learning [20] để tăng tính hiệu quả cho mô hình tính xác suất được chọn của các câu. Các yếu tố quyết định trong học tăng cường là thông tin về trạng thái hiện tại, hành động tương ứng, điểm thưởng và chiến lược học được cài đặt như sau:

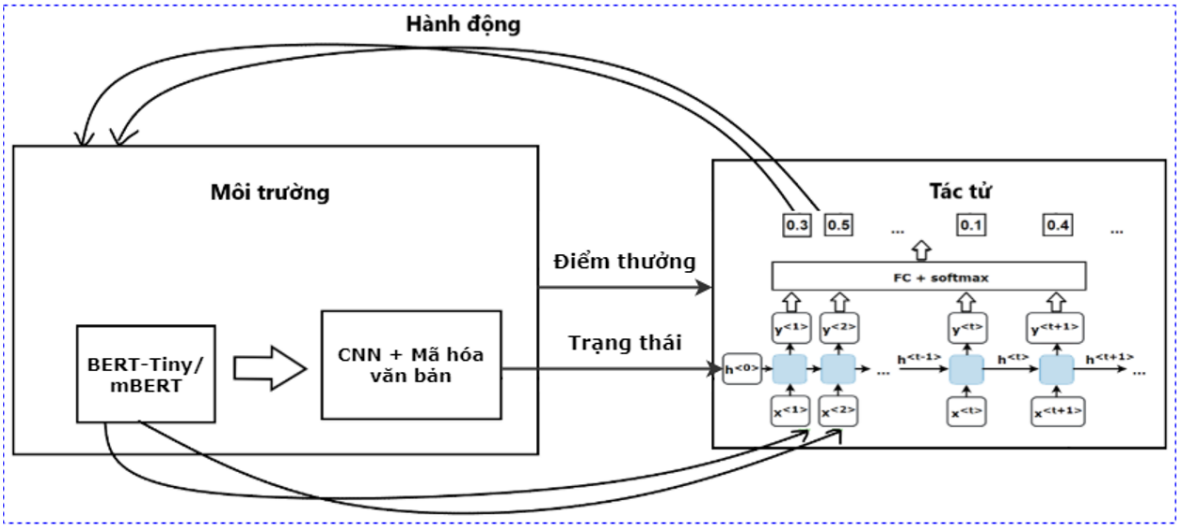
Trạng thái: Mỗi trạng thái s_t biểu diễn cho câu đang xét gồm hai thành phần: h_t là trạng thái ẩn mã hóa cho các trạng thái trước đó, được tạo bởi mạng GRU của tác tử đang tương tác với môi trường; x_t là mã hóa trạng thái đang xét, là véctơ câu đầu ra sau phép toán Average Pooling.

Hành động: Có 2 hành động tương ứng dựa trên xác suất đầu ra của lớp FC đối với một trạng thái: "1" - chọn câu đang xét, "0" - không chọn câu đang xét.

Điểm thưởng: Ở trạng thái *t*, nếu không chọn câu đang xét $sent_t$ thì nhận điểm thưởng bằng **0**. Nếu chọn câu đang xét $sent_t$ thì sẽ nhận điểm thưởng R_t được tính theo công thức: $R_t = Rouge\_L(sent_t, D) - \delta$ (bằng điểm *Rouge-L* của câu đang xét $sent_t$ so với bản tóm tắt hiện có *D* trừ đi giá trị δ để tránh chọn các câu quá khác biệt so với bản tóm tắt hiện có).

Chiến lược: Ở trạng thái s_t, tác tử thực hiện một hành động để chuyển đến trạng thái s_{t+1}, nhận điểm thưởng R_t từ môi trường và mục tiêu là tìm chiến lược có tổng điểm thưởng lớn nhất.

Mô hình huấn luyện đề xuất với kỹ thuật học tăng cường Deep Q-Learning của chúng tôi được thiết lập như Hình 2.



Hình 2. *Mô hình huấn luyện với kỹ thuật học tăng cường Deep Q-Learning*

3. Thử nghiệm và đánh giá

3.1. Dữ liệu thử nghiệm

Phương pháp đề xuất được thử nghiệm trên hai bộ dữ liệu: CNN của bộ dữ liệu CNN/Daily Mail [25] cho tiếng Anh và Baomoi cho tiếng Việt. Bộ dữ liệu CNN/Daily Mail gồm 312.085 bài báo tin tức (bộ dữ liệu CNN có 92.579 bài báo) và các câu nổi bật đi kèm trong mỗi bài báo được sử dụng để đánh giá độ chính xác của bản tóm tắt đầu ra. Số câu nổi bật trung bình xấp xỉ 3 nên bản tóm tắt cũng chọn 3 câu cho tương ứng. Bộ dữ liệu **Baomoi** được thu thập từ các bài báo tin tức của trang báo điện tử Việt Nam *(http://baomoi.com)* gồm 1.000.847 bài báo tin tức. Mỗi bài báo gồm 3 phần: tiêu đề, tóm tắt và nội dung. Phần tóm tắt có trung bình xấp xỉ 2 câu, được sử dụng làm cơ sở để sinh bản tóm tắt gồm 2 câu và đánh giá độ chính xác của bản tóm tắt đầu ra.

3.2. Tiền xử lý dữ liệu

Trước hết, các bộ dữ liệu CNN, Baomoi được xử lý tách phần nội dung, tóm tắt và đánh số thứ tự cho các câu. Các thư viện **StanfordNLP**[3], **VnCoreNLP**[4] được sử dụng để tách câu của văn bản cho bộ dữ liệu CNN, Baomoi tương ứng. Tiếp theo, các câu được gán nhãn dựa trên tối đa tổng của R-2 và R-L sử dụng thư viện **Rouge-score 0.0.4**[5]. Sau đó, các câu này được đưa vào mô hình BERT-Tiny, mBERT tương ứng để thu được các véctơ mã hóa từ của các câu. Đồng thời, các véctơ mã hóa từ của mỗi câu được xử lý sử dụng thư viện **PyTorch**[6] để được véctơ mã hóa câu 128 chiều, 768 chiều cho tiếng Anh, tiếng Việt tương ứng.

3.3. Thiết kế thử nghiệm

Trước hết, chúng tôi thực hiện thử nghiệm một số phương pháp cơ bản trên hai bộ dữ liệu CNN và Baomoi. Các độ đo Rouge-1 (R-1), Rouge-2 (R-2) và Rouge-L (R-L) tính dựa trên thư viện **Rouge-score 0.0.4** được sử dụng để đánh giá độ chính xác của các phương pháp tóm tắt thử nghiệm. R-1, R-2 là tỉ lệ % số 1-gram, 2-gram chung giữa bản tóm tắt của hệ thống và bản tóm

[3] https://stanfordnlp.github.io/CoreNLP/
[4] https://github.com/vncorenlp/VnCoreNLP/
[5] https://github.com/google-research/google-research/tree/master/rouge/
[6] https://github.com/pytorch/pytorch/

tắt tham chiếu, còn R-L là tỉ lệ % dãy con chung dài nhất giữa hai bản tóm tắt đó. Các kết quả thử nghiệm được trình bày như trong Bảng 1.

Bảng 1. *Kết quả thử nghiệm một số phương pháp cơ bản*

Phương pháp	CNN			Baomoi		
	R-1	R-2	R-L	R-1	R-2	R-L
LexRank	22,9	6,6	17,2	38,5	17,0	28,9
TextRank	26,0	7,3	19,2	44,7	19,2	32,9
LEAD	29,0	10,7	19,3	46,5	20,3	30,8

Tiếp theo, chúng tôi triển khai thử nghiệm bốn mô hình kịch bản trên hai bộ dữ liệu CNN và Baomoi để lựa chọn mô hình hiệu quả nhất cho phương pháp đề xuất. Các kịch bản mô hình thử nghiệm được trình bày sau đây.

(i) Kịch bản 1 (BERT-Tiny/mBERT + CNN + seq2seq): Sử dụng mô hình BERT-Tiny (đối với CNN), mBERT (đối với Baomoi) kết hợp với mạng CNN và mạng seq2seq để huấn luyện mô hình tính xác suất được chọn của các câu đưa vào bản tóm tắt.

(ii) Kịch bản 2 (BERT-Tiny/mBERT + CNN + seq2seq + MMR): Mô hình kịch bản 1 kết hợp với phương pháp MMR để lựa chọn câu đưa vào bản tóm tắt.

(iii) Kịch bản 3 (BERT-Tiny/mBERT + CNN + seq2seq + DeepQL): Mô hình kịch bản 1 kết hợp với kỹ thuật học tăng cường Deep Q-Learning để huấn luyện bộ trích rút câu để lựa chọn câu đưa vào bản tóm tắt.

(iv) Kịch bản 4 (BERT-Tiny/mBERT + CNN + seq2seq + DeepQL + MMR): Mô hình kịch bản 3 kết hợp với phương pháp MMR để lựa chọn câu đưa vào bản tóm tắt.

Chúng tôi sử dụng thư viện **Transformers**[7] để kế thừa các mô hình BERT-Tiny, mBERT và thư viện PyTorch để xây dựng mô hình phân loại câu. Các mô hình kịch bản được huấn luyện sử dụng Google Colab với cấu hình máy chủ GPU V100, 25GB RAM được cung cấp bởi Google Research. Kết quả thử nghiệm của các mô hình kịch bản thu được như trong Bảng 2.

Bảng 2. *Kết quả thử nghiệm của các mô hình kịch bản*

1	CNN			Baomoi		
	R-1	R-2	R-L	R-1	R-2	R-L
BERT-Tiny/mBERT + CNN + seq2seq	29,55	11,67	27,12	51,17	23,83	36,54
BERT-Tiny/mBERT + CNN + seq2seq + MMR	30,09	11,95	27,80	51,41	24,01	36,92
BERT-Tiny/mBERT + CNN + seq2seq + DeepQL	30,49	12,22	27,89	51,73	24,10	37,18
BERT-Tiny/mBERT + CNN + seq2seq + DeepQL + MMR	**31,36**	**12,84**	**28,33**	**51,95**	**24,38**	**37,56**

Với các kết quả thử nghiệm trong Bảng 2, mặc dù mô hình trong kịch bản 1 chưa xử lý loại bỏ các thông tin trùng lặp nhưng đã cho kết quả khả quan và tốt hơn các phương pháp như LexRank, TextRank, LEAD (Bảng 1) trên cả hai bộ dữ liệu CNN và Baomoi. Trong mô hình kịch bản 2, phương pháp MMR được sử dụng để loại bỏ các thông tin trùng lặp đã cho kết quả tốt hơn mô hình kịch bản 1. Mô hình trong kịch bản 3 mặc dù chưa xử lý loại bỏ các thông tin trùng lặp nhưng việc kết hợp kỹ thuật học tăng cường Deep Q-Learning đã cho kết quả tốt hơn so với mô hình kịch bản 1 và tốt hơn cả mô hình kịch bản 2. Với việc sử dụng phương pháp MMR, mô hình trong kịch bản 4 đã cho các kết quả tốt hơn rõ rệt so với mô hình kịch bản 3 trên cả hai bộ dữ liệu CNN và Baomoi nên mô hình trong kịch bản 4 được lựa chọn cho phương pháp tóm tắt đề xuất.

3.4. So sánh và đánh giá kết quả

Chúng tôi so sánh kết quả thử nghiệm của phương pháp tóm tắt đề xuất với kết quả thử nghiệm của các hệ thống mà chúng tôi đã thử nghiệm và các hệ thống hiện đại khác đã công bố

[7] https://huggingface.co/transformers/

trên cùng bộ dữ liệu thử nghiệm. Kết quả so sánh và đánh giá được trình bày như trong Bảng 3 *(ký hiệu '*', '-' biểu diễn hệ thống mà chúng tôi đã thử nghiệm, hệ thống không được thử nghiệm trên các bộ dữ liệu tương ứng)*.

Bảng 3. *So sánh và đánh giá kết quả của các phương pháp*

Phương pháp	CNN			Baomoi		
	R-1	R-2	R-L	R-1	R-2	R-L
LexRank*	22,9	6,6	17,2	38,5	17,0	28,9
TextRank*	26,0	7,3	19,2	44,7	19,2	32,9
LEAD*	29,0	10,7	19,3	46,5	20,3	30,8
Cheng và Lapata (2016) [12]	28,4	10,0	25,0	-	-	-
REFRESH [12]	30,4	11,7	26,9	-	-	-
BERT-Tiny/mBERT + CNN + seq2seq + DeepQL + MMR *(phương pháp đề xuất)*	**31,36**	**12,84**	**28,33**	**51,95**	**24,38**	**37,56**

Kết quả trong Bảng 3 cho thấy, phương pháp tóm tắt sử dụng mô hình BERT-Tiny/mBERT, CNN, seq2seq, kỹ thuật học tăng cường và phương pháp MMR cho kết quả tốt hơn đáng kể so với các hệ thống hiện đại khác trên hai bộ dữ liệu CNN và Baomoi tương ứng. Điều này chứng tỏ phương pháp tóm tắt đề xuất đã đạt hiệu quả tốt cho tóm tắt văn bản tiếng Anh và tiếng Việt.

4. Kết luận và hướng phát triển

Trong nghiên cứu này, chúng tôi đã đề xuất một phương pháp tóm tắt văn bản hướng trích rút sử dụng các mô hình học sâu kết hợp với kỹ thuật học tăng cường và phương pháp MMR để sinh bản tóm tắt đầu ra. Mô hình được huấn luyện trên toàn bộ văn bản bằng cách tối ưu hóa điểm ROUGE. Phương pháp đề xuất đã cho kết quả thử nghiệm tốt hơn các hệ thống hiện đại khác trên cùng bộ dữ liệu thử nghiệm. Trong phương pháp đề xuất, văn bản được mã hóa sử dụng các mô hình pretrained BERT bị giới hạn về độ dài. Trong tương lai, chúng tôi nghiên cứu áp dụng mô hình GPT (Generative Pre-Training) [26] để cải thiện chất lượng của bản tóm tắt đầu ra nhằm nâng cao hiệu quả cho phương pháp đề xuất.

Lời cám ơn

Nghiên cứu này được tài trợ bởi Trường Đại học Bách khoa Hà Nội (HUST) trong khuôn khổ đề tài mã số T2020-PC-208.

TÀI LIỆU THAM KHẢO/ REFERENCES

[1] M. Wasson, "Using leading text for news summaries: Evaluation results and implications for commercial summarization applications," *Proceedings of COLING 1998 vol. 2: The 17th International Conference on Computational Linguistics*, 1998, pp. 1364-1368.

[2] G. Erkan and D. R. Radev, "LexRank: Graph-based Lexical Centrality as Salience in Text Summarization," *Journal of Artificial Intelligence Research*, vol. 22, pp. 457-479, 2004.

[3] R. Mihalcea and P. Tarau, "TextRank: Bringing Order into Texts," *Proceedings of the 2004 Conference on Empirical Methods in Natural Language Processing*, 2004, pp. 404-411.

[4] J. Carbonell and J. Goldstein, "The Use of MMR, Diversity-Based Reranking for Reordering Documents and Producing Summaries," *Proceedings of the 21st annual international ACM SIGIR conference on Research and development in information retrieval*, 1998, pp. 335-336.

[5] Y. Zhang, J. E. Meng, and M. Pratama, "Extractive Document Summarization Based on Convolutional Neural Networks," *In IECON 2016 - 42nd Annual Conference of the IEEE Industrial Electronics Society*, 2016, pp. 918-922.

[6] J. Cheng and M. Lapata, "Neural summarization by extracting sentences and words," *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics*, vol. 1, 2016, pp. 484-494.

[7] Q. Zhou, N. Yang, F. Wei, S. Huang, M. Zhou, and T. Zhao, "Neural Document Summarization by Jointly Learning to Score and Select Sentences," *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics*, vol. 1, 2018, pp. 654-663.

[8] K. Al-Sabahi, Z. Zuping, and M. Nadher, "A Hierarchical Structured Self-Attentive Model for Extractive Document Summarization (HSSAS)," *IEEE Access*, vol. 6, pp. 24205-24212, 2018.

[9] M. Zhong, P. Liu, Y. Chen, D. Wang, X. Qiu, and X. Huang, "Extractive Summarization as Text Matching," *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, 2020, pp. 6197-6208.

[10] C. Rioux, S. A. Hasan, and Y. Chali, "Fear the REAPER: A system for automatic multidocument summarization with reinforcement learning," *Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, 2014, pp. 681-690.

[11] S. Hen, M. Mieskes, and I. Gurevych, "A reinforcement learning approach for adaptive single and multi-document summarization," *Proceedings of International Conference of the German Society for Computational Linguistics and Language Technology*, 2015, pp. 3-12.

[12] S. Narayan, S. B. Cohen, and M. Lapata, "Ranking Sentences for Extractive Summarization with Reinforcement Learning," *Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, vol. 1, 2018, pp. 1747-1759.

[13] Q. U. Nguyen, T. A. Pham, C. D. Truong, and X. H. Nguyen, "A Study on the Use of Genetic Programming for Automatic Text Summarization," *Proceedings of 2012 Fourth International Conference on Knowledge and Systems Engineering*, 2012, pp. 93-98.

[14] Q. T. Lam, T. P. Pham, and D. H. Do, "Automatic Vietnamese Text Summarization with Model Sequence-to-sequence," (in Vietnamese), *Scientific Journal of Can Tho University*, Special topic: Information Technology, pp. 125-132, 2017.

[15] T. Mikolov, I. Sutskever, K. Chen, G. Corrado, and J. Dean, "Distributed representations of words and phrases and their compositionality," *Proceedings of the 26th International Conference on Neural Information Processing Systems*, vol. 2, 2013, pp. 3111-3119.

[16] J. Pennington, R. Socher, and C. D. Manning, "Glove: Global vectors for word representation," *Proceedings of the 2014 Conference on EMNLP*, 2014, pp. 1532-1543.

[17] J. Devlin, M. W. Chang, K. Lee, and K. Toutanova, "Bert: Pre-training of deep bidirectional transformers for language understanding," *Proceedings of NAACL-HLT 2019*, 2019, pp. 4171-4186.

[18] I. Turc, M. W. Chang, K. Lee, and K. Toutanova, "Well-Read Students Learn Better: On the Importance of Pre-training Compact Models," arXiv:1908.08962 [cs.CL], 2019.

[19] T. Pires, E. Schlinger, and D. Garrette, "How multilingual is Multilingual BERT?," *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, 2019, pp. 4996-5001.

[20] V. Mnih, K. Kavukcuoglu, D. Silver, A. Graves, I. Antonoglou, D. Wierstra, and M. R. Miller, "Playing Atari with Deep Reinforcement Learning," arXiv:1312.5602v1 [cs.LG], 2013.

[21] C. Y. Lin, "Rouge: A package for automatic evaluation of summaries," 2004. [Online]. Available: https://aclanthology.org/W04-1013.pdf. [Accessed July 11, 2021].

[22] Y. Kim, "Convolutional neural networks for sentence classification," *Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, 2014, pp. 1746-1751.

[23] I. Sutskever, O. Vinyals, and Q. V. Le, "Sequence to Sequence Learning with Neural Networks," *Proceedings of the 27th International Conference on Neural Information Processing Systems*, vol. 2, 2014, pp. 3104-3112.

[24] K. Cho, B. V. Merrienboer, C. Gulcehre, D. Bahdanau, F. Bougares, H. Schwenk, and Y. Bengio, "Learning Phrase Representations using RNN Encoder-Decoder for Statistical Machine Translation," *Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, 2014, pp. 1724-1734.

[25] K. M. Hermann, T. Kocisky, E. Grefenstette, L. Espeholt, W. Kay, M. Suleyman, and P. Blunsom, "Teaching machines to read and comprehend," *Proceedings of the 28th International Conference on Neural Information Processing Systems*, vol. 1, 2015, pp. 1693-1701.

[26] A. Radford, K. Narasimhan, T. Salimans, and I. Sutskever, "Improving Language Understanding by Generative Pre-Training," 2018. [Online]. Available: https://s3-us-west-2.amazonaws.com/openai-assets/research-covers/language-unsupervised/language_understanding_paper.pdf. [Accessed April 23, 2021].