Xác định thứ tự thời gian giữa hai câu tiếng Việt chỉ quá trình để tóm lược

Determining The Temporal Order Between Two Vietnamese Process Sentences for Summarizing

Trần Trung, Nguyễn Tuấn Đăng

Abstract: *In this paper we introduce a method for summarizing the meaning of two continual Vietnamese sentences manifesting a sequence of processes which belongs to one of three process types (according to Functional Grammar [26, 41]): the state of subject is changed, the position of subject is changed, and the state or position of the subject is affected by an agent. The sentence-generation method is performed in two main processes: (i) resolve anaphoric pronoun and represent the semantics of the source pair of sentences; (ii) determine the ordinal relationship of processes and generate new reduced Vietnamese sentence. To evaluate the quality of summarization, we compare our generated sentences with sentence fusions which generated using K. Filippova [31]'s method as well as an enhancement by F. Boudin and E. Morin [16]. Using ROUGE measures [6 - 9], the results show that our method's summaries are more precise and natural in overall.*

Keywords: *sentence generation, summarization, semantic representation.*

I. GIỚI THIỆU

Khởi đầu từ năm 1958 bằng những hoạt động tiên phong của H. P. Luhn [20] và P. Baxendale [44], vấn đề mà K. S. Jones định nghĩa là việc thực hiện "*một tiến trình biến đổi rút gọn một văn bản nguồn thành một văn bản tóm lược bằng cách lựa chọn và / hoặc tổng quát hóa những gì là quan trọng trong văn bản nguồn*" [35, 36] hay còn được gọi ngắn gọn là "tóm lược văn bản" đã trở thành một lĩnh vực nghiên cứu quan trọng trong cộng đồng Xử lý ngôn ngữ tự nhiên trong suốt hơn nửa thế kỷ qua. Trong số những nghiên cứu đầu tiên nhằm mục tiêu tóm lược các văn bản khoa học, H. P. Luhn [20] đã đề xuất phương pháp xếp hạng và trích xuất câu từ văn bản nguồn dựa trên mức độ xuất hiện thường xuyên của các từ vựng và ngữ đoạn. Với ý tưởng tương tự, P. Baxendale [44] đã đề xuất ý tưởng trích xuất dựa trên vị trí trong đoạn văn bản. Đáng chú ý nhất là nghiên cứu của H. P. Edmunson [21] vào năm 1969 đã đề xuất giả thiết xem xét giá trị thông tin cao của những ngữ đoạn tiêu đề, những câu đầu và cuối của văn bản.

Về cơ bản, K. S. Jones đã đề xuất một ý tưởng dựa trên việc thực hiện ba tiến trình liên tiếp để chuyển đổi một văn bản nguồn thành một văn bản tóm lược [35, 36]:

- Tiến trình thứ nhất: thực hiện mô tả văn bản đầu vào bởi một dạng biểu diễn thứ nhất.
- Tiến trình thứ hai: thực hiện chuyển đổi dạng biểu diễn thứ nhất sang dạng biểu diễn thứ hai là một mô tả của văn bản tóm lược.
- Tiến trình thứ ba: thực hiện tạo sinh ngôn ngữ và hoàn chỉnh văn bản tóm lược từ dạng biểu diễn thứ hai.

Từ những năm cuối thế kỷ XX và đầu thế kỷ XXI, ý tưởng của K. S. Jones [35, 36] đã được nhiều nhóm nghiên cứu triển khai để đề xuất những phương pháp khác nhau nhằm nâng cao hiệu quả trong việc chuyển đổi một văn bản nguồn thành một văn bản tóm lược [5, 10, 12, 13, 28, 29, 34-36, 40]. Các phương pháp được đề xuất được phân loại theo hai hướng nghiên cứu chính [5, 10]: (i) hướng thứ nhất được gọi là "tóm

lược trích xuất" – "extractive summarization"; (ii) hướng thứ hai được gọi là "tóm lược trừu tượng" – "abstractive summarization".

Trong hướng tiếp cận "extractive summarization", từng câu trong văn bản ban đầu sẽ được tính toán để xác định mức độ quan trọng của nó trong văn bản bằng các phương pháp máy học thống kê [5, 10, 12, 13, 23-25, 28, 29, 34-36, 40, 65]. Những đặc điểm thường được sử dụng để tính toán mức độ quan trọng của câu là từ khóa, tiêu đề, vị trí hoặc độ dài của câu, những ngữ đoạn đặc thù. Từ đó, những câu hay ngữ đoạn được cho là quan trọng nhất là những câu có điểm tính toán cao hơn ngưỡng sẽ được chọn để tạo thành văn bản tóm lược.

Mặc dù có nhiều giải pháp được đề xuất và đạt được những kết quả quan trọng, một số vấn đề cơ bản của hướng tiếp cận "extractive summarization" vẫn đang được các nhà khoa học nghiên cứu để khắc phục [5, 10, 12, 13, 23-25, 28, 29, 34-36, 40, 65]:

- Do những thông tin liền mạch được thể hiện xuyên suốt thông qua các câu trong văn bản nguồn nên việc trích xuất các câu quan trọng nhưng không liên tiếp có thể khiến văn bản tóm lược mất đi sự liền mạch này.
- Nhiều câu trong văn bản nguồn có sự xuất hiện của đại từ hồi chỉ. Việc trích xuất sẽ khiến mối liên hệ giữa đại từ và đối tượng tiền ngữ sẽ bị mất đi, và ngữ cảnh thực sự của văn bản ban đầu sẽ không được thể hiện chính xác.

Trong hướng tiếp cận "abstractive summarization", những vấn đề quan trọng cần giải quyết là đề xuất được những cơ chế để hiểu và biểu diễn được ý nghĩa của văn bản nguồn cũng như tạo sinh được văn bản tóm lược. Để thực hiện những điều này, những nghiên cứu theo hướng tiếp cận này cần phải có sự kết hợp những kỹ thuật và kiến thức thuộc các lĩnh vực về khoa học máy tính là hiểu văn bản và tạo sinh văn bản cũng như các lý thuyết ngôn ngữ học. Trong những năm gần đây, hướng tiếp cận dựa trên "abstractive summarization" bắt đầu được chú ý nhiều hơn với một số phương pháp được đề xuất [1, 5, 42]: các phương pháp dựa trên tiếp cận cấu trúc "structure-based" như phương pháp cây phụ thuộc [50, 51] hay các phương pháp trích xuất thông tin [48]; các phương pháp dựa trên tiếp cận ngữ nghĩa như phương pháp biểu diễn ngữ nghĩa theo những "Information Item" [46] hay đồ thị ngữ nghĩa [27]. Một số vấn đề được đặt ra là những phương pháp này được đề xuất chủ yếu nhằm tóm lược đa văn bản và cũng chưa có sự kết hợp với các lý thuyết ngôn ngữ học. (Xem [1, 5, 42]).

Một hướng tiếp cận hẹp mới được tập trung nghiên cứu trong những năm gần đây dựa trên "abstractive summarization" là tạo thành một câu nhiều thông tin bằng việc kết hợp nhiều câu khác nhau và được gọi là tiếp cận trộn câu "sentence fusion". Tiếp cận trộn câu cho phép tạo ra một câu mới từ sự gom nhóm những thông tin có trong những câu nguồn khác nhau và có thể được cải tiến theo nhiều cách. Hướng tiếp cận trộn câu được khởi đầu bởi R. Barzilay và K. R. McKeown [51] bằng việc phát triển một hệ thống tóm lược đa văn bản thực thi theo hai quá trình chính: (i) trong quá trình thứ nhất, nhiều phương pháp máy học khác nhau có thể được áp dụng để gom cụm các câu có cùng chủ đề; (ii) trong quá trình thứ hai, hệ thống trộn các cây phụ thuộc của các câu trong từng cụm và tạo sinh các câu mới rồi lựa chọn kết quả trộn tốt nhất. Dựa trên cùng ý tưởng sử dụng cấu trúc cây phụ thuộc, K. Filippova và M. Strube [32, 33] đề xuất phương pháp cải tiến để tạo sinh các câu mới đúng ngữ pháp hơn bằng cách "trộn hợp nhất" ("union fusion") thay vì chỉ trộn giao nhau "intersection fusion" như của R. Barzilay và K. R. McKeown [51]. Một nghiên cứu khác của K. Filippova [31] kết hợp trộn câu và nén câu "sentence compression", trong đó tác giả sử dụng một đồ thị từ vựng của các câu được trộn và lựa chọn đường đi trong đồ thị chứa đựng những thông tin chung để tạo câu mới. Phương pháp này của K. Flippova [31] được tiếp tục cải tiến bởi F. Boudin và E. Morin [16] để tạo ra những câu có chứa nhiều thông tin hơn bằng cách đánh giá lại dựa theo những cụm từ khóa. (Xem [1, 5, 16, 31-33, 42, 51]).

Theo hướng tiếp cận dựa trên "abstractive summarization" và thực hiện ba tiến trình bên trên,

chúng tôi đặt ra vấn đề tổng quát là xây dựng một mô hình biểu diễn nội dung ngữ nghĩa của toàn bộ văn bản nguồn và đề xuất một phương pháp để tạo sinh ra một đoạn văn bản mới ngắn gọn nhất có thể để tóm lược nội dung của văn bản nguồn đã được mô hình hóa. Để giải quyết vấn đề tổng quát này và thực hiện kết hợp với ý tưởng trong lĩnh vực tạo sinh ngôn ngữ tự nhiên [15], trong những nghiên cứu gần đây [59 - 62], chúng tôi đã đề xuất một số giải pháp, kỹ thuật nhằm tóm lược những dạng cặp câu tiếng Việt đơn giản có đặc điểm khác nhau.

Ở giai đoạn biểu diễn nội dung ngữ nghĩa của văn bản nguồn, trong công trình [59] và nghiên cứu này, ngữ nghĩa của một cặp câu tiếng Việt sẽ được biểu diễn bởi một cấu trúc Discourse Representation Structure (DRS). Theo lý thuyết Discourse Representation Theory [19, 38, 39, 45], DRS là một cấu trúc biểu diễn cho biết hai dạng thông tin: (i) thông tin về những đối tượng – biểu thị bởi những danh từ – xuất hiện trong đoạn văn bản; (ii) thông tin về những thuộc tính – biểu thị bởi những danh từ, động từ hay tính từ – mà những đối tượng này có và sự tương quan giữa chúng. DRS lưu trữ hai dạng thông tin này dưới dạng một cặp danh sách hữu hạn <U, Con>: danh sách U chứa những chỉ số riêng biệt cho biết từng đối tượng và danh sách Con chứa những vị từ (là những thuộc tính hay còn được gọi là điều kiện) gắn với những chỉ số này.

Ở giai đoạn thực hiện tạo sinh đoạn văn bản mới, để tóm lược nội dung của văn bản nguồn đã được mô hình hóa bởi cấu trúc DRS, cách tiếp cận hiện tại của chúng tôi là: chúng tôi giả sử rằng sẽ tóm lược từng cặp câu liên tiếp có liên quan, nếu câu không có liên quan thì không tóm lược. Quá trình tóm lược sẽ diễn ra theo nhiều bước, ở nhiều cấp (sau mỗi bước là một cấp tóm lược), cho đến khi không còn cặp câu nào có thể tóm lược được nữa. Trong [59], áp dụng cho những đoạn văn bản gồm hai câu tiếng Việt đơn giản, chúng tôi xác định hai câu được cho là có liên quan nếu có mối quan hệ đại từ hồi chỉ liên câu. Dựa trên mối quan hệ này, chúng tôi thực hiện phân tích cấu trúc DRS và tạo sinh cấu trúc cú pháp của câu tiếng Việt rút gọn mới. Cuối cùng, những thành phần trong cấu trúc cú pháp sẽ được thay thế bởi bộ từ vựng tiếng Việt phù hợp để hoàn chỉnh câu tiếng Việt tóm lược.

Tiếp tục phát triển hướng tiếp cận, để nâng cao chất lượng của câu tiếng Việt được tạo sinh, trong [60 - 62] chúng tôi xem xét thêm các mối quan hệ liên câu giữa cặp câu tiếng Việt ban đầu: mối quan hệ về thứ tự xem xét giữa hành động ở câu thứ nhất với hành động ở câu thứ hai. Dựa trên những mối quan hệ này, chúng tôi thực hiện một số cải tiến so với [59] nhằm: (i) tạo dựng cấu trúc DRS để mô hình hóa cụ thể hơn ngữ nghĩa của những cặp câu tiếng Việt được xem xét có đặc điểm phù hợp; và (ii) tạo sinh câu tiếng Việt rút gọn mới có chất lượng tốt hơn.

Một vấn đề quan trọng trong cách tiếp cận của chúng tôi khi thực hiện tóm lược các cặp câu tiếng Việt là làm sao xác định chính xác đối tượng tiền ngữ cho đại từ hồi chỉ xuất hiện ở câu thứ hai trong những ngữ cảnh có sự nhập nhằng. Để giải quyết vấn đề này và áp dụng cho một số dạng cặp câu tiếng Việt có cấu trúc đặc biệt, trong [63, 64], chúng tôi đề xuất những chiến lược nhằm xử lý chính xác hơn đại từ "nó" và những đại từ chỉ người. Chúng tôi cũng kết hợp áp dụng cấu trúc mệnh đề quan hệ trong ngữ pháp tiếng Việt để tạo sinh câu tiếng Việt rút gọn mới thỏa mãn yêu cầu đặt ra.

Trong nghiên cứu này, chúng tôi tập trung áp dụng phương pháp tạo sinh câu để tóm lược ý nghĩa một số dạng đoạn văn bản bao gồm hai câu tiếng Việt chỉ quá trình. Theo lý thuyết Functional Grammar [26, 41], một quá trình là một chuỗi biến cố trong đó chủ thể, thông thường là một tĩnh vật, phải trải qua một cách không tự nguyện. Để tóm lược nghĩa của những đoạn văn bản bao gồm những câu chỉ quá trình, chúng ta phải trả lời hai câu hỏi: (i) Chủ thể nào trải qua các quá trình?; và (ii) Thứ tự thời gian xảy ra các quá trình?

Đối tượng nghiên cứu chính của chúng tôi trong bài báo là những cặp câu tiếng Việt, được xem như những đoạn văn bản đơn giản nhất, trong đó có một chủ thể là tĩnh vật trải qua hai quá trình: một quá trình

được diễn đạt bởi động từ ở câu thứ nhất, và một quá trình được diễn đạt bởi động từ ở câu thứ hai.

Dựa trên sự phân loại động từ chỉ quá trình trong các lý thuyết Functional Grammar [26, 41], chúng tôi xử lý ba dạng câu chỉ quá trình:

- **Dạng 1**: quá trình trong đó chủ thể bị thay đổi trạng thái.

Ví dụ 1: "*Cái bình bị nứt.*"

- **Dạng 2**: quá trình trong đó chủ thể bị thay đổi vị trí.

Ví dụ 2: "*Chiếc lá rụng.*"

- **Dạng 3**: quá trình trong đó chủ thể bị một tác động bởi một tác nhân, khiến cho nó bị thay đổi trạng thái hoặc vị trí.

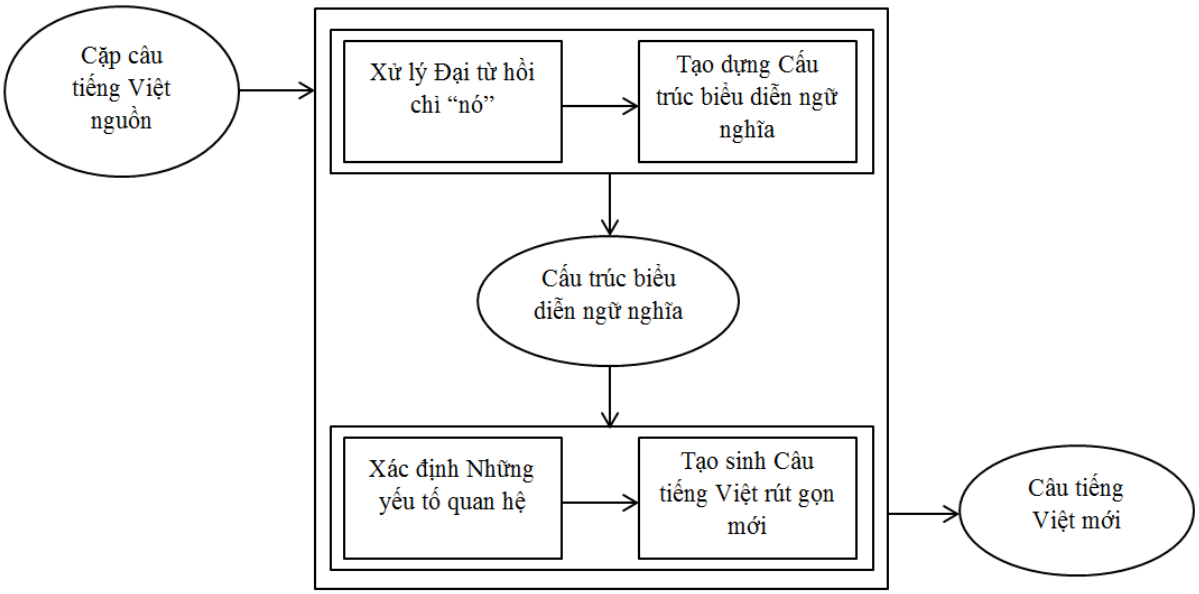
Ví dụ 3: "*Sét đánh cành cây.*"

Chúng tôi giả thiết rằng có một thứ tự thời gian để xảy ra các quá trình: quá trình dạng 3 xảy ra trước tiên, quá trình dạng 2 xảy ra tiếp theo, quá trình dạng 1 xảy ra sau cùng. Việc xác định quan hệ thứ tự thời gian giữa quá trình ở câu thứ nhất với quá trình ở câu thứ hai sẽ là tiền đề để tóm lược ý nghĩa của đoạn văn bản. Cùng với đó, một yêu cầu quan trọng cũng được đặt ra: câu tiếng Việt được tạo sinh phải mang tính phổ dụng trong giao tiếp thông thường.

Kiến trúc tổng quát của phương pháp tạo sinh câu được minh họa trong Hình 1.

Kiến trúc tổng quát này bao gồm những giai đoạn chính sau:

- **Giai đoạn 1**: Xử lý đại từ hồi chỉ "nó". Trong tiếng Việt, đại từ "nó" tùy ngữ cảnh có thể chỉ người, động vật hoặc tĩnh vật. Với mục tiêu của nghiên cứu này, chúng tôi xác định tiền ngữ của đại từ "nó" là một đối tượng tĩnh vật.
- **Giai đoạn 2**: Tạo dựng một cấu trúc biểu diễn ngữ nghĩa của cặp câu tiếng Việt nguồn.
- **Giai đoạn 3**: Xác định những yếu tố quan hệ: chủ thể của các quá trình, hiện tượng tác động lên chủ thể, thứ tự thời gian xảy ra các quá trình. Việc xác định được thực hiện thông qua phân tích cấu trúc biểu diễn trên.
- **Giai đoạn 4**: Tạo sinh câu tiếng Việt rút gọn mới. Chúng tôi kết hợp từ vựng thuộc cặp câu nguồn và từ vựng thể hiện mối quan hệ dựa trên thứ tự thời gian xảy ra các quá trình.



Hình 1. Kiến trúc tổng quát của phương pháp tạo sinh câu với các giai đoạn thực hiện chính

Cấu trúc của bài báo như sau: trong Phần II, chúng tôi sẽ trình bày chi tiết những giai đoạn xử lý của phương pháp tạo sinh câu; trong Phần III, chúng tôi sẽ trình bày thử nghiệm và phương pháp đánh giá chất lượng câu tiếng Việt rút gọn mới.

II. TIẾN TRÌNH TÓM LƯỢC

II.1. Phân loại đoạn văn bản dựa trên giả thiết về thứ tự thời gian xảy ra các quá trình

Nghiên cứu được thực hiện với mục tiêu tóm lược những đoạn văn bản gồm hai câu tiếng Việt đơn giản chỉ quá trình bằng phương pháp tạo sinh câu. Những cặp câu được nghiên cứu có đặc điểm là một đối tượng tĩnh vật trải qua hai quá trình ở hai câu. Từng câu trong đó thuộc một trong ba dạng: dạng 1 trong đó đối tượng có sự biến chuyển về trạng thái; dạng 2 trong đó đối tượng có sự biến chuyển về vị trí; dạng 3 trong đó chủ thể bị một tác động bởi một tác nhân, khiến cho nó bị thay đổi trạng thái hoặc vị trí. Dựa trên giả thiết về thứ tự thời gian xảy ra các quá trình (được trình bày trong phần Giới thiệu), những cặp câu được phân loại thành ba loại lớn:

- **Loại 1**: Quá trình ở câu thứ nhất xảy ra trước quá trình ở câu thứ hai. Dựa trên những ngữ cảnh thông thường trong thực tế, chúng tôi giả định rằng quá trình ở câu thứ nhất là nguyên nhân của quá trình ở câu thứ hai.

Ví dụ 4: "*Sét đánh cành cây. Nó bị gãy.*"

- **Loại 2**: Quá trình ở câu thứ nhất xảy ra sau quá trình ở câu thứ hai. Dựa trên những ngữ cảnh thông thường trong thực tế, chúng tôi giả định rằng quá trình ở câu thứ nhất là hệ quả của quá trình ở câu thứ hai.

Ví dụ 5: "*Cái bình bị nứt. Nó bị rơi.*"

- **Loại 3**: Quá trình ở câu thứ nhất xảy ra đồng thời quá trình ở câu thứ hai.

Ví dụ 6: "*Chiếc lá bị úa. Nó bị héo.*"

Những kiểu cặp câu thuộc ba loại trên được tổng hợp trong Bảng 1 với những ký hiệu được sử dụng:

- **X, Y, Z**: lần lượt chỉ các câu thuộc các dạng 1, 2, 3.
- **↳, ↰, ⊕**: Lần lượt chỉ các cặp câu thuộc loại 1, 2, 3.

Bảng 1. Tổng hợp những kiểu cặp câu tiếng Việt đơn giản được nghiên cứu dựa trên giả định về thứ tự thời gian xảy ra các quá trình

	X	**Y**	**Z**
X	⊕	↰	↰
Y	↳	⊕	↰
Z	↳	↳	⊕

II.2. Xử lý đại từ hồi chỉ "nó" và tạo dựng cấu trúc biểu diễn ngữ nghĩa

Do đặc điểm của những cặp câu được nghiên cứu, có tối đa 2 đối tượng thuộc hai loại trong một cặp câu: tĩnh vật, hiện tượng. Chiến lược để xác định tiền ngữ cho một đại từ "nó" ở câu thứ hai: xác định đối tượng tĩnh vật ở câu thứ nhất làm tiền ngữ.

Các bước xử lý để thực hiện chiến lược trên như sau:

- **Bước 1**: Phân tích cấu trúc đoạn văn bản thành hai câu riêng biệt. Đánh chỉ vị trí từng câu: `[first]` đối với câu thứ nhất, `[second]` đối với câu thứ hai. Dựa trên lý thuyết Unification-Based Grammar [37, 55], chỉ số này được truyền lên xuống trên cây cú pháp.
- **Bước 2**: Phân tích cấu trúc câu thành những ngữ đoạn nhỏ hơn. Có hai dạng cấu trúc cú pháp câu trong nghiên cứu này:
  - `Sentence` → `Noun Phrase + [bị] + Predicate Phrase`. Cấu trúc này của câu thuộc dạng 1 hoặc 2.
  - `Sentence` → `Noun Phrase + Predicate Phrase`. Cấu trúc này của câu thuộc dạng 3.
- **Bước 3**: Mô tả đặc điểm từ vựng. Những đặc điểm này được sử dụng vào hai mục đích: (i) xác định đối tượng tiền ngữ cho đại từ "nó"; (ii) tạo dựng cấu trúc biểu diễn ngữ nghĩa của cặp câu nguồn. Dựa trên đặc điểm những cặp câu được nghiên cứu, chúng tôi phân loại từ vựng thành ba lớp chính: đối tượng gồm hai lớp con là tĩnh vật và hiện tượng; động từ chỉ quá trình gồm hai lớp con là chuyển thái và chuyển vị; động từ chỉ hành

động gồm một lớp con là transitive. Bảng 2 trình bày những thông tin được mô tả.

Xét từ vựng đối tượng "cành cây" trong đoạn văn bản ở Ví dụ 4. Mô tả đặc điểm của đối tượng này với nền tảng GULP [37] trong Prolog như Hình 2.

Bảng 2. Những thông tin được mô tả của từ vựng

	Đặc điểm từ vựng	**Vị từ**
Đối tượng	• Chỉ số riêng biệt. • Nội dung từ vựng. • Loại từ. • Lớp con từ loại.	• Chỉ vị trí trong câu. • Chỉ loại từ. • Chỉ ngữ nghĩa.
Quá trình	• Chỉ số gắn với đối tượng chủ thể. • Loại từ. • Lớp con từ loại.	• Chỉ ngữ nghĩa.
Hành động	• Chỉ số gắn với đối tượng chủ thể. • Loại từ. • Lớp con từ loại.	• Chỉ ngữ nghĩa.

```
n(N) --> [cành,cây], {
   append([position(I,FP),
       species(I,FCLASS),
       cành_cây(I,CO,CAT,FCLASS)],
       Con,NewCon),
   unique_integer(I),
   CO = [cành,cây],
   CAT = [object],
   FCLASS = [nonanimated],
   N = syn~(flag_index~I ..
            flag_position~FP) ..
       sem~(in~[drs(U,Con)|Super] ..
            out~ [drs([I|U],NewCon)|
                        Super])
}.
```

Hình 2. Mô tả đặc điểm đối tượng "cành cây" trong Ví dụ 4 với nền tảng GULP [37] trong Prolog.

```
p(P) --> [gãy],{
   append([gãy(Arg,CO,CAT,FCLASS)],
       Con,NewCon),
   CO = [gãy],
   CAT = [process],
   FCLASS = [state_changed],
   P = syn~(flag_arg1~Arg) ..
       sem~(in~[drs(U,Con)|Super] ..
       out~[drs(U,NewCon)|Super])
}.
```

Hình 3. Mô tả đặc điểm động từ chỉ quá trình chuyển thái "gãy" trong Ví dụ 4 với nền tảng GULP [37] trong Prolog.

⇨ Những đặc điểm từ vựng gồm: chỉ số riêng biệt `I` được tạo sinh riêng biệt cho từng đối tượng; chỉ số nội dung `CO` nhận giá trị `[cành,cây]`; chỉ số loại từ vựng `CAT` nhận giá trị `[object]` cho biết đây là đối tượng; chỉ số lớp con từ loại `FCLASS` nhận giá trị `[nonanimated]` cho biết là đối tượng tĩnh vật.

⇨ Những vị từ gắn với chỉ số `I` mà sẽ được dùng để tạo dựng cấu trúc DRS: vị từ chỉ vị trí `position()`; vị từ chỉ loại từ `species()`; vị từ chỉ ngữ nghĩa `cành_cây()`.

Xét từ vựng động từ chỉ quá trình chuyển thái "gãy" trong đoạn văn bản ở Ví dụ 4. Mô tả đặc điểm của đối tượng này với nền tảng GULP [37] trong Prolog như Hình 3.

⇨ Những đặc điểm từ vựng gồm: chỉ số `Arg` gắn với đối tượng chủ thể; chỉ số nội dung `CO` nhận giá trị `[gãy]`; chỉ số loại từ vựng `CAT` nhận giá trị `[process]` cho biết đây là quá trình; chỉ số lớp con từ loại `FCLASS` nhận giá trị `[state_changed]` cho biết là quá trình chuyển thái.

⇨ Những vị từ gắn với chỉ số `Arg` mà sẽ được dùng để tạo dựng cấu trúc DRS: vị từ chỉ ngữ nghĩa `gãy()`.

- **Bước 4**: Tìm kiếm tiền ngữ cho đại từ hồi chỉ "nó".

Ý tưởng chính của giải thuật là tìm kiếm trong danh sách Con của cấu trúc DRS, xác định đối tượng có chỉ số `Index` gắn với hai vị từ: vị từ `position()` nhận giá trị `[first]` cho biết đối tượng ở câu thứ nhất và vị từ `species()` nhận giá trị `[nonanimated]` cho biết đối tượng là tĩnh vật. Giải thuật được thể hiện với nền tảng GULP [37] trong Prolog như Hình 4.

```
np(NP,H,H) -->  ([nó]),{
   NP=sem~in~DrsList,
   member(drs(U,Con),DrsList),
   member(Index,U),
   member(
       position(Index2,
             [first]),
       Con),
   member(
       species(Index2,
             [nonanimated]),
       Con),
   Index == Index2,
   NP=syn~flag_index~Index,
   NP=sem~scope~in~DrsList,
   NP=sem~scope~out~DrsOut,
   NP=sem~out~DrsOut
}.
```

Hình 4. Tìm kiếm tiền ngữ cho đại từ hồi chỉ "nó".

Kết quả thực hiện các bước trên là một cấu trúc DRS biểu diễn ngữ nghĩa của cặp câu tiếng Việt. Xét cặp câu trong Ví dụ 4, cấu trúc DRS của cặp câu này với hai danh sách U và Con như sau:

```
[1,2]
sét(1,[sét],[object],[phenomenon])
species(1,[phenomenon])
position(1,[first])
cành_cây(2,[cành,cây],[object],
             [nonanimated])
species(2,[nonanimated])
position(2,[first])
đánh(1,2,[đánh],[action],
             [transitive])
gãy(2,[gãy],[process],
             [state_changed])
```

Hình 5. Cấu trúc DRS của cặp câu "Sét đánh cành cây. Nó bị gãy." với hai danh sách: danh sách U gồm các chỉ số của các đối tượng; danh sách Con gồm các vị từ gắn với các chỉ số trong danh sách U.

II.3. Xác định những yếu tố quan hệ để tạo sinh cấu trúc cú pháp của câu tiếng Việt rút gọn mới

Trong giai đoạn xử lý này, chúng tôi xác định những yếu tố quan hệ làm tiền đề tạo sinh cấu trúc cú pháp của câu tiếng Việt rút gọn mới. Với yêu cầu đặt ra là câu tiếng Việt được tạo sinh không chỉ tóm lược ý nghĩa của cặp câu chỉ quá trình ban đầu mà còn phải mang tính phổ dụng trong giao tiếp thông thường, việc xác định được dựa trên giả thiết ban đầu về thứ tự thời gian xảy ra các quá trình (được trình bày trong phần Giới thiệu và II.1).

Sau khi tạo dựng được cấu trúc DRS biểu diễn ngữ nghĩa của cặp câu tiếng Việt nguồn, chúng tôi phân tích để xác định các yếu tố quan hệ theo các bước sau:

- **Bước 1**: Xác định những thông tin mang nội dung chính trong cấu trúc DRS. Những thông tin này bao gồm:
  - Những chỉ số riêng biệt trong danh sách U. Những chỉ số này cho biết đối tượng tĩnh vật trải qua hai quá trình và hiện tượng tác động.
  - Vị từ ngữ nghĩa của từ vựng. Vị từ này cho biết thông tin về đặc điểm của đối tượng cũng như quá trình và mối liên hệ giữa các đối tượng.

Xét đoạn văn bản trong Ví dụ 4 thuộc loại cặp câu 1, cấu trúc DRS sau khi được xác định những nội dung chính:

```
[1,2]
sét(1,[sét],[object],[phenomenon])
cành_cây(2,[cành,cây],[object],
             [nonanimated])
đánh(1,2,[đánh],[action],
             [transitive])
gãy(2,[gãy],[process],
             [state_changed])
```

Hình 6. Cấu trúc DRS của cặp câu "Sét đánh cành cây. Nó bị gãy." với những thông tin mang nội dung chính.

Xét đoạn văn bản trong Ví dụ 5 thuộc loại cặp câu 2, cấu trúc DRS sau khi được xác định những nội dung chính:

```
[1]
cái_bình(1,[cái,bình],[object],
             [nonanimated])
nứt(1,[nứt],[process],
             [state_changed])
rơi(1,[rơi],[process],
             [position_changed])
```

Hình 7. Cấu trúc DRS của cặp câu "Cái bình bị nứt. Nó bị rơi" với những thông tin mang nội dung chính.

Xét đoạn văn bản trong Ví dụ 6 thuộc loại cặp câu 3, cấu trúc DRS sau khi được xác định những nội dung chính:

```
[1]
chiếc_lá(1,[chiếc,lá],[object],
            [nonanimated])
úa(1,[úa],[process],
            [state_changed])
héo(1,[héo],[process],
            [state_changed])
```

Hình 8. Cấu trúc DRS của cặp câu "Chiếc lá bị úa. Nó bị héo" với những thông tin mang nội dung chính.

- **Bước 2**: Xác định những yếu tố quan hệ: chủ thể của các quá trình, hiện tượng tác động lên chủ thể, thứ tự thời gian xảy ra các quá trình. Việc xác định được thực hiện theo các bước con sau:
  - **Bước 2.1**: Lần lượt xét vị từ ngữ nghĩa của động từ thứ nhất và thứ hai.
    - Nếu thông tin CAT nhận giá trị [action] và thông tin FCLASS nhận giá trị [transitive], đây là vị từ ngữ nghĩa của động từ chỉ hành động. Vị từ này có hai chỉ số: chỉ số thứ nhất gắn với đối tượng hiện tượng giữ vai trò tác động, chỉ số thứ hai gắn với đối tượng tĩnh vật giữ vai trò chủ thể trải qua quá trình.
    - Nếu thông tin CAT nhận giá trị [process] và thông tin FCLASS nhận giá trị [state_changed] hay [position_changed], đây là vị từ ngữ nghĩa của động từ chỉ quá trình. Vị từ này có một chỉ số gắn với đối tượng tĩnh vật giữ vai trò chủ thể trải qua quá trình.
  - **Bước 2.2**: Dựa vào giá trị của thông tin FCLASS trong vị từ ngữ nghĩa của động từ thứ nhất và động từ thứ hai, xác định mối quan hệ thứ tự thời gian xảy ra quá trình theo sự phân loại trong phần II.1. Quan hệ này được tổng hợp tương ứng như trong Bảng 1 với sự điều chỉnh ký hiệu cụ thể:
    - Dòng là những giá trị của thông tin FCLASS trong vị từ ngữ nghĩa của động từ thứ nhất.
    - Cột là những giá trị của thông tin FCLASS trong vị từ ngữ nghĩa của động từ thứ hai.
    - Điều chỉnh ký hiệu: X chỉ giá trị [state_changed], Y chỉ giá trị [position_changed], Z chỉ giá trị [transitive].

Sau khi xác định được những yếu tố quan hệ, chúng tôi tạo sinh cấu trúc cú pháp của câu tiếng Việt mới với giải thuật tổng quát sau:

- **Bước 1**: Xác định vị từ ngữ nghĩa của đối tượng tĩnh vật làm trung tâm. Thêm vị từ này vào cấu trúc cú pháp ở vị trí đầu tiên.
- **Bước 2**: Thêm <bị> vào cấu trúc cú pháp.
- **Bước 3**: Thêm các vị từ ngữ nghĩa của quá trình thứ nhất vào cấu trúc cú pháp.
- **Bước 4**: Thêm yếu tố quan hệ thứ tự thời gian vào cấu trúc cú pháp.
- **Bước 5**: Thêm <bị> vào cấu trúc cú pháp.
- **Bước 6**: Thêm các vị từ ngữ nghĩa của quá trình thứ hai vào cấu trúc cú pháp.

Bảng 3 trình bày cấu trúc cú pháp tổng quát của câu tiếng Việt rút gọn mới cho các kiểu cặp câu trong Bảng 1. Ký hiệu [ON] chỉ đối tượng tĩnh vật, [OP] chỉ đối tượng hiện tượng, (P) chỉ động từ chỉ quá trình hay hành động.

Xét cấu trúc DRS trong Hình 6, cấu trúc cú pháp câu tiếng Việt rút gọn mới:

```
cành_cây(2) + <bị> + sét(1) +
đánh(1,2) + ↳ + <bị> + gãy(2)
```

Xét cấu trúc DRS trong Hình 7, cấu trúc cú pháp câu tiếng Việt rút gọn mới

```
cái_bình(1) + <bị> + nứt(1) + ↰ +
<bị> + rơi(1)
```

Bảng 3. Cấu trúc cú pháp tổng quát của câu tiếng Việt rút gọn mới cho các kiểu cặp câu trong Bảng 1

Loại cặp câu	Cấu trúc cú pháp tổng quát của câu tiếng Việt rút gọn mới
X ⊕ X	[ON_1] + <bị> + (P_1) + ⊕ + <bị> + (P_2)
X ↰ Y	[ON_1] + <bị> + (P_1) + ↰ + <bị> + (P_2)
X ↰ Z	[ON_1] + <bị> + (P_1) + ↰ + <bị> + [OP_2] + (P_2)
Y ↳ X	[ON_1] + <bị> + (P_1) + ↳ + <bị> + (P_2)
Y ⊕ Y	[ON_1] + <bị> + (P_1) + ⊕ + <bị> + (P_2)
Y ↰ Z	[ON_1] + <bị> + (P_1) + ↰ + <bị> + [OP_2] + (P_2)
Z ↳ X	[ON_1] + <bị> + [OP_1] + (P_1) + ↳ + <bị> + (P_2)
Z ↳ Y	[ON_1] + <bị> + [OP_1] + (P_1) + ↳ + <bị> + (P_2)
Z ⊕ Z	[ON_1] + <bị> + [OP_1] + (P_1) + ⊕ + <bị> + [OP_2] + (P_2)

Xét cấu trúc DRS trong Hình 8, cấu trúc cú pháp câu tiếng Việt rút gọn mới:

```
chiếc_lá(1) + <bị> + úa(1) + ⊕ + <bị>
+ héo(1)
```

II.4. Hoàn chỉnh câu tiếng Việt rút gọn mới

Việc hoàn chỉnh câu tiếng Việt rút gọn mới đòi hỏi lựa chọn từ vựng đáp ứng hai yêu cầu: (i) phù hợp cấu trúc cú pháp đã được tạo sinh; và (ii) giúp câu tiếng Việt rút gọn mới mang tính tự nhiên đối với sự tri nhận của người Việt bản ngữ. Việc lựa chọn từ vựng được thực hiện theo nguyên tắc với những điểm chính:

- Giữ nguyên vị trí các phần tử trong cấu trúc cú pháp khi được thay thế bằng từ vựng.
- Thay thế vị từ ngữ nghĩa của từ vựng bằng hình thái từ được sử dụng trong thực tế.
- Thay thế yếu tố quan hệ thứ tự thời gian bằng những bộ từ vựng tương ứng trong giao tiếp tiếng Việt thông thường.

Trong Bảng 4, chúng tôi trình bày những bộ từ vựng tương ứng trong tiếng Việt để thể hiện yếu tố quan hệ thứ tự thời gian trong nghiên cứu này.

Đối với yếu tố "⊕", chúng tôi ưu tiên sử dụng bộ từ vựng "vừa … vừa" trong ba bộ từ vựng đối với yếu tố này trong Bảng 4.

Bảng 4. Bộ từ vựng tiếng Việt thể hiện yếu tố quan hệ thứ tự thời gian trong nghiên cứu này

Quan hệ	Bộ từ vựng tương ứng
⊕	• và • vừa … vừa • không những … mà còn
↳	• nên
↰	• vì

Xét ba cấu trúc cú pháp của câu tiếng Việt mới được tạo sinh trong phần II.3 đối với những đoạn văn bản trong Ví dụ 4, 5, 6. Câu tiếng Việt rút gọn mới được hoàn chỉnh lần lượt:

- Đoạn văn bản trong Ví dụ 4:
  "*Cành cây bị sét đánh nên bị gãy.*"
- Đoạn văn bản trong Ví dụ 5:
  "*Cái bình bị nứt vì bị rơi.*"
- Đoạn văn bản trong Ví dụ 6:
  "*Chiếc lá vừa bị úa vừa bị héo.*"

III. THỬ NGHIỆM VÀ ĐÁNH GIÁ

III.1. Xây dựng bộ ngữ liệu thử nghiệm

Để thử nghiệm mô hình tóm lược được đề xuất trong bài báo này, chúng tôi tiến hành tập hợp các cặp câu tiếng Việt chỉ quá trình. Theo mục tiêu nghiên cứu của bài báo này, một yêu cầu được đặt ra đối với những cặp câu được dùng trong thử nghiệm này là phải có đại từ hồi chỉ "nó" để liên hệ giữa hai câu.

Trên thực tế, số lượng những cặp câu tiếng Việt thỏa mãn yêu cầu này là rất ít và khó thu thập đủ để tiến hành thử nghiệm. Do vậy, chúng tôi đề xuất phương pháp xây dựng bộ ngữ liệu thử nghiệm theo các bước sau:

- **Bước 1:** Tập hợp những động từ chỉ quá trình được liệt kê trong [26]. Chúng tôi phân loại những động từ này theo ba dạng câu chỉ quá trình được trình bày trong mục I. Chúng tôi cũng tập hợp một số từ vựng chỉ các hiện tượng tự nhiên và nhân tạo trong thực tế. Ví dụ, động từ chỉ quá trình chuyển vị "nghiêng", động từ chỉ quá trình chuyển thái "móp", động từ chỉ quá trình tác động "tàn phá", hiện tượng tự nhiên "lũ".
- **Bước 2:** Tập hợp những câu tiếng Việt đơn giản chỉ quá trình. Chúng tôi sử dụng những từ vựng là động từ chỉ quá trình làm từ khóa để tìm kiếm các câu tiếng Việt được sử dụng làm ví dụ minh họa cho định nghĩa của những từ tương ứng trong những trang web từ điển trực tuyến[1,2,3,4,5,6,7,8,9].

Với cách thức này, chúng tôi tập hợp được 115 câu tiếng Việt chỉ quá trình và có cấu trúc đơn giản. Những câu này có cấu trúc cú pháp thuộc một trong hai dạng được trình bày trong Bước 2 ở Phần II.2.

Ví dụ 7: Đối với động từ chỉ quá trình chuyển thái "móp", một câu chỉ quá trình có thể được tham khảo trong từ điển tiếng Việt *Cồ Việt tra từ*[9]:

"*Cái ấm bị móp.*"

- **Bước 3:** Tạo thủ công thêm một số câu tiếng Việt chỉ quá trình có sử dụng đại từ "nó". Những dạng câu này được xây dựng như sau:
  - Với những từ vựng là động từ chỉ quá trình mà đối tượng chủ thể của nó bị thay đổi trạng thái hay vị trí, chúng tôi tạo thủ công thêm những câu tiếng Việt có dạng:

"*Nó + bị + [động_từ]*"

Ví dụ 8: "*Nó bị móp.*"

  - Đối với những từ vựng là động từ chỉ quá trình mà đối tượng chủ thể của nó bị tác động bởi một hiện tượng, chúng tôi tạo thủ công những câu tiếng Việt có dạng:

"*[hiện_tượng] + [động_từ] + nó*"

Ví dụ 9: "*Lốc cuốn nó.*"

- **Bước 4:** Tổ hợp thủ công những câu ở Bước 2 và Bước 3 để tạo thành những cặp câu tiếng Việt dùng cho thử nghiệm. Đối với từng câu tiếng Việt được tập hợp từ các nguồn tài liệu tham khảo ở Bước 2, chúng tôi lần lượt ghép vào sau đó 1 trong 9 câu được chúng tôi tạo thủ công ở Bước 3, bao gồm: 3 câu quá trình thay đổi trạng thái, 3 câu quá trình thay đổi vị trí, 3 câu quá trình tác động.

Xét câu "*Cái ấm bị móp*" trong Ví dụ 7, chúng tôi thực hiện bước 4 để tạo thành 3 cặp câu ví dụ như sau:

➔ **Ví dụ 10:** Ghép 1 câu chỉ quá trình chuyển thái được tạo thủ công ở bước 3 vào sau câu này để tạo thành cặp:

"*Cái ấm bị móp. Nó bị nứt.*"

➔ **Ví dụ 11:** Ghép 1 câu chỉ quá trình chuyển vị được tạo thủ công ở bước 3 vào sau câu này để tạo thành cặp:

"*Cái ấm bị móp. Nó bị rơi.*"

➔ **Ví dụ 12:** Ghép 1 câu chỉ quá trình tác động được tạo thủ công ở bước 3 vào sau câu này để tạo thành cặp:

"*Cái ấm bị móp. Lửa đốt nó.*"

Với bốn bước thực hiện bên trên, chúng tôi xây dựng được bộ ngữ liệu thử nghiệm bao gồm 1035 cặp câu tiếng Việt, phân loại theo các loại quan hệ trong phần II.1 như sau: 145 cặp câu có quan hệ ↦, 564 cặp câu có quan hệ ↤, 326 cặp câu có quan hệ ⊕.

[1] http://rongmotamhon.net/mainpage/tudien_tiengviet_0_8.html#1

[2] http://vdict.com/

[3] http://tratu.soha.vn

[4] http://www.informatik.uni-leipzig.de/~duc/Dict/

[5] https://vi.glosbe.com/

[6] http://3.vndic.net

[7] http://www.rung.vn/dict/vn_vn/Trang_Ch%C3%ADnh#

[8] http://dict.vietfun.com/

[9] http://tratu.coviet.vn/hoc-tieng-anh/tu-dien/lac-viet/V-V/-all.html

III.2. Thử nghiệm và đánh giá

Để đánh giá chất lượng các câu tiếng Việt rút gọn mới được tạo sinh dựa trên phương pháp được trình bày trong bài báo, chúng tôi tiến hành thử nghiệm và so sánh chúng với các câu tiếng Việt được tạo sinh bởi mô-đun `takahe`[10]. Trong mô-đun này, tác giả F. Boudin đã triển khai phương pháp của K. Filippova [31] khi thực hiện trộn câu bằng cách xác định đường đi chứa thông tin chung trong đồ thị. Một cải tiến dựa trên việc đánh giá lại những ứng viên là những câu trộn dựa theo các ngữ đoạn khóa của F. Boudin và E. Morin [16] cũng được thực thi trong mô-đun này.

Việc thử nghiệm mô-đun `takahe`[10] được chúng tôi thực hiện trên hệ thống Linux Ubuntu phiên bản 12.04LTS 64bits. Hệ thống đã được cài đặt sẵn môi trường phát triển và thực thi cho ngôn ngữ Python với phiên bản Python 2.7.3. Do mô-đun `takahe`[10] là một bộ mã nguồn mở nên để thực thi, chúng tôi tích hợp trong bộ công cụ lập trình `NetBeansIDE`[11] phiên bản 8.0.2 với một plugin `python4netbeans8.0.2`[12] dành riêng để lập trình ngôn ngữ Python.

Chúng tôi thực thi mô-đun `takahe`[10] trong bộ công cụ `NetBeansIDE`[11] theo các bước chính:

- **Bước 1:** Thực hiện gán nhãn từ vựng từng câu với nhãn thích hợp trong bộ nhãn của dự án Penn Treebank [2]. Ở bước này, chúng tôi phân tách thành hai trường hợp để thử nghiệm: (i) trường hợp thứ nhất là giữ nguyên đại từ hồi chỉ "nó"; (ii) trường hợp thứ hai là tiền xử lý đại từ hồi chỉ "nó" dựa theo các kỹ thuật được trình bày trong phần II.2.

Xét cặp câu trong Ví dụ 12, chúng tôi thực hiện gán nhãn từ vựng theo Bước 1 với hai trường hợp như sau:

➔ Trường hợp giữ nguyên đại từ hồi chỉ "nó":

```
Pair1a = ["Cái_ấm/NN bị/VB móp/JJ ./PUNCT", "Lửa/NN đốt/VB nó/PRP ./PUNCT"]
```

➔ Trường hợp tiền xử lý đại từ hồi chỉ "nó":

```
Pair1b = ["Cái_ấm/NN bị/VB móp/JJ ./PUNCT", "Lửa/NN đốt/VB cái_ấm/NN ./PUNCT"]
```

- **Bước 2:** Thực thi lần lượt Pair1a và Pair1b với mô-đun `takahe`[10], nhận được 4 kết quả như sau:

➔ Kết quả thứ nhất. Thực thi trộn cặp câu Pair1a với phương pháp của K. Filippova [31]. Kết quả nhận được là hai câu trộn:

- "*cái_ấm bị móp .*"
- "*lửa đốt nó .*"

➔ Kết quả thứ hai. Thực thi trộn cặp câu Pair1a với phương pháp của F. Boudin và E. Morin [16]. Kết quả nhận được là hai câu trộn:

- "*cái_ấm bị móp .*"
- "*lửa đốt nó .*"

➔ Kết quả thứ ba. Thực thi trộn cặp câu Pair1b với phương pháp của K. Filippova [31]. Kết quả nhận được là ba câu trộn:

- "*cái_ấm bị móp .*"
- "*lửa đốt cái_ấm .*"
- "*lửa đốt cái_ấm bị móp .*"

➔ Kết quả thứ tư. Thực thi trộn cặp câu Pair1b với phương pháp của F. Boudin và E. Morin [16]. Kết quả nhận được là ba câu trộn:

- "*cái_ấm bị móp .*"
- "*lửa đốt cái_ấm .*"
- "*lửa đốt cái_ấm bị móp .*"

Thực hiện so sánh những câu tiếng Việt rút gọn mới được tạo sinh từ phương pháp được trình bày trong bài báo với những kết quả đạt được khi thực thi mô-đun `takahe`[10], chúng tôi áp dụng độ đo ROUGE với công cụ `Rouge2.0`[13]. Công cụ `Rouge2.0`[13] là phiên bản xây dựng trên nền ngôn ngữ Java của công cụ được C. Y. Lin [6, 7, 8, 9] đề xuất, thực hiện tính toán các chỉ số F-score, Recall, Precision [11] với hai

[11]NetBeans IDE 8.0.2 (tại https://netbeans.org/)

[12]Python in NetBeans IDE 8.0.2 (tại http://plugins.netbeans.org/plugin/56795/python4netbeans802)

dạng tóm lược: văn bản tóm lược "reference summary" được tạo thủ công bởi con người; văn bản tóm lược "system summary" được tạo tự động bởi hệ thống. Thiết lập hệ thống và thực thi công cụ `Rouge2.0`[13] như sau:

- **Bước 1:** Với từng cặp câu trong số 1035 cặp câu nguồn được xây dựng trong phần III.1, chúng tôi thực hiện tập hợp một số lượng câu tóm lược thủ công. Số lượng câu tóm lược thủ công có thể khác nhau đối với từng cặp câu nguồn. Danh sách tất cả các câu tóm lược thủ công sẽ trở thành "reference summary" cho từng lần thực thi công cụ `Rouge2.0`[13].
- **Bước 2:** Thực thi công cụ `Rouge2.0`[13] với các câu tóm lược tự động từ phương pháp được trình bày trong bài báo, trở thành "system summary" thứ nhất. Các câu "reference summary" được tập hợp ở Bước 1 được phân chia thành từng tập tin trong thư mục reference, các câu "system summary" thứ nhất được phân chia thành từng tập tin trong thư mục system[13]. Chúng tôi lần lượt thực thi `Rouge2.0`[13] theo uni-gram và bi-gram.
- **Bước 3:** Thực thi tương tự Bước 2 trong đó "system summary" là các câu kết quả của việc thực thi module `takahe`[10] với phương pháp của K. Filippova [31] cho các cặp câu nguồn giữ nguyên đại từ hồi chỉ "nó".
- **Bước 4:** Thực thi tương tự Bước 2 trong đó "system summary" là các câu kết quả của việc thực thi module `takahe`[10] với phương pháp của F. Boudin và E. Morin [16] cho các cặp câu nguồn giữ nguyên đại từ hồi chỉ "nó".
- **Bước 5:** Thực thi tương tự Bước 2 trong đó "system summary" là các câu kết quả của việc thực thi module `takahe`[10] với phương pháp của K. Filippova [31] cho các cặp câu nguồn đã được tiền xử lý đại từ hồi chỉ "nó".
- **Bước 6:** Thực thi tương tự Bước 2 trong đó "system summary" là các câu kết quả của việc thực thi module `takahe`[10] với phương pháp của F. Boudin và E. Morin [16] cho các cặp câu nguồn đã được tiền xử lý đại từ hồi chỉ "nó".

Kết quả thực hiện đánh giá bằng công cụ `Rouge2.0`[13] được thể hiện trong Bảng 5.

Phân tích kết quả trong Bảng 5, chúng tôi ghi nhận các chỉ số đạt được của hệ thống cao hơn so với các chỉ số đạt được khi thực thi mô-đun `takahe`[10] trong hầu hết các trước hợp là do một số yếu tố chính:

- Phương pháp của K. Filippova [31] hay cải tiến của Boudin và E. Morin [16] cũng như những phương pháp khác theo hướng tiếp cận "sentence fusion" chủ yếu trộn những thông tin chung trong những câu nguồn để tạo câu rút gọn mới. Câu rút gọn được tạo ra theo hướng như vậy có thể dẫn đến sự rời rạc do chưa tính đến mối liên hệ về thời gian cũng như không gian trong ngữ cảnh mà các sự việc xảy ra. Vấn đề này càng trở nên phức tạp khi chưa có sự tiền xử lý đại từ hồi chỉ.
- Trong nghiên cứu này, chúng tôi đặt mục tiêu chỉ xem xét những cặp câu chỉ quá trình có cấu trúc đơn giản, không chứa những từ vựng mang ý nghĩa chỉ thời gian. Do vậy, một yếu tố quan trọng giúp xác định mối quan hệ giữa hai câu chỉ quá trình là tiền giả định về thứ tự thời gian được đề xuất trong phần Giới thiệu.
- Đối với những cặp câu thuộc Loại 1 và Loại 2, trong đó một câu chỉ quá trình tác động, yếu tố tiền giả định về thứ tự thời gian được thể hiện rõ nét. Xét trường hợp cặp câu trong Ví dụ 4, khi không có yếu tố ngoại cảnh nào khác, thì quá trình cành cây bị sét đánh sẽ được xác định là xảy ra trước và là nguyên nhân của quá trình cành cây bị gãy.

[13]ROUGE 2.0 – Java Package For Evaluation Of Summarization Tasks With Updated ROUGE Measures – được phát triển bởi Kavita Ganesan cho ngôn ngữ Java (tại http://kavita-ganesan.com/content/rouge-2.0).

Bảng 5. Kết quả thực hiện đánh giá bằng công cụ Rouge2.0[13]

Hệ thống	Tiền xử lý đại từ "nó"	Uni-gram	Bi-gram	Trung bình Recall	Trung bình Precision	Trung bình F-score
Hệ thống được xây dựng theo phương pháp của chúng tôi		X		0.8986	0.8695	0.8800
Takahe[10] theo phương pháp của K. Filippova [31]		X		0.379	0.9177	0.5133
Takahe[10] theo phương pháp của F. Boudin và E. Morin [16]		X		0.379	0.9177	0.5133
Takahe[10] theo phương pháp của K. Filippova [31]	X	X		0.5605	0.9042	0.6812
Takahe[10] theo phương pháp của F. Boudin và E. Morin [16]	X	X		0.5605	0.9042	0.6812
Hệ thống được xây dựng theo phương pháp của chúng tôi			X	0.7334	0.7191	0.7241
Takahe[10] theo phương pháp của K. Filippova [31]			X	0.1788	0.4266	0.244
Takahe[10] theo phương pháp của F. Boudin và E. Morin [16]			X	0.1788	0.4266	0.244
Takahe[10] theo phương pháp của K. Filippova [31]	X		X	0.3303	0.5934	0.4126
Takahe[10] theo phương pháp của F. Boudin và E. Morin [16]	X		X	0.3303	0.5934	0.4126

- Đối với những cặp câu thuộc Loại 1 và Loại 2, trong đó một câu chỉ quá trình đối tượng bị thay đổi trạng thái, một câu chỉ quá trình đối tượng bị thay đổi vị trí, trong đa phần các trường hợp thử nghiệm, việc xác định quá trình đối tượng thay đổi vị trí xảy ra trước quá trình đối tượng thay đổi trạng thái dựa theo tiền giả định là hợp lý. Xét trường hợp cặp câu trong Ví dụ 5, khi không có yếu tố ngoại cảnh nào khác, thì quá trình cái bình bị rơi được xác định là xảy ra trước và là nguyên nhân của quá trình chiếc bình bị nứt. Trong trường hợp này, mặc dù có thể nhưng sẽ là miễn cưỡng và gượng gạo nếu xác định quá trình chiếc bình bị nứt xảy ra trước hoặc xảy ra đồng thời hay là quá trình trái ngược với quá trình chiếc bình bị rơi.
- Đối với những cặp câu thuộc Loại 3, khi một đối tượng trải qua hai quá trình cùng dạng (theo phân loại trong phần Giới thiệu), trong đa phần các trường hợp thì việc xác định hai quá trình này xảy ra đồng thời dựa theo tiền giả định là phù hợp. Xét trường hợp cặp câu trong Ví dụ 6, khi không có yếu tố ngoại cảnh nào khác, thì hai quá trình héo và úa của chiếc lá được xác định là xảy ra đồng thời.

Bên cạnh đó, chúng tôi cũng ghi nhận một tỉ lệ nhất định những cặp câu tiếng Việt chỉ quá trình chưa được tóm lược đúng bởi một câu tiếng Việt mới phù hợp. Nguyên nhân chính được xác định là do trong một số những ngữ cảnh thực tế, tiền giả định được đề xuất không phù hợp với thứ tự thời gian mà hai quá trình xảy ra.

Ví dụ 13: "*Mui xe bị móp. Nó bị quá tải.*"

➔ Trong trường hợp này, khi áp dụng phương pháp tóm lược trong phần II với những tiền giả định được đề xuất, câu tiếng Việt được tạo sinh tự động:

"*Mui xe vừa bị móp vừa bị quá tải.*"

➔ Tuy nhiên, trong thực tế, thông thường quá trình bị quá tải xảy ra sẽ là nguyên nhân của quá trình bị móp. Do vậy, một câu tiếng Việt tóm lược hợp lý hơn sẽ là:

"*Mui xe bị móp vì bị quá tải.*"

Ngoài ra, một vấn đề còn tồn tại đó là việc áp dụng phương pháp tóm lược cho những cặp câu trong đó từng câu có cấu trúc cú pháp phức tạp hơn. Dựa trên kết quả đã đạt được, chúng tôi sẽ xem xét thêm những yếu tố ngoại cảnh tác động về không gian và thời gian, đồng thời đề xuất thêm những tiền giả định về quan hệ giữa hai câu.

IV. THẢO LUẬN VÀ KẾT LUẬN

Trong nghiên cứu này, việc đề xuất một phương pháp tạo sinh câu kết hợp với các tiền giả định dựa trên sự phân loại các dạng câu quá trình theo tiêu chí của Functional Grammar [26, 41] tỏ ra có hiệu quả trong việc tóm lược những cặp câu được xem xét. Đánh giá chất lượng câu tóm lược tiếng Việt bằng phương pháp mới cho thấy tỉ lệ chấp nhận đạt được khá cao khi so sánh với hai phương pháp gần đây theo hướng tiếp cận "sentence fusion" [16, 31].

Với những kết quả đạt được, chúng tôi có thể mở rộng hướng tiếp cận đã đề nghị để áp dụng cho những đoạn văn bản tiếng Việt phức tạp hơn.

TÀI LIỆU THAM KHẢO

[1] A. KHAN and N. SALIM, "*A Review on Abstractive Summarization Methods*", Journal of Theoretical and Applied Information Technology, vol. 59, no.1, 2014, pp. 64–72.

[2] B. SANTORINI, *Part-of-speech Tagging Guidelines for the Penn Treebank Project*, Technical Report MS-CIS-90--47, Department of Computer and Information Science, University of Pennsylvania, 1990.

[3] C. D. MANNING and H. SCHUTZE, *Foundations of Statistical Natural Language Processing*, MIT Press, Cambridge, MA USA, 1999.

[4] C. S. LEE, Z. W. JIAN and L. K. HUANG, "*A Fuzzy Ontology and Its Application to News Summarization*", IEEE Transaction on Systems, Man and Cybernetics, Part B: Cybernetics, vol. 35, no. 5, 2005, pp. 859–880.

[5] C. S. SARANYAMOL and L. SINDHU, "*A Survey on Automatic Text Summarization*", International Journal of Computer Science and Information Technologies, vol. 5, no. 6, 2014, pp. 7889–7893.

[6] C. Y. LIN, "*ROUGE: A Package for Automatic Evaluation of Summaries*", Proceedings of the Workshop on Text Summarization Branches Out, Post-Conference Workshop of ACL 2004, Barcelona, Spain, 2004.

[7] C. Y. LIN, "*Looking for a Few Goods Metrics: ROUGE and its Evaluation*", Proceedings of NTCIR Workshop 2004, Tokyo, Japan, 2004.

[8] C. Y. LIN and E. H. HOVY, "*Automatic Evaluation of Summaries Using N-gram Co-occurrence Statistics*", Proceedings of 2003 Language Technology Conference (HLT-NAACL 2003), Edmonton, Canada, 2004.

[9] C. Y. LIN and F. J. OCH, "*Automatic Evaluation of Machine Translation Quality Using Longest Common Subsequence and Skip-Bigram Statistics*", Proceedings of the 42nd Annual Meeting of ACL (ACL 2004), Barcelona, Spain, 2004.

[10] D. DAS and A. F. T. MARTINS, *A survey on automatic text summarization*, Language Technologies Institute, Carnegie Mellon University, 2007.

[11] D. M. W. POWERS, "*Evaluation: From Precision, Recall and F-MEASURE to ROC, Informedness, Markedness & Correlation*", Journal of Machine Learning Technologies, vol. 2, no. 1, 2011, pp. 37–63.

[12] D. R. RADEV, E. HOVY and K. MCKEOWN, "*Introduction to the special issue on summarization*", Computational Linguistics, vol. 28, no. 4, 2002, pp. 399–408.

[13] E. LLORET, *Text summarization: an overview*, paper supported by the Spanish Government under the project TEXT-MESS (TIN2006-15265-C06-01), 2008.

[14] E. LLORET and M. PALOMAR, "*Analyzing the Use of Word Graphs for Abstractive Text Summarization*", Proceedings of the 1st International Conference on Advances in Information Mining and Management (IMMM 2011), Barcelona, Spain, 2011, pp. 61–66.

[15] E. REITER and R. DALE, *Building Natural Language Generation System*, Cambridge University Press, 1997.

[16] F. BOUDIN and E. MORIN, "*Keyphrase extraction for n-best reranking in multi-sentence compression*", Proceedings of the 2013 Conference of the North American Chapter of the Association for

Computational Linguistics: Human Language Technologies (NAACL-HLT 2013), Atlanta, Georgia, 2013, pp. 298–305.

[17] F. LIU, J. FLANIGAN, S. THOMSON, N. SADEH and N. A. SMITH, "*Toward Abstractive Summarization Using Semantic Representations*", Accepted by the Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies (NAACL 2015).

[18] G. CARENINI and J. C. K. CHEUNG, "*Extractive vs. NLG-based Abstractive Summarization of Evaluative Text: The Effect of Corpus Controversiality*", Proceedings of the 5th International Natural Language Generation Conference, Salt Fork, Ohio, 2008.

[19] H. KAMP, "*A theory of truth and semantic representation*", in: Groenendijk, Jeroen, Janssen, Theo M. V and Stokhof, Martin (eds.), Formal Methods in the Study of Language, Part 1, pp. 277–322, 1981, Mathematical Centre Tracts.

[20] H. P. LUHN, "*The automatic creation of literature abstracts*", IBM Journal of Research Development, vol.2, no. 2, 1958, pp. 159–165.

[21] H. P. EDMUNDSON, "*New methods in automatic extracting*", Journal of the ACM, vol. 1, no. 2, 1969, pp. 264–285.

[22] H. SAGGION and G. LAPALME, "*Generating Indicative-Informative Summaries with SumUM*", Computational Linguistics, vol. 28, no. 4, 2002, pp. 497–526.

[23] H. T. LE, R. C. SAM and P. T. NGUYEN, "*Extracting Phrases in Vietnamese Document for Summary Generation*", Proceedings International Conference on Asian Language Processing (IALP), Harbin, China, 2010, pp. 207–210.

[24] H. T. T. NGUYEN and Q. H. NGUYEN, "*A semi-supervised learning method combined with dimensionality reduction in vietnamese text summarization*", International Journal of Innovative Computing, Information and Control, vol. 9, no. 12, pp. 4903–4915.

[25] H. T. T. NGUYEN, Q. H. NGUYEN and T. N. T. NGUYEN, "*A supervised learning method combine with dimensionality reduction in vietnamese text summarization*", Proceedings 2013 Computing, Communications and IT Applications Conference (ComComAp), Hong Kong, 2013, pp. 69–73.

[26] H. X. CAO, *Tiếng Việt: Sơ thảo ngữ pháp chức năng*, Nhà xuất bản giáo dục, 2006.

[27] I. F. MOAWAD and M. AREF, "*Semantic graph reduction approach for abstractive Text Summarization*", Proceedings of Computer Engineering & Systems (ICCES), 2012 Seventh International Conference on, 2012, pp. 132-138.

[28] I. MANI, *Automatic Summarization*, John Benjamins Publishing Company, 2001.

[29] I. MANI and M. T. MAYBURY, *Advances in Automatic Text Summarization*, MIT Press, 1999.

[30] K. A. GANESAN, C. X. ZHAI and J. HAN, "*Opinosis: A Graph-Based Approach to Abstractive Summarization of Highly Redundant Opinions*", Proceedings of the 23rd International Conference on Computational Linguistics (COLING 2010), Beijing, China, 2010, pp. 340–348.

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

[32] K. FILIPPOVA and M. STRUBE, "*Dependency Tree Based Sentence Compression*", Proceedings of the 5th International Natural Language Generation Conference, Salt Fork, Ohio, 2008.

[33] K. FILIPPOVA and M. STRUBE, "*Sentence Fusion via Dependency Graph Compression*", Proceedings of the Conference on Empirical Methods in Natural Language Processing, Honolulu, Hawaii, 2008.

[34] K. JEZEK and J. STEINBERGER, "*Automatic Text summarization*", Vaclav Snasel (Ed.): Znalosti 2008, ISBN 978-80-227-2827-0, FIIT STU Brarislava, Ustav Informatiky a softveroveho inzinierstva, pp. 1–12, 2008.

[35] K. S. JONES, "*Automatic summarizing: factors and directions*", in: I. Mani and M. Marbury, editors, Advances in Automatic Text Summarization, MIT Press, 1999.

[36] K. S. JONES, *Automatic summarising: a review and discussion of the state of the art*, Technical Report 679, Computer Laboratory, University of Cambridge, 2007.

[37] M. A. COVINGTON, *GULP 4: An Extension of Prolog for Unification Based Grammar*, Research Report AI-1994-06. USA: Artificial Intelligence Center, The University of Georgia, 2007.

[38] M. A. COVINGTON and N. SCHMITZ, *An Implementation of Discourse Representation Theory*, ACMC Research Report 01-0023. USA: Advanced Computational Methods Center, The University of Georgia, 1989.

[39] M. A. COVINGTON, D. NUTE, N. SCHMITZ and D. GOODMAN, *From English to Prolog via Discourse Representation Theory*, ACMC Research Report 01-0024. USA: The University of Georgia, 1988.

[40] M. A. FATTAH and F. REN, "*Automatic Text Summarization*", Proceedings of World Academy of Science, Engineering and Technology, vol. 27, ISSN 13076884, 2008, pp. 192–195.

[41] M. A. K. HALLIDAY and C. M. I. M. MATTHIESSEN, *An Introduction to Functional Grammar*, Third Edition, Hodder Arnold, 2004.

[42] N. R. KASTURE, N. YARGAL, N. N. SINGH, N. KULKARNI and V. MATHUR, "*A Survey on Methods of Abstractive Text Summarization*", International Journal for Research in Merging Science and Technology, vol. 1, iss. 6, 2014, pp. 53–57.

[43] O. CHAOWALIT and O. SORNIL, "*An Automatic Approach to Generating Abstractive Summary for Thai Opinions*", International Journal of Advancements in Computing Technology, vol. 6, no. 3, 2014, pp. 142–150.

[44] P. BAXENDALE, "*Machine-made index for technical literature - an experiment*", IBM Journal of Research Development, vol. 2, no. 4, 1958, pp. 354–361.

[45] P. BLACKBURN and J. BOS, *Representation and Inference for Natural Language – Volume II: Working with Discourse Representation Structures*, Germany: Department of Computational Linguistics, University of Saarland, 1999.

[46] P. E. GENEST and G. LAPALME, "*Framework for Abstractive Summarization using Text-to-Text Generation*", Proceedings of the Workshop on Monolingual Text-to-Text Generation, Oregon, Portland, 2011, pp. 64–73.

[47] P. E. GENEST and G. LAPALME, "*Text Generation for Abstractive Summarization*", Proceedings of the 3rd Text Analysis Conference, Gaithersburg, Maryland, USA, 2010.

[48] P. E. GENEST and G. LAPALME, "*Fully Abstractive Approach to Guided Summarization*", Proceedings of the 50th Annual Meeting of the Association for Computational Linguistics: Short Papers – Volum 2, Jeju Island, Korea, 2012, pp. 354–358.

[49] P. T. NGUYEN and H. T. LE, "*Vietnamese text summarisation using discourse structures*", ICT.rda Conference, Hanoi, Vietnam, 2008.

[50] R. BARZILAY, K. R. MCKEOWN and M. ELHADAD, "*Information fusion in the context of multi-document summarization*", Proceedings of the 37th annual meeting of the Association for Computational Linguistics on Computational Linguistics, 1999, pp. 550–557.

[51] R. BARZILAY and K. R. MCKEOWN, "*Sentence fusion for multidocument news summarization*", Computational Linguistics, vol. 31, 2005, pp. 297–328.

[52] S. GERANI, Y. MEHDAD, G. CARENINI, T. NG. RAYMOND and B. NEJAT, "*Abstractive Summarization of Product Reviews Using Discourse Structure*", Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing (EMNLP 2014), Doha, Qatar, 2014, pp. 1602–1613.

[53] S. K. JAGADISH, K. G. SRINIVASA and R. B. ESWARA, "*A Comprehensive Analysis of Guided Abstractive Text Summarization*", International Journal of Computer Science Issues, vol. 11, iss. 6, no. 1, 2014, pp. 115–121.

[54] S. NIVATTANAKUL, J. SINGTHONGCHAI, E. NAENUDORN and S. WANAPU, "*Using of Jaccard coefficient for keywords similarity*", Proceedings of the International Muti Conference Engineers and Computer Scientists, Hong Kong, 2013, pp. 380–384.

[55] S. M. SHIEBER, *An introduction to unification-based approaches to grammar*, Massachusetts: Microtome Publishing Brookline, 2003.

[56] T. T. TANIMOTO, *An element mathematical theory of classification*, Technical report, I.B.M. Research, New York, NY USA, 1958. Internal report.

[57] T. TRAN and D. T. NGUYEN, "*A Solution for Resolving Inter-sentential Anaphoric Pronouns for Vietnamese Paragraphs Composing Two Single Sentences*", Proceedings of The 5th IEEE International Conference of Soft Computing and Pattern Recognition (SoCPaR 2013), Hanoi, Vietnam, 2013, pp. 172–177.

[58] T. TRAN and D. T. NGUYEN, "*The Solution for Resolving Inter-Sentential Anaphoric Pronoun "nó" in Vietnamese Paragraphs Composing 3 to 5 Simple Sentences*", International Journal of Advanced Science and Technology, vol. 65, 2014, pp. 95–112.

[59] T. TRAN and D. T. NGUYEN, "*Merging Two Vietnamese Sentences Related by Inter-sentential Anaphoric Pronouns for Summarizing*", Proceedings of The 1st NAFOSTED Conference on Information and Computer Science, Hanoi, Vietnam, 2014, pp. 371–381.

[60] T. TRAN and D. T. NGUYEN, "*Improving Techniques for Summarizing the Meaning of Two Vietnamese Sentences by Adding a Meaningful Relationship between Two Actions*", Proceedings of The 16th ACM International Conference on Information Integration and Web-based Applications & Services (iiWAS 2014), Hanoi, Vietnam, 2014, pp. 484–488.

[61] T. TRAN and D. T. NGUYEN, "*Enhancement of Sentence-Generation Based Summarization Method By Modelling Inter-Sentential Consequent-Relationships*", Proceedings of the 16th ACM International Conference on Information Integration and Web-based Applications & Services (iiWAS 2014), Hanoi, Vietnam, 2014, pp. 302–309.

[62] T. TRAN and D. T. NGUYEN, "*Modelling Consequence Relationships between Two Action, State or Process Vietnamese Sentences for Improving the Quality of New Meaning-Summarizing Sentence*", International Journal of Pervasive Computing and Communications, vol. 11, no. 2, 2015, pp. 169–190. Emerald Group Publishing Limited. ISBN 1742-7371.

[63] T. TRAN and D. T. NGUYEN, "*Semantic Predicative Analysis for Resolving Some Cases of Ambiguous Referents of Pronoun "Nó" in Summarizing Meaning of Two Vietnamese Sentences*", Proceedings of the 17th UKSIM-AMSS International Conference on Modelling and Simulation (UKSIM 2015), Cambridge, United Kingdom, 2015, pp. 340–345.

[64] T. TRAN and D. T. NGUYEN, "*Combined Method of Analyzing Anaphoric Pronouns and Inter-sentential Relationships between Transitive Verbs for Enhancing Pairs of Sentences Summarization*", Proceedings of the 4th Computer Science On-line Conference (CSOC 2015) – Vol 1: Artificial Intelligence Perspectives and Applications, in: R. Silhavy et al. (eds), Advances in Intelligent Systems and Computing – Vol. 347, 2015, pp. 67–77.

[65] V. GUPTA and G. S. LEHAL, "*A Survey of Text Summarization Extractive Techniques*", Journal of Emerging Technologies in Web Intelligence, vol. 2, no. 3, 2010, pp. 258–268.

[66] V. SORNLERTLAMVANICH, T. POTIPITI and T. CHAROENPORN, "*UNL Document Summarization*", Proceedings of the 1st International Workshop on Multimedia Annotation (MMA 2001), Tokyo, Japan, 2001.

Nhận bài ngày: 11/12/2014

SƠ LƯỢC VỀ TÁC GIẢ

TRẦN TRUNG

Sinh năm 1985 tại Hải Dương.

Tốt nghiệp ĐH ngành CNTT năm 2007 tại Trường ĐH Khoa học Tự nhiên, ĐH Quốc gia TP. HCM, Thạc sĩ chuyên ngành Khoa học máy tính năm 2012 tại Trường ĐH CNTT, ĐH Quốc gia TP. HCM.

Làm Nghiên cứu sinh chuyên ngành Khoa học máy tính tại Trường ĐH CNTT, ĐH Quốc gia TP. HCM từ tháng 07/2012.

Lĩnh vực nghiên cứu: Xử lý ngôn ngữ tự nhiên, Ngôn ngữ học máy tính.

Điện thoại: 0908 599 738.

Email: ttrung@nlke-group.net

NGUYỄN TUẤN ĐĂNG

Sinh năm 1972 tại Sài Gòn.

Nhận bằng Cử nhân ngành Tin học tại Trường ĐH Mở Bán công TP. HCM năm 1996, Thạc sĩ ngành Tin học tại Viện Tin học sử dụng tiếng Pháp năm 2000, Thạc sĩ ngành Tin học tại Trường ĐH Khoa học Tự nhiên, ĐH Quốc gia TP. HCM năm 2003. Bảo vệ luận án Tiến sĩ ngành Tin học tại Trường ĐH Caen Basse-Normandie, Pháp năm 2006.

Hiện là giảng viên tại Khoa Khoa học Máy tính, Trường ĐH CNTT, ĐH Quốc gia TP. Hồ Chí Minh.

Chuyên ngành nghiên cứu: Xử lý ngôn ngữ tự nhiên, Ngôn ngữ học máy tính

Điện thoại: 0913 655 977

Email: dangnt@uit.edu.vn