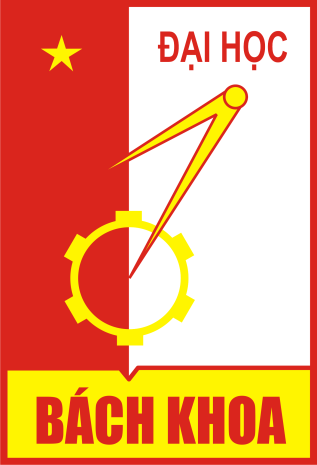


**ĐẠI HỌC BÁCH KHOA HÀ NỘI**

VIỆN CÔNG NGHỆ THÔNG TIN & TRUYỀN THÔNG

-----🙞🙜🕮🙞🙜-----

****

**BÁO CÁO BÀI TẬP LỚN**

HỌC PHẦN XỬ LÝ NGÔN NGỮ TỰ NHIÊN

ĐỀ TÀI: TÓM TẮT VĂN BẢN

BẰNG PHƯƠNG PHÁP TRÍCH RÚT CÂU

*Giảng viên hướng dẫn:* **PGS. TS. LÊ THANH HƯƠNG**

*Nhóm sinh viên thực hiện:* **NHÓM 11**

**Nguyễn Văn Công Linh – 20152223**

**Lê Trịnh Thành – 20153380**

**Đặng Văn Hà – 20151137**

**Nguyễn Quốc Lâm – 20151362**

Hà Nội, 12/2019

**MỤC LỤC**

[PHẦN 1. GIỚI THIỆU ĐỀ TÀI 2](#_Toc27561191)

[1.1. Lý do lựa chọn đề tài 2](#_Toc27561192)

[1.2. Mô tả bài toán 2](#_Toc27561193)

[PHẦN 2. GIẢI QUYẾT BÀI TOÁN 3](#_Toc27561194)

[2.1. Hướng giải quyết 3](#_Toc27561195)

[2.2. Các bước giải quyết bài toán 3](#_Toc27561201)

[2.2.1. Tiền xử lý và tách câu 3](#_Toc27561202)

[2.2.2. Vector hoá câu 4](#_Toc27561203)

[2.2.3. Phân cụm các câu 5](#_Toc27561204)

[2.2.4. Tìm câu đại diện cho cụm 6](#_Toc27561205)

[2.2.5. Sắp xếp các câu đại diện cụm thành đoạn tóm tắt 6](#_Toc27561206)

[PHẦN 3. CÁC KỸ THUẬT SỬ DỤNG 7](#_Toc27561207)

[3.1. Word2Vec 7](#_Toc27561208)

[3.2. K-Means Clustering 9](#_Toc27561209)

[3.3. Hierarchical Clustering 9](#_Toc27561210)

[PHẦN 4. KẾT QUẢ ĐẠT ĐƯỢC 11](#_Toc27561211)

[4.1. Văn bản đầu vào 11](#_Toc27561212)

[4.2. Minh hoạ kết quả tóm tắt 12](#_Toc27561213)

[4.2.1. Phân cụm bằng Hierarchical Clustering 12](#_Toc27561214)

[4.2.1. Phân cụm bằng K-Means Clustering 13](#_Toc27561215)

[PHẦN 5. TÀI LIỆU THAM KHẢO 14](#_Toc27561216)

# **PHẦN 1. GIỚI THIỆU ĐỀ TÀI**

## **1.1. Lý do lựa chọn đề tài**

Trong thời đại bùng nổ thông tin hiện nay, lượng thông tin được sinh ra mỗi phút, mỗi giây là vô cùng nhiều. Nguồn thông tin đó nếu được khai thác hiệu quả sẽ đem lại lợi ích to lớn. Thế nhưng, giới hạn về thời gian và khả năng đọc của con người khiến chúng ta rất khó nắm bắt được nhiều thông tin. Chúng ta thường gặp rất nhiều khó khăn trong quá trình thu nhận thông tin. Bài toán đặt ra là: làm thế nào có thể tóm tắt ngắn gọn và chính xác nguồn thông tin lớn ấy, để dễ dàng trong việc nắm bắt các thông tin quan trọng nhất.

Thế nên, tự động tóm tắt sẽ là một trong những công cụ tuyệt vời giúp con người giảm thiểu được thời gian, công sức đọc báo, email, truyện,… Lượng thông tin ngắn ngọn, cô đọng sẽ giúp con người dễ dàng nắm được vấn đề, từ đó giúp họ giải quyết công việc một cách hiệu quả hơn.

Chính vì vậy, nhóm em đã quyết định lựa chọn đề tài Tóm tắt văn bản tiếng Việt dựa vào việc trích rút các câu quan trọng của đoạn văn, xây dựng một hệ thống có khả năng tóm tắt ngắn gọn, chính xác một đoạn văn.

## **1.2. Mô tả bài toán**

**Đầu vào:** Một văn bản dài gồm N câu.

**Đầu ra:** Đoạn văn bản ngắn K câu, gồm các ý quan trọng của đoạn văn bản đầu vào.

Ý tưởng:

Cố gắng phân tích các câu của đoạn văn bản đầu vào, sử dụng các thuật toán học máy để lựa chọn ra những câu quan trọng. Ghép các câu quan trọng đó thành đoạn văn bản tóm tắt.

# **PHẦN 2. GIẢI QUYẾT BÀI TOÁN**

## 

## **2.1. Hướng giải quyết**

Cụm

các câu

Văn bản

đầu vào

Tập vector

câu

Tiền xử lý

và tách câu

Vector hoá câu

Tập câu

## 

## 

Phân cụm

## 

Sắp xếp các câu đại diện cụm

Tìm câu

gần tâm cụm nhất

Văn bản

tóm tắt

## 

Các câu đại diện cho cụm

## 

Hình 1. Các bước giải quyết bài toán

## **2.2. Các bước giải quyết bài toán**

### **2.2.1. Tiền xử lý và tách câu**

***Tiền xử lý văn bản:***

Biến đổi về chữ thường.

Loại bỏ khoảng trắng thừa.

Tách từ (sử dụng công cụ tách từ ViTokenizer)

A screenshot of a cell phone

Description automatically generated

Hình 2. Tiền xử lý văn bản

***Tách câu:***

Sử dụng thư viện nltk để tách câu:A screenshot of a cell phone

Description automatically generated

Hình 3. Tách câu

### **2.2.2. Vector hoá câu**

**Bước 1:** Xây dựng mô hình vector hoá từ:

Xây dựng mô hình word embedding – mô hình chuyển một tập từ vựng về một không gian vector mà vẫn giữ được đặc trưng ý nghĩa của từ.

Bộ dữ liệu sử dụng: https://github.com/duyvuleo/VNTC

*Input:* 2.385.532 câu tiếng Việt

*Công cụ:* thư viện gensim.

*Ouput:* mô hình word2vec:

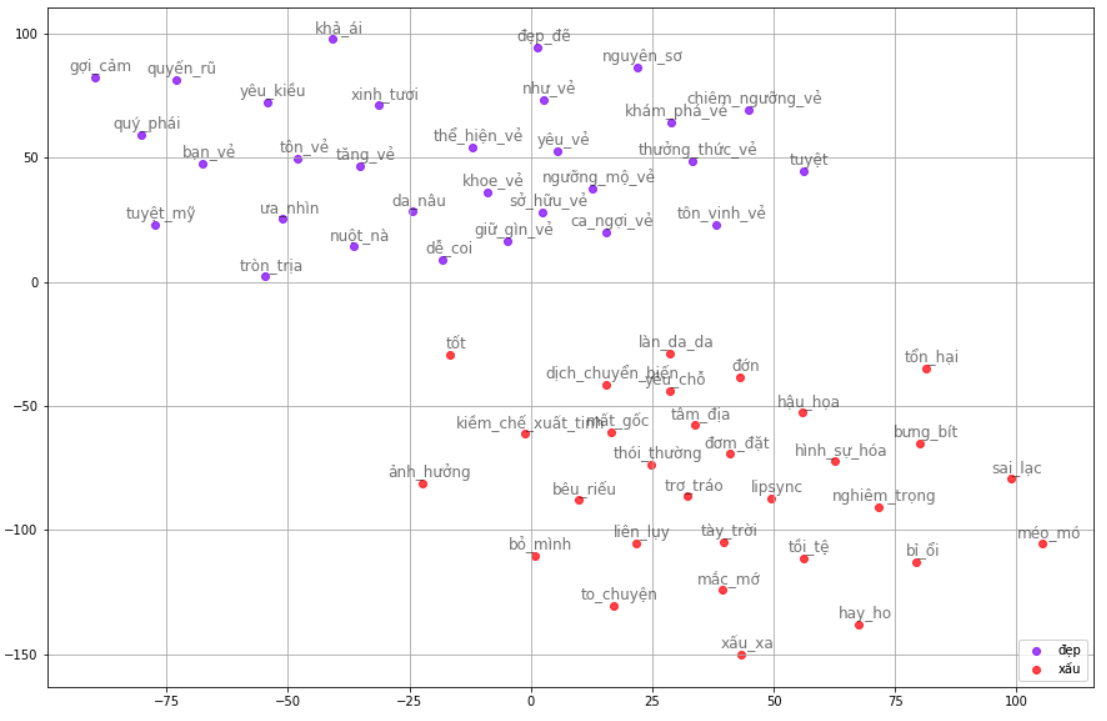
Kích thước từ điển: 225.822 từ

Chức năng: biến đổi một từ có trong bộ từ điển thành một vector 128 chiều.

**A picture containing text

Description automatically generated**

Hình 4. Mô phỏng mô hình word2vec

****

Hình 5. Các từ gần nghĩa trong mô hình word2vec

**Bước 2:** Tiến hành vector hoá câu:

*Input:* Câu đã được tiền xử lý và tách từ.

*Phương pháp:* Vector hoá từng từ của câu (bỏ qua những từ không có trong từ điển của mô hình word2vec). Sau đó cộng các vector đó lại.

*Output:* Vector 128 chiều đại diện cho câu.

Sau bước này, ta có N vector 128 chiều là biểu diễn của N câu trong đoạn văn bản đầu vào.

### **2.2.3. Phân cụm các câu**

Nhắc lại:

N là số câu của văn bản ban đầu.

K là số câu của văn bản tóm tắt.

Sau bước trước, ta có N vector 128 chiều biểu diễn cho N câu trong đoạn văn input, tiến hành phân N vector này vào K cụm, sử dụng 2 thuật toán phân cụm:

K-Means Clustering.

Hierarchical Clustering.

### **2.2.4. Tìm câu đại diện cho cụm**

Xác định K câu gần K tâm cụm nhất. Các câu đó mang ý nghĩa tổng quát nhất cho cụm, có thể đại diện cho K cụm

### **2.2.5. Sắp xếp các câu đại diện cụm thành đoạn tóm tắt**

Tính thứ tự trung bình của từng cụm. Xếp k câu đại diện theo đúng thứ tự trung bình của cụm chứa câu đó.

Thứ tự trung bình =

***Ví dụ:*** Thứ tự các câu trong input: 0, 1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11.

Kết quả phân cụm (câu **in đậm và gạch chân** là câu đại diện cho cụm):

Cụm 1: [ 2, **8** ] => Thứ tự trung bình: T1 = (2 + 8)/2 = 5.0

Cụm 2: [ **0**, 1, 10 ] => Thứ tự trung bình: T2 = (0 + 1 + 10)/3 = 3.67

Cụm 3: [ 3, 4, **5**, 6, 7, 9, 11 ] => Thứ tự trung bình: T3 = (3+4+5+6+7+9+11)/7=6.43

Ta thấy: T2 < T1 < T3

Do đó, thứ tự sắp xếp: Cụm 2 => Cụm 1 => Cụm 3

Vậy đoạn tóm tắt là: Câu 0 => Câu 8 => Câu 5

# **PHẦN 3. CÁC KỸ THUẬT SỬ DỤNG**

## 

## **3.1. Word2Vec**

Word2vec là một mạng neural 2 lớp với duy nhất 1 tầng ẩn, lấy đầu vào là một corpus lớn và sinh ra không gian vector(với số chiều khoảng vài trăm), với mỗi từ duy nhất trong corpus được gắn với một vector tương ứng trong không gian.

Các word vectors được xác định trong không gian vector sao cho những từ có chung ngữ cảnh trong corpus được đặt gần nhau trong không gian. Dự đoán chính xác cao về ý nghĩa của một từ dựa trên những lần xuất hiện trước đây.

Word2vec có 2 mô hình: Skip-gram, Continuous Bag of Words (CBOW).

**Skip - gram**

Sử dụng một từ để đoán ngữ cảnh mục tiêu.

**Continuous Bag of Words (CBOW)**

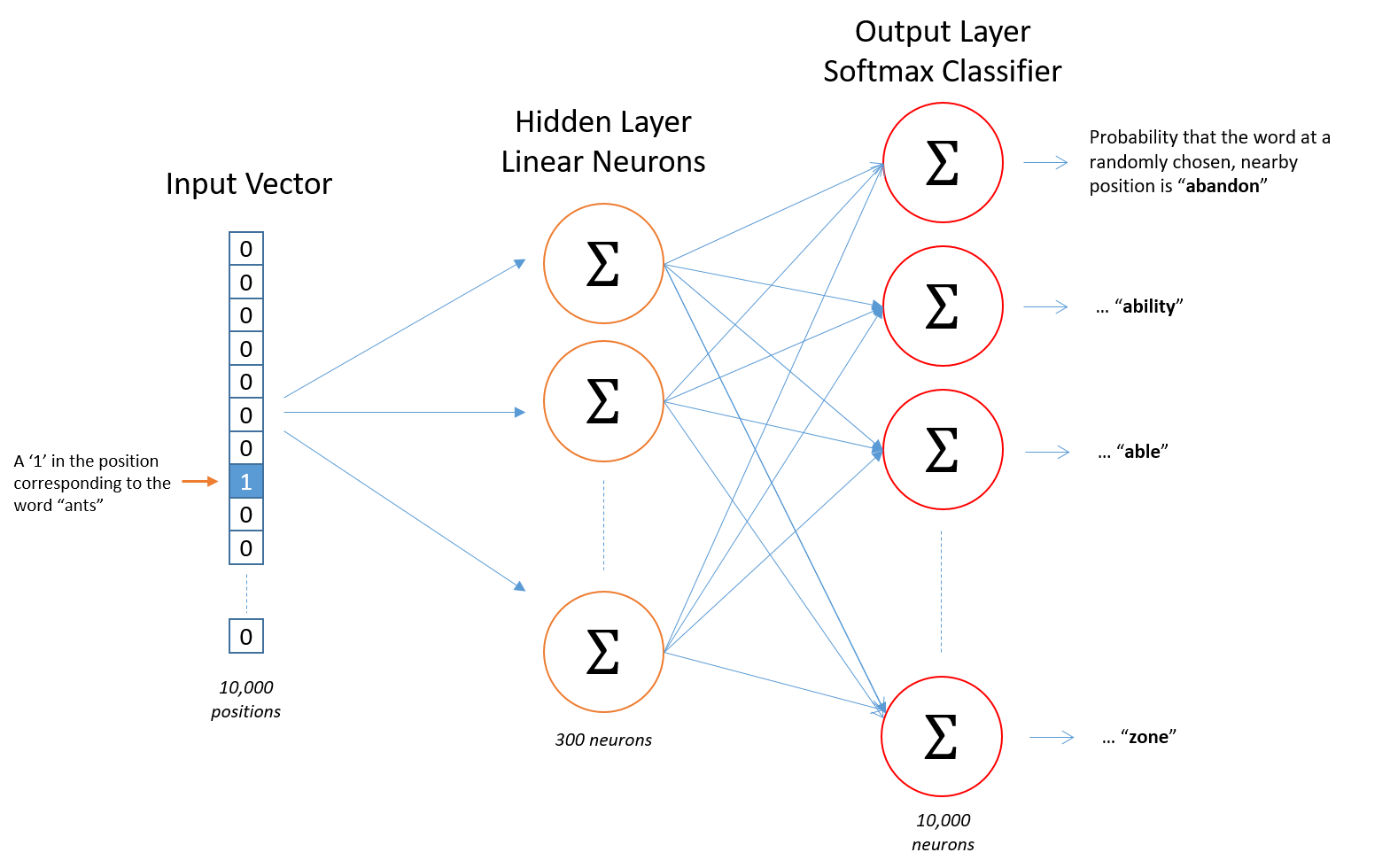
CBOW sử dụng ngữ cảnh để đoán mục tiêu.

A close up of a map

Description automatically generated

Hình 6. Mô hình CBOW và Skip-gram (Nguồn: https://towardsdatascience.com)

Mô hình train của skip-gram (của CBOW thì ngược lại):



Hình 7. Mô hình train Skip-gram (Nguồn: https://medium.com)

– Input là one-hot-vector mỗi word sẽ có dạng x1,x2,..xv trong đó V là số vocabulary, là một vector trong đó mỗi word sẽ có giá trị 1 tương đương với index trong vocabulary và còn lại sẽ là 0.

– Weight matrix giữa input và hidden layer là matrix W(có dimention VxN) có active function là linear, weight giữa hidden và out put là W′ (có dimention là NxV) active function của out put là soft max.

– Mỗi row của W là vector N chiều đại diện cho vw là mỗi word trong input layer.Mỗi row của W là w.

– Từ hidden layer đến output là matrix W′=w′i,j . Tính score ui cho mỗi từ trong vocabulary.

– Trong đó vwj là vector colum j trong W′. Tiếp đó ta sử dụng soft max funtion.

– Ta sẽ thu được tập các vector mô tả các từ vựng, mà vẫn lưu ngữ nghĩa.

## **3.2. K-Means Clustering**

K-Means clustering là thuật toán phân cụm phổ biến hiện nay. Đầu vào là tập điểm dữ liệu và số cụm, đầu ra là kết quả phân chia các điểm dữ liệu vào các cụm.

**Đầu vào:** Dữ liệu X và số lượng cluster cần tìm K.

**Đầu ra:** Các center M và label vector cho từng điểm dữ liệu Y.

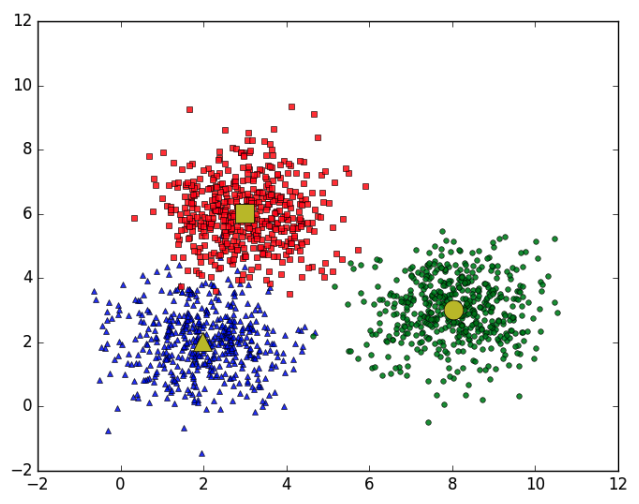
1. Chọn KK điểm bất kỳ làm các center ban đầu.

2. Phân mỗi điểm dữ liệu vào cluster có center gần nó nhất.

3. Nếu việc gán dữ liệu vào từng cluster ở bước 2 không thay đổi so với vòng lặp trước nó thì ta dừng thuật toán.

4. Cập nhật center cho từng cluster bằng cách lấy trung bình cộng của tất các các điểm dữ liệu đã được gán vào cluster đó sau bước 2.

5. Quay lại bước 2.



Hình 8. Minh hoạ kết quả phân cụm K-Means (Nguồn: https://machinelearningcoban.com)

## **3.3. Hierarchical Clustering**

Hierarchical Clustering là thuật toán phân cụm dựa trên sự phân tầng.

Có 2 loại Hierarchical Clustering: Bottom up và Top down

**Bottom up**

Bắt đầu với những cluster chỉ là 1 phần tử.

Ở mỗi bước, kết hợp 2 clusters gần nhau thành 1 cluster.

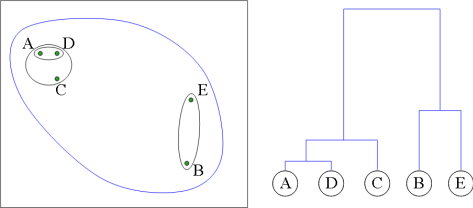
Khoảng cách giữa 2 clusters : 2 điểm gần nhất từ 2 clusters, hoặc khoảng cách trung bình.

**Top down**

Bắt đầu với 1 cluster là tất cả dữ liệu.

Tìm 2 clusters con.

Tiếp tục đệ quy trên 2 clusters con.



Hình 9. Minh hoạ thuật toán Hierarchical Clustering (Nguồn: cs.umd.edu)

# 

# **PHẦN 4. KẾT QUẢ ĐẠT ĐƯỢC**

Xây dựng thành công hệ thống tóm tắt văn bản.

## **4.1. Văn bản đầu vào**

A screenshot of a cell phone

Description automatically generated

**A screenshot of a cell phone

Description automatically generated**

**A screenshot of a cell phone

Description automatically generated**

**A screenshot of a cell phone

Description automatically generated**

## **4.2. Minh hoạ kết quả tóm tắt**

Tóm tắt văn bản đầu vào thành đoạn văn bản ngắn gồm 3 câu, kết quả được minh hoạ dưới đây:

### **4.2.1. Phân cụm bằng Hierarchical Clustering**

**A screenshot of a cell phone

Description automatically generated**

**A screenshot of a cell phone

Description automatically generated**

**A screenshot of a cell phone

Description automatically generated**

### **4.2.1. Phân cụm bằng K-Means Clustering**

**A screenshot of a cell phone

Description automatically generated**

**A screenshot of a cell phone

Description automatically generated**

# **PHẦN 5. TÀI LIỆU THAM KHẢO**

**1. Dữ liệu huấn luyện word2vec:**

<https://github.com/duyvuleo/VNTC>

**2. Hierarchical Clustering:**

<https://towardsdatascience.com/understanding-the-concept-of-hierarchical-clustering-technique-c6e8243758ec>

<https://www.cs.umd.edu/hcil/hce/hce3-manual/hce3_manual.html>

**3. K-Means Clustering:**

<https://machinelearningcoban.com/2017/01/01/kmeans/>

**4. Bài hướng dẫn:**

Tác giả Nguyễn Thành Hậu: <https://github.com/thanhhau097/Natual-Language-Processing/blob/master/Text%20Summarization/Text%20Summarization.ipynb>

Tác giả Phạm Hoàng Anh: <https://viblo.asia/p/xay-dung-chuong-trinh-tom-tat-van-ban-tieng-viet-don-gian-voi-machine-learning-YWOZrgAwlQ0>