ĐẠI HỌC QUỐC GIA HÀ NỘI
TRƯỜNG ĐẠI HỌC CÔNG NGHỆ

KHỔNG BÙI TRUNG

PHÂN LOẠI GIỚI TÍNH NGƯỜI DÙNG MẠNG XÃ HỘI DỰA VÀO TIN NHẮN VĂN BẢN VÀ WORD2VEC

Ngành: Công nghệ thông tin
Chuyên ngành: Kỹ thuật phần mềm
Mã số: 60480103

TÓM TẮT LUẬN VĂN THẠC SĨ KỸ THUẬT PHẦN MỀM

Hà Nội – Năm 2016

CHƯƠNG 1: TỔNG QUAN KHAI PHÁ DỮ LIỆU VÀ MẠNG XÃ HỘI

1.1. Khai phá dữ liệu

1.1.1. Khai phá dữ liệu là gì?

Khai phá dữ liệu (datamining) được định nghĩa như là một quá trình chắt lọc hay khai phá tri thức từ một lượng lớn dữ liệu. Một ví dụ hay được sử dụng là việc khai thác vàng từ đá và cát, Dataming được ví như công việc "Đãi cát tìm vàng" trong một tập hợp lớn các dữ liệu cho trước. Thuật ngữ Datamining ám chỉ việc tìm kiếm một tập hợp nhỏ có giá trị từ một số lượng lớn các dữ liệu thô. Có nhiều thuật ngữ hiện được dùng cũng có nghĩa tương tự với từ Datamining như Knowledge Mining (khai phá tri thức), knowledge extraction (chắt lọc tri thức), data/patern analysis (phân tích dữ liệu/mẫu), data archaeoloogy (khảo cổ dữ liệu), datadredging (nạo vét dữ liệu), ... [1].

1.1.2. Quá trình khai phá dữ liệu

Khai phá dữ liệu là một bước trong bảy bước của quá trình KDD (Knowleadge Discovery in Database) và KDD được xem như 7 quá trình khác nhau theo thứ tự sau [1]:

1. Làm sạch dữ liệu (data cleaning & preprocessing).

2. Tích hợp dữ liệu.

3. Trích chọn dữ liệu (data selection).

4. Chuyển đổi dữ liệu.

5. Khai phá dữ liệu (data mining).

6. Ước lượng mẫu (knowledge evaluation).

7. Biểu diễn tri thức (knowledge presentation).

1.1.3. Các chức năng chính của khai phá dữ liệu

• Mô tả khái niệm (concept description).

• Luật kết hợp (association rules).

• Phân loại và dự đoán (classification & prediction).

• Phân cụm (clustering).

• Khai phá chuỗi (sequential/temporal patterns).

1.1.4. Các kỹ thuật khai phá dữ liệu

1.1.4.1. Phân loại (phân loại - classification)

1.1.4.2. Hồi qui (regression)

1.1.4.3. Phân cụm (clustering)

1.1.4.4. Tổng hợp (summarization)

1.1.4.5. Mô hình hoá sự phụ thuộc (dependency modeling)

1.1.4.6. Phát hiện sự biến đổi và độ lệch (change and deviation dectection)

1.2. Mạng xã hội

1.2.1. Mạng xã hội là gì?

Mạng xã hội là việc thực hiện mở rộng một số lượng các mối quan hệ của doanh nghiệp hoặc các quan hệ xã hội bằng cách tạo ra các kết nối thông qua các cá nhân người dùng, thường là thông qua các trang web mạng xã hội như Facebook, Twitter, LinkedIn và Google+[16].

1.2.2. Lợi ích và tác hại của mạng xã hội

1.2.2.1. Lợi ích của mạng xã hội

Mạng xã hội ngày nay có một số lợi ích như sau [4]:

a. Giới thiệu bản thân mình với mọi người.

b. Kết nối bạn bè.

c. Tiếp nhận thông tin, học hỏi kiến thức và kỹ năng.

d. Kinh doanh.

e. Bày tỏ quan niệm cá nhân.

f. Mang đến lợi ích về sức khoẻ.

1.2.2.2. Tác hại của mạng xã hội

Ta không thể phủ nhận những lợi ích mà mạng xã hội đã mang đến cho con người hiện nay như giúp ích cho công việc, cho việc tìm kiếm thông tin, thiết lập các mối quan hệ cá nhân hay giải trí… Tuy nhiên, nó cũng chứa đựng nhiều nguy

cơ, rủi ro tiềm ẩn có thể ảnh hưởng xấu tới công việc, mối quan hệ cá nhân và cuộc sống của người sử dụng [4]:

a. Giảm tương tác giữa người với người.

b. Lãng phí thời gian và xao lãng mục tiêu thực của cá nhân.

c. Nguy cơ mắc bệnh trầm cảm.

d. Giết chết sự sáng tạo.

e. Không trung thực và bạo lực trên mạng.

f. Thường xuyên so sánh bản thân với người khác.

g. Mất ngủ.

h. Thiếu riêng tư.

1.2.3. Các mạng xã hội phổ biến

1.2.3.1. Facebook[1]

1.2.3.2. Instagram[2]

1.2.3.3. Twitter[3]

1.2.3.4. Zalo[4]

---

[1] https://www.facebook.com/
[2] https://www.instagram.com/
[3] https://twitter.com
[4] http://zalo.me/

CHƯƠNG 2: WORD2VEC VÀ MÔ HÌNH "TỪ" THÀNH "VECTOR"

2.1. Vector từ là gì

Để máy tính có thể hiểu được các từ thì chúng ta phải biểu diễn các từ đó dưới dạng vector từ. Vector từ là một vector của các trọng số biểu diễn cho từ. Trong dạng biểu diễn 1-of-N (hay "one-hot") việc mã hóa các thành phần trong vector được liên kết với một từ trong bộ từ vựng. Việc mã hóa một từ cho trước là đưa ra một vector, trong đó các phần tử liên quan được thiết lập giá trị là 1, tất cả các phần tử khác là 0.

Trong Word2Vec, một biểu diễn phân tán của một từ được sử dụng. Tạo ra một vector với kích thước vài trăm chiều. Mỗi từ được biểu diễn bởi tập các trọng số của từng phần tử trong nó. Vì vậy, thay vì sự kết nối 1-1 giữa một phần tử trong vector với một từ, biểu diễn từ sẽ được dàn trải trên tất cả các thành phần trong vector, và mỗi phần tử trong vector góp phần định nghĩa cho nhiều từ khác nhau.

Như vậy một vector trở thành đại diện một cách tóm lược ý nghĩa của một từ. Và như ta sẽ thấy tiếp theo, đơn giản bằng việc kiểm tra một tập văn bản lớn, nó có thể học các vector từ, ta có thể nắm bắt mối quan hệ giữa các từ theo một cách đáng ngạc nhiên. Ta cũng có thể sử dụng các vector như các đầu vào cho một mạng Nerual.

2.2. Lập luận với Vector từ

Ta thấy rằng các đại diện từ được nghiên cứu trong thực tế nắm bắt quy tắc cú pháp và ngữ nghĩa có ý nghĩa theo một cách rất đơn giản. Cụ thể, các quy tắc được quan sát như các giá trị bù vector không đổi giữa các cặp từ chia sẻ một mối quan hệ đặc biệt. Ví dụ, nếu ta ký hiệu vector cho chữ i là X_i, và tập trung vào mối quan hệ số ít/số nhiều, ta sẽ quan sát thấy rằng $X_{apple} - X_{apples} \approx X_{car} - X_{cars}$, $X_{family} - X_{families} \approx X_{car} - X_{cars}$, v.v. Ta thấy rằng đây cũng là trường hợp cho một loạt các quan hệ ngữ nghĩa được đo bởi mối quan hệ tương đồng [7].

Các vector rất tốt khi trả lời câu hỏi tương tự dạng a là dành cho b như c là dành cho?. Ví dụ, Man (đàn ông) là dành cho Woman (phụ nữ) như uncle (chú) là dành cho? Aunt (thím, dì) sử dụng một phương pháp các giá trị bù vector đơn giản dựa vào khoảng cách cosin.

Đây là sự hợp thành vector cũng cho phép ta trả lời câu hỏi "Vua – Đàn ông + Phụ nữ =?" và đi đến kết quả "Hoàng hậu"! Tất cả đều thực sự đáng chú ý khi bạn nghĩ rằng các kiến thức này chỉ đơn giản là xuất phát từ việc nhìn vào rất nhiều từ trong ngữ cảnh (ta sẽ thấy ngay) mà không có thông tin khác được cung cấp về ngữ nghĩa của nó.

2.3. Nghiên cứu các vector từ vựng

Sự phức tạp trong các mô hình ngôn ngữ mạng neural (Truyền thẳng hay tái diễn) xuất phát từ lớp ẩn phi tuyến tính. Trong khi đây là những gì làm cho mạng neural trở nên rất hấp dẫn, vì vậy tôi quyết định tìm hiểu những mô hình đơn giản hơn, có thể không có khả năng đại diện cho các dữ liệu chính xác như các mạng neural, nhưng có thể được tạo trên nhiều dữ liệu hiệu quả hơn. Mikolov và cộng sự [11] đã đề xuất ra hai

mô hình mới để sinh ra Word2Vec: Mô hình Continuous Bag-of-Words và mô hình Skip-gram.

2.4. Mô hình Continuous Bag-of-word/Mô hình túi từ liên tục (CBOW)

Mục tiêu huấn luyện của mô hình Continuous Bag-of-word là để dự đoán một từ khi biết các từ lân cận (ngữ cảnh) sử dụng mạng neural 3 tầng. Phần này tôi sẽ giới thiệu về ngữ cảnh của một từ và ngữ cảnh của một cụm từ.

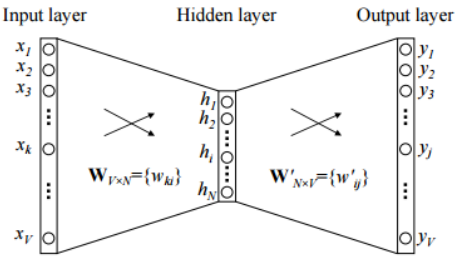
2.4.1. Ngữ cảnh của một từ

Hình 2.6 sau đây biểu diễn mô hình mạng, sự định nghĩa ngữ cảnh đã được đơn giản hóa. Trong thiết lập của ta, quy mô từ vựng là V, và quy mô lớp ẩn là N. Các đơn vị trên lớp liền kề được kết nối đầy đủ Đầu vào là một vector được mã hóa one – hot, có nghĩa là cho một từ trong ngữ cảnh đầu vào được nhắc đến, chỉ có một trong số các đơn vị V, $\{x_1,\ldots,x_V\}$, sẽ là 1, và tất cả các đơn vị khác là 0.

Input layer: Lớp đầu vào

Hidden layer: Lớp ẩn

Output layer: Lớp đầu ra



Hình 2.6: Mô hình CBOW đơn giản với chỉ một từ trong ngữ cảnh

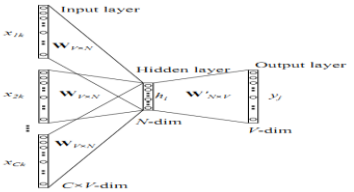
2.4.2. Ngữ cảnh của cụm từ

Hình 2.7 sau đây cho thấy mô hình CBOW với thiết lập ngữ cảnh của cụm từ. Khi tính toán đầu ra của lớp ẩn, thay vì trực tiếp sao chép vector đầu vào của nhóm từ cùng ngữ cảnh đầu vào, thì mô hình CBOW lấy trung bình các vector của các nhóm từ cùng ngữ cảnh đầu vào, và sử dụng các kết quả của ma trận trọng số đầu vào → ma trận trọng số ẩn và vector trung bình như đầu ra.

Input layer: Lớp đầu vào

Hidden layer: Lớp ẩn

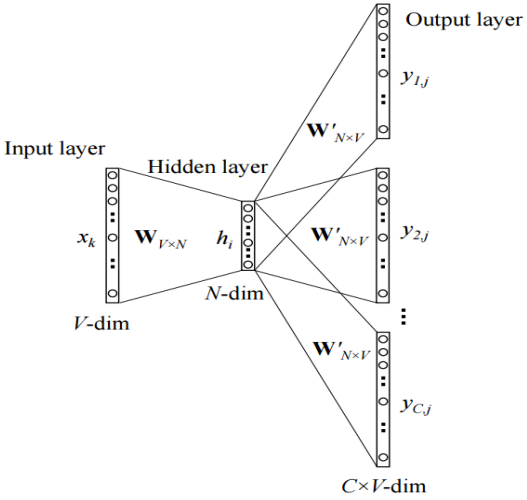
Output layer: Lớp đầu ra



Hình 2.7: Mô hình túi từ liên tục (CBOW)

2.5. Mô hình Skip-gram

Mô hình Skip-gram được đưa ra bởi Mikolov và các cộng sự [10,11]. Mô hình này trái ngược lại với mô hình CBOW. Các từ mục tiêu bây giờ lại ở lớp đầu vào và các từ cùng ngữ cảnh lại ở lớp đầu ra.



Hình 2.8: Mô hình Skip-gram

Mục tiêu huấn luyện của mô hình Skip-gram là để tìm ra đại diện từ vựng hữu ích để dự đoán các từ xung quanh trong một câu hay một tài liệu. Chính thức hơn, đưa ra một chuỗi các từ huấn luyện $\omega_1, \omega_2, \omega_3, \ldots, \omega_T$, mục tiêu của mô hình Skip-gram là tối đa hóa xác suất log trung bình.

2.5.1. Hierarchical Softmax (Softmax phân cấp)

2.5.2. Negative Sampling (Mẫu phủ định)

2.5.3. Subsampling of Frequent Words (Lựa chọn mẫu phụ của các từ thường gặp).

CHƯƠNG 3: ỨNG DỤNG WORD2VEC VÀO PHÂN LOẠI GIỚI TÍNH NGƯỜI DÙNG MẠNG XÃ HỘI

3.1. Mở đầu

Hiện nay, đã xuất hiện một số phương pháp và kỹ thuật khai phá dữ liệu văn bản tiếng Việt tuy nhiên với mỗi phương pháp đều có ưu, nhược điểm khác nhau. Trong mỗi lĩnh vực khác nhau thì các phương pháp cũng cho kết quả phân tích, phân loại là khác nhau. Và để cải thiện kết quả của việc khai phá dữ liệu văn bản tiếng Việt tôi đã đề xuất sử dụng Word2Vec để đưa vào quá trình phân tích và phân loại cho văn bản.

Đối với các kỹ thuật không sử dụng Word2Vec, khi phân loại văn bản người ta sẽ trích chọn các đặc trưng tương ứng với các nhãn được gán. Sau đó các đặc trưng này sẽ được lưu vào bộ từ vựng cùng với các trọng số của nó. Tuy nhiên bộ từ vựng này sẽ có thể xảy ra tình trạng thưa dữ liệu và vấn đề kích thước của mô hình ngôn ngữ là rất lớn nếu tập văn bản có kích thước lớn. Còn đối với kỹ thuật có sử dụng thêm Word2Vec, các từ sẽ được biểu diễn bởi các vector. Các từ có ý nghĩa gần giống nhau thì có kích thước vector gần bằng nhau. Chính vì vậy mà Word2Vec có thể tự học được đối với các từ cùng ngữ cảnh.

3.2. Giải pháp cho bài toán phân loại giới tính người dùng mạng xã hội

Đối với bài toán phân loại giới tính người dùng mạng xã hội tôi sử dụng phương pháp học máy để xử lý. Trong học máy có hai vấn đề cần được giải quyết, đó là:

**** Lựa chọn bộ phân lớp:***

Trong quá trình thực nghiệm tôi đã áp dụng bài toán này trên cả hai bộ phân lớp phổ biến và được đánh giá hiệu quả hiện nay là bộ phân lớp SVM và Logistic. Kết quả thu được cho thấy bộ phân lớp Logistic cho kết quả thực nghiệm tốt hơn. Chính vì vậy mà tôi đã lựa chọn bộ phân lớp Logistic để sử dụng cho bài toán phân loại giới tính người dùng mạng xã hội. Tuy nhiên tôi không đi sâu vào tìm hiểu cách làm việc của bộ phân lớp này mà chỉ sử dụng công cụ GraphLab Create[5].

**** Trích chọn đặc trưng:***

Khi đã có được một bộ phân lớp tốt thì việc phân loại hiện giờ sẽ phụ thuộc rất nhiều vào các đặc trưng đưa vào để phân loại. Đặc trưng càng chính xác thì việc phân loại càng nhận được kết quả tốt. Chính vì vậy mà việc trích chọn đặc trưng vô cùng quan trọng. Đối với quá trình thực nghiệm trong luận văn này tôi chủ yếu tìm hiểu để xây dựng và trích chọn được các đặc trưng tốt nhằm cải thiện kết quả phân loại. Do đó mà bước lấy dữ liệu thô, sau đó tiền xử lý và xây dựng các bộ dữ liệu sẽ quyết định rất nhiều đối với việc trích chọn đặc trưng. Dữ liệu càng mịn, càng ít nhiễu thì đặc trưng càng chính xác. Đối với bài toán phân lớp văn bản ban đầu sẽ chọn đặc trưng theo mô hình n-gram với n=1,2,3. Sau đó các đặc trưng này sẽ được sử dụng cho bộ phân lớp.

Khi bộ phân lớp thực hiện việc phân lớp nó sẽ sử dụng các đặc trưng lấy được từ tập dữ liệu kiểm thử rồi tìm

---

[5] https://turi.com/products/create/

liên kết đến các đặc trưng được trích chọn từ tập dữ liệu huấn luyện theo mô hình n-gram. Tuy nhiên khi sử dụng kỹ thuật này hay xảy ra tình trạng thưa dữ liệu, phân bố không đồng đều. Bên cạnh đó, khi kích thước tập văn bản huấn luyện lớn, số lượng các cụm n-gram và kích thước của mô hình ngôn ngữ cũng rất lớn. Chính vì vậy tôi sử dụng thêm Word2Vec để đưa thêm được ngữ cảnh từ vào cho các đặc trưng.

Giả sử khi các đặc trưng của tập dữ liệu kiểm thử không tìm thấy bất kỳ một liên kết nào với các đặc trưng đã được trích chọn trong tập dữ liệu huấn luyện. Khi đó bộ phân lớp sẽ tìm kiếm trong Word2Vec các từ có nghĩa gần với các từ của các đặc trưng trong tập dữ liệu kiểm thử, sau khi tìm được các từ gần nghĩa hoặc giống nhau nó sẽ coi đây là các đặc trưng của bộ dữ liệu kiểm thử và đưa vào tìm các liên kết với các đặc trưng được trích chọn của tập huấn luyện. Chính vì có khả năng tự học được các từ có nghĩa tương đồng như vậy mà khi sử dụng thêm Word2Vec thì kết quả phân lớp của chúng ta sẽ tăng lên.

3.2.1. Phân loại theo mô hình n-gram

**** Áp dụng mô hình n-gram cho bài toán phân loại giới tính người dùng mạng xã hội ta thực hiện như sau:***

Ban đầu tôi lấy dữ liệu trên tường của từng người dùng mạng xã hội lưu thành một file và gán nhãn cho dữ liệu này theo thông tin trên tài khoản của họ là nam hay nữ. Sau đó tiền xử lý dữ liệu rồi sử dụng công cụ vn.vitk-master để thực hiện tách từ và thu được các dữ liệu đã chuẩn hóa. Từ dữ liệu đã chuẩn hóa tôi trích chọn đặc trưng 1-gram và 2-gram cùng giá trị nhãn đã được gán để sử dụng làm tập dữ liệu huấn

luyện. Nghĩa là, các đặc trưng của tập dữ liệu huấn luyện được phân thành hai lớp giới tính nam và giới tính nữ.

Trong quá trình trích chọn đặc trưng (1-gram và 2-gram) đối với từng người dùng tôi sẽ tìm xem các từ người dùng hay sử dụng (đã loại bỏ các từ vô nghĩa - stopword) và đưa vào làm đặc trưng cho lớp tương ứng đã được gán nhãn của người dùng đó. Ví dụ với một người dùng tôi thấy hay sử dụng cụm từ "đi đá_bóng" và người này đã được gán nhãn là Nam thì tôi sẽ đưa cụm từ này vào làm đặc trưng cho lớp giới tính Nam của tập huấn luyện. Với một người dùng tôi thấy hay sử dụng cụm từ "đi mua_sắm" và người này đã được gán nhãn là Nữ thì tôi sẽ đưa cụm từ "đi mua_sắm" vào làm đặc trưng cho lớp giới tính Nữ của tập huấn luyện. Tương tự như vậy đối với các người dùng khác thì tôi sẽ xây dựng được một tập dữ liệu huấn luyện chứa các đặc trưng đã được phân lớp.

Để phân loại giới tính của một người dùng tôi sẽ đưa lần lượt từng đặc trưng đã trích chọn trên tường của người này (dữ liệu kiểm thử) vào bộ phân lớp Logistic để so sánh với các đặc trưng trong tập dữ liệu huấn. Bộ phân lớp này sẽ tìm kiếm một liên kết giữa đặc trưng của dữ liệu kiểm thử với các đặc trưng trong tập huấn luyện. Sau đó sẽ kiểm tra xem đặc trưng trong tập huấn luyện được tìm thấy nằm ở lớp nào sẽ có được kết quả. Sau đó lại tiếp tục với các đặc trưng còn lại của dữ liệu kiểm thử. Cuối cùng giới tính của người dùng này sẽ được xác định dựa vào những kết quả thu được. Kết quả đặc trưng nằm ở lớp nào chiếm đa số sẽ được sử dụng làm kết quả cuối cùng.

3.2.2. Phân loại khi sử dụng thêm Word2Vec

Mô hình n-gram ở trên hiện đang được sử dụng phổ biến trong xử lý ngôn ngữ tự nhiên. Tuy nhiên nó lại gặp phải một số khó khăn.

Một là, dữ liệu phân bố không đều.

Hai là, kích thước bộ nhớ của mô hình ngôn ngữ lớn.
**** Áp dụng phân loại khi đưa thêm Word2Vec làm đặc trưng cho mô hình n-gram ta thực hiện như sau:***

Để có thể cải thiện những hạn chế nêu trên ta sẽ sử dụng Word2Vec biểu diễn cho các từ về dạng vector. Đầu tiên tôi lấy nội dung các bài báo trên các trang web như; 24h.com.vn, vnexpress.net, eva.vn, dantri.vn,... Sau đó tiến hành tiền xử lý rồi sử dụng công cụ vn.vitk-master để thực hiện tách từ và thu được dữ liệu đã chuẩn hóa. Tiếp theo tôi sử dụng công cụ Gensim[6] cho bộ dữ liệu đã chuẩn hóa để sinh ra một file Word2Vec dùng làm đặc trưng vector. Trong đặc trưng vector này các từ có nghĩa gần nhau sẽ được biểu diễn với giá trị gần bằng nhau.

Tiếp theo tôi thực hiện giống như với mô hình n-gram. Tuy nhiên khi tìm kiếm một liên kết giữa đặc trưng của dữ liệu kiểm thử với các đặc trưng trong tập huấn luyện. Nếu không tìm thấy bất kỳ một liên kết nào thì tôi sẽ đưa thêm Word2Vec vào bộ phân lớp để tìm kiếm. Tôi tìm từ của đặc trưng vector có giá trị biểu diễn gần nhất với từ của đặc trưng trong dữ liệu

[6] https://pypi.python.org/pypi/gensim

kiểm thử để sử dụng làm đặc trưng kiểm thử rồi tìm kiếm sự liên kết với các đặc trưng của tập huấn luyện. Cứ tiếp tục như vậy cho đến khi tìm được một liên kết với tập huấn luyện. Sau đó lại tiếp tục với các đặc trưng còn lại của dữ liệu kiểm thử. Cuối cùng giới tính của người dùng này sẽ được xác định dựa vào kết quả thu được.

3.3. Thực nghiệm

3.3.1. Dữ liệu thực nghiệm

* Chuẩn bị dữ liệu của người dùng mạng xã hội

- Sử dụng công cụ Graph API Explorer[7] để truy xuất vào và lấy nội dung trên tường của người dùng mạng xã hội.

- Các nội dung trên tường của từng người được lưu vào 1 file theo định dạng:

<Blog>

<date> </date>

<post> </post>

</Blog>

- Sử dụng công cụ vn.vitk-master để tách từ, tách câu. Đối với dữ liệu tiếng Anh thì không phải thực hiện bước này còn đối với tiếng Việt thì bước này sẽ vô cùng cần thiết.

[7] https://developers.facebook.com/tools/explorer

- Lưu tên file theo định dạng: mã.**giới tính**.tuổi.nghề nghiệp.tên.xml

Với mã, giới tính, tuổi, nghề nghiệp, tên là các nhãn được gán theo thông tin của người dùng mạng xã hội.

- Đưa tất cả các file vào một thư mục chung .

- Tạo ra và lưu vào 1 đối tượng SFrame[8] tất cả dữ liệu của người dùng mạng xã hội dưới dạng các đặc trưng và trọng số của các đặc trưng. Mỗi dòng là dữ liệu của một người dùng mạng xã hội.

* Xây dựng Word2Vec để sử dụng làm 1 đặc trưng trong quá trình dự đoán.

- Tôi sử dụng thư viện Jsoup để viết mã lấy dữ liệu từ các trang web: 24h.com.vn, vnexpress.net, dantri.vn, eva.vn.

- Sau đó tôi sử dụng thêm công cụ vn.vitk-master để tách từ cho dữ liệu được lấy về.

- Lấy dữ liệu được sinh ra sau khi đã tách từ đưa hết vào thư mục /home/trungkb/data/txt.

- Sử dụng Gensim với tham số mặc định để sinh ra một file word2vec theo mô hình Skip-gram với số chiều của vector là 300, và bỏ qua các từ xuất hiện ít hơn 40 lần. Bộ dữ liệu để sinh ra file word2vec có dung lượng 1,2Gb bao gồm nội dung các bài báo được đăng trên các trang web như 24h.com.vn, vnexpress.net, dantri.vn, eva.vn,... Việc lấy nội

[8] https://github.com/turi-code/SFrame

dung này là tôi sử dụng thư viện Jsoup để lấy nội dung từ các trang web trên.

3.3.2. Cấu hình thực nghiệm

3.3.3. Mô tả thực nghiệm

Đối với các tập dữ liệu trên tường người dùng mạng xã hội ở trên tôi lần lượt chia tập dữ liệu thành 2 phần có tỷ lệ như sau:

Bảng 3.2: Tỷ lệ chia tập dữ liệu huấn luyện và kiểm thử

Lần chia	*Tập huấn luyện*	*Tập kiểm thử*
1	*75%*	*25%*
2	*80%*	*20%*
3	*85%*	*15%*

Với lần chia thứ nhất tỷ lệ tập dữ liệu huấn luyện/tập dữ liệu kiểm thử là 75%-25% thì tôi sẽ thực nghiệm 10 lần. Mỗi lần thực nghiệm tôi sẽ lấy ngẫu nhiên dữ liệu theo tỷ lệ trên. Điều này sẽ giúp cho quá trình chạy thực nghiệm 10 lần thì cả 10 lần tập dữ liệu huấn luyện cũng như tập dữ liệu kiểm thử sẽ khác nhau. Sau đó tôi lấy trung bình cộng kết quả 10 lần chạy sẽ được kết quả thực nghiệm cho từng lần chia tỷ lệ tập dữ liệu.

Tương tự như trên với các lần chia tỷ lệ tập dữ liệu huấn luyện/tập dữ liệu kiểm thử là 80%-20% và 85%-15%.

Với mỗi lần thực nghiệm tôi đều chạy theo cả 2 kỹ thuật thực hiện là: n-gram khi không có Word2Vec và khi có Word2Vec. Với mô hình n-gram tôi có các đặc trưng đầu vào là đặc trưng [1gram] và đặc trưng [2gram]. Khi sử dụng thêm Word2Vec tôi đưa thêm đặc trưng [vector] vào cho bộ phân lớp.

3.3.4. Đánh giá

Tiêu chuẩn đánh giá trong thực nghiệm là độ đo chính xác, tỉ lệ phần trăm mẫu phân lớp chính xác trên tổng số mẫu kiểm thử, độ chính xác được tính bằng công thức sau:

$$\text{Độ chính xác} = \frac{|\{x|x \in D_{tst} \cap f(x) = y\}|}{|\{x|x \in D_{tst}\}|} \quad (3.1)$$

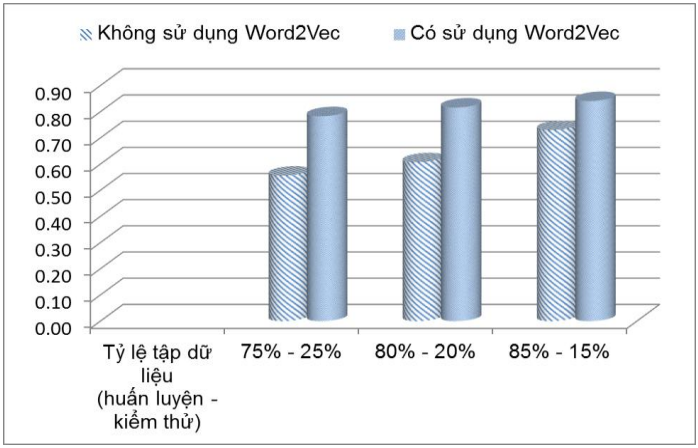
Trong đó, D_{tst} là dữ liệu kiểm thử, y là cực quan điểm ban đầu, $f(x)$ là cực quan điểm dự đoán.

3.3.5. Kết quả thực nghiệm

Bảng 3.6: Tổng hợp so sánh kết quả thực nghiệm

Tỷ lệ tập dữ liệu (huấn luyện - kiểm thử)	***[1gram]; [2gram]***	***[1gram]; [2gram];[vector]***
75% - 25%	*0.556*	*0.783*
80% - 20%	*0.607*	*0.816*
85% - 15%	*0.730*	*0.841*

Để thấy rõ và dễ so sánh kết quả thực nghiệm hơn ta sẽ xem biểu đồ sau:



Hình 3.3: Biểu đồ biểu diễn kết quả thực nghiệm

TÀI LIỆU THAM KHẢO

Tài liệu tiếng Việt:

[1] Nguyễn Thị Thanh Thảo, ***Tìm hiểu các ứng dụng của datamining trong kinh doanh***, 2012.

[2] Nhóm tác giả: Kim Đình Sơn, Đặng Ngọc Thuyên, Phùng Văn Chiến, Ngô Thành Đạt, ***Các mô hình ngôn ngữ N-gram và Ứng dụng***, 2013.

[3] Bộ môn hệ thống thông tin, Khoa công nghệ thông tin, Đại học hàng hải Việt Nam, ***Bài giảng khai phá dữ liệu***, 2011.

[4] Bộ phận tư vấn – hỗ trợ và giới thiệu việc làm SV, ***Tác động của mạng xã hội đến học sinh sinh viên***, 2015.

https://www.kgtec.edu.vn/component/k2/1440-tac-dong-cua-mang-xa-hoi-den-hoc-sinh-sinh-vien.

Tài liệu tiếng Anh:

[5] Andriy Mnih and Geoffrey E Hinton. ***A scalable hierarchical distributed language model. Advances in neural information processing systems, 21:1081–1088***, 2009.

[6] Andriy Mnih and Yee Whye Teh. ***A fast and simple algorithm for training neural probabilistic language models. arXiv preprint arXiv:1206.6426***, 2012.

[7] David A. Jurgens, Saif M. Mohammad, Peter D. Turney, Keith J. Holyoak, ***SemEval-2012 Task 2: Measuring Degrees of Relational Similarity***, 2012.

[8] Frederic Morin and Yoshua Bengio. ***Hierarchical probabilistic neural network language model. In Proceedings of the international workshop on artificial intelligence and statistics, pages 246–252***, 2005.

[9] Michael U Gutmann and Aapo Hyv¨arinen. ***Noise-contrastive estimation of unnormalized statistical models, with applications to natural image statistics. The Journal of Machine Learning Research, 13:307–361***, 2012.

[10] Mikolov et al, ***Distributed Representations of Words and Phrases and their Compositionality***, 2013.

[11] Mikolov, Kai Chen, Greg Corrado, and Jeffrey Dean. ***Efficient estimation of word representations in vector space. ICLR Workshop***, 2013.

[12] Mikolov, Stefan Kombrink, Lukas Burget, Jan Cernocky, and Sanjeev Khudanpur. ***Extensions of recurrent neural network language model. In Acoustics, Speech and Signal Processing (ICASSP), 2011 IEEE International Conference on, pages 5528–5531. IEEE***, 2011.

[13] Richard Socher, Yoshua Bengio and Chris Manning, ***Deep Learning for NLP (without Magic)***, ACL2012.

[14] Ronan Collobert and Jason Weston. ***A unified architecture for natural language processing: deep neural networks with multitask learning. In Proceedings of the 25th international conference on Machine learning, pages 160–167. ACM***, 2008.

[15] Rong, ***Word2vec Parameter Learning Explained***, 2014.

[16] Margaret Rouse, ***Social networking***, 2016.

http://whatis.techtarget.com/definition/social-networking.