Mục Lục

[1. Cơ sở lý thuyết và phương pháp: 2](#_Toc265732385)

[***a.*** ***Phân loại câu hỏi :*** 2](#_Toc265732386)

[***b.*** ***Các phương pháp phân loại*** 4](#_Toc265732387)

[***c.*** ***Đặc trưng phân loại*** 8](#_Toc265732388)

[2. Các nghiên cứu và ứng dụng liên quan 11](#_Toc265732389)

[***a.*** ***LIBSVM*** 11](#_Toc265732390)

[***b.*** ***Wordnet*** 11](#_Toc265732394)

[**c.** **POS tagger** 13](#_Toc265732395)

[**d.** **Một số vấn đề liên quan khác** 13](#_Toc265732396)

[3. Hướng tiếp cận của nhóm 15](#_Toc265732398)

[***a.*** ***Dữ liệu huấn luyện và kiểm thử*** 15](#_Toc265732399)

[***b.*** ***Phương pháp*** 15](#_Toc265732400)

[***c.*** ***Đặc trưng*** 15](#_Toc265732401)

[***d.*** ***Mô hình cho bài toán phân loại câu hỏi*** 16](#_Toc265732402)

[Phụ lục 17](#_Toc265732403)

[A. Danh sách các stop words 17](#_Toc265732404)

[B. Nhãn từ loại 19](#_Toc265732405)

[C. Cú pháp trong Regular Expression 20](#_Toc265732406)

[Tài liệu tham khảo: 23](#_Toc265732407)

1. **Cơ sở lý thuyết và phương pháp:**
   1. ***Phân loại câu hỏi :***

*QA system (Question & Answer system): là một* hệ thống đóng vai trò phổ biến trong việc tìm kiếm thông tin chính xác và hiệu quả. Nó đưa ra câu trả lời đầy đủ và chính xác ứng với yêu cầu của người dùng và câu trả lời được thể hiện bằng ngôn ngữ tự nhiên. Người dùng nhanh chóng lấy được thông tin cần thiết thay vì tìm kiếm thông tin trong một khối lượng lớn các văn bản.

Trong hệ thống QA bao gồm nhiều thành phần : thành phần phân tích câu hỏi , thành phần phân loại câu hỏi , thành phần truy vấn dựa trên những tập tài liệu liên quan đến câu truy vấn , …

Trong các thành phần nêu trên , thành phần phân loại câu hỏi đóng vai trò quan trọng trong một hệ thống QA [1-4]. Người dùng đưa vào hệ thống một câu hỏi, câu hỏi sau đó sẽ được phân tích để tìm từ khóa tìm kiếm hoặc định dạng thành một loại câu truy vấn xuống cơ sở dữ liệu chứa thông tin. Kết quả trả về của quá trình đó không thể trả về ngay lập tức một câu trả lời duy nhất và chính xác. Trên thực tế nó sẽ đưa ra một tập các câu trả lời đề cử có liên quan đến vấn đề được hỏi. Vậy, công việc chính của phân loại câu hỏi là lọc và loại bỏ những câu trả lời nào không phù hợp, hơn thế nữa có thể dựa trên một số tiêu chí để đánh giá xếp hạng các câu trả lời theo mức độ liên quan đến câu hỏi.

Ngoài ra, phân loại câu hỏi còn có thể giúp ta xác định được kiểu câu trả lời một khi ta biết được phân loại của nó. Việc này giúp hệ thống đưa ra một câu trả lời gần gũi với ngôn ngữ tự nhiên.

Ví dụ : “Who is the author of Harry Potter ? “ là một loại câu hỏi về con người (tác giả) . Câu trả lời mặc nhiên phải là một danh từ riêng liên quan đến một người nào đó (tức tên tác giả của cuốn Harry Potter) “J. K. Rowling is …”

Thông thường các câu hỏi được phân loại theo 6 mục thô và 50 mục mịn (Bảng 1) . Các mục này được đưa ra bởi Li và Roth (2002). Các bài báo nghiên cứu sau này về phân loại câu hỏi cũng đựa trên các mục này để nghiên cứu và đánh giá [1,2,12,14].

|  |  |
| --- | --- |
| Mục thô | Mục mịn |
| ABBR | abbreviation,expansion |
| DESC | definition, description, manner, reason |
| ENTY | animal, body, color, creation, currency, disease, event, food, instrument, language, letter, other, plant, product, religion, sport, subtance, symbol, technique, term, vehicle, word |
| HUM | description, group, individual, title |
| LOC | city, country, mountain, other, state |
| NUM | Code, count, date, distance, money, order, other, percent, period, speed, temperature, size, weight |

Bảng 1 – Phân loại 6 mục thô và 50 mục mịn

Hiện có một số hệ thống QA system được biết tới như :

+ NSIR : một hệ thống hỏi đáp phát triển từ CLAIR – một nhóm thuộc đại học Michigan . Hệ thống câu hỏi mẫu được lấy từ TREC . (<http://tangra.si.umich.edu/clair/NSIR/html/html/about.html>)

+AnswerBus : là một hệ thống QA system cho nhiều lĩnh vực , hỗ trợ những câu hỏi cho nhiều thứ tiếng như Anh, Đức, Pháp , Tây Ban Nha,… câu trả lời là những đường link đến câu trả lời chứ không đưa ra một đáp áp cụ thể . (<http://www.answerbus.com/about/index.shtml>)

* 1. ***Các phương pháp phân loại***

Việc tiếp cận phân loại câu hỏi có thể phân biệt qua ba nhóm chính : dựa trên luật (rule-based) , mô hình hóa ngôn ngữ (language modeling) và dựa trên máy học ( machine learning based). Đó là ba hướng tiếp cận chính hiện nay [2]

*Đối với các tiếp cận dựa trên luật :* việc phân loại câu hỏi dựa vào một số các luật bằng tay. Các luật này có được là do đề xuất từ các chuyên gia. Đối với cách tiếp cận này , một loạt các biểu thức thông dụng (regular expression) được tạo ra để so khớp với câu hỏi từ đó quyết định phân loại của câu hỏi và loại câu trả lời . Nhưng nó có một số hạn chế như sau :

+ Sự phân loại dựa trên các luật viết gặp nhiều khó khăn và tốn nhiều thời gian xử lý . Do dựa trên kiến thức chủ quan của con người trên một tập dữ liệu câu hỏi .

+ Có sự giới hạn về mức độ bao quát và phức tạp trong việc mở rộng phạm vi của các loại câu trả lời .

*Trong cách tiếp cận máy học :* kiến thức chuyên môn được thay thế bằng một tập lớn các câu hỏi được gán nhãn kết quả sẵn (một tập dữ liệu mẫu ) . Từ tập này , classifier (bộ phân lớp) sẽ được học một cách có kiểm soát hoặc không kiểm soát trên tập mẫu. Các thuật toán máy học thường dùng bao gồm : Mạng nơ-ron (Neural NetWork), tính xác suất Naïve Bayes, Maximum Entropy , cây quyết định (decision Tree) , lân cận (Nearest-Neighbors), Sparse Network of Winnows(SNoW), Support Vector machine(SVM) ... Với cách tiếp cận máy học , ta dễ dàng thấy nó sẽ giải quyết được nhiều hạn chế từ cách tiếp cận dựa trên những luật . Những thuận lợi trong cách tiếp cận này gồm :

+ Thời gian tạo dựng ngắn , không tốn thời gian để đề ra các luật.

+ Bộ phân loại được tạo ra tự động thông qua việc học từ một tập dữ liệu huấn luyện ; việc cung cấp các luật giờ không cần thiết nữa .

+ Mở rộng độ bao phủ : bằng cách thu được từ các ví dụ huấn luyện .

+ Nếu có nhu cầu , bộ phân loại có thể tái cấu trúc lại (học lại) một cách linh hoạt để phù hợp với quy luật mới.

Hiện tại , bộ phân loại thường được sử dụng trong phương pháp máy học là Support Vector Machine . Dựa trên kiểm thử và đánh giá với một số phương pháp máy học khác Zhang và Lee [12] cho thấy Support Vector Machine có phần vượt trội hơn so với các phương pháp máy học khác.

***Nearest-Neighbors***

Ý tưởng của thuật toán này là dựa trên mức độ giống nhau nhất giữa một trường hợp chưa được gán nhãn phân loại với các thực thể trong tập dữ liệu học ( đã được gán nhãn ) . Tức là một câu hỏi cần phân loại sẽ được đem ra so sánh với các thực thể trong tập huấn luyện dựa trên các đặt trưng . Câu hỏi đó có độ tương tự gần giống với thực thể nào nhất sẽ lấy nhãn của thực thể đó . Điều này , đơn giản chỉ là so sánh sự trùng lặp các đặc trưng giữa hai câu hỏi .

***Naive Bayes***

Là mô hình tiếp cận cho việc phân loại dựa trên định lý sác xuất Bayes . Mô hình này giả định rằng các đặc trưng tồn tại độc lập trong một phân loại của câu hỏi , vì thế sác xuất của chúng cũng độc lập . Mô hình sẽ tính sác xuất của các đặc trưng trong câu hỏi và xem tỉ lệ sác xuất nào cao nhất để phân loại câu hỏi đó. Công thức tính sác xuất có điều kiện của Bayes như sau:

P(A/B) = P(B/A) x P(A) / P (B)

Xác xuất của biến cố A khi tồn tại một biến cố B . Tức , với một đặc trưng B và phân loại A, ta tính sác xuất của *P(A) trên tập dữ liệu huấn luyện = số trường hợp là A / tổng số các trường hợp* . Tiếp theo tính xác xuất của đặc trưng B trên phân loại A : *P(B/A) = số đặc trưng B xuất hiện trong A / Số trường hợp là A* .Xác xuất của P(B) là 1. ( Tham khảo thêm ví dụ tại <http://www.statsoft.com/textbook/naive-bayes-classifier/> )

***Decision Tree***

Cây quyết định sẽ mô tả một cấu trúc cây , trong đó , các lá đại diện cho các phân loại còn cành đại diện cho các thuộc tính kết hợp dẫn đến phân loại đó . Các dữ liệu đưa vào học có dạng :

(x,y) = (x1,x2,x3...xk,y)

Trong đó y là giá trị mà chúng ta hướng tới để phân loại . Tập các biến x1,x2,x3...xk là các thuộc tính mà ta dựa vào để thực hiện việc phân loại .

Vd : Một tập dữ liệu mẫu về thói quen đi chơi cuối tuần :

|  |  |  |  |
| --- | --- | --- | --- |
| Set of Atributes (x1,x2,x3) | | | Variable y |
| Weekend | (Example)WeatherParentsMoney | Decision | (Category) |
| Sunny | Yes | Rich | Cinema |
| Sunny | No | Rich | Tennis |
| Windy | Yes | Rich | Cinema |
| Rainy | Yes | Poor | Cinema |

Sau khi có một tập dữ liệu mẫu , một số thuật toán được áp dụng để xây dựng cây quyết định như ID3 của Quinlan (ý tưởng chung là chọn thuộc tính sau cho nó phân các thực thể thành các tập con (các nhánh) sao cho có nhiều thực thể cùng loại nhất ) . (tham khảo thêm tại <http://www.doc.ic.ac.uk/~sgc/teaching/v231/lecture11.html> )

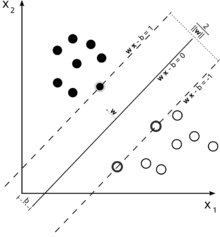
***Support Vector Machine***

Phương pháp này dựa trên ý tưởng từ một tập huấn luyện cho trước, dùng một mặt phẳng để chia các điểm trong không gian thành hai lớp riêng biệt. Trong đó các điểm là các đặc trưng mà ta rút ra được trong quá trình huấn luyện.

Khoảng cách giữa mặt phẳng tới điểm dữ liệu gần nhất của một phân loại sẽ quyết định chất lượng của mặt phẳng . Khoảng cách này gọi là biên.

Nếu chúng ta có một tập dữ liệu :

\mathcal{D} = \{ (\mathbf{x}_i, c_i)|\mathbf{x}_i \in \mathbb{R}^p, c_i \in \{-1,1\}\}_{i=1}^n

[](http://en.wikipedia.org/wiki/File:Svm_max_sep_hyperplane_with_margin.png)Là một tập các vector xi trong đó ci sẽ có giá trị là -1 hoặc 1. Chúng ta sẽ có được một phương trình mặt phẳng :

\mathbf{w}\cdot\mathbf{x} - b=0,\,

Vector w là một vector pháp tuyến: nó vuông góc với siêu phẳng. Biến xác định độ lệch của siêu phẳng từ gốc theo vector pháp tuyến.

Ta có thêm hai mặt phẳng tại điểm biên :

\mathbf{w}\cdot\mathbf{x} - b=1\, và \mathbf{w}\cdot\mathbf{x} - b=-1.\,

Bằng phương pháp hình học, người ta đã tính được khoảng cách giữa hai mặt phẳng này là . Vì thế, để cực đại khoảng cách biên thì phải phải cực tiểu hóa ||w||.

Chúng ta có ràng buộc sau :

\mathbf{w}\cdot\mathbf{x}_i - b \ge 1\qquad\text{ for }\mathbf{x}_i  thuộc về phân loại thứ 1

\mathbf{w}\cdot\mathbf{x}_i - b \le -1\qquad\text{ for }\mathbf{x}_i  thuộc về phân loại thứ 2

Có thể viết lại : c_i(\mathbf{w}\cdot\mathbf{x}_i - b) \ge 1,  \quad \text{ for all }  1 \le i \le n.\qquad\qquad(1)

Cuối cùng bài bài toán đưa về việc cực tiểu hóa *w* và tìm *b* sao cho thỏa điều kiện (1) .

* 1. ***Đặc trưng phân loại***

Trong phương pháp tiếp cận máy học, từ tập dữ liệu có sẵn ta rút ra những đặc trưng phân loại đề từ đó đưa ra huấn luyện. Các đặc trưng này đơn giản chỉ là một hoặc nhiều từ nằm đâu đó trong câu hỏi. Chúng không quyết định câu hỏi đó thuộc về phân loại nào, chỉ là cơ sở để qua qua trình học dự đoán một câu hỏi thuộc về một phân loại.

Trong phân loại câu hỏi, các đặc trưng này đều là các đặc trưng nhị phân. Điều này có nghĩa là giá trị của một đặc trưng là 1hoặc -1: 1 tức là đặc trưng đó thuộc về phân loại đó và ngược lại là -1. Vì thế, các đặc trưng khi rút được ra từ một câu hỏi để huấn luyện đều có giá trị nhị phân là 1.

Có nhiều loại đặc trưng cho bài toán phân loại câu hỏi, có một số đặc trưng thông dụng và thường được sử dụng. Một số khác do một số tác giả đề xuất, sau đó đánh giá và kiểm thử để biết được mức độ chính xác của bộ phân loại khi áp dụng đặc trưng đó. Các đặc trưng đó sẽ được trình bày rõ hơn ở phần bên dưới.

* + 1. ***Bag of words***

BOW là đặc trưng thường được sử dụng nhất. Vì nó khá đơn giản và dễ hiểu. Các từ vựng trong một câu hỏi đều được coi như là đặc trưng phân loại. Một câu hỏi được coi như là một túi chứa các đặc trưng.

Thông thường các stopwords như : what, is ... có ít vai trò trong việc tìm kiếm dữ liệu vì chủ yếu việc này dựa trên các key-word. Nhưng đối với phân loại câu hỏi thì các stop words đóng vai trò quan trọng và cần được giữ lại. Điều này có lẽ quá rõ ràng vì đối với một câu hỏi thì các stop words như *What, Who, When, Where...* góp phần lớp trong việc quyết định phân loại. (danh sách các stop words có thể tham khảo tại phục lục A )

* + 1. ***N-gram***

Câu hỏi được coi như là một danh sách các từ liên tiếp nhau. Dựa trên ý tưởng như thế, ta có 3 loại n-gram thường được sử dụng là unigram, bigram, trigram.

+unigiram : tương tự bag of words.

+bigram : lấy lần lượt 2 từ liên tiếp nhau trong câu.

+trigram : lấy lần lượt 3 từ liên tiếp nhau trong câu.

* + 1. ***Tận dụng tính năng ngữ nghĩa của Wordnet***

Wordnet là một kho từ điển ngữ nghĩa của tiếng Anh. Các danh từ, động từ, tính từ được nhóm thành các các nhóm từ đồng nghĩa gọi là synset. Ngoài ra, còn kể đến cấu trúc từ bao hàm hypernym của Wordnet, một từ có liên quan đến một từ khác mang một nghĩa bao hàm rộng hơn.

Ví dụ cho một cấu trúc hypernym :

*dog, domestic dog, Canis familiaris*

*=> canine, canid*

*=> carnivore*

*=> placental, placental mammal, eutherian, eutherian mammal*

*=> mammal*

*=> vertebrate, craniate*

*=> chordate*

*=> animal, animate being, beast, brute, creature, fauna*

*=> ...*

Như ví dụ trên thì {dog, domestic dog, Canis familiaris} là một tập synset. Hypernym của *dog* là *canine.*Hypernym của *canine* là *carnivore.* Cấu trúc cứ thế đi từ nghĩa cụ thể đến nghĩa khái quát hơn :

*=> canine => carnivore => ... => animal => ...*

Do cách thức tổ chức từ vựng của Wordnet mà nó được sử dụng nhiều trong xử lý ngôn ngữ tự nhiên. Hypernym là một tính năng mà được sử dụng nhiều trong việc chọn làm đặc trưng phân loại. Nó có ý nghĩa khái quát hóa ý nghĩa của một từ, những từ có thể khác nhau nhưng có cùng một nghĩa khái quát chung nào đó.

* + 1. ***Nhận diện thực thể đặt tên***

Trong câu hỏi, các thực thể đặt tên là những danh từ riêng chỉ đến một nhân vật nào đó, tên một tổ chức, một ngôn ngữ, một từ viết tắt ... Để có thể hỗ trợ câu trả lời tổng quát, mà các câu trả lời có thể là những thực thể đặt tên (Named Entity) hay danh từ chung, thì có một hướng tiếp cận là sử dụng từ điển từ vựng Wordnet. Wordnet là một mô hình lớn, một Ontology đuợc xây dựng bằng tay thường được sử dụng rộng rãi trong việc xử lý ngôn ngữ tự nhiên. Nó cho phép phân loại nghĩa của từ và quan hệ cùng với 155327 từ vựng căn bản mà có thể được sử dụng cho việc phân loại ngữ nghĩa giúp cho việc phân loại thực thể.

Tuy nhiên kho từ điển vẫn còn hạn chế về số lượng các từ vựng. Cho nên, kho dữ liệu Wikipedia là một bách khoa toàn thư được sử dụng để mở rộng vốn từ vựng cho việc nhận diện thực thể đặt tên.

Các thực thể đặt tên sẽ được xếp vào một trong 25 lexicophaper trong Wordnet. Một danh từ bất kì trong wordnet đều thuộc một trong 25 nhóm này (bàng 2). Việc lấy đặc trưng sẽ dựa trên 25 nhóm này.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| person | cognition | time | event | feeling |
| communication | possession | attribute | quantity | shape |
| arfitact | location | object | motive | plant |
| act | subtance | process | animal | relation |
| food | state | phenomenon | body | group |

Bảng 2 – 25 lexicographer trong Wordnet

* + 1. **Các đặc trưng khác**

Trong bài báo gần đây, tác giả Huang đã đưa ra đặc trưng Headword dựa trên ý tưởng một từ trong câu hỏi đại diện cho một đối tượng cần hỏi đến [14]. Ngoài ra, Huang còn kết hợp với hypernym trong wordnet, đặc trưng hypernym phụ thuộc vào headword lấy được.Thêm vào đó, còn có các đặc trưng khác được nhắc tới như wh-word, word shape.

Trước đó, Skowron và Araki (2005) đã đề xuất ra ba đặc trưng : subordinate word category, Question focus, Syntactic-Semantic Structure [2]. Trong đó, đặc trưng subordinate word category cũng dựa trên hypernym của các danh từ trong câu hỏi, tìm ra một hypernym có mức độ khái quát cao nhất. Kế tiếp là đặt trưng Question focus, kết quả của đặc trưng này là rút ra một từ trọng tâm trong câu hỏi (focus word) thông qua một số các pattern (regular expression) và lấy từ đó làm đặc trưng. Cuối cùng là đặc trưng Syntactic-Semantic Structure dựa trên cấu trúc ngữ nghĩa của câu hỏi được lặp đi lặp lại, lấy cấu trúc của một câu hỏi làm đặc trưng.

Các đặc trưng trên đều do các tác giả tự đề xuất. Sau đó, chúng được đưa vào thử nghiệm và đánh giá. Các kết quả từ các đặc trưng trên khá thuyết phục. Vậy không có một sự giới hạn nào về số lượng các loại đặc trưng trong phân loại câu hỏi.

1. **Các nghiên cứu và ứng dụng liên quan** 
   1. ***LIBSVM***

LIBSVM là một bộ thư viện đơn giản dễ sử dụng và hiệu quả dành cho bộ phân loại SVM. Đây là một mã nguồn mở cung cấp cho nhiều ngôn ngữ khác nhau : Java, Python, Perl, Ruby ... Phiên bản mới nhất hiện nay là libsvm 2.91 dành cho Java. (<http://www.csie.ntu.edu.tw/~cjlin/libsvm/> )

Để bắt đầu sử dụng với bộ thư viện này, ta cần phải xây dựng một tập tin huấn luyện theo đúng dịnh dạng. Định dạng của tập tin chứa dữ liệu huấn luyện và tập tin kiểm thử là:

*<label> <index1>:<value1> <index2>:<value2>*

Trong đó:

*<label>* là giá trị đích của tập huấn luyện. Đối với việc phân loại, nó là một số nguyên xác định một lớp.

*<index>* là một số nguyên bắt đầu từ 1. Cụ thể trong bài toán phân loại nó đại diện cho các đặt trưng.

*<value>* là một số thực. Giá trị này thể hiện mức độ liên quan của đặc trưng đối với một phân loại nằm trong khoảng [-1,1]. Do các đặc trưng trong phân loại câu hỏi đều là đặc trưng nhị phân nên lúc huấn luyện giá trị này sẽ là 1.

Sau khi có được tập tin huấn luyện đúng định dạng, nhiệm vụ của libsvm là sẽ huấn luyện dựa trên tập tin định dạng và cho kết quả trả về là một tập tin có đuôi là *.model*. Tập tin này là mô hình xây dựng dựa trên việc huấn luyện. Từ đó ,ta chỉ việc xử dụng lại mô hình này để dự đoán các dữ liệu kiểm thử. ( Quá trình đưa dữ liệu kiểm thử cũng giống như huấn luyện, vẫn phải xây dựng tập tin kiểm thử theo định dạng như trên).

Đối với việc thiết lập các biến thông số, do số lượng đặc trưng trong phân loại câu hỏi lớn hơn số lượng trường hợp huấn luyện, ta nên để mặc định các thông số và chọn kernel cho SVM là LINENEAR (Huang 2008) [14].

* 1. ***Wordnet***

WordNet là một dạng từ điển ngữ nghĩa tiếng Anh, được tạo ra từ năm 1985 và đang được duy trì ở các phòng thí nghiệm khoa học nhận thức của Đại học Princeton dưới sự hướng dẫn của giáo sư tâm lý học A. George Miller. WordNet đã được hỗ trợ từ các Quỹ khoa học quốc gia, DARPA, các DTO (trước đây là Advanced Research and Development Activity), và REFLEX.

Đến năm 2006, cơ sở dữ liệu chứa khoảng 150.000 từ được tổ chức vào hơn 115.000 synsets cho tổng số 207.000 cặp word-sense; ở dạng nén, nó khoảng 12 M. WordNet phân biệt giữa danh từ, động từ, tính từ và trạng từ. Mỗi Synset chứa một nhóm các từ đồng nghĩa hoặc collocations (collocation một là một chuỗi các từ mà đi với nhau để tạo một ý nghĩa cụ thể, chẳng hạn như "car pool"); những nghĩa khác nhau của một từ nằm trong những synsets khác nhau. Ý nghĩa của synsets được làm rõ hơn với việc định nghĩa lời chú thích ngắn (gloss) (định nghĩa và / hay các câu ví dụ).

Các mối quan hệ trong Wordnet :

*+Danh từ*

*Holonym* (chứa): Y được gọi là Holonym với X nếu X là một phần của Y. Ví dụ: “book” có quan hệ holonym với “page”.

*Meronym* (thành phần): X được gọi là meronym với Y nếu X là một phần của Y. Ví dụ “page” có quan hệ meronym với “book”.

*Hypernym*: Y được xem là hypernym với X nếu X là một dạng của Y. Ví dụ: “animal” có quan hệ hypernym với “lion”.

*Hyponym*: X được xem là có quan hệ hyponym với Y nếu X là một dạng của Y. Ví dụ: “lion” có quan hệ hyponym với “animal”.

*Coordinate*: X và Y có quan hệ Coordinate nếu X và Y có cùng hypernym. Ví dụ: “lion” và “cat” có quan hệ coordinate với nhau vì có cùng hypernym là “animal”

*+Động từ*

*Hypernym*: Y hypernym với X khi hành động X là 1 dạng của Y, ví dụ: perceive có quan hệ hypernym với listen vì listen là 1 dạng perceive

*Coordinate*: X và Y có quan hệ Coordinate nếu X và Y có cùng hypernym.

*Entailment*: X có quan hệ Entailment với Y nếu X không thể hoàn thành cho đến khi Y hoàn thành. Ví dụ: “tiêu hóa” có quan hệ entailment với “ăn”.

*Troponym* (hyponym): X có quan hệ troponym với Y nếu hành động X là trường hợp (dạng, cách) khi làm hành động Y, ví dụ “nói ngọng” có quan hệ troponym với “nói”.

*+Tính từ*

*Simila*r:X và Y có quan hệ đồng nghĩa.

*Participle of verb*: mô tả tính từ có nguồn gốc từ động từ theo dạng hiện tại hoặc quá khứ, đây là quan hệ giữa tính từ và động từ. Ví dụ: collected là participle của collect.

*Related noun*: mô tả tính từ có nguồn gốc từ danh từ, là quan hệ giữa tính từ và danh từ. Ví dụ: weekly có quan hệ này với week.

*+Trạng từ*

*Root adj*: mô tả trạng từ có nguồn gốc từ tính từ, đây là quan hệ giữa trạng từ và tính từ. Ví dụ: kindly có quan hệ này với kind.

* 1. **POS tagger**

Trong phân loại câu hỏi, việc gán nhãn từ loại (POS tagger) cũng đóng một vai trò quan trọng. Các danh từ trong câu hỏi đại diện cho các đối tượng hay các thực thể cần hỏi tới. Vì thế, ta cần xác định từ loại của các từ trong câu hỏi. Đó là nhiệm vụ chính của việc gán nhãn từ loại.

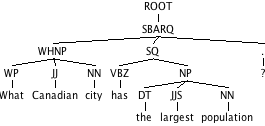
Wordnet là một trong số những công cụ ta có thể sử dụng trong việc gán nhãn từ loại. Ngoài ra chương trình GATE cũng có hỗ trợ tron vấn đề này. Wordnet và GATE đều là hai công cụ phổ biến trong xử lý ngôn ngữ tự nhiên. Vậy gán nhãn từ loại cũng đóng một vai trò khá quan trọng trong xử lý ngôn ngữ tự nhiên. (Các nhãn từ loại ở đây theo hệ thống Penn Treebank , tham khảo phụ lục B).

* 1. **Một số vấn đề liên quan khác**

*+Bộ phân tích cú pháp (parser)*

Nhiệm cụ của bộ phân tích cú pháp là phân tích một câu đưa vào thành các thành phần như chủ từ, động từ, chủ ngữ, động ngữ, .... Kết quả trả về của một bộ phân tích cú pháp sẽ là một cây cú pháp có nút gốc là ROOT. Các nút khác là các thành phần trong câu như đã nói trên kèm theo đó là các nhãn từ loại. Mỗi từ trong câu đóng vai trò như một nút lá.

Ví dụ : một cây cú pháp cho câu “What Canadian city has the largest population ?”



Vai trò bộ phân tích cú pháp trong việc phân loại là nó sẽ phân tích các thành phần cú pháp trong câu và chỉ ra mối quan hệ giữa các từ trong câu. Điều này sẽ giúp ích phần nào trong việc rút trích các đặc trưng trong câu.

Một số bộ phân tích cú pháp mà nhóm biết tới là Stanford Parser và Berkerley Parser. Cả hai bộ phân tích cú pháp này đều là mã nguồn mở và được viết trên ngôn ngữ java.

*+Biểu thức thông dụng ( regular expression) :*

Khái niệm regular expression xuất hiện trong khá nhiều ngôn ngữ lập trình như Java, Ruby, Perl, PHP ... Trong mỗi ngôn ngữ, nó lại được định nghĩa khác nhau. Tuy nhiên xét về mặt chung nhất, regular expression là một chuỗi được dùng để miêu tả hoặc so khớp với một tập các chuỗi khác dựa trên những luật cú pháp. (Theo định nghĩa của Wikipedia <http://en.wikipedia.org/wiki/Regular_expression_examples> ).

Việc tạo ra các biểu thức nhằm để so sánh, hay nói đúng hơn là so khớp nó với một đoạn văn bản hoặc một chuỗi nào đó hoặc tìm kiếm xem vị trí của chuỗi nào trong văn bản phù hợp với điều kiện mà biểu thức đề ra .Trong xử lý ngôn ngữ tự nhiên, regular expression được sử dụng để kiểm trang các định dạng như : ngày tháng, một địa chỉ email, kiểu số ...

Việc tạo ra các biểu thức giúp ích rất nhiều trong việc tìm kiếm một câu, một chữ trong văn bản , hoặc thực hiện thao tác như đếm tầng số xuất hiện chúng .Nhưng để tạo ra những biểu thức này cần nắm được những cú pháp về regular expression.(Cú pháp của regular expression có thể tham khảo tại phụ lục C ).

1. **Hướng tiếp cận của nhóm**
   1. ***Dữ liệu huấn luyện và kiểm thử***

Hiện nay, có hai tập dữ liệu câu hỏi thường được dùng là TREC và UIUC. Đối với tập dữ liệu TREC cung cấp các loại câu hỏi dưới dạng các tập tin theo định dạng giống như XML. Trên trang web của UIUC thì cung cấp tập tin danh sách các câu hỏi mà trong đó các câu hỏi đã được gán nhãn phân loại sẵn. Ví dụ : “DESC:def What is compounded interest ?”. Ngoài ra, các tập tin được sắp xếp theo thứ tự 1000,2000, 3000,4000 và 5500 câu hỏi đã được gán nhãn. Thêm vào đó, UIUC cung cấp một tập tin để kiểm tra gồm 500 câu hỏi trong TREC 10. Từ đó, nhóm quyết định chọn tập huấn luyện dựa trên kho dữ liệu câu hỏi UIUC vì những tiện dụng nêu trên.

* 1. ***Phương pháp***

Phương pháp máy học là một phương pháp được sử dụng nhiều trong các bài báo [1,2,3,12]. Việc xây dựng bộ phân loại trên phương pháp máy học sẽ ít tốn thời gian hơn so với phương pháp đề ra các luật tay thủ công. Và khi cần áp dụng một đặc trưng mới, ta chỉ cần huấn luyện lại bộ phân loại. Vì thế, phương pháp máy học được chọn làm hướng tiếp cận của nhóm trong phân loại câu hỏi.

Đối với bộ phân loại, nhóm chọn SVM để phân loại câu hỏi. Hai tác giả Lee và Zhang (2003) đã chứng minh được sự vượt trội của nó so với các bộ phân loại khác. Ngoài ra, SVM cũng là một phương pháp máy học thường được các tác giả chọn để sử dụng.

Để xây dựng bộ phân loại SVM, thư viện LIBSVM được áp dụng trong quá trình huấn luyện và kiểm thử.

* 1. ***Đặc trưng***

Rõ ràng, giới hạn về số lượng các loại đặc trưng là không có. Vì thế, nhóm sẽ đề ra một số đặc trưng mới dựa theo một số đặc trưng của các tác giả. Kho từ điển Wordnet rất được coi trọng và được sử dụng nhiều trong việc rút đặc trưng. Cụ thể nhóm sẽ tận dụng mối quan hệ hypernym trong wordnet. Huang (2008) đã cho thấy mức độ hiệu quả khi sử dụng trực tiếp hypernym kết hợp với đặc trưng Head word ( đạt độ chính xác 89% trong phân lớp mịn). Skowron và Araki (2005) cho thấy việc hiệu quả khi kết hợp các đặc trưng với nhau .Cho nên, hướng tiếp cận của nhóm cũng sẽ kết hợp các đặt trưng lại với nhau, hy vọng sẽ nâng cao độ chính xác từ 1-2%.

Ngoài ra, một số đặc trưng khác như : thực thể đặt tên, wh-word, n-gram ... cũng đáng được lưu ý và tận dụng.

* 1. ***Mô hình cho bài toán phân loại câu hỏi***

# Phụ lục

## A. Danh sách các stop words

Bảng danh sách các stopword trong tiếng anh :

(lấy từ trang <http://armandbrahaj.blog.al/2009/04/14/list-of-english-stop-words/> )

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| STT | List of stop words | | | |
| 1  2  3  4  5  6  7  8  9  10  11  12  13  14  15  16  17  18  19  20  21  22  23  24  25  26  27  28  29  30  31  32  33  34  35  36  37  38  39  40  41  42  43  44  45  46  47  48  49  50  51  52  53  54  55  56  57  58  59  60  61  62  63  64  65  66  67  68  69  70  71  72  73  74  75  76  77  78  79  80 | a  about  above  across  after  afterwards  again  against  all  almost  alone  along  already  also  although  always  am  among  amongst  amoungst  amount  an  and  another  any  anyhow  anyone  anything  anyway  anywhere  are  around  as  at  back  be  became  because  become  becomes  becoming  been  before  beforehand  behind  being  below  beside  besides  between  beyond  bill  both  bottom  but  by  call  can  cannot  cant  co  computer  con  could  couldnt  cry  de  describe  detail  do  done  down  due  during  each  eg  eight  either  eleven  else | elsewhere  empty  enough  etc  even  ever  every  everyone  everything  everywhere  except  few  fifteen  fify  fill  find  fire  first  five  for  former  formerly  forty  found  four  from  front  full  further  get  give  go  had  has  hasnt  have  he  hence  her  here  hereafter  hereby  herein  hereupon  hers  herse”  him  himse”  his  how  however  hundred  i  ie  if  in  inc  indeed  interest  into  is  it  its  itse”  keep  last  latter  latterly  least  less  ltd  made  many  may  me  meanwhile  might  mill  mine  more | moreover  most  mostly  move  much  must  my  myse”  name  namely  neither  never  nevertheless  next  nine  no  nobody  none  noone  nor  not  nothing  now  nowhere  of  off  often  on  once  one  only  onto  or  other  others  otherwise  our  ours  ourselves  out  over  own  part  per  perhaps  please  put  rather  re  same  see  seem  seemed  seeming  seems  serious  several  she  should  show  side  since  sincere  six  sixty  so  some  somehow  someone  something  sometime  sometimes  somewhere  still  such  system  take  ten  than  that | the  their  them  themselves  then  thence  there  thereafter  thereby  therefore  therein  thereupon  these  they  thick  thin  third  this  those  though  three  through  throughout  thru  thus  to  together  too  top  toward  towards  twelve  twenty  two  un  under  until  up  upon  us  very  via  was  we  well  were  what  whatever  when  whence  whenever  where  whereafter  whereas  whereby  wherein  whereupon  wherever  whether  which  while  whither  who  whoever  whole  whom  whose  why  will  with  within  without  would  yet  you  your  yours  yourself  yourselves |

## B. Nhãn từ loại

Danh sách các nhãn từ loại trong hệ thống Penn Treebank :

(<http://www.ims.uni-stuttgart.de/projekte/CorpusWorkbench/CQP-HTMLDemo/PennTreebankTS.html> )

|  |  |  |
| --- | --- | --- |
| STT | Từ loại | Giải thích |
| 1. | CC | Coordinating conjunction |
| 2. | CD | Cardinal number |
| 3. | DT | Determiner |
| 4. | EX | Existential there |
| 5. | FW | Foreign word |
| 6. | IN | Preposition or subordinating conjunction |
| 7. | JJ | Adjective |
| 8. | JJR | Adjective, comparative |
| 9. | JJS | Adjective, superlative |
| 10. | LS | List item marker |
| 11. | MD | Modal |
| 12. | NN | Noun, singular or mass |
| 13. | NNS | Noun, plural |
| 14. | NP | Proper noun, singular |
| 15. | NPS | Proper noun, plural |
| 16. | PDT | Predeterminer |
| 17. | POS | Possessive ending |
| 18. | PP | Personal pronoun |
| 19. | PP$ | Possessive pronoun |
| 20. | RB | Adverb |
| 21. | RBR | Adverb, comparative |
| 22. | RBS | Adverb, superlative |
| 23. | RP | Particle |
| 24. | SYM | Symbol |
| 25. | TO | to |
| 26. | UH | Interjection |
| 27. | VB | Verb, base form |
| 28. | VBD | Verb, past tense |
| 29. | VBG | Verb, gerund or present participle |
| 30. | VBN | Verb, past participle |
| 31. | VBP | Verb, non-3rd person singular present |
| 32. | VBZ | Verb, 3rd person singular present |
| 33. | WDT | Wh-determiner |
| 34. | WP | Wh-pronoun |
| 35. | WP$ | Possessive wh-pronoun |
| 36. | WRB | Wh-adverb |

## C. Cú pháp trong Regular Expression

Danh sách các cú pháp bên dưới dựa trên cú pháp của ngôn ngữ Perl Đối với các ngôn ngữ lập trình khác hầu như là giống nhau. Các phiên bản mới của regular expression theo chuẩn ISO/IEC 9945-2:1993.

(<http://www.wdvl.com/Authoring/Languages/Perl/PerlfortheWeb/perlintro2_table1.html>)

|  |  |  |
| --- | --- | --- |
| Cú pháp | Ý nghĩa | Ví dụ |
| *.* | Bất kỳ một ký tự nào | "do." -> "dog", "dot", "doe” ,... "d..r" -> "door","deer",… |
| \* | Không hoặc có bất kỳ ký tự nào trước đó | "do.\*" -> "dog", "done", "doppleganger",…  "to\*" -> "to" ,"too" |
| + | Một hoặc bất kỳ ký tự nào trước đó | "fre+.." -> "freak", "freeze", "fresh" |
| ? | Không có hoặc có một ký tự trước đó | "ton?e" -> "toe" ,"tone" |
| () | Nhóm lại | (dog|cat) -> "dog" ,"cat" |
| [] | Bất kì ký tự nào trong tập | "ta[pb]" -> "tap" , "tab"  "r[aeiou]t" -> "rat", "ret", "rot", "rut" |
| [^] | Không chứ bất kì ký tự nào trong tập | "t[^aeiou]+.\*s" -> "thanks", "this", "trappings", ... |
| {min,max} | Phạm vi xuất hiện | [a-z]{3} : ba ký tự thường liên tiếp  [0-9]{3} : ba chữ số liên tiếp  [A-Z]{2,5} : các chữ cái in hoa xuấ hiện liên tiếp ở thời điểm 2,3,4,5 |

Ngoài các cú pháp cơ bản trên , regular expression còn có các lớp ký tự được gọi là Character Class mà ta rất thường sử dụng .

|  |  |
| --- | --- |
| Character Classes | Ý nghĩa |
| \d | Bất kì ký tự nào thuộc 0-9 |
| \D | Bất kì ký tự nào không thuộc 0-9 , ngược lại với \d |
| \w | Bất kì ký tự nào cả chữ lẫn số [a-zA-Z0-9] |
| \W | Ngược lại với \w |
| \s | Khoảng trắng |
| \S | Ngược lại với \s |

Vậy ta có thể thấy rằng :

\D tương đương [^\d]

\W tương đương [^\w]

\S tương đương [^\s]

Ngoài ra còn có một số ký hiệu khác gọi là Anchor Sequences (tạm dịch là “đánh dấu trình tự” )

|  |  |
| --- | --- |
| Anchor Sequences | Ý nghĩa |
| ^ | Bắt đầu của một chuỗi ký tự |
| $ | Kết thúc chuỗi ký tự |
| \b | Biên (giới) của từ |
| \B | Ngược lại , bất kỳ đâu trừ biên của từ |

# Tài liệu tham khảo:

**[1] Question Classification by Ensemble Learning, LiXin, Huang Xuan Jing,Wu Lid-de, 2006**

**[2] Effectiveness of Combined features for machine learning based question classification ,Marcin Skowron and Kenji Araki , 2005**

**[3] Question Classification in Social Media – 2009**

**[4] Minimally supervised question classification and answering base on wordnet and wikipedia\_2009**

**[5] Learning Surface Text Patterns for a Question Answering System,Deepak Ravichandran and Eduard Hovy, 2002**

**[6] YAGO: A Core of Semantic Knowledge Unifying WordNet and Wikipedia**

**[7]** [**http://gate.ac.uk/releases/gate-5.1-build3431-ALL/doc/tao/splitap7.html#x32-609000G**](http://gate.ac.uk/releases/gate-5.1-build3431-ALL/doc/tao/splitap7.html#x32-609000G)

**[8]** [**http://www.wdvl.com/Authoring/Languages/Perl/PerlfortheWeb/perlintro2\_table1.html**](http://www.wdvl.com/Authoring/Languages/Perl/PerlfortheWeb/perlintro2_table1.html)

**[9]** [**http://java.sun.com/docs/books/tutorial/essential/regex/intro.html**](http://java.sun.com/docs/books/tutorial/essential/regex/intro.html)

**[10] Xử Lý Ngôn Ngữ Tự Nhiên, Đinh Điền, 2006 , trang 239**

**[11] Trang chủ của GATE** [**http://gate.ac.uk/**](http://gate.ac.uk/)

**[12] Question Classification using Support Vector Machine, Dell Zhang and Wee Sun Lee, 2003**

**[13] Learning Question Classifier, Li and Roth, 2002**

**[14] Question classification using Head word and their Hypernyms, Zhiheng Huang, Marcus Thint and Zeng Chang Qin ,2008**