Chương 2: LÝ THUYẾT NỀN TẢNG

1. Lý thuyết nền tảng của dịch máy:

1.1. Định nghĩa:

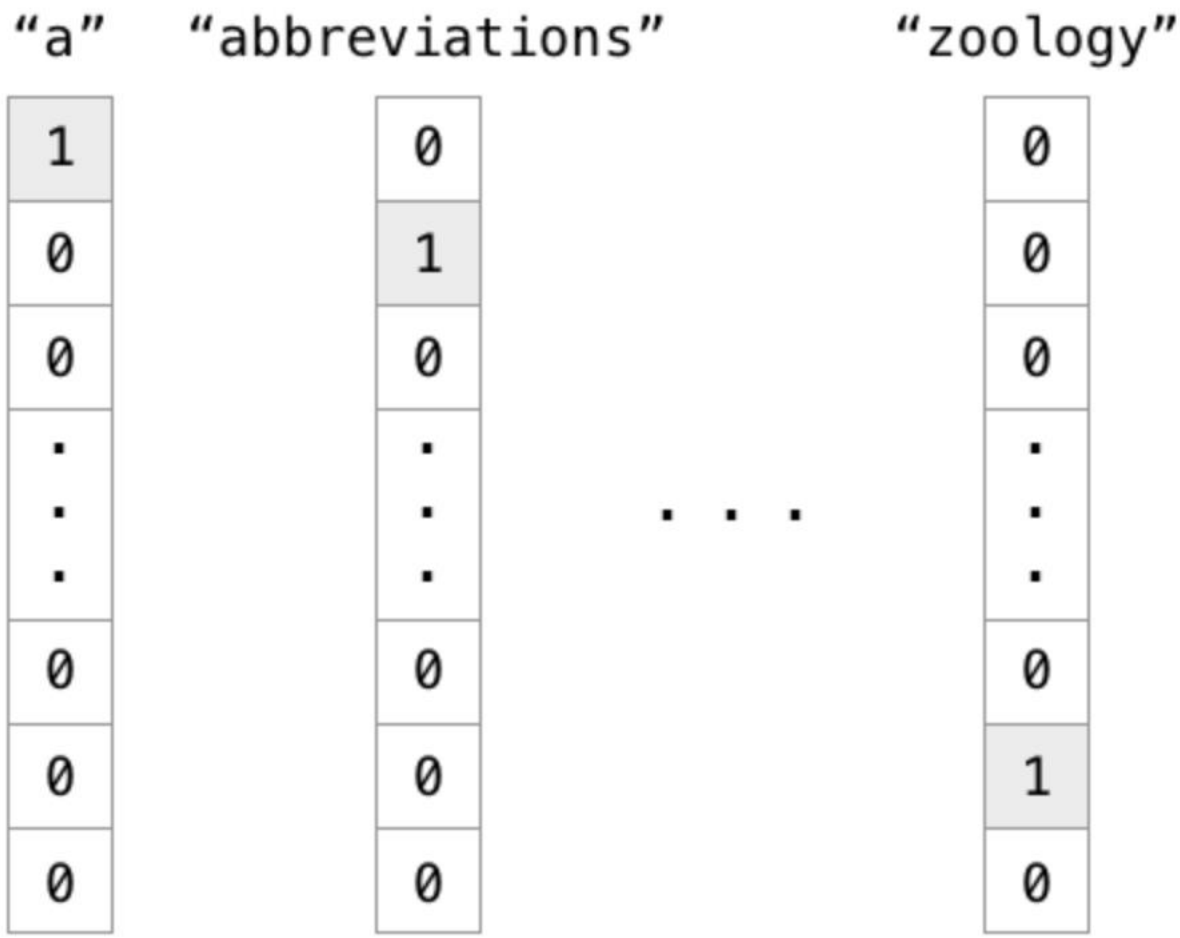
Phần này trình bày một số khái niệm cốt lõi về dịch máy, các mô hình ngôn ngữ và phương pháp học theo đặc trưng trong xử lý ngôn ngữ tự nhiên(NLP).

1.1.1. Định nghĩa dịch máy:

Dịch máy (machine translation) là một quá trình thay đổi văn bản từ ngôn ngữ này sang ngôn ngữ khác (gọi là ngôn ngữ đích) một cách tự động, không có sự can thiệp của con người trong quá trình dịch.

1.1.2. Word embeddings:

- Xử lý đầu vào cho bài toán dịch máy là một bước rất quan trọng, các thuật toán, vì các kiến trúc Machine learning, Deep learning chúng chỉ có thể hiểu được đầu vào ở dạng là số nên cần chuyển đầu vào ở dạng text sang dạng số để chúng có thể hiểu được.
- Nhưng nếu chỉ đơn giản biểu diễn từ bằng một con số có thể dẫn đến sai lệch mối quan hệ ngữ nghĩa giữa các từ.Ví dụ như nếu đánh dấu “mèo” là số 1 và “chó” là số 2, như vậy “mèo” + “mèo” = “chó”.
- Một kỹ thuật đơn giản được sử dụng để khác phục là One-hot vector, chúng chuyển các từ thành vector có số chiều bằng số từ của bộ từ vựng đầu vào, trong đó chỉ có duy nhất một phần tử bằng 1 (các phần tử khác bằng 0) tương ứng với vị trí từ đó trong bộ từ vựng. Tuy nhiên cách biểu diễn này là số chiều của vector lại rất lớn, ảnh hưởng đến quá trình xử lý và lưu trữ.



Hình 1: Hình mô tả cách mã hóa one-hot-vector

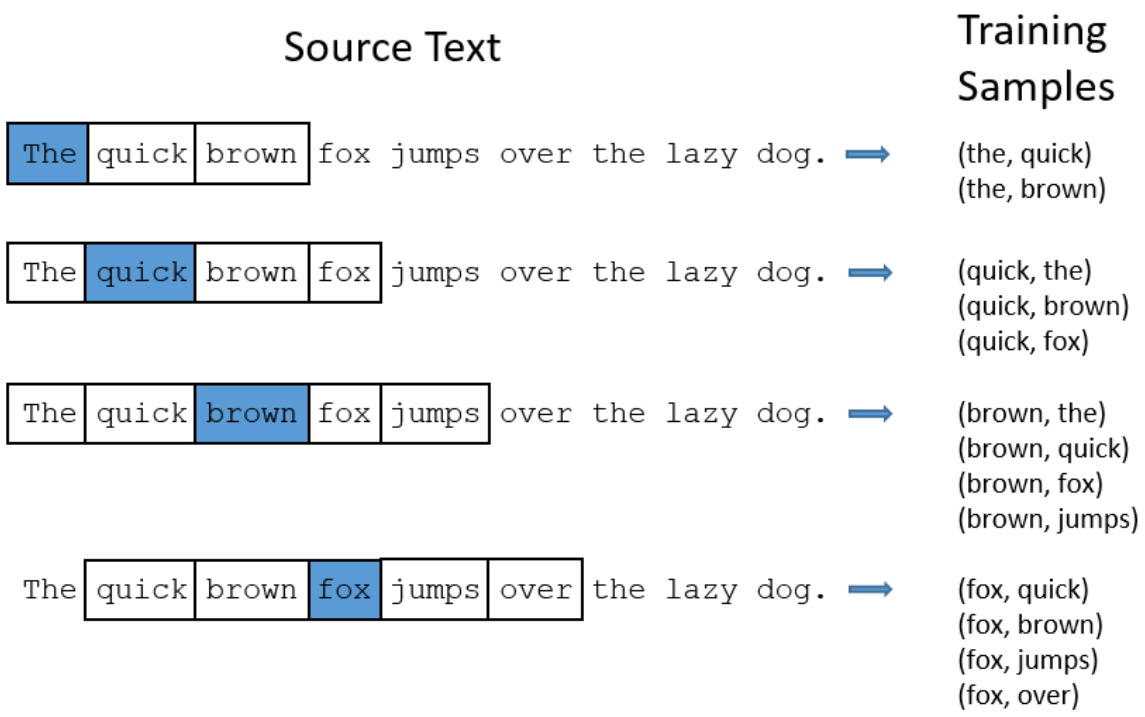
- Một cách khác là xử dụng vector ngẫu nhiên, mỗi từ được biểu thị bằng một vector có giá trị các chiều là ngẫu nhiên, mỗi từ là một điểm trong không gian 3D, do đó làm giảm số chiều vector, tuy nhiên nó lại không biểu diễn quan hệ tương đồng giữa các từ.
- Sử dụng **Word embeddings** được coi là cách tốt nhất để thể hiện các từ trong văn bảnm nó cũng gán mỗi từ với một vector nhưng các vector được tính toán để biểu diễn quan hệ tương đồng giữa các từ.
- Word embeddings có 2 model nổi tiếng là Word2vec và Glove.

1.1.3. Word2vec:

Word2vec là một model unsupervised learning nó dùng để thể hiện mỗi quan hệ giữa các từ, nó được kết hợp từ hai thuật toán Skip-gram và Continuous bag of words (CBOW).

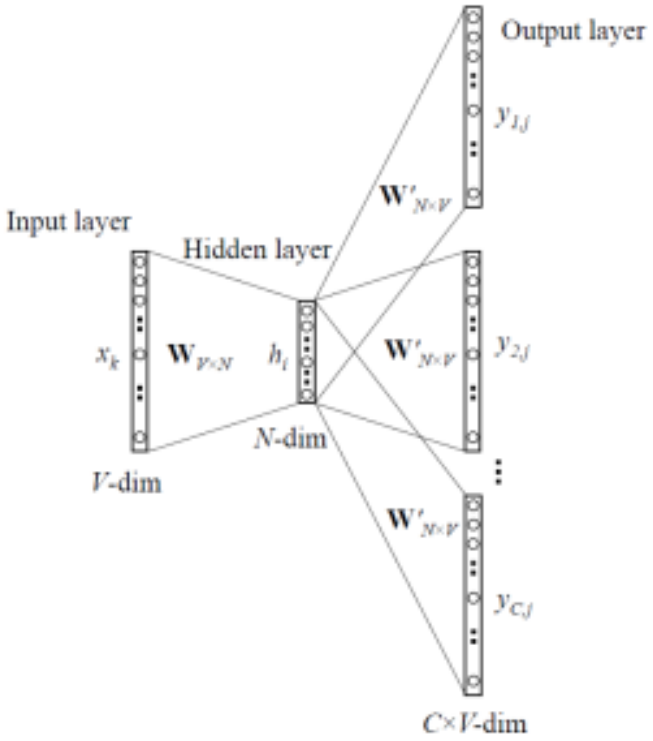
1.1.3.1. Skip-gram:

Ý tưởng chính của mô hình này là xác định các từ xung quanh từ mục tiêu trong một khoảng nhất định gọi là 'window'.



Hình 2: Hình Mô Tả Trainning Với Window Bằng 2. (Nguồn: Leonardo Barazza)

Đối với Skip-gram, đầu vào là từ đích, trong khi đầu ra là các từ xung quanh từ đích. Tất cả dữ liệu đầu vào và đầu ra có cùng kích thước được mã hóa bằng one-hot. Mạng chứa một lớp ẩn có kích thước bằng kích thước nhúng, nhỏ hơn vector đầu vào và đầu ra. Ở cuối lớp đầu ra, một hàm kích hoạt softmax được áp dụng sao cho mỗi phần tử của vector đầu ra mô tả khả năng một từ cụ thể sẽ xuất hiện trong ngữ cảnh.



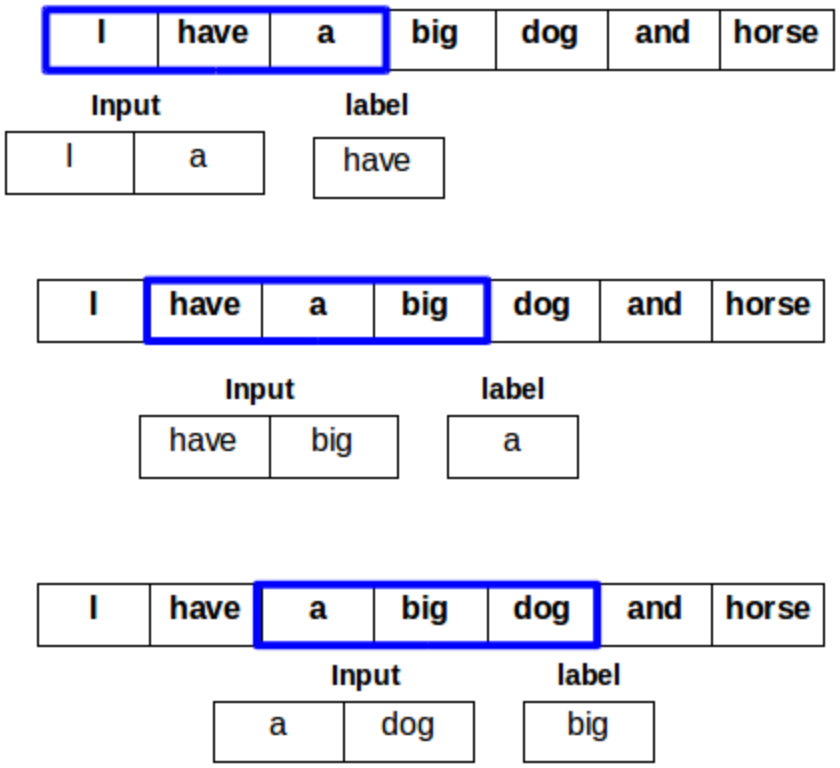
Hình 3: Hình mô tả cấu trúc mạng của Skip-gram

(Nguồn https://www.analyticsvidhya.com/blog/2017/06/word-embeddings-count-word2veec/)

Với skip-gram, kích thước biểu diễn từ giảm từ kích thước bằng số từ trong bộ từ vựng xuống bằng chiều dài lớp ẩn. Hơn nữa các vector có ý nghĩa nhiều hơn về mặt mô tả mối quan hệ giữa các từ.

1.1.3.2. Continuous bag of words (CBOW).

Ngược lại với Skip-gram nó hoán đổi đầu vào và đầu ra, ý tưởng của thuật toán CBOW là đưa ra một bối cảnh và cho biết từ nào có khả năng xuất hiện nhiều nhất trong đó.



Hình 4: Hình Mô Tả Trainning Với Window Bằng 2. (Nguồn: Nguyễn Trường Long Blog)

1.1.4. Beam Search

1.1.5. Bleu Score

1.2. Lý thuyết nền tảng mạng nơ-ron(Neural Networds)

1.2.1. Mô Tả mạng nơ-ron:

Mạng nơ-ron là một tập hợp các mô hình toán học được được xây dựng dựa trên tập hợp các nút, được kết nối với các hàm kích hoạt phi tuyến tính cùng các tham số có khả năng học. Mạng nơ-ron hiện là mô hình phổ biến nhất được sử dụng cho các ứng dụng máy học trong một loạt các lĩnh vực như thị giác máy tính, nhận dạng giọng nói, xử lý ngôn ngữ tự nhiên,…

Một tế nào (nút) của mạng nơ-ron là một hàm của tập các trọng số tương ứng với các giá trị đầu vào (inputs) $\{x_0, \dots, x_N\}$.

$$y = a\left(\sum_{i}^{N} w_i x_i + b\right)$$

Trong đó:

- w_i : trọng số của đầu vào x_i
- a: hàm kích hoạt (activation function)
- b: độ sai lệch (bias)

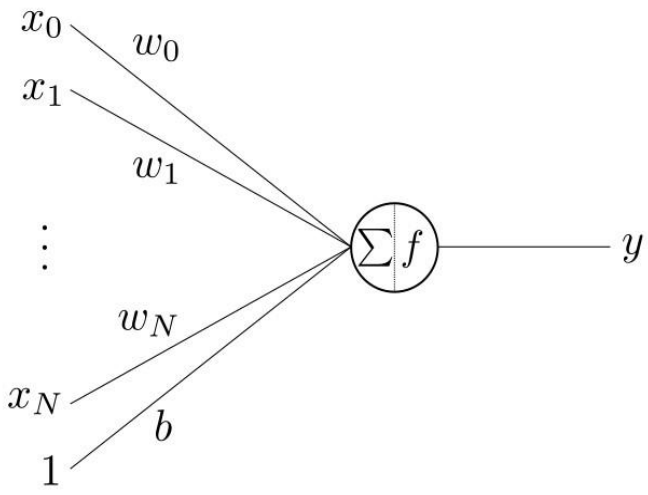
Ta sẽ sử dụng kí hiệu ma trận để làm đơn giản cách thể hiện, trong đó mỗi tế bào nơ-ron bao gồm một vector đầu vào $x = \{x_0, \dots, x_N\}$, một vector trọng số $w = \{w_0, \dots, w_N\}$ và một vector sai lệch b, khi đó đầu ra là:

$$y = a(w^T x + b)$$

Nếu hàm kích hoạt a là một biến thể của hàm Heaviside,

$$a(x) = \begin{cases} 1, & x \geq 0 \\ 0 \text{ hoặc } -1, & x < 0 \end{cases}$$

thì tế bào nơ-ron này được gọi là một perceptron, một bộ phân loại nhị phân đơn giản, là một trong những phương pháp học kết nối sớm nhất được phát minh bởi Rosenblatt.



Hình 5: Minh họa kiến trúc điển hình của một tế bào mạng nơ-ron

1.2.2. Hàm kích hoạt (Activation function)

Hàm kích hoạt là phần rất quan trọng trong mạng nơ-ron, đặc biệt là mạng nơ-ron nhiều lớp ẩn. Nếu không có hàm kích hoạt phi tuyến tính, cho dù mạng nơ-ron có nhiều lớp ẩn đến cỡ nào thì cũng chỉ có sức mạnh đại diện cho phân loại tuyến tính, điều này tương đương với một mạng mà không có lớp ẩn nào. Vì bản chất tổng hợp các hàm tuyến tính là một hàm tuyến tính. Do đó, hàm kích hoạt a là một hàm phi tuyến tính được áp dụng cho đầu ra tại mỗi nút và input data cho tầng tiếp theo, cho phép mạng nơ-ron nhiều lớp ẩn học các hàm phi tuyến phức tạp.

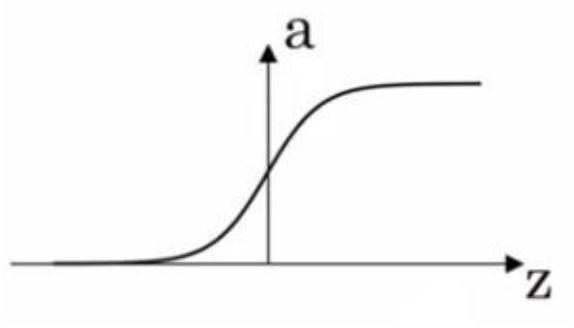
Hàm kích hoạt phổ biến là hàm sigmon, nó là một hàm phi tuyến với đầu vào là các số thực cho kết quả nằm trong khoảng từ 0 đến 1, phù hợp cho các mạng phân loại nhị phân (nổi tiếng như là thuật toán Logistic Regression), nó có công thức theo phương trình:

$$z^{[i]} = W^{[i]}x + b^{[i]}$$

$$a = \frac{1}{1 + e^{-z}}$$

Trong đó:

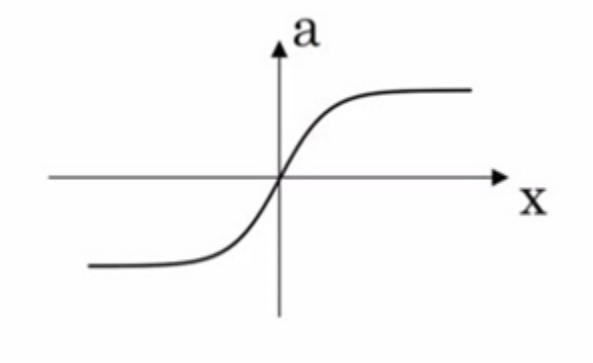
- $W^{[i]}$: trọng số tại lớp thứ i
- $b^{[i]}$: tham số sai lệch tại lớp thứ i
- a: hàm kích hoạt



Hình 6: Minh họa hàm kích hoạt sigmoid. (Nguồn: Coursera Sequence Models)

Ngoài ra, có một hàm kích hoạt luôn hoạt động tốt hơn hàm sigmoid là hàm tiếp tuyến hyperbolic (hyperbolic tangent function), ánh xạ các giá trị đầu vào vào khoảng biên từ -1 đến 1. Có công thức là:

$$a = \tanh(x) = \frac{e^{z} - e^{-z}}{e^{z} + e^{z}}$$

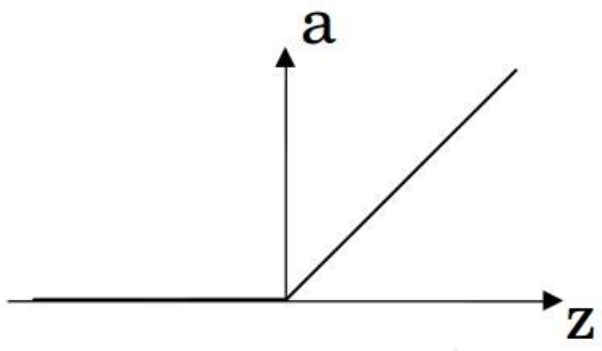


Hình 7 Minh họa hàm kích hoạt tanh. (Nguồn: Coursera Sequence Models)

Tuy nhiên, một vấn đề với hàm kích hoạt sigmoid và tanh nếu z quá lớn hoặc quá nhỏ thì độ dốc của hàm sẽ rất nhỏ, điều này làm chậm quá trình tìm điểm cực tiểu của hàm chi phí, dẫn đến làm chậm quá trình học. Vì lý do

này, dựa vào các kết quả thực nghiệm được cải thiện, mạng nơ-ron hiện đại có xu hướng sử dụng hàm kích hoạt đơn vị tuyến tính chỉnh lưu (ReLu - Rectified Linear Unit). Có công thức như sau:

$$a = max\ (0,z)$$



Hình 8: Minh họa hàm kích hoạt ReLu. (Nguồn: Coursera Sequence Models)

Vì vậy, đạo hàm luôn bằng 1 nếu z dương, và bằng 0 nếu z âm. Dựa trên thực nghiệm, sử dụng hàm kích hoạt ReLu, mạng nơ-ron sẽ học nhanh hơn so với khi dùng với hàm sigmoid hoặc hàm tanh. Lý do chính là có ít hơn sự ảnh hưởng của độ dốc hàm bằng 0 làm chậm việc học. Vì mặc dù, có một nửa phạm vi của z làm độ dốc hàm ReLu bằng 0, nhưng trong thực tế, đủ các đơn vị ẩn thì ta sẽ có z lớn hơn 0, vì vậy việc học vẫn khá nhanh với hầu hết các ví dụ đào tạo.

1.2.3. Lan truyền ngược (Back propagation)

Các thuật toán học sâu tương phản với các thuật toán học nông bởi số biến đổi được tham số hóa một tín hiệu gặp phải khi nó lan truyền từ các lớp đầu vào đến các lớp đầu ra. Mỗi chuỗi các biến đổi từ đầu vào đến đầu ra gọi là một đường gán kế thừa (CAP - Credit Assignment Path). Vấn đề gán kế thừa (Credit Assignment Problem) được giải quyết với khám phá lan truyền ngược (backpropagation), cho phép học với mạng nơ-ron nhiều lớp.

1.2.4. Lan truyền tới (Forward Propagation)

1.2.5. Phương pháp giảm độ dốc với Gradient desent và các biến thể

1.3. Phương pháp giảm độ dốc với động lượng (Momentum)

1.4. Các phương pháp huấn luyện mạng nơ-ron hiện đại

1.4.1. Hàm kích hoạt đơn vị tuyến tính chỉnh lưu (Rectified Linear Unit)

1.4.2. Phương pháp chuẩn hóa theo lô (Batch Nomalization)

1.4.3. Phương pháp cắt giảm (Dropout)

1.5. Các kiến trúc mạng nơ-ron hồi quy:

1.5.1. Mạng nơ-ron hồi quy (RNN – Recurrent Neural Network)

1.5.2. Mạng bộ nhớ dài ngắn (LSTM - Long Short Term Memory)

1.5.3. Mạng nơ-ron hồi quy hai chiều (BIRNN - Bidirectional Recurrent Neural Network)

1.5.4. Mạng nơ-ron hồi quy học sâu (Deep RNN – Deep Reccurent Neural Network)

1.6. Các kỹ thuật trong dịch máy

Cơ chế Attention (Attention Mechanism)

2. Hệ thống dịch máy