北京大學
PEKING UNIVERSITY

Basics in Language Models

Kun Yuan

Center for Machine Learning Research @ Peking University

Feb. 29, 2024

Contents

- Word embedding
- Language models
- Recurrent neural network
- Sequence-to-sequence model

Some of the materials are from a great course [1]

[1] The deep learning specialization

1-hot word representation

- Vocabulary set = {a, aaron, …, zulu}; the size is typically on the order of 10,000

- 1-hot representation is the most natural idea to represent word

$$
\begin{array}{cccccc}
\text{Man} & \text{Woman} & \text{King} & \text{Queen} & \text{Apple} & \text{Orange} \\
(5391) & (9853) & (4914) & (7159) & (456) & (6527) \\
\begin{bmatrix} 0 \\ 0 \\ 0 \\ 0 \\ \vdots \\ 1 \\ \vdots \\ 0 \\ 0 \end{bmatrix} &
\begin{bmatrix} 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ \vdots \\ 1 \\ \vdots \\ 0 \end{bmatrix} &
\begin{bmatrix} 0 \\ 0 \\ 0 \\ \vdots \\ 1 \\ \vdots \\ 0 \\ 0 \\ 0 \end{bmatrix} &
\begin{bmatrix} 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ \vdots \\ 1 \\ \vdots \\ 0 \end{bmatrix} &
\begin{bmatrix} 0 \\ \vdots \\ 1 \\ \vdots \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \end{bmatrix} &
\begin{bmatrix} 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ \vdots \\ 1 \\ \vdots \\ 0 \end{bmatrix}
\end{array}
$$

1-hot word representation

- 1-hot representation ignores semantic relationship

I like orange juice. → I like apple ____.

- “Orange” and “apple” should be close to each other. Language model should fill in “juice”

- But in one-hot representation, apple and orange are not close to each other

$$
\begin{array}{cccccc}
\text{Man} & \text{Woman} & \text{King} & \text{Queen} & \text{Apple} & \text{Orange} \\
(5391) & (9853) & (4914) & (7159) & (456) & (6527) \\
\begin{bmatrix} 0 \\ 0 \\ 0 \\ 0 \\ \vdots \\ 1 \\ \vdots \\ 0 \\ 0 \end{bmatrix} &
\begin{bmatrix} 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ \vdots \\ 1 \\ \vdots \\ 0 \end{bmatrix} &
\begin{bmatrix} 0 \\ 0 \\ 0 \\ \vdots \\ 1 \\ \vdots \\ 0 \\ 0 \\ 0 \end{bmatrix} &
\begin{bmatrix} 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ \vdots \\ 1 \\ \vdots \\ 0 \end{bmatrix} &
\begin{bmatrix} 0 \\ \vdots \\ 1 \\ \vdots \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \end{bmatrix} &
\begin{bmatrix} 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ \vdots \\ 1 \\ \vdots \\ 0 \end{bmatrix}
\end{array}
$$

Semantic representation

- Each word is represented with vectors that involve semantics

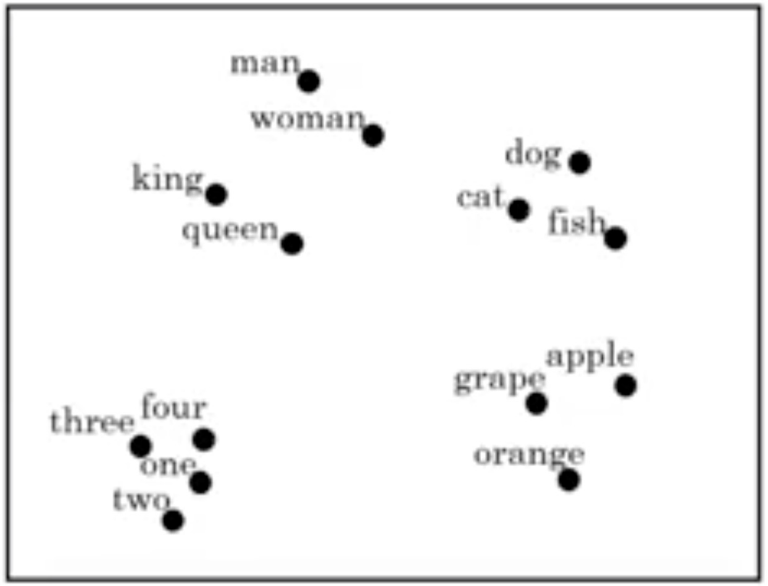
	Man (5391)	Woman (9853)	King (4914)	Queen (7159)	Apple (456)	Orange (6527)
Gender	-1.00	1.00	-0.95	0.97	0.00	0.01
Royal	0.01	0.02	0.93	0.95	-0.01	0.00
Age	0.51	0.47	0.7	0.69	0.03	-0.02
Food	0.04	0.01	0.02	0.01	0.95	0.97

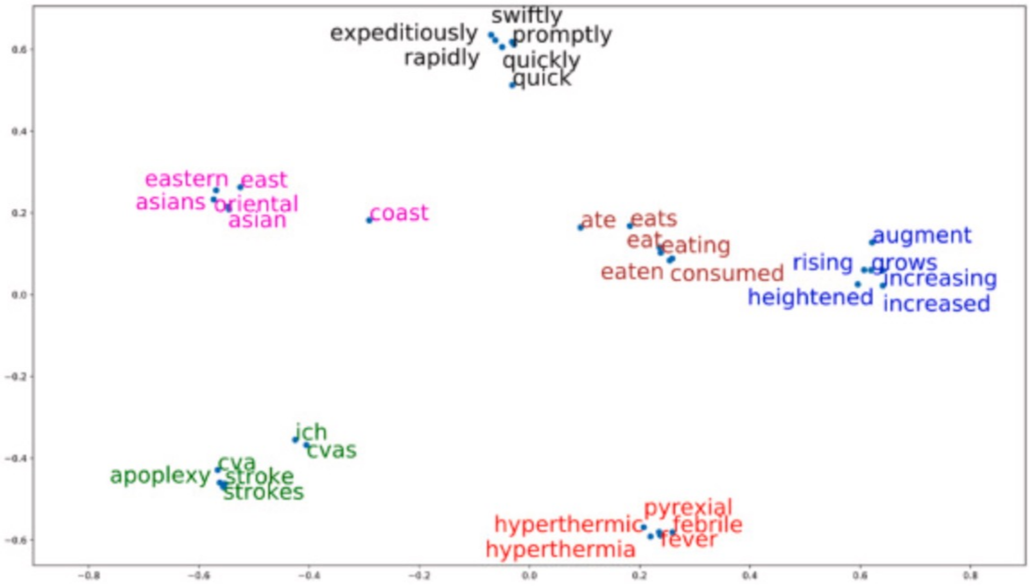
- “Man” is close to “Woman”, “King” is close to “Queen”, and “Apple” is close to “Orange”

- Semantic representation can be much shorter than 1-hot representation

Semantic representation

- Visualization in semantic representation





Semantic representation

- Sematic representation helps the down-stream tasks

I like orange juice. → I like apple ____.

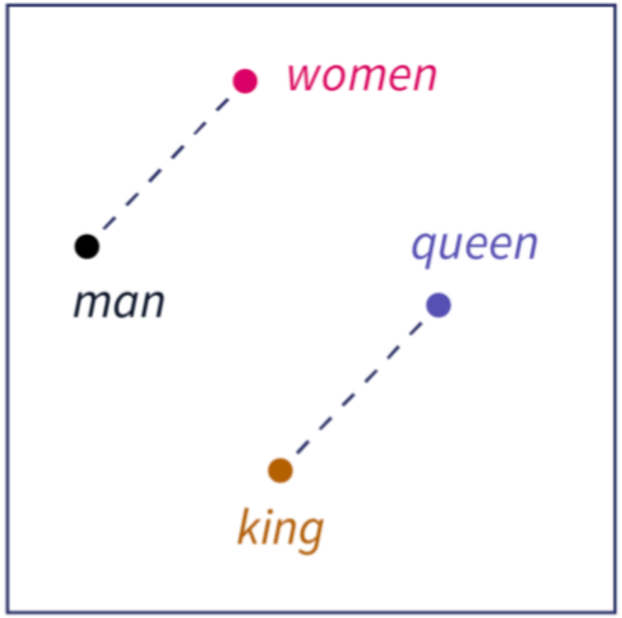
- Since “orange” and “orange” are close to each other， language model should fill in juice

- [Play semantic games in ChatGPT]

Semantic representation

- Sematic representation helps find synonyms or antonyms

- Example: given “man” vs “women”, fill in “king” vs “____”



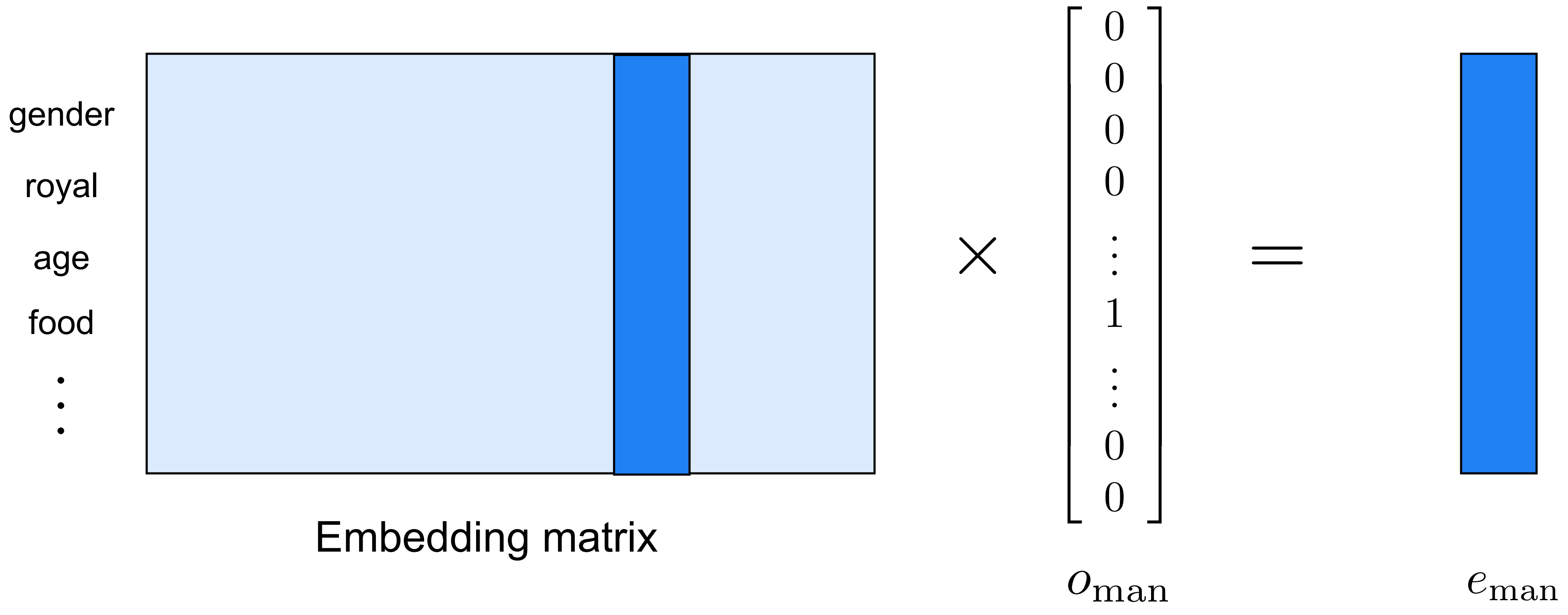
$$e_{\text{man}} - e_{\text{woman}} \approx e_{\text{king}} - e_w$$

$$\rightarrow \quad e_w \approx e_{\text{king}} - e_{\text{man}} + e_{\text{woman}}$$

$$\rightarrow \quad w = \arg\max_u \{e_u, e_{\text{king}} - e_{\text{man}} - e_{\text{woman}}\}$$

1-hot representation to semantic representation

- Given the embedding matrix, we can easily achieve the semantic representation as follows



Embedding matrix

- How to get the embedding matrix?

- We first collect the dataset from the corpus. Many ways; we use the simplest one to highlight the idea

- Given any word in a sentence, find a nearby (say, with window 2) word to construct the word pair

I want a glass of orange juice → (orange, juice), (orange, glass)

He likes watching TV → (watching, TV), (watching, likes)

corpus	dataset
I want a glass of orange juice	(orange, juice) (orange, glass)
He likes watching TV	(watching, TV) (watching, likes)

Embedding matrix

- Given the dataset, we use the first word to predict the second word