Little History of Word Representation

Yichu Zhou
University of Utah

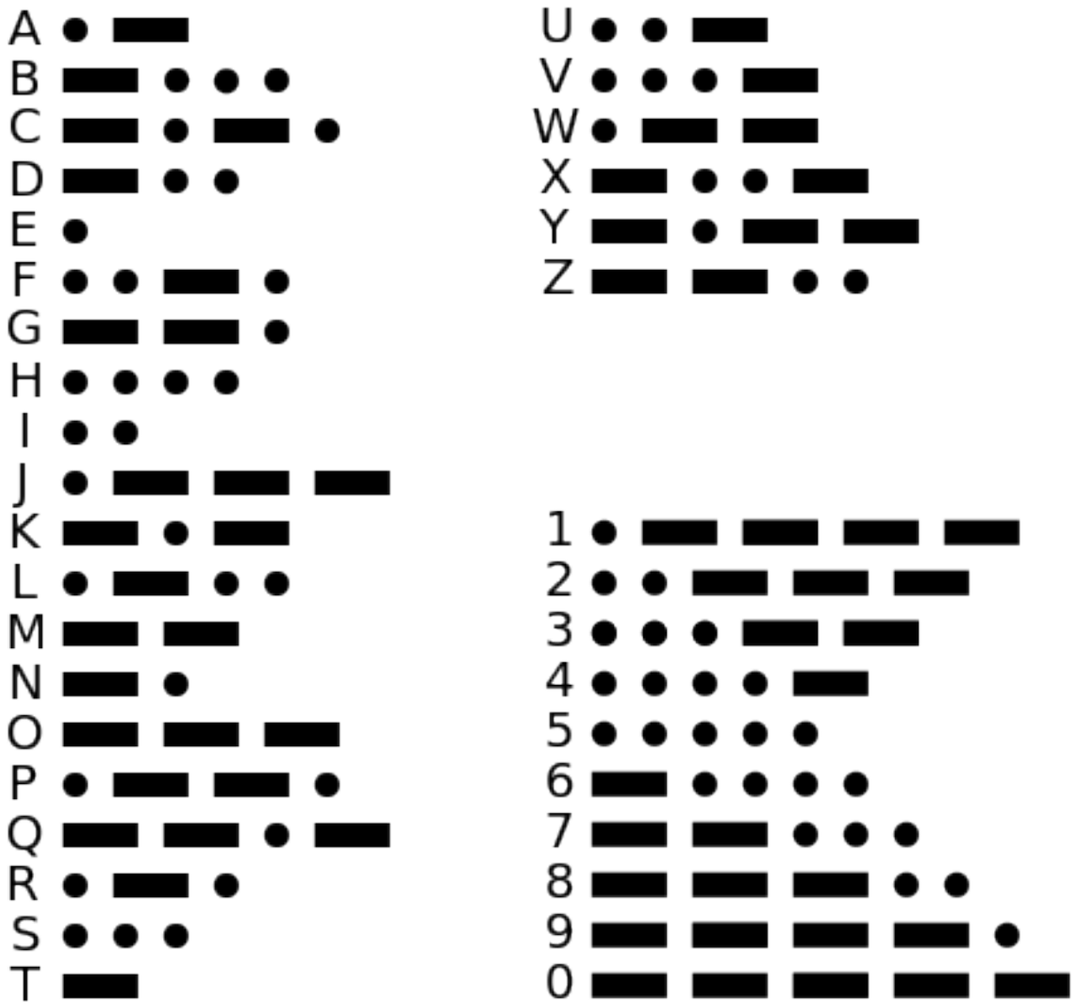
Little History of Word Representations

- Prehistory: Dictionary Lookup
- Middle Age: One-hot representation
- The Enlightenment: Word Vectors
- Industrial Age: Type vector(word2vec, fastText)
- Modern Age: Token vector(BERT)
- Summary

Prehistory: Dictionary Lookup

Words	ID
I	0
am	1
groot	2
unknown	3

Prehistory: Dictionary Lookup



Problems

Words	ID
I	0
am	1
groot	2
unknown	3

- Imposing ordering of words
  - $1 > 0 \Rightarrow$ **am > I**
- Integers do not contain information
  - **Integers are still symbols**
- The dictionary will grow to infinite
  - **We are creating new words everyday**

Little History of Word Representations

- Prehistory: Dictionary Lookup
- Middle Age: One-hot representation
- The Enlightenment: Word Vectors
- Industrial Age: Type vector(word2vec, fastText)
- Modern Age: Token vector(BERT)
- Summary

Middle Age: One-hot Representation

	I	am	groot	unknown
I	1	0	0	0
am	0	1	0	0
groot	0	0	1	0
any other words	0	0	0	1

One-hot Representation

- Advantages
  - Eliminate the wrong ordering assumption
- Problmes:
  - Carry no sementic information
  - Dimension can be infinite

Little History of Word Representations

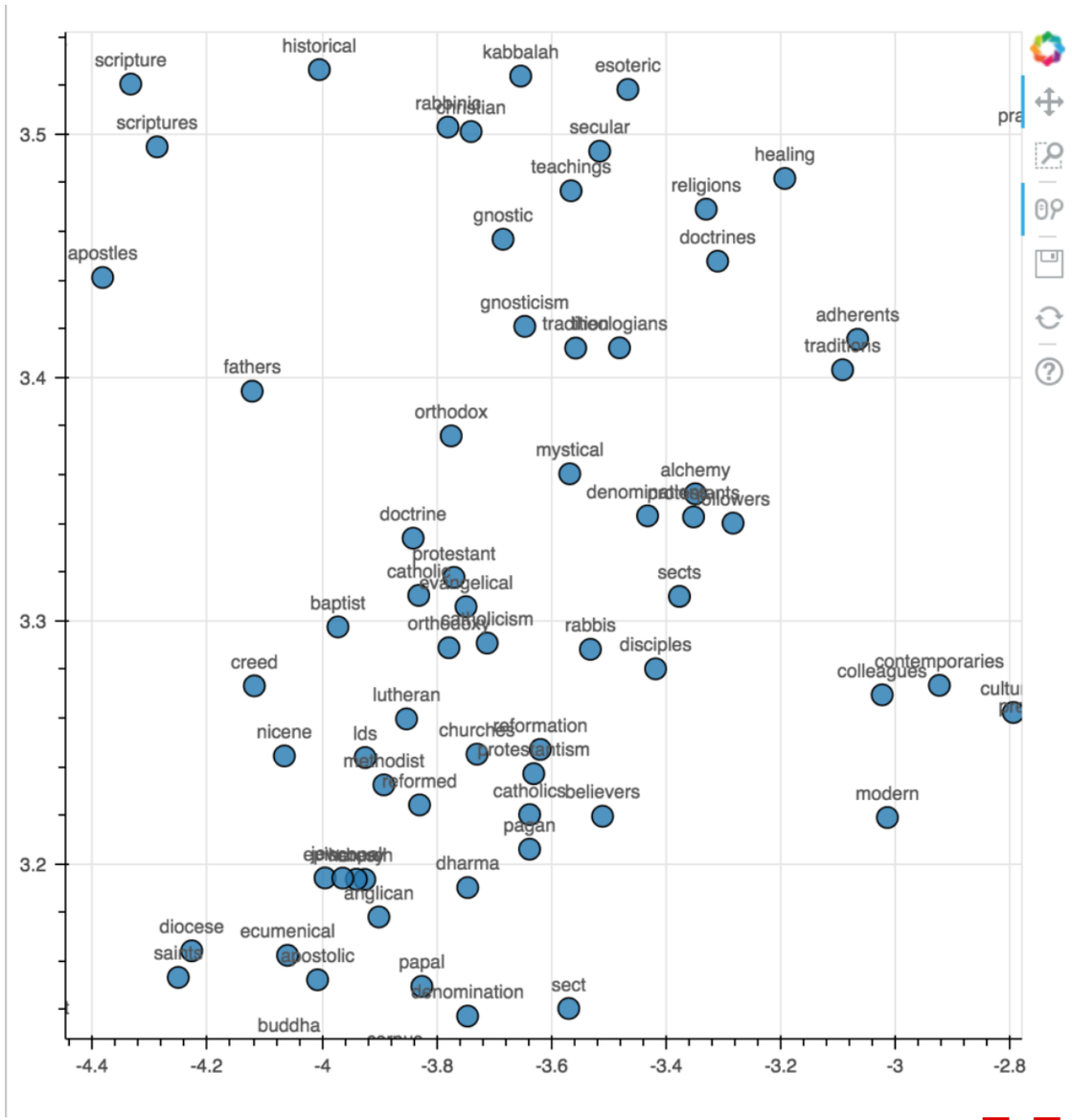
- Prehistory: Dictionary Lookup
- Middle Age: One-hot representation
- The Enlightenment: Word Vectors
- Industrial Age: Type vector(word2vec, fastText)
- Modern Age: Token vector(BERT)
- Summary

The Enlightenment: Word Vectors

Use a dense low dimension vector to represent words

boy 0 0 0 …1…0 0 0

boy -2.5693 -0.5565 -5.3025 -4.0634 0.8257



The Enlightenment: Building Word Vectors

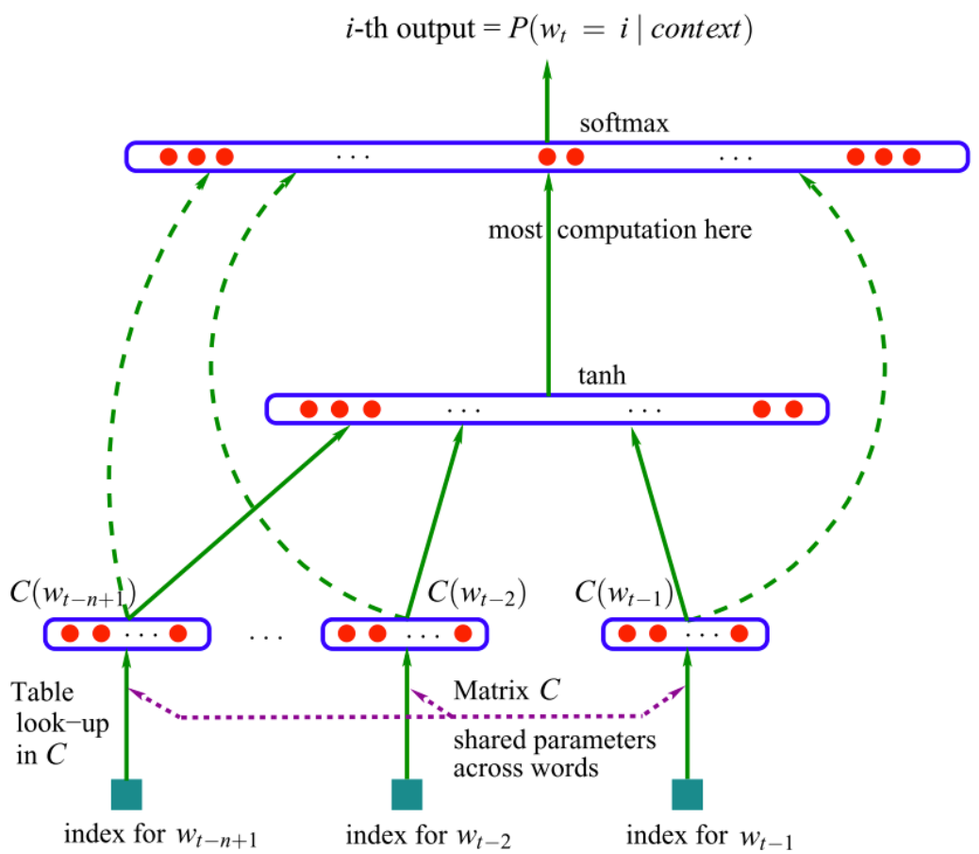


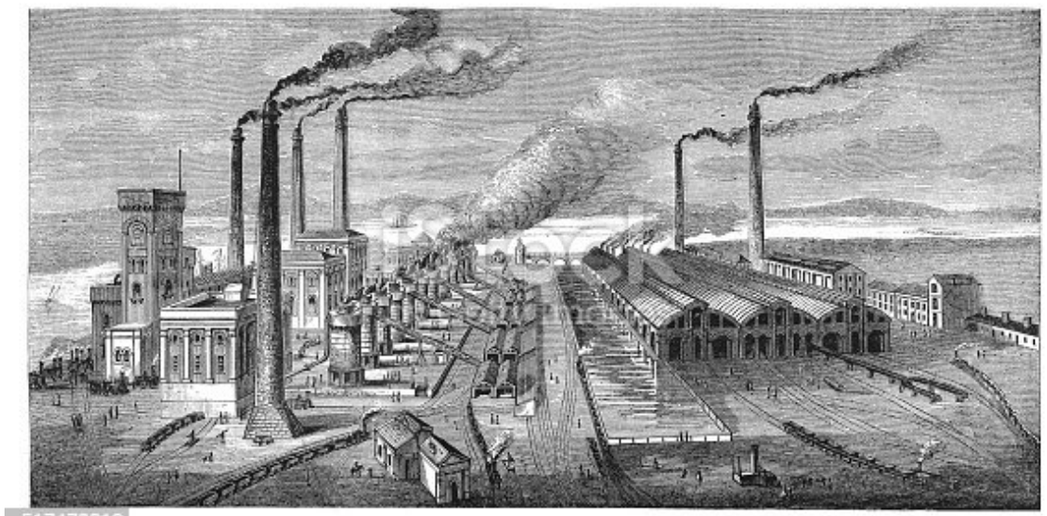
Figure 1: Neural architecture: $f(i, w_{t-1}, \cdots, w_{t-n+1}) = g(i, C(w_{t-1}), \cdots, C(w_{t-n+1}))$ where g is the neural network and $C(i)$ is the i-th word feature vector.

A neural probabilistic language model(Bengio et al.,2003)

The Enlightenment: Word Vectors

- Advantages
  - No ordering assumptions
  - Carry semantic information
  - Fixed dimension
- Disadvantages
  - Expensive to train such a vector space

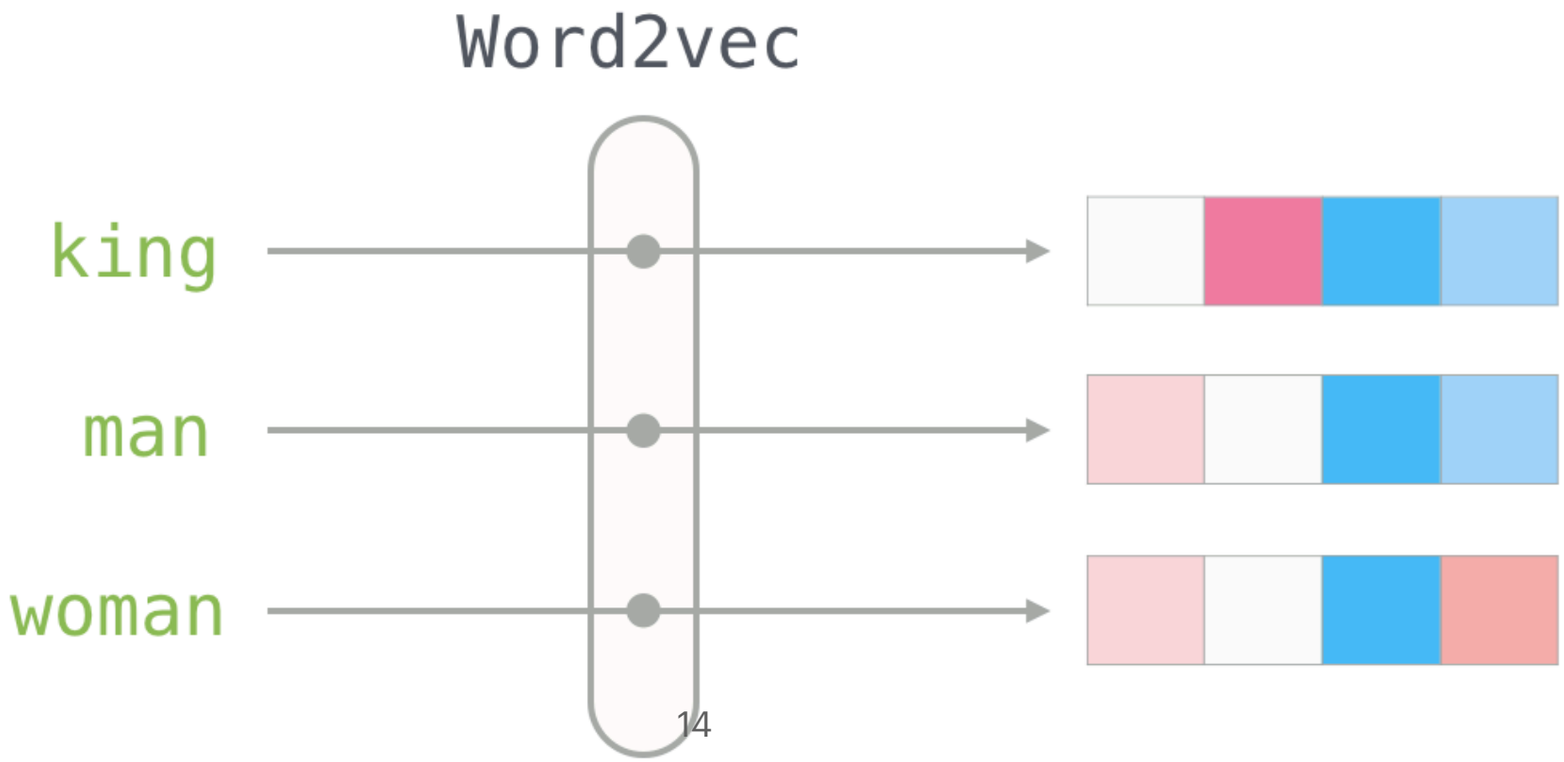
Little History of Word Representations



- Prehistory: Dictionary Lookup
- Middle Age: One-hot representation
- The Enlightenment: Word Vectors
- Industrial Age: Type vector(word2vec, fastText)
- Modern Age: Token vector(BERT)
- Summary

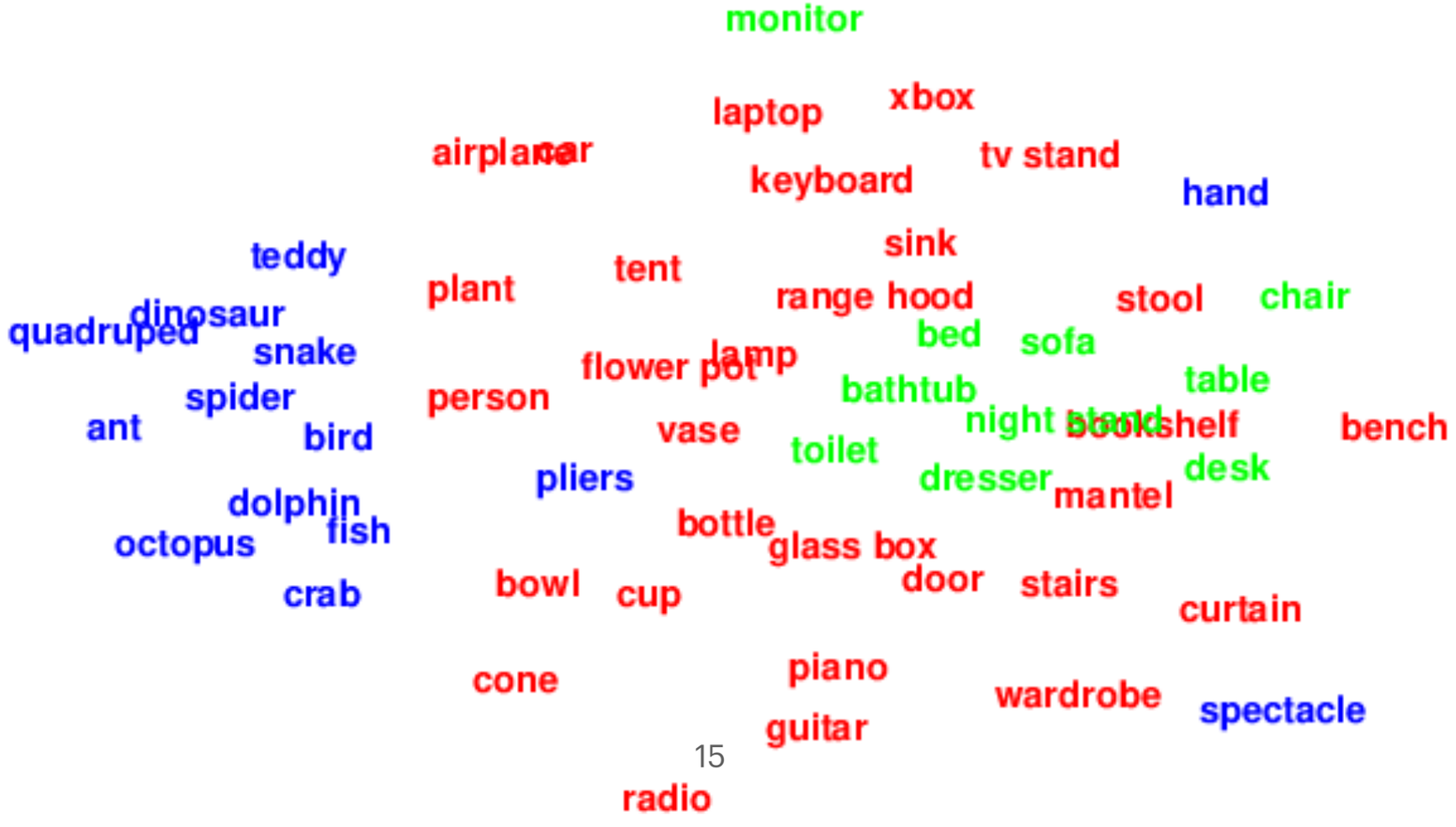
Industrial Age: Word2vec

- In 2013, a team at Google led by Tomas Mikolov published a word embedding toolkit word2vec (Mikolov et al., 2013), which can train word embeddings faster than previous approaches.



Industrial Age: Word2vec

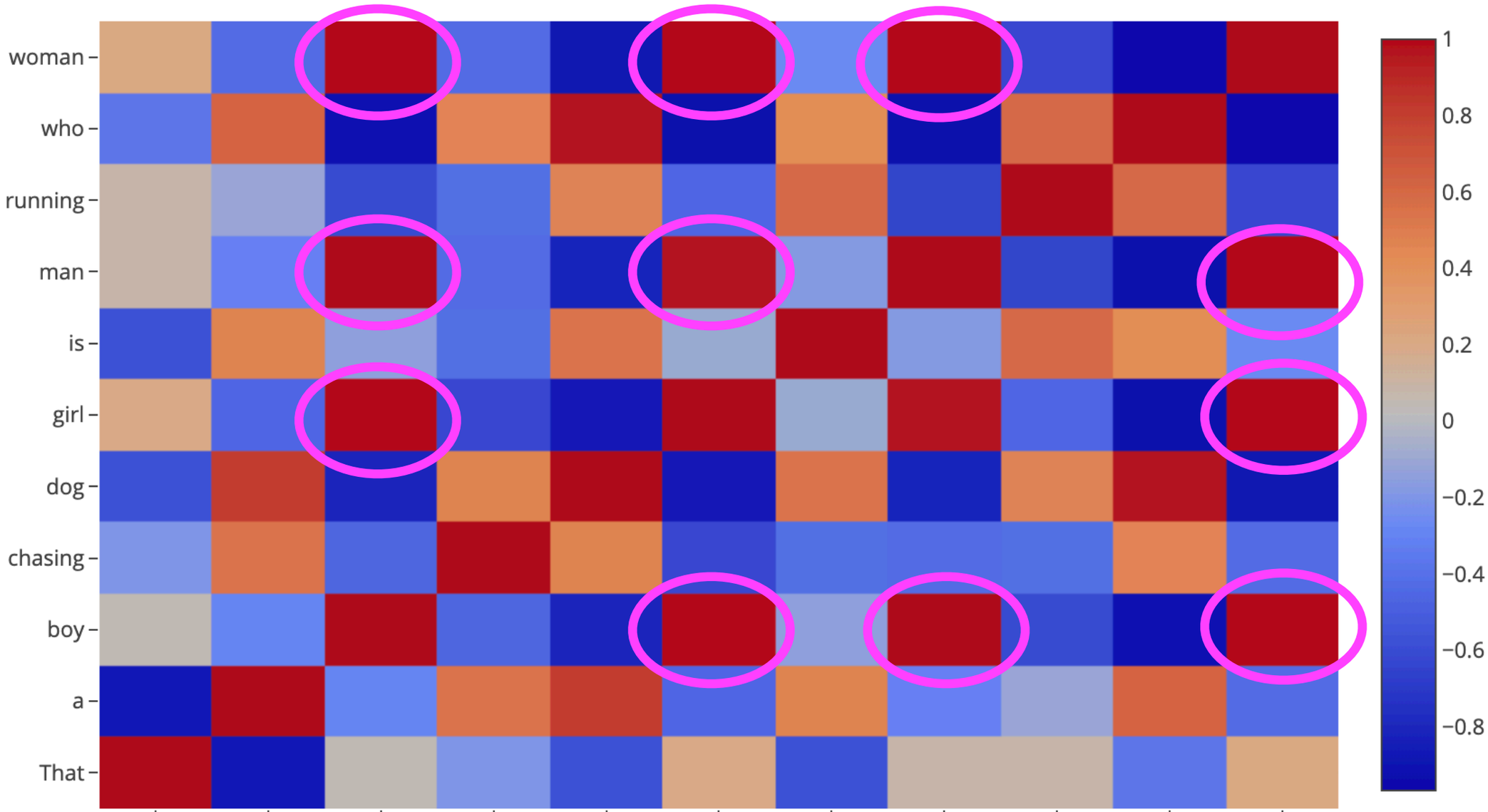
$$\overrightarrow{\text{king}} - \overrightarrow{\text{man}} + \overrightarrow{\text{woman}} \approx \overrightarrow{\text{queen}}$$



Toy Example

- That boy is chasing a dog who is running
- That girl is chasing a dog who is running
- That man is chasing a dog who is running
- That woman is chasing a dog who is running

Toy Example


woman
who
running
man
is
girl
dog
chasing
boy
a
That
That
a
boy
chasing
dog
girl
is
man
running
who
woman
1
0.8
0.6
0.4
0.2
0
−0.2
−0.4
−0.6
−0.8

Limitations of word2vec

- A fixed dictionary
- Polysemy and homonymy are not handled properly
- Out-of-Vocabulary(OOV problem)

Industrial Age: fastText

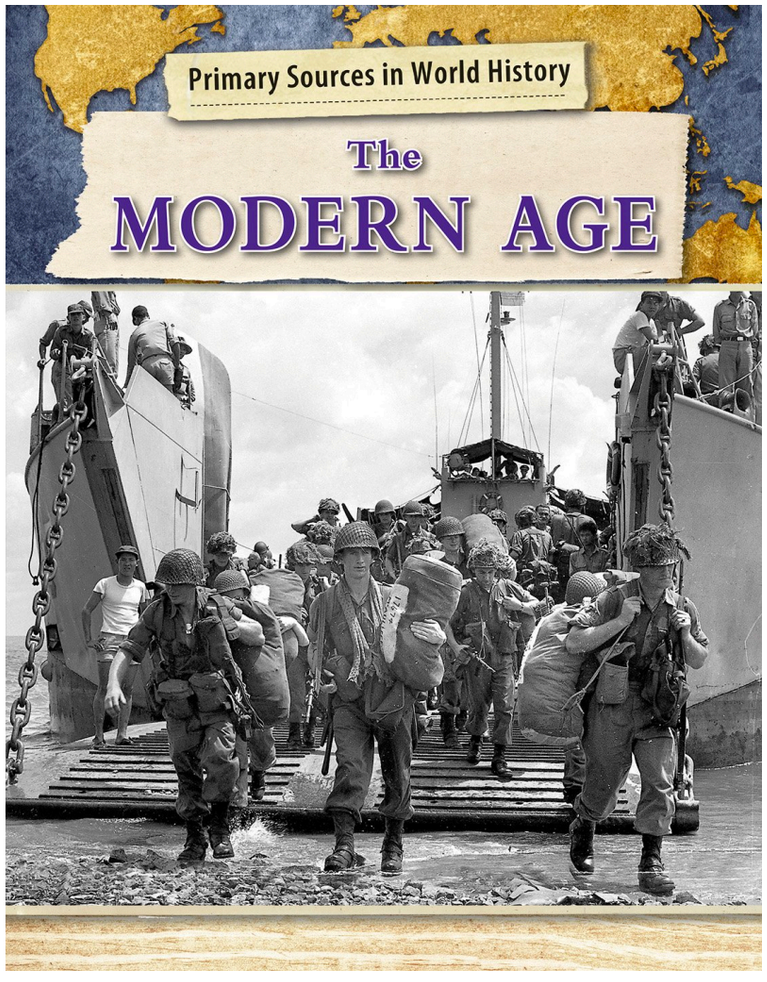
- Facebook's AI Research (FAIR) lab created another important embedding model in 2017: **fastText** (Bojanowski et al., 2017)
- Use subword in the dictionary instead of the whole word.

Where: <wh, #whe, #her, #ere, #re>

Solve the OOV problem.

Little History of Word Representations

- Prehistory: Dictionary Lookup
- Middle Age: One-hot representation
- The Enlightenment: Word Vectors
- Industrial Age: Type vector(word2vec, fastText)
- Modern Age: Token vector(BERT)
- Summary



Modern Age

- **Type Vector**
  - Type vector means **context independent**. The **same** word in different sentences (contexts) has the **same** representation.
- **Token Vector**
  - Token vector means **context dependent**. The **same** word in different sentence (contexts) has **different** representations.

Type Vector v.s. Token Vector

- I need to deposit some money to the **bank.**
- I am walking along the river **bank.**

Type Vector: $\overrightarrow{\text{bank}} = \overrightarrow{\text{bank}}$

Token Vector: $\overrightarrow{\text{bank}} \neq \overrightarrow{\text{bank}}$

Modern Age: BERT

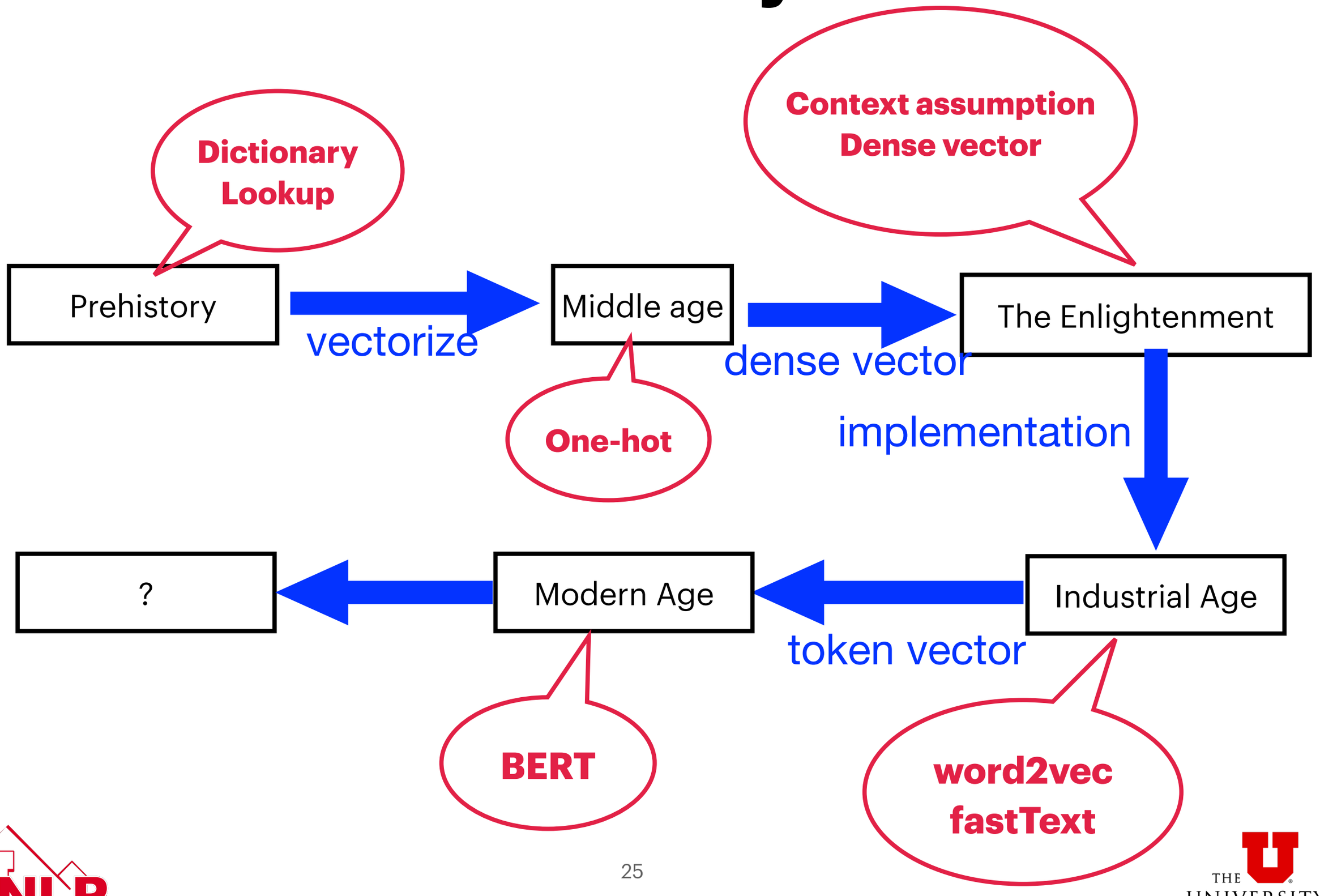
- If the creation of type vector (word2vec model) marks the age of industry, then the creation of BERT marks the modern age.

- In 2019, Google published another important paper in word representation history: "*BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding*"(Devlin et al., 2019).

Little History of Word Representations

- Prehistory: Dictionary Lookup
- Middle Age: One-hot representation and Bag-of-Words
- The Enlightenment: Word Vectors
- Industrial Age: Type vector(word2vec, fastText)
- Modern Age: Token vector(BERT)
- Summary

Summary


Dictionary Lookup
Context assumption
Dense vector
Prehistory
vectorize
Middle age
dense vector
The Enlightenment
One-hot
implementation
?
Modern Age
token vector
Industrial Age
BERT
word2vec
fastText

Takeaways

- Simple idea is not simple before it is proposed.
- Research is always built on the previous work.
- Computers and human see the world differently.