Computing for Medicine

Google Classroom Code: dnd5qkt5

Monsoon 2025

Lecture 6

Word Vectors

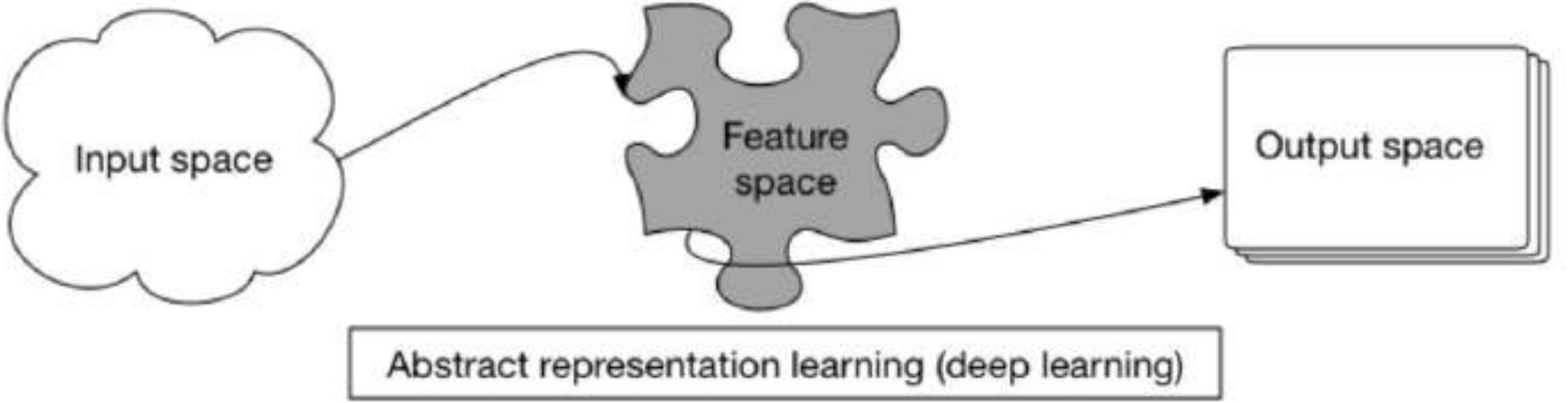
International Classification of Diseases (ICD)

ICD-10 Chapter	Code range
I Certain infectious and parasitic diseases	A00-B99
II Neoplasms	C00-D48
III Diseases of the blood and blood-forming organs and certain disorders involving the immune system	D50-D89
IV Endocrine, nutritional and metabolic diseases	E00-E90
V Mental and behavioral disorders	F00-F99
VI Diseases of the nervous system	G00-G32
VII Diseases of the eye and adnexia	H00-H59
VIII Diseases of the ear and mastoid process	H60-H95
IX Diseases of the circulatory system	I00-I99
X Diseases of the respiratory system	J00-J99
XI Diseases of the digestive system	K00-K93
XII Diseases of the skin and subcutaneous tissue	L00-L99
XIII Diseases of the musculoskeletal system and connective tissue	M00-M99

ICD Coding for Respiratory

Section	Code range
Acute upper respiratory infections	J00-J06
Influenza and pneumonia	J10-J18
Other acute lower respiratory infections	J20-J22
Other diseases of the upper respiratory tract	J30-J39
Chronic lower respiratory diseases	J40-J47
Lung diseases due to external agents	J60-J70
Other respiratory diseases principally affecting the interstitium	J80-J84
Suppurative and necrotic conditions of the lower respiratory tract	J85-J86
Other diseases of the pleura	J90-J94
Other diseases of the respiratory system	J95-J99

Basic Idea Behind All Modern Representations



How will you represent words?

Word Embeddings

Mathematical representation of language units

1. Basic vectorization approaches
2. Distributed representations
3. Universal language representation
4. Handcrafted features

Motivation: Capture the meaning of the language rather than structure

1. Break the sentence into lexical units such as lexemes, words, and phrases

2. Derive the meaning for each of the lexical units

3. Understand the syntactic (grammatical) structure of the sentence

4. Understand the context in which the sentence appears

Featurization

One Hot Encoding:

Dog Bites Man = [[1 0 0 0 0 0] [0 1 0 0 0 0] [0 0 1 0 0 0]]

BoW:

Dog Bites Man = [1 1 1 0 0 0]

TF-IDF: $TF(w) * IDF(w)$

$$\mathrm{tf}(t,d) = \frac{f_{t,d}}{\sum_{t' \in d} f_{t',d}} \quad ; \quad IDF(w) = \log \frac{N}{n_w}$$

One Hot Encoding

- Map each word w -> a unique integer w_{id} between 1 and |V|
- Each **word** then becomes a V-dimensional binary vector
- E.g. Dog = [1 0 0 0 0 0]
- “Dog Bites Man” = [[1 0 0 0 0 0] [0 1 0 0 0 0] [0 0 1 0 0 0]]

```python
sentences = ["It was the best of times",
             "it was the worst of times",
             "it was the age of wisdom",
             "it was the age of foolishness"]

tokenized_sentences = [[t for t in sentence.split()] for sentence in sentences]
vocabulary = set([w for s in tokenized_sentences for w in s])

import pandas as pd
pd.DataFrame([[w, i] for i,w in enumerate(vocabulary)])
```

```python
def onehot_encode(tokenized_sentence):
    return [1 if w in tokenized_sentence else 0 for w in vocabulary]

onehot = [onehot_encode(tokenized_sentence)
          for tokenized_sentence in tokenized_sentences]

for (sentence, oh) in zip(sentences, onehot):
    print("%s: %s" % (oh, sentence))
```

Out:

```
[0, 1, 1, 0, 0, 1, 1, 0, 0, 1, 1]: It was the best of times
[1, 1, 1, 0, 0, 0, 1, 1, 0, 1, 0]: it was the worst of times
[0, 1, 1, 0, 1, 0, 0, 1, 1, 1, 0]: it was the age of wisdom
[0, 1, 1, 1, 1, 0, 0, 1, 0, 1, 0]: it was the age of foolishness
```

Challenges of One Hot Encoding

- Size of a one-hot vector $\propto$ size of the |V|
- Sparse representations- matrix full of zeroes
- Storage constraints and overfitting due to sparsity
- Does not give fixed length representation
- Assumes independence between words
- Out of Vocabulary problem- needs retraining every time a new word is added

Bag of Words

- assumes that the text belonging to a given class in the dataset is characterized by a unique set of words
- Knowing the words present in a text, tells about the class (bag)
- Each **document** is a V-dimensional vector
- “Dog Bites Man” = [1 1 1 0 0 0]
- Gives a fixed length representation
- Captures some semantic similarity of documents

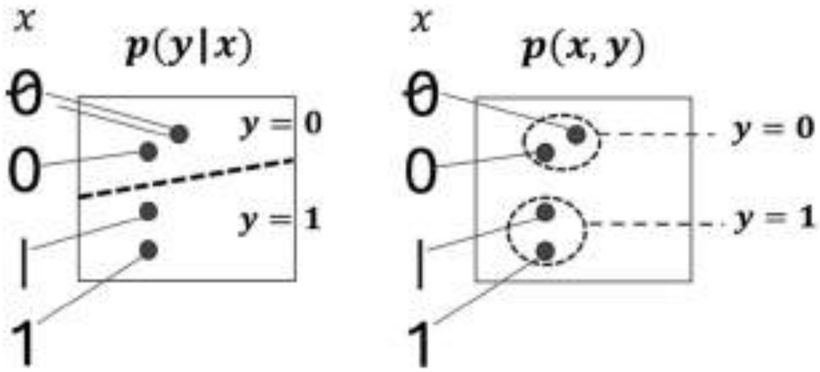
Some challenges with BoW representation

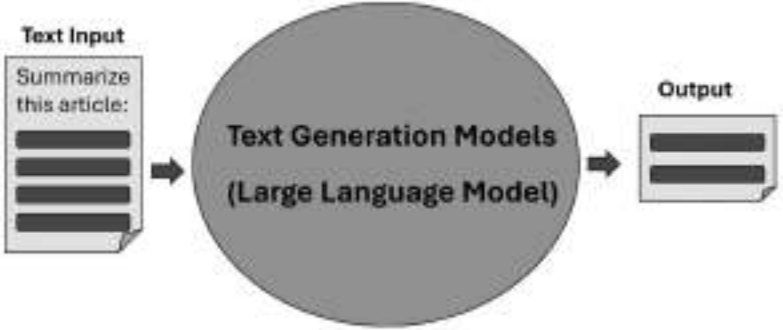
- Sparsity
- Same word may mean different things in different contexts
- Out of Vocabulary (OOV) words
- **Order** information is lost

Bag of N-Grams (BoN)

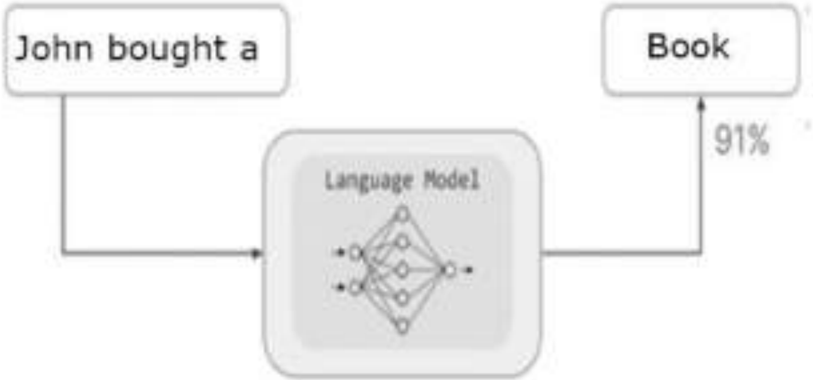
- Attempts to preserve context and order information
- Bigrams = {dog bites, bites man, man bites, bites dog, dog eats, eats meat, man eats, eats food}
- Dog Bites Man = [1,1,0,0,0,0,0,0]
- We have Bigram, Trigram, … n-gram feature selection models
- What are the challenges?

Discriminative vs Generative AI





Probabilistic Autoregressive (Traditional Models)



$$p(John, bought, a, book) = 0.02$$
$$p(book, bought, a, John) = 0.01$$
$$p(book, a, John, bought) = 0.0001$$

```
John bought a
John bought a book
John bought a book for
John bought a book for coloring
```

$$p(John, bought, a, book) = p(John)\, p(bought \mid John)$$

Vector Space Paradigm: Distributional Hypothesis

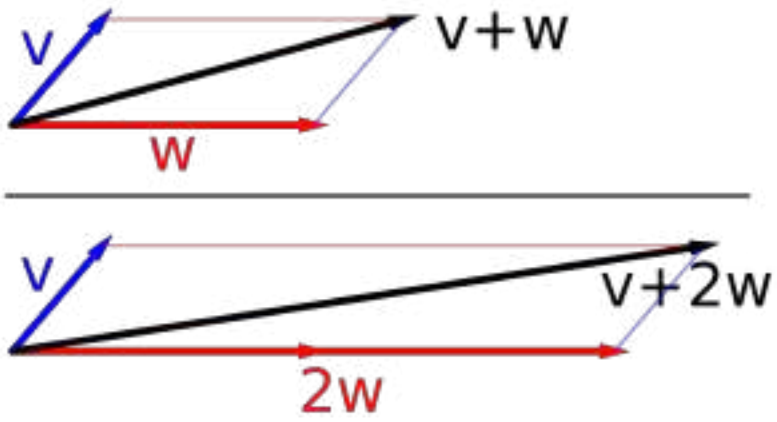
"You shall know a word by the company it keeps!"

Firth (1957)

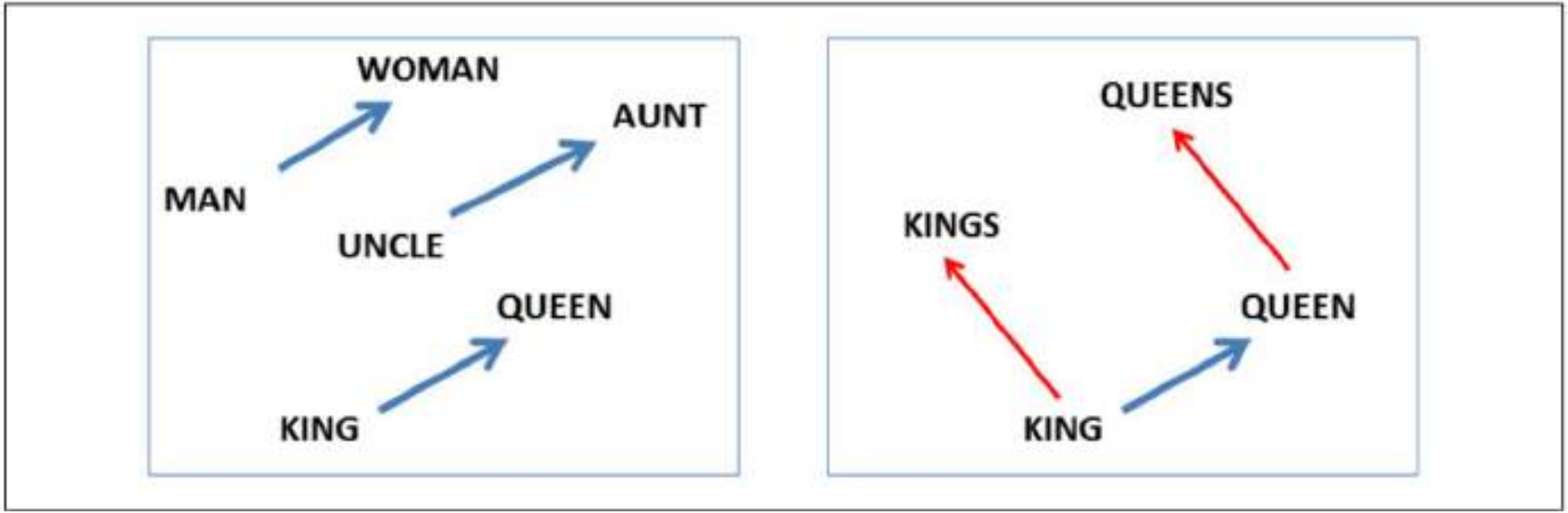
- Words that occur in similar contexts have similar meanings
- Distributional Representation: Inducing Distributional Property from context to generate a representation

Core Idea: Vector Space

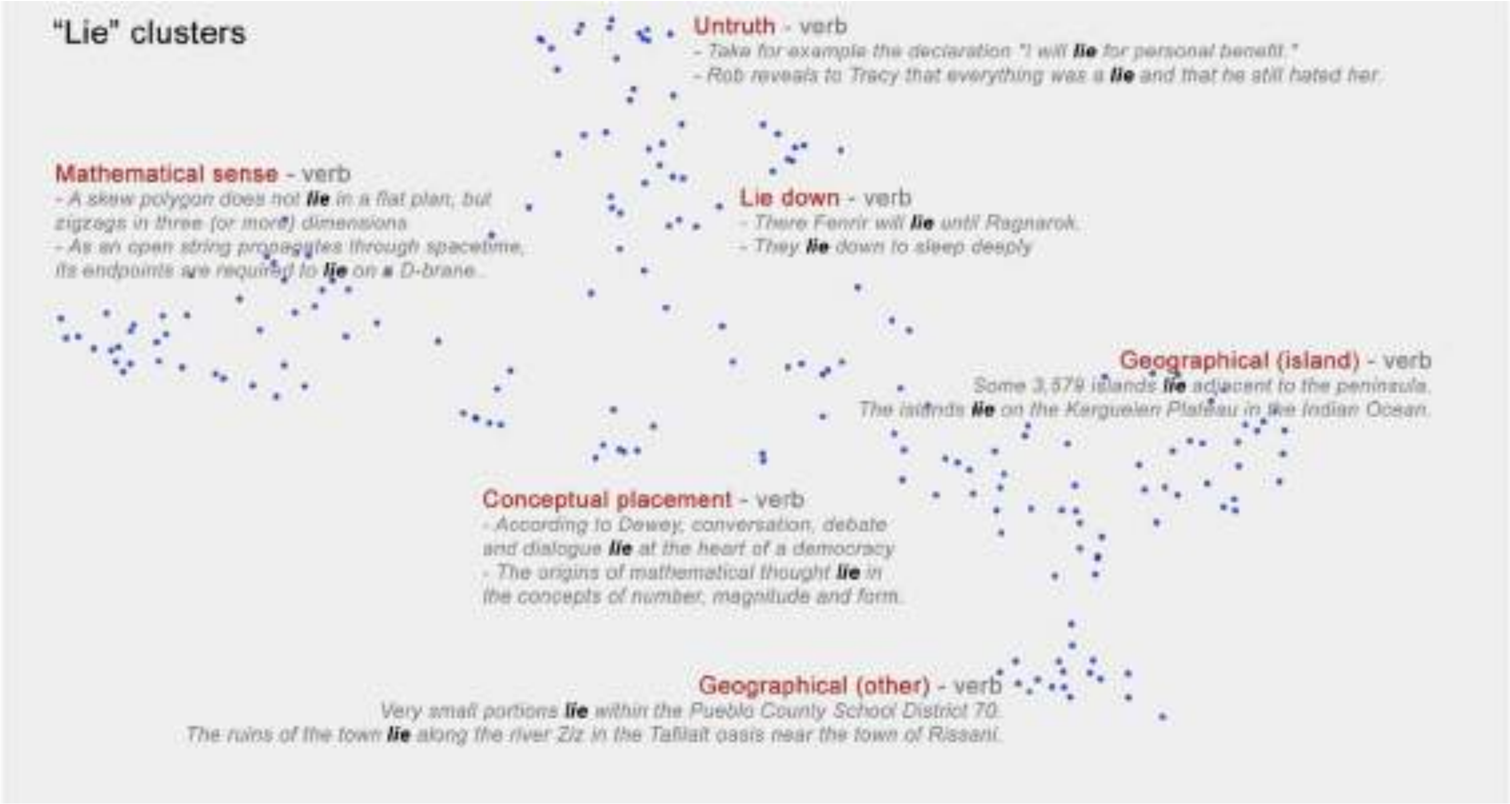
- Represent words (or tokens) as vectors
- Words with similar meanings have related vector representations
- Associations between words are captured in shared weights
- Vector weights can be trained using neural networks



Performing Algebra with Words (& other things)

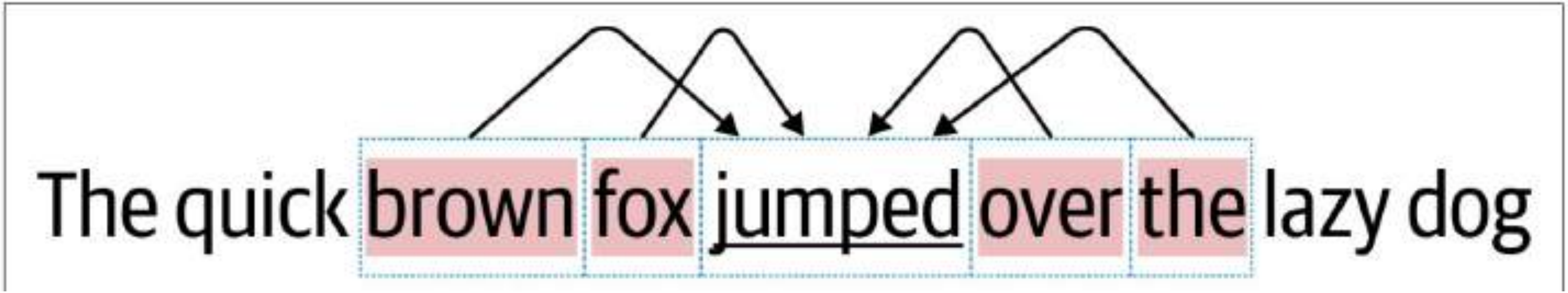


How Embeddings Capture Context

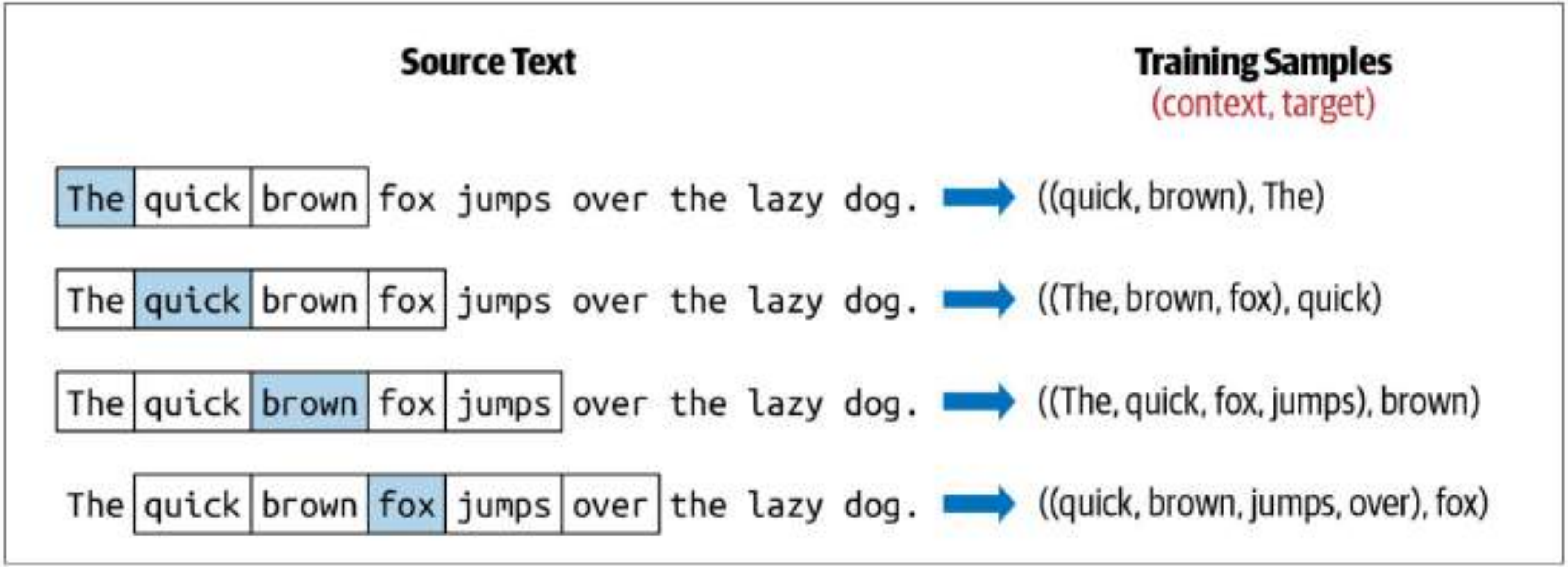


Generation 1 (w2v)

- Continuous bag of words (CBOW)
- Predicts the middle word given the context
- Assigns probability to sentences such that “good” sentences are maximized

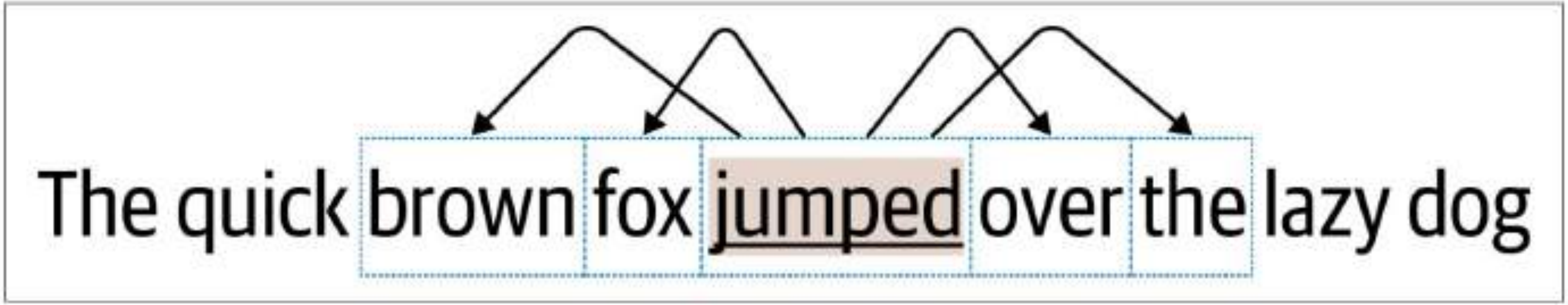


Training with CBOW

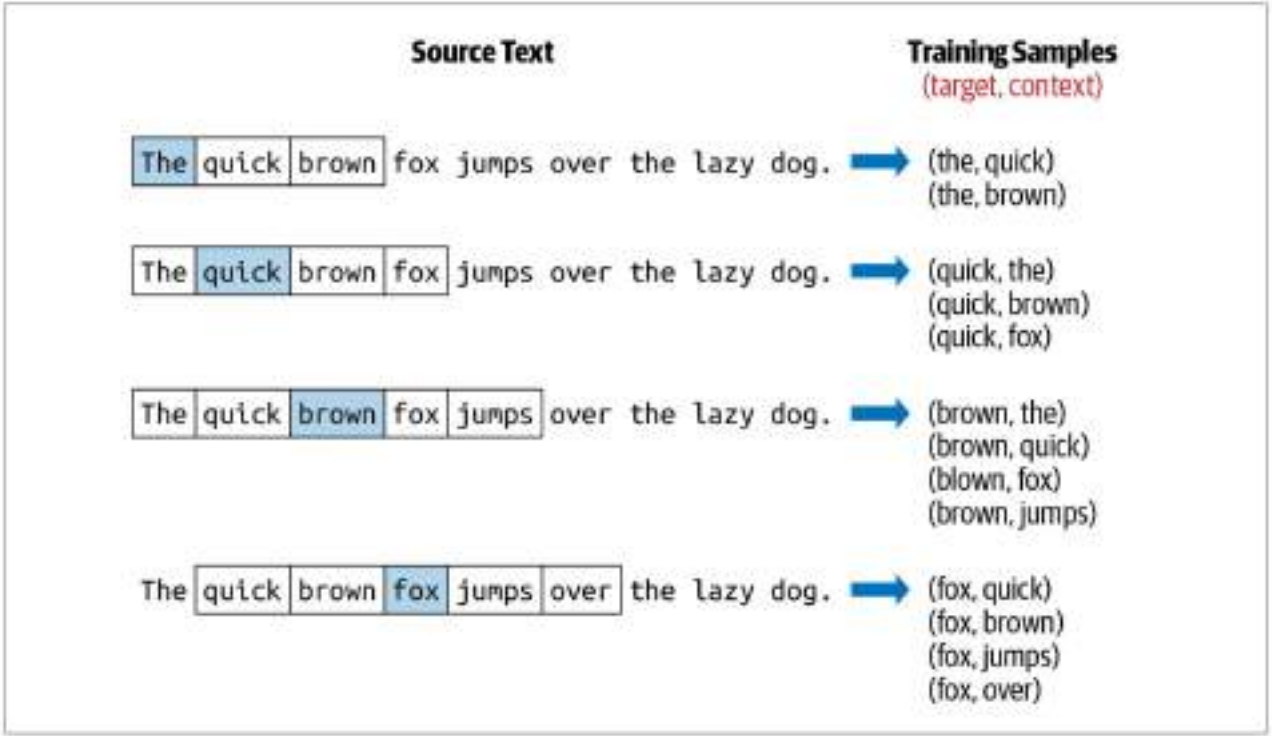


Skip Gram Variant

- Predicts the context given the middle word



Training with Skip Gram



Thanks for attending the class!