Inspire…Educate…Transform.

Word embeddings

Dr. Kishore Reddy Konda

Mentor, International School of Engineering

Text processing

Words and sentences are of varying length, is this a problem for using them as input to a machine learning algorithm?

Images have pixel intensities which can act as direct inputs to a neural network.

While text needs to be encoded into a vector form.

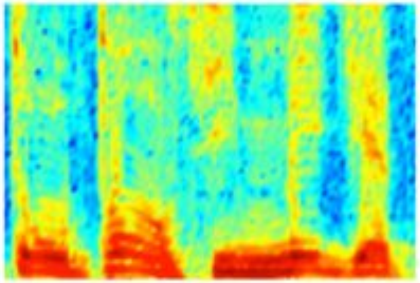
Popular methods for generating word embeddings:

Word2Vec - Mikolov, Tomas; et al. "Efficient Estimation of Word Representations in Vector Space

GloVe - http://nlp.stanford.edu/projects/glove/

Text processing

AUDIO



Audio Spectrogram

DENSE

IMAGES

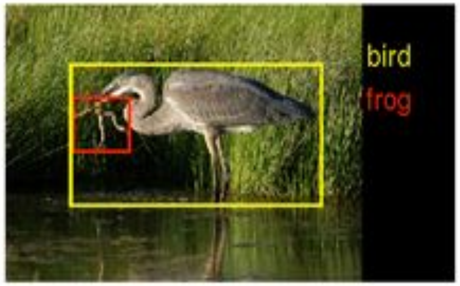


Image pixels

DENSE

TEXT

0	0	0	0.2	0	0.7	0	0	0	...	...

Word, context, or document vectors

SPARSE

Learning word embeddings

Word2Vec and GloVe

- Unsupervised learning neural network based algorithms for obtaining vector representations for words.
- Trained on large corpus of text data.
- representations showcase interesting linear substructures of the word vector space.
- Generates a fixed length vector embedding for each word.

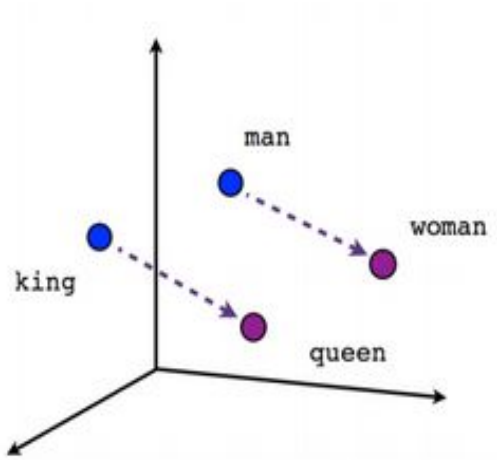
If the length of a given sentence is s, then the dimensionality of the sentence matrix is s×d (where d is the word2vec dimensions).

The parameter d can be in range of 100 to 1000, typically. This is decided when training the unsupervised models (*Word2Vec or GloVe*).

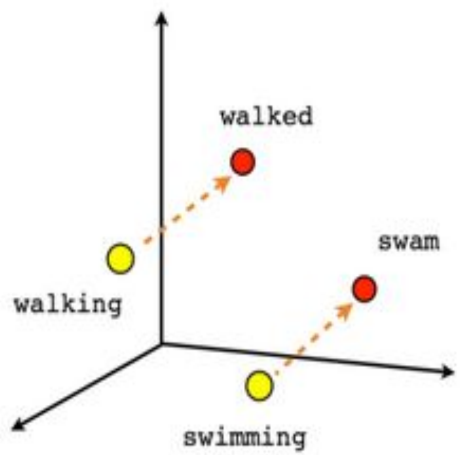
Word2Vec

- Word2vec is a group of related models that are used to produce word embeddings.
-
- Word2vec was created by a team of researchers led by Tomas Mikolov at Google.
-
- Algorithm uses a large amount of text to create high-dimensional (50 to 300 dimensional) representations.
-
- representations of words capturing relationships between words unaided by external annotations.
-
- Captures many linguistic regularities,
- vec(‘Rome’) = vec(‘Paris’) – vec(‘France’) + vec(‘Italy’).

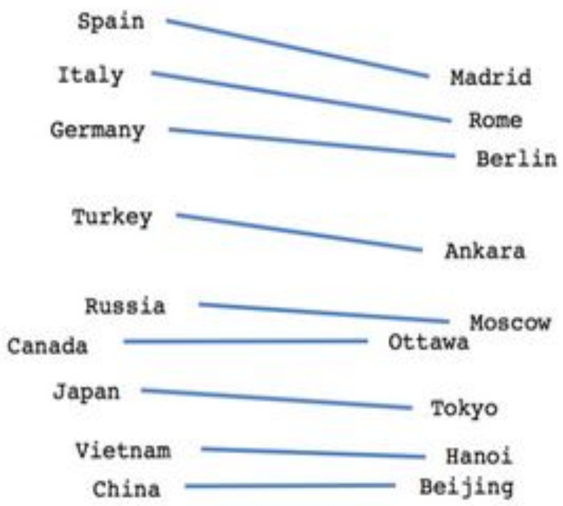
Word2Vec



Male-Female



Verb tense



Country-Capital

Word2Vec: Skip-Gram Model

- A single hidden layer neural network is trained to perform a fake task.
- After training the fake task is dumped and only hidden weights are used.
-

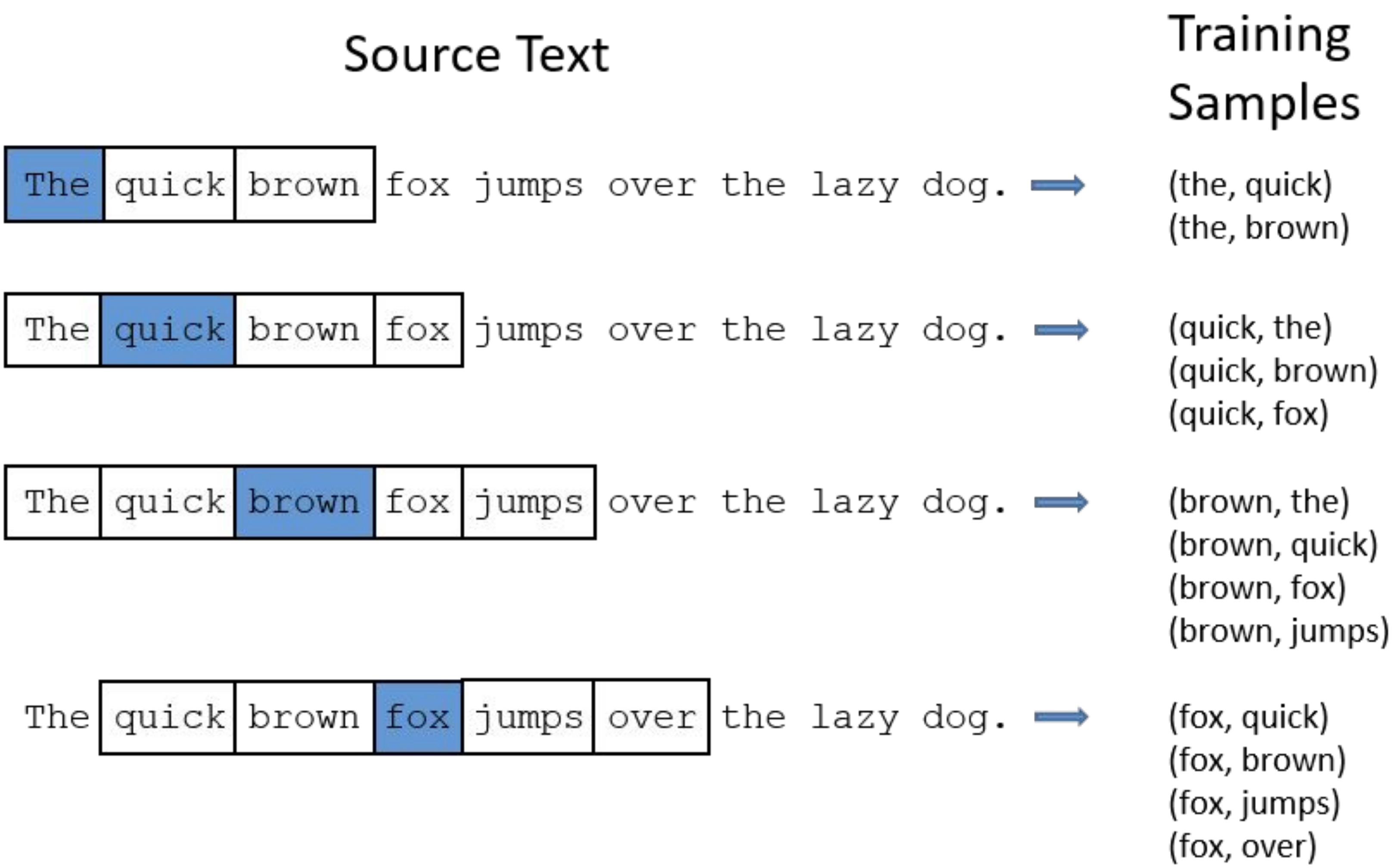
Fake Task: Given a sentence,

“The quick $\boxed{\text{brown}}$ fox jumps over the lazy dog.”

Predict the probabilities of different words from vocabulary occurring in a fixed window size around the chosen word.

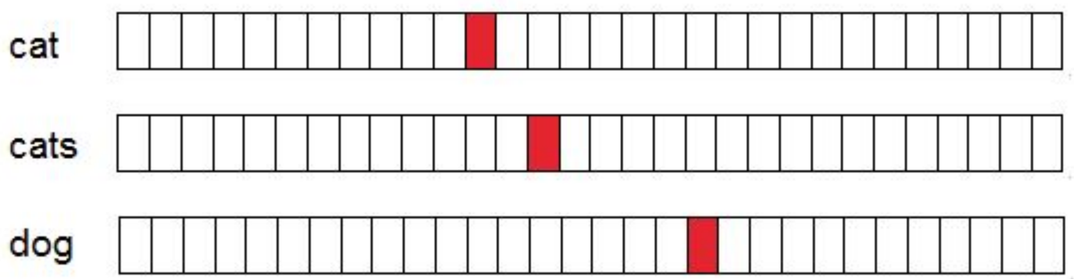
http://mccormickml.com

Word2Vec: Skip-Gram Model



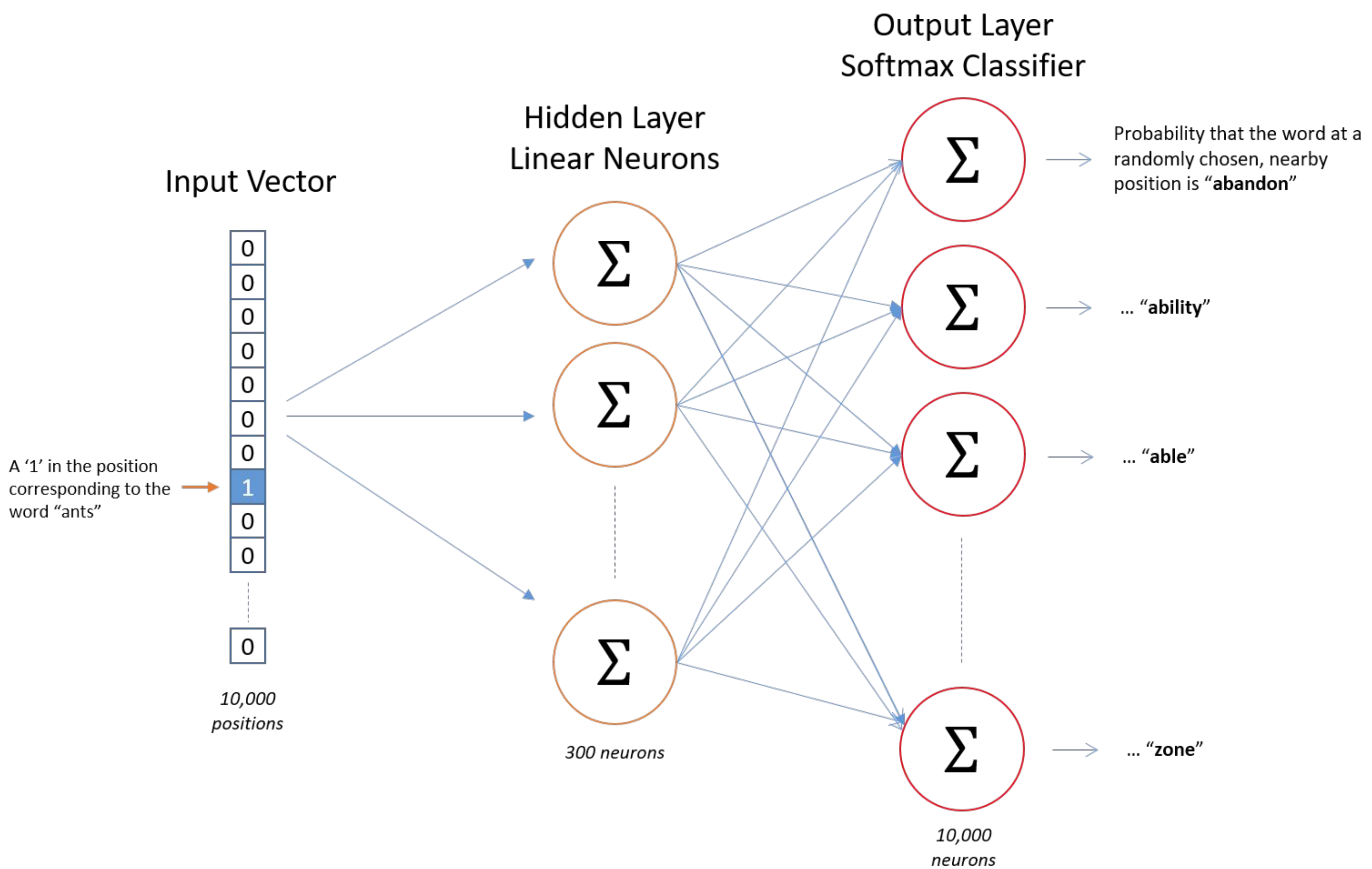
http://mccormickml.com

- **One hot encoding of a word:**
-
- We need a numerical representation for each word to train our skip-gram model.
- If you have a vocabulary of 10000 words treat each word as a state of categorical variable and dummify it.



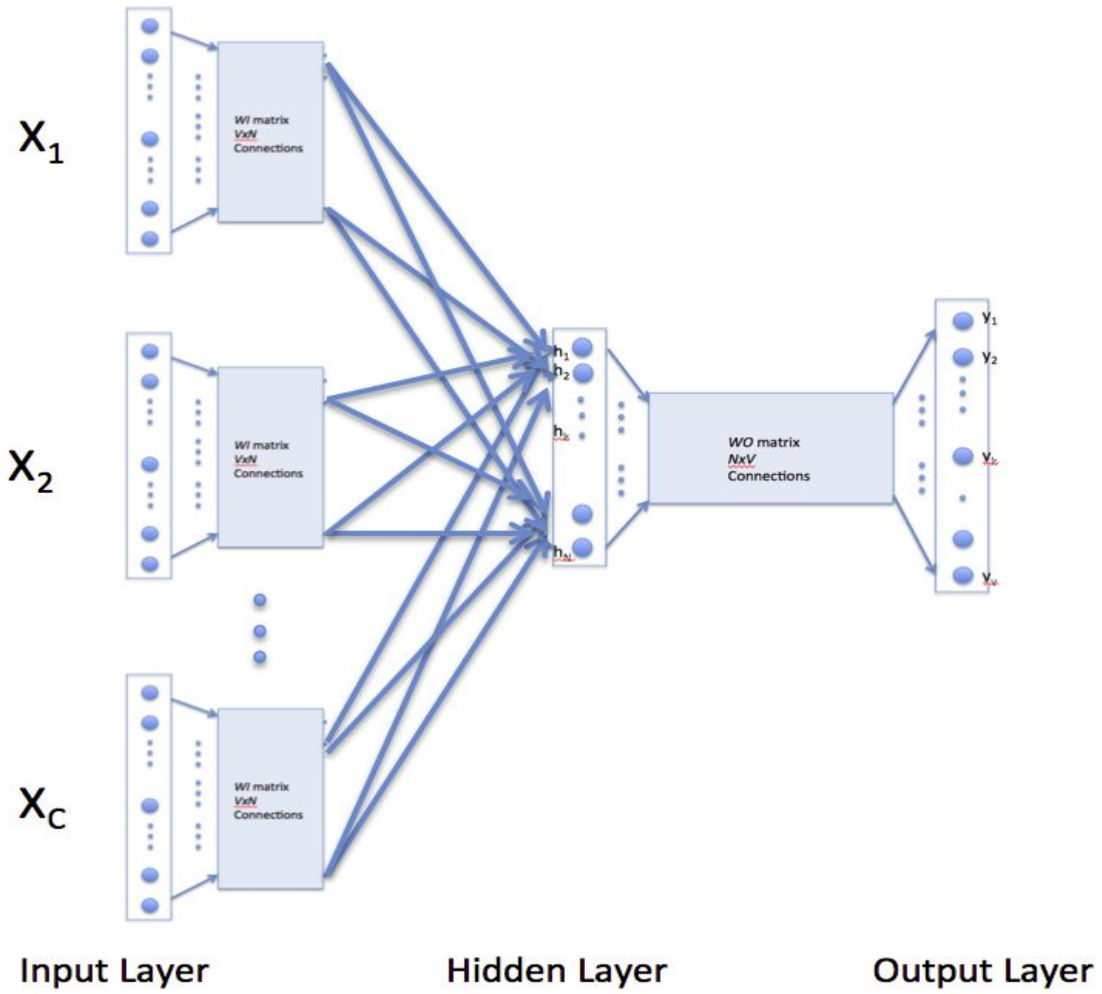
http://mccormickml.com

Word2Vec: Skip-Gram Model



http://mccormickml.com

Word2Vec: CBOW



Training sample:

Given Sentence:

"The quick brown fox jumps over the lazy dog"

(quick,brown,jumps:fox)

(jumps,the,dog: lazy)

http://mccormickml.com

Skip-gram Vs CBOW

- Skip-gram: works well with small amount of the training data, represents well even rare words or phrases.
-
- CBOW: several times faster to train than the skip-gram, slightly better accuracy for the frequent words

https://code.google.com/archive/p/word2vec/

GloVe: Global Vectors for Word Representation

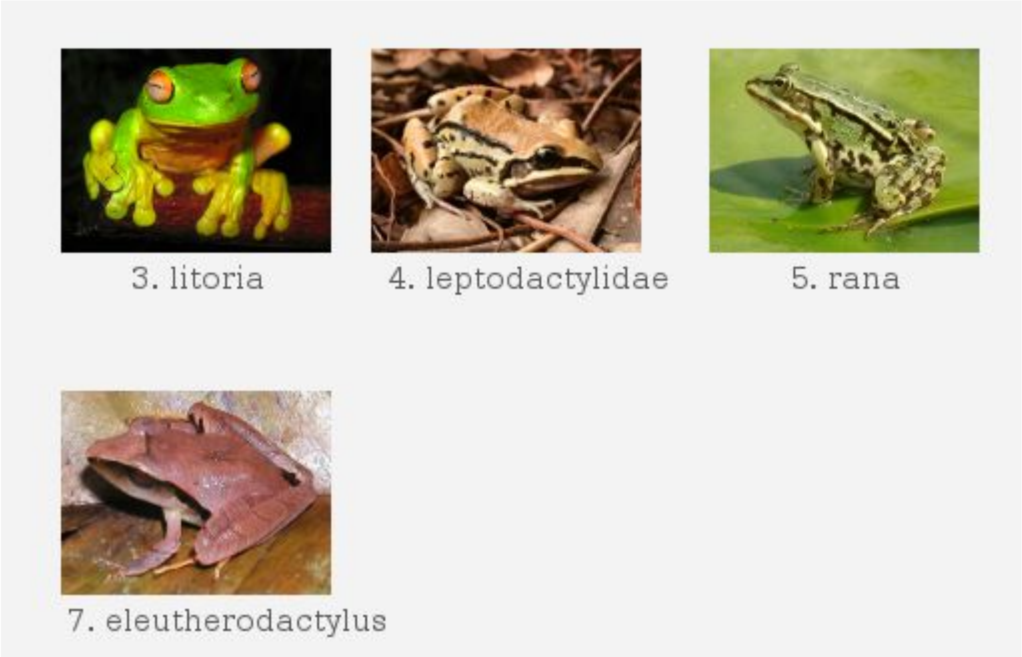
Jeffrey Pennington, Richard Socher, and Christopher D. Manning. 2014

GloVe

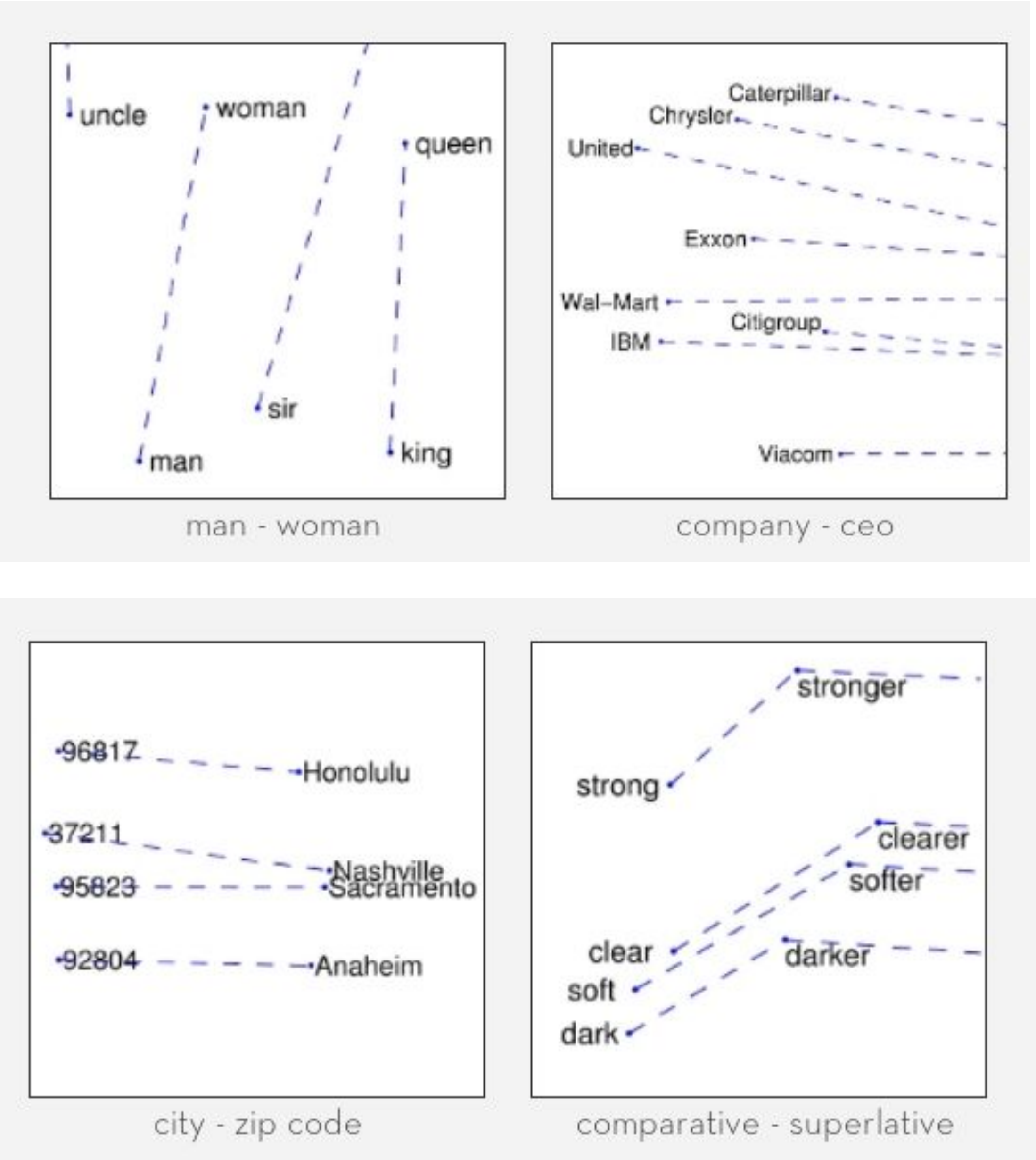
- GloVe is an unsupervised learning algorithm for obtaining vector representations for words.
-
- Training is performed on aggregated global word-word co-occurrence statistics from a corpus
-
- Resulting representations showcase interesting linear substructures of the word vector space.

GloVe

- Sometimes, the nearest neighbors according to this metric reveal rare but relevant words that lie outside an average human's vocabulary.
- For example, here are the closest words to the target word frog:
- Frog, frogs, toad, litoria, leptodactylidae, rana, lizard, eleutherodactylus



GloVe



Linear substructures:The similarity metrics used for nearest neighbor evaluations produce a single scalar that quantifies the relatedness of two words

GloVe: Training

- The GloVe model is trained on the non-zero entries of a global word-word co-occurrence matrix.
- tabulates how frequently words co-occur with one another in a given corpus
- For large corpora, this pass can be computationally expensive, but it is a one-time up-front cost.

GloVe: Training

- Define a constraint,

$$w_i w_j + b_i + b_j = \log(X_{ij})$$

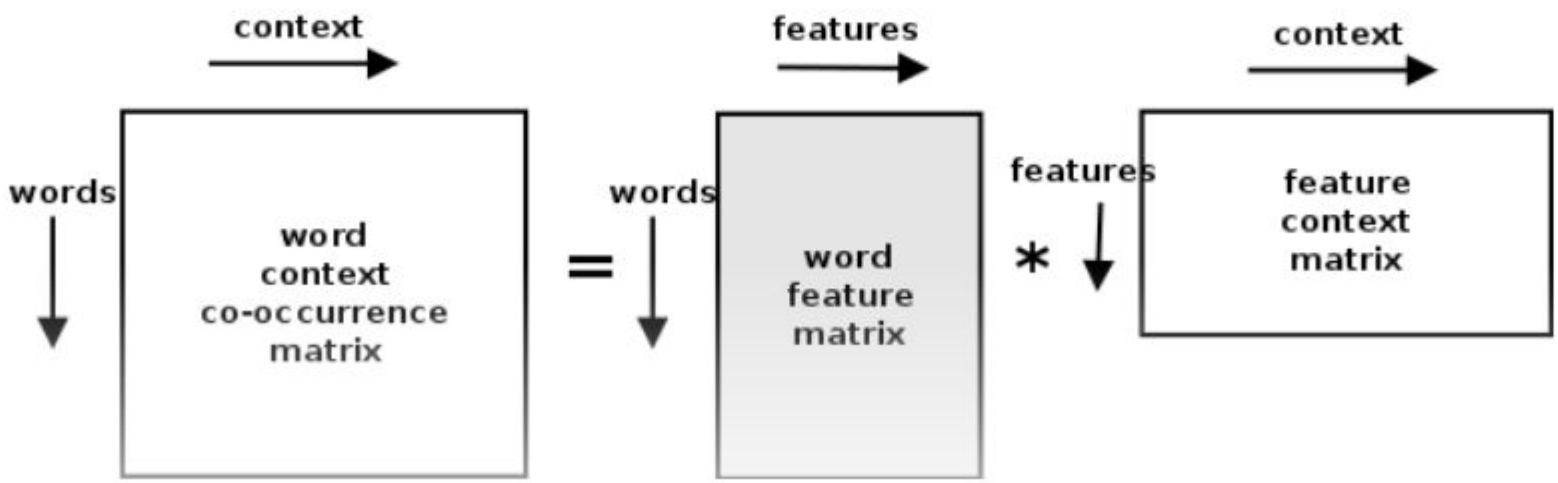
- Now we need a cost function to optimize,

$$J = \sum_{i=1}^{V} \sum_{j=1}^{V} f(X_{ij})(w_i^T w_j + b_i + b_j - \log X_{ij})^2$$

$$f(X_{ij}) = \begin{cases} (\frac{X_{ij}}{x_{max}})^{\alpha} & \text{if } X_{ij} < XMAX \\ 1 & \text{otherwise} \end{cases}$$

GloVe: Training

- Training objective of GloVe is to learn word vectors such that their dot product equals the logarithm of the words' probability of co-occurrence.



GloVe Vs Word2Vec

- In word2vec, Skip-gram models tries to capture co-occurrence one window at a time
- In Glove it tries to capture the counts of overall statistics how often its appears.
- Both capture linear substructures and tend to perform equally good.

Sentence/Paragraph/Document 2 Vec

PV-DM model

- Introduced by Tomas Mikolov (https://arxiv.org/pdf/1405.4053v2.pdf)
- Based on simple idea of using the word2vec (CBOW) model, and adding another vector (Paragraph ID below) to the input.

PV-DM model

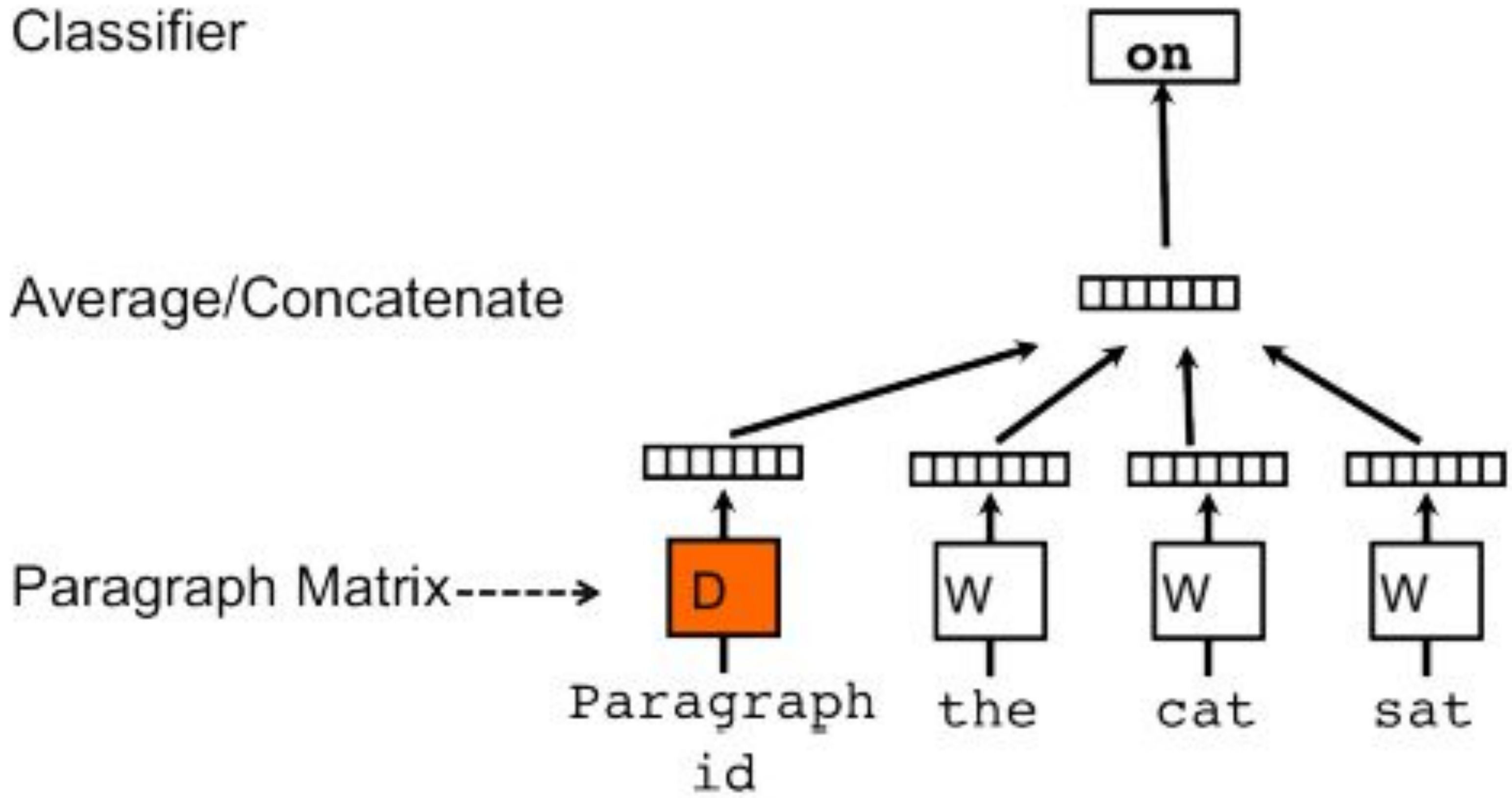


fig 3: PV-DM model

PV-DBOW model

- Introduced by Tomas Mikolov (https://arxiv.org/pdf/1405.4053v2.pdf)
- Another way is to ignore the context words in the input, but force the model to predict words randomly sampled from the paragraph in the output

PV-DBOW model

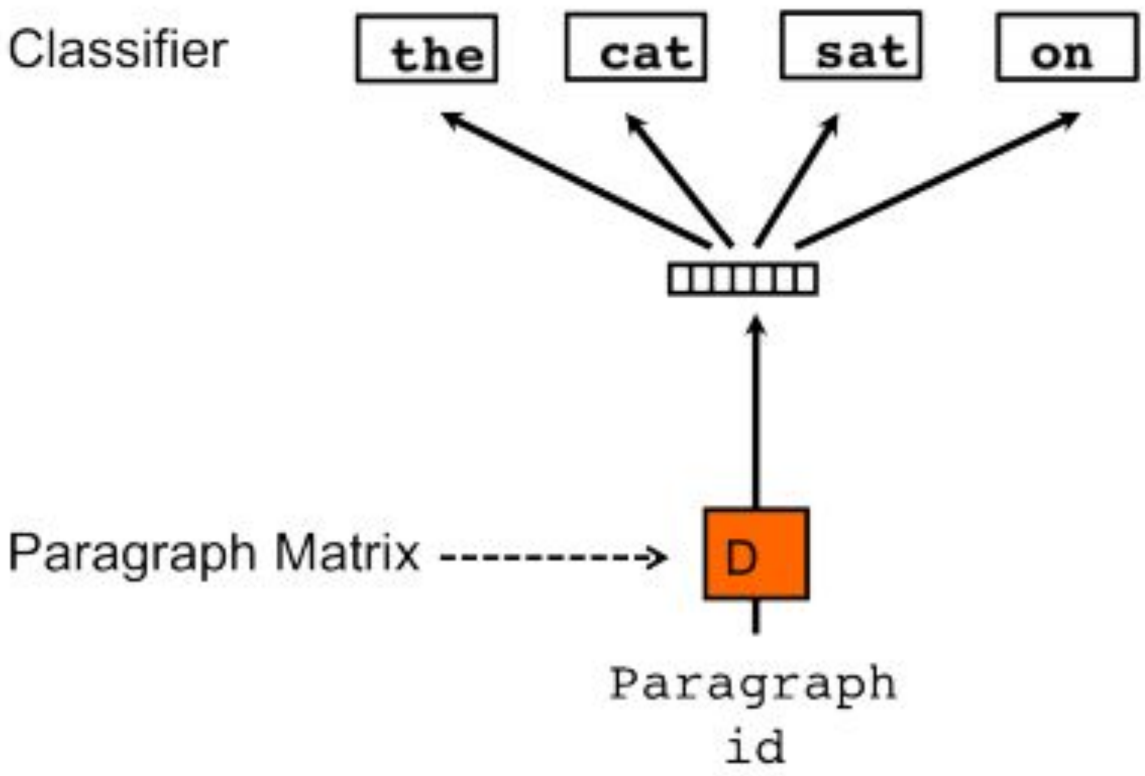


fig 4: PV-DBOW model