

# Natural Language Processing

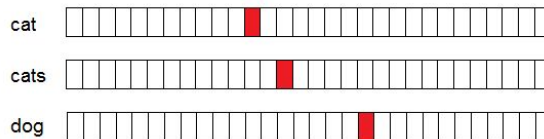
## Word Representation

Sudeshna Sarkar

18 Mar 2020

# Word embeddings

- Each word = a vector
- Examples of early approaches:
  - One-hot vector
  - Joint distribution



## Criticism

Example:

- Vocabulary size: 100,000 unique words.
1. *One-hot vector*: 100,000 free parameters, and **no knowledge of words semantic or syntactic relations**.

# Word Distributed Representation

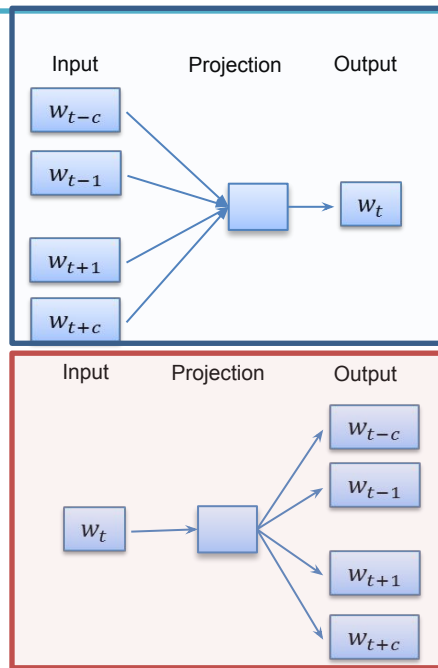
- What represents the meanings of a word?

“You shall know a word by the company it keeps” – J.R. Firth, 1957

- Words are represented by **real** valued **dense** vectors of significantly **smaller dimensions** (e.g. 100 – 1000).
- The goal is to assign each word a vector such that similar words have similar vectors (by dot-product).
- Intuition: consider each vector cell as a representative of some feature.

# word2vec approach to represent the meaning of word

- Key idea:
  - Predict center word, given surrounding word (CBOW)
  - OR
  - Predict surrounding words of every word (SKIPGRAM)
- Train a classifier on a binary **prediction** task:
  - Is  $w$  likely to show up near "*desert*"?
  - Take the learned classifier weights as the word embeddings
- Use running text as implicitly supervised training data!

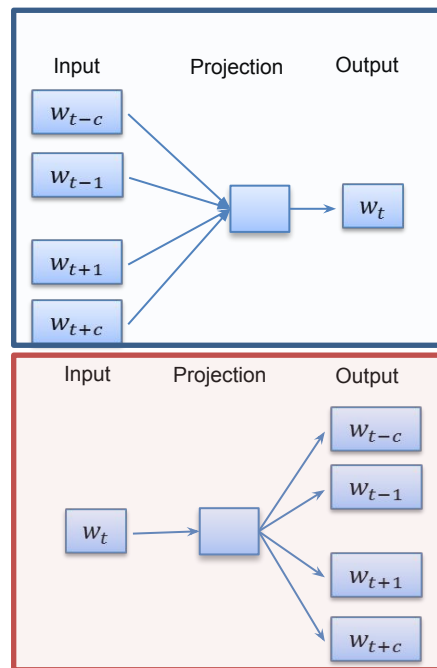


# Word2vec

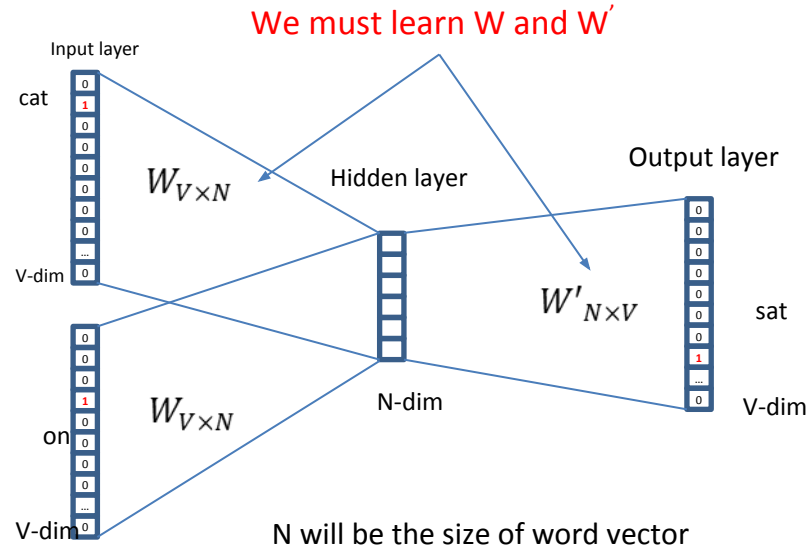
- Representation of words

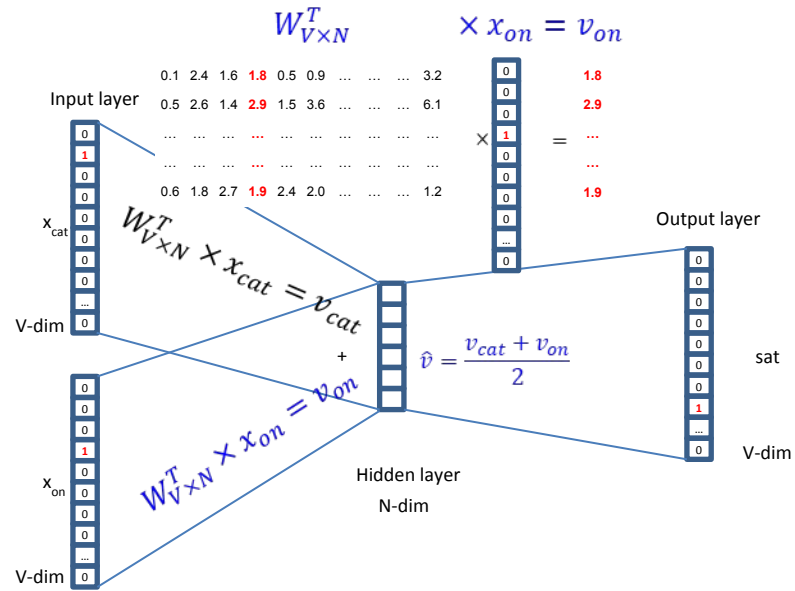
“Similar words have similar contexts”

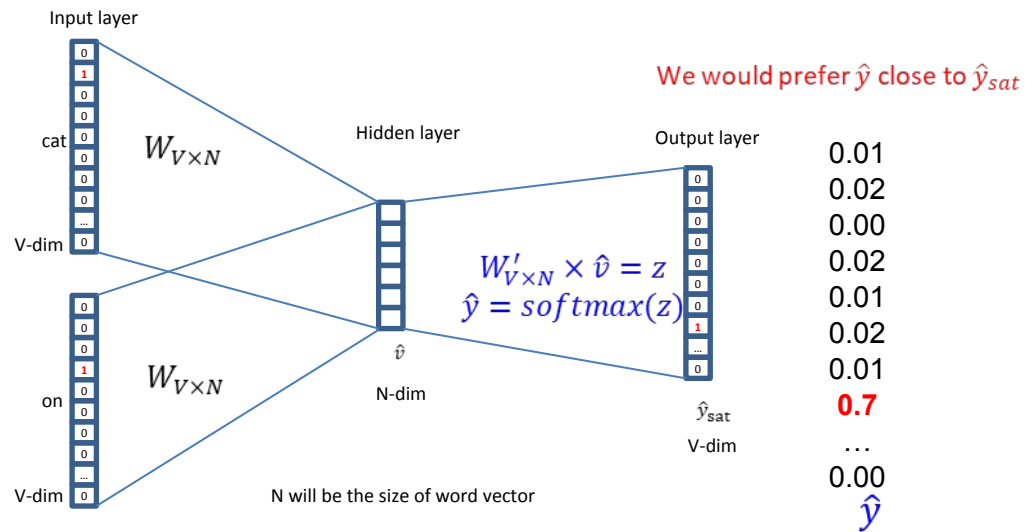
1. CBOW:  $P(\text{Word}|\text{Context})$
2. Skipgram:  $P(\text{Context}|\text{Word})$



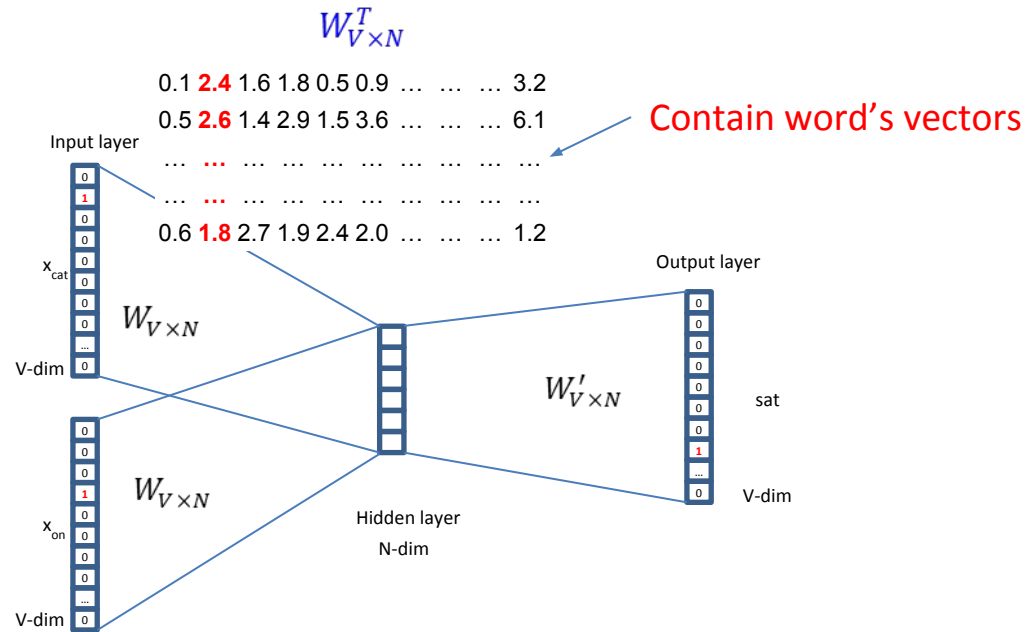
# CBOW











We can consider either  $W$  or  $W'$  as the word's representation. Or even take the average.

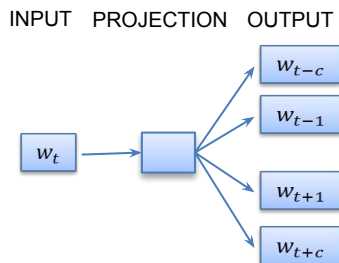
# Word2Vec Objective

$$J(\theta) = \frac{1}{T} \sum_{t=1}^T \sum_{-m \leq j \leq m, j \neq 0} \log p(w_j | c)$$

$$p(w_j | c) = \frac{\exp(w_j^T c)}{\sum_{i=1}^N \exp(w_i^T c)}$$

# Skipgram Model

- Input: Central word  $w_t$
- Output: Words in its context:  $\mathbf{w}_{\text{con}}$   
 $\{w_{t-c}, \dots, w_{t-1}, w_{t+1}, \dots, w_{t+c}\}$
- Each input word represented by a 1-hot encoding of size  $V$



1. Treat the target word and a neighboring context word as positive examples.
2. Randomly sample other words in the lexicon to get negative samples
3. Use logistic regression to train a classifier to distinguish those two cases
4. Use the weights as the embeddings

Source Text:

**Deep Learning attempts to learn multiple levels of representation from data.**

Input output pairs :

**Positive samples:**

(representation, levels)  
(representation, of)  
(representation, from)  
(representation, data)

**Negative samples:**

(representation, x)  
[x: all other words except the 4 positive]

# Skip-Grams with Negative Sampling (SGNS)

Marco saw a furry little **wampimuk** hiding in the tree.

# Skip-Grams with Negative Sampling (SGNS)

Marco saw a furry little wampimuk hiding in the tree.

## words

wampimuk

wampimuk

wampimuk

wampimuk

...

## contexts

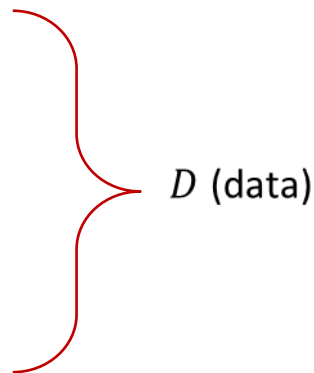
furry

little

hiding

in

...



“word2vec  
Explained...”  
Goldberg & Levy, arXiv  
2014

# Skip-Grams with Negative Sampling (SGNS)

**Maximize:**  $\sigma(\vec{w} \cdot \vec{c})$

- $c$  was **observed** with  $w$

words

wampimuk

wampimuk

wampimuk

wampimuk

contexts

furry

little

hiding

in

**Minimize:**  $\sigma(\vec{w} \cdot \vec{c}')$

- $c'$  was **hallucinated** with  $w$

words

wampimuk

wampimuk

wampimuk

wampimuk

contexts

Australia

cyber

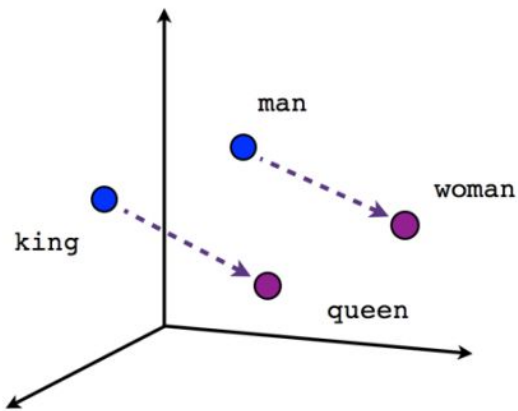
the

1985

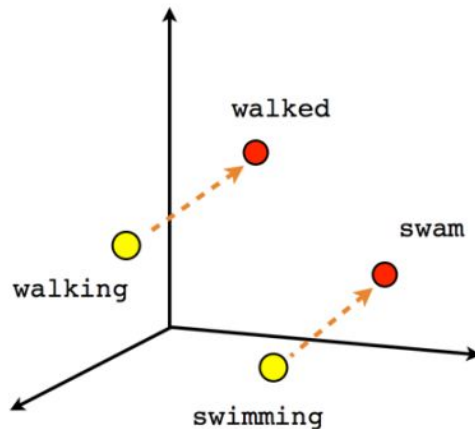
# Setup and Learning

- Let's represent words as vectors of some length (say 200), randomly initialized.
- Iteratively adjust the weights
  - Maximize the similarity of the **target word**, **context word** pairs (t,c) drawn from the positive data
  - Minimize the similarity of the (t,c) pairs drawn from the negative data.

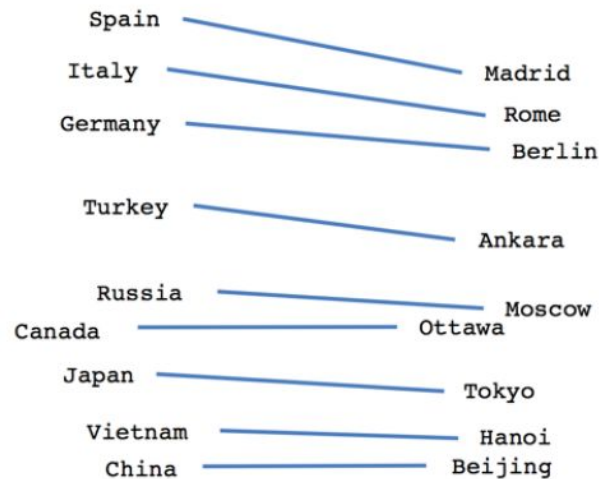
# Embeddings capture relational meaning!



Male-Female



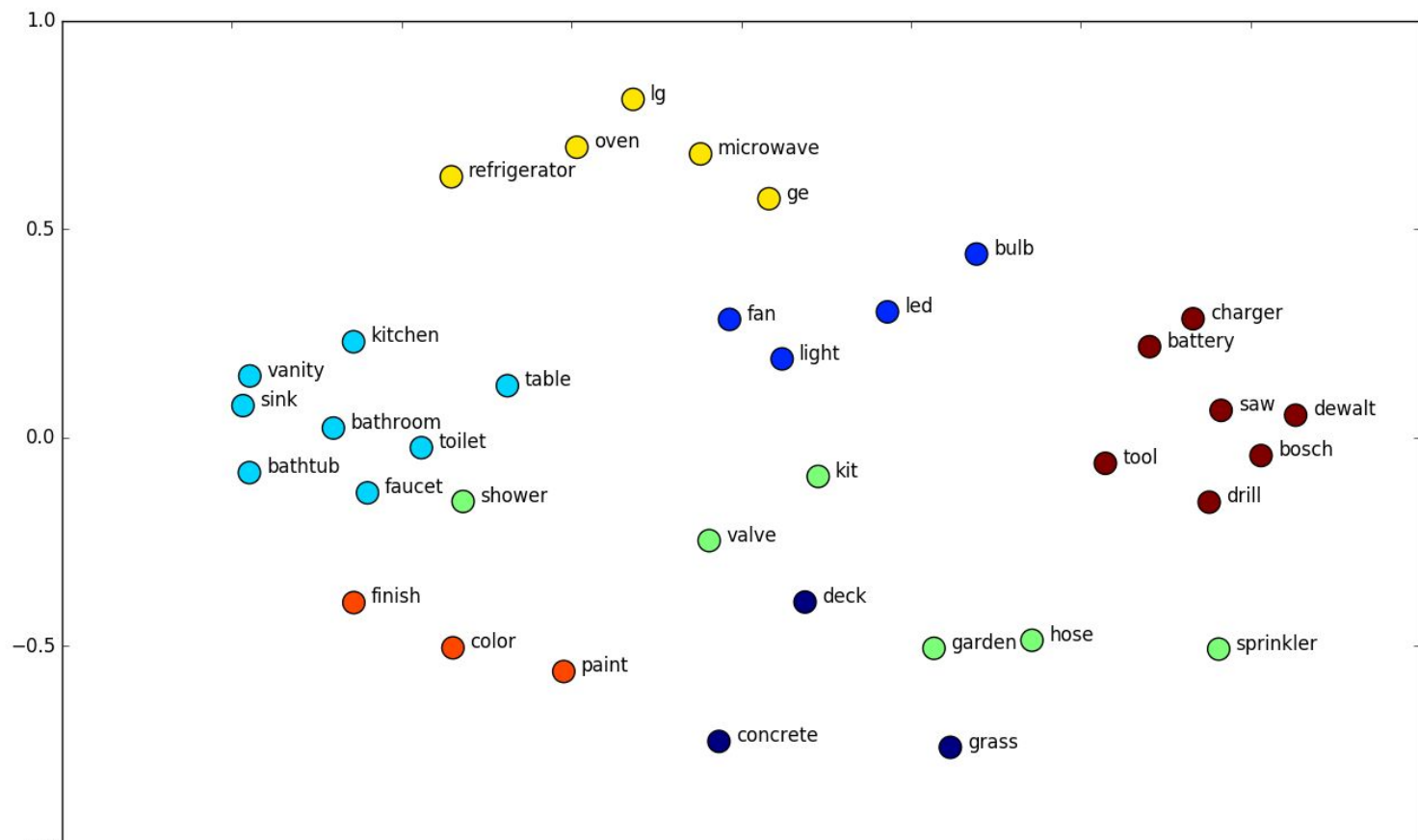
Verb tense



Country-Capital



# Results



# An interesting application

- Nature, July 2019 V. Tshitonya et al, “Unsupervised word embeddings capture latent knowledge from materials science literature”.
- Lawrence Berkeley lab material scientists applied word embedding to 3.3 million scientific abstracts published between 1922-2018. V=500k. Vector size: 200 dimension, used skip-gram model
- Captured things like periodic table and structure-property relationship in materials:  
ferromagnetic – NiFe + IrMn  $\approx$  antiferromagnetic
- Discovered new thermoelectric materials

# Contextualized Word Vectors

- Independent Word Vectors Fails to account for lexical ambiguity or dependence of word meaning on context.
- Contextualized Word Vectors produce a vector representation for a specific occurrence of a word, by using textual context to compute its meaning.
  - ELMo (Embeddings from Language Models, Peters et al., 2018)
  - Bidirectional Encoder Representations from Transformers (BERT, Devlin et al., 2018)
  - XLNet
  - Ernie