# Vector Space Models:
## Word, Sentence, Document Levels

Ch.10 Text Analytics with Python. Dipanjan Sankar. 2019. Apress
Advanced Deep Learning with Python. Ivan Vasilev. 2019. Packt Publishing

# Word Embedding Revisited

**Distributed Representation** = Word Vector
= Word Embedding

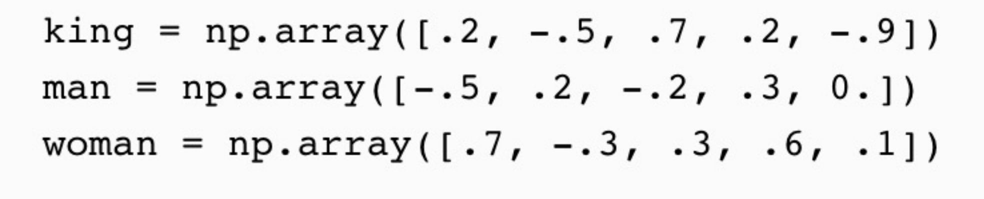
**Bag-of-Words Models**

- Count-based: TF, TF-IDF, N-grams

**Prediction-Based Models**

- Based on distributed representations (a dense representations of words in a low-dimensional vector space): Word2Vec, FastText

Word = one point in the embedding space

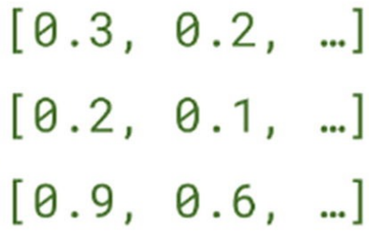


```
king = np.array([.2, -.5, .7, .2, -.9])
man = np.array([-.5, .2, -.2, .3, 0.])
woman = np.array([.7, -.3, .3, .6, .1])
```

n dimensions = 5: 1 x 5 vector

Word is associated with a continuous vector representation

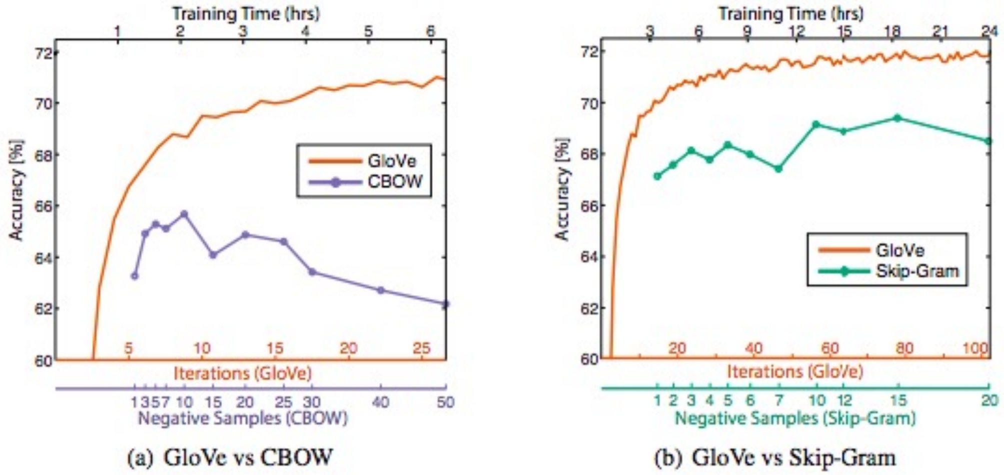
Dimensions = latent characteristics of a word (grammatical or semantic property)



```
[0.3, 0.2, …]
[0.2, 0.1, …]
[0.9, 0.6, …]
```

Sarkar, D. 2018. Deep Learning Methods for Text Data – Word2Vec, GloVe, FastText. Towards Data Science

# Word Embeddings: Various Approaches



(a) GloVe vs CBOW

(b) GloVe vs Skip-Gram

(Pennington et al., 2014, p.10)

**Word2Vec (2013)**
**Google**

- Smallest unit = word
- Fixed-size vocabulary
- Local context window
- Predictive model

**GloVe (2014)**
**Stanford**

- Smallest unit = word
- Frequent co-occurrences carry additional information
- Fixed-size vocabulary
- Count-based model (Global co-occurrence counts)

**FastText (2016)**
**Facebook**

- Smallest unit = character    <where> : <wh, whe,her,ere,re>
- Generalization to unknown words (OOV: out-of-vocabulary)

Sarkar, D. 2019. Deep Learning Methods for Text Data – Word2Vec, GloVe, FastText. Towards Data Science

Pennington J. et al. 2014. GloVe: Global Vectors for Word representation. In the Proceedings of the 24th EMNLP.

Bohanovski, P. et al. 2017. Enriching Word Vectors with Subword Information. In Transactions of the ACL, 5, 135-146.

# Embedding

**Embedding**

A fixed-length vector typically used to encode and represent an entity (document, sentence, word, graph).

**Word Embedding** - Encoding words in fixed-length dense vectors

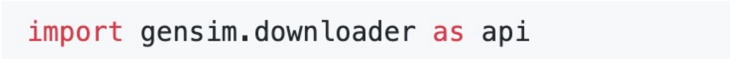
**Sentence Embedding** - Encoding sentences in fixed-length dense vectors

**Universal Embedding** - Pre-trained embedding = **Transfer Learning**

## Learn an Embedding

a large amount of text data

1

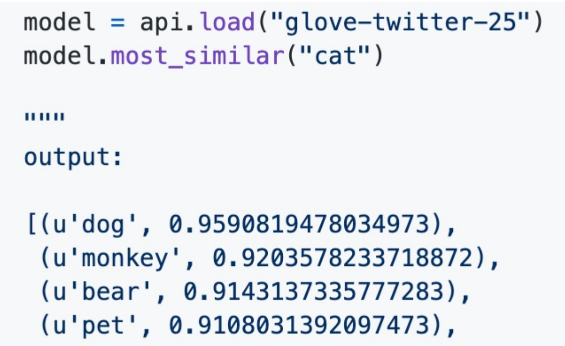


```
import gensim.downloader as api
```

## Reuse an Embedding

Using pre-trained models to generalized representations on new text data

2

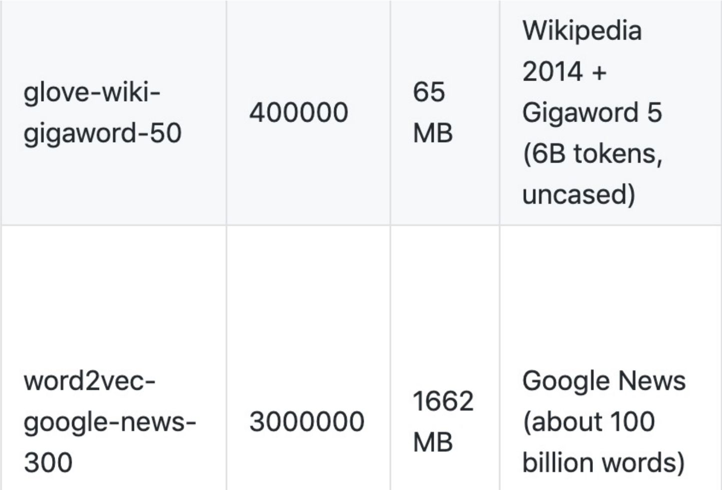


```
model = api.load("glove-twitter-25")
model.most_similar("cat")

"""
output:

[(u'dog', 0.9590819478034973),
 (u'monkey', 0.9203578233718872),
 (u'bear', 0.9143137335777283),
 (u'pet', 0.9108031392097473),
```

https://github.com/RaRe-Technologies/gensim-data



| | | | |
|---|---|---|---|
| glove-wiki-gigaword-50 | 400000 | 65 MB | Wikipedia 2014 + Gigaword 5 (6B tokens, uncased) |
| word2vec-google-news-300 | 3000000 | 1662 MB | Google News (about 100 billion words) |

Sarkar, D. 2019. Chapter 10. The Promise of Deep Learning. Apress

Sarkar, D. 2019. Deep Learning Methods for Text Data – Word2Vec, GloVe, FastText. Towards Data Science

# Sentence Embedding

## Doc2Vec

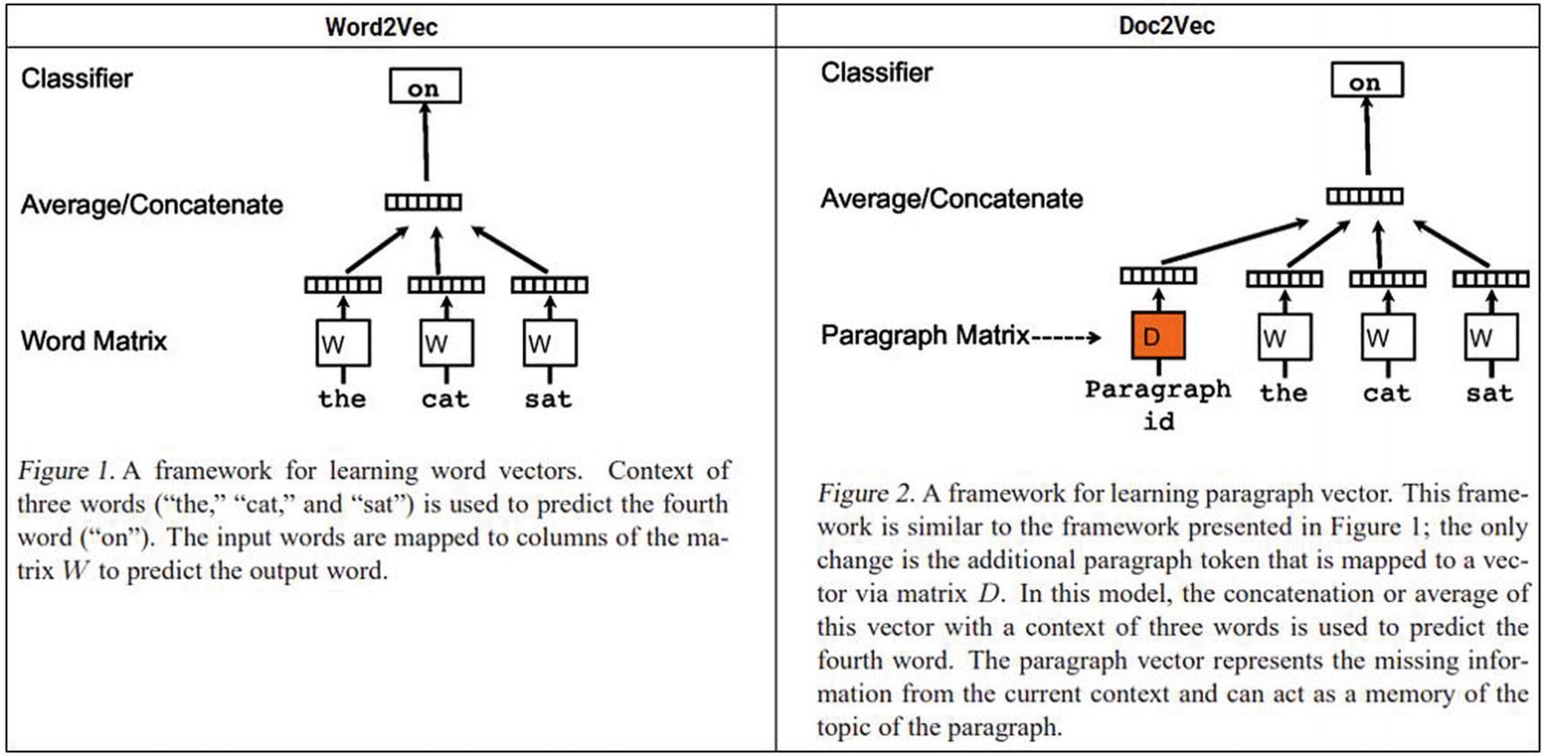
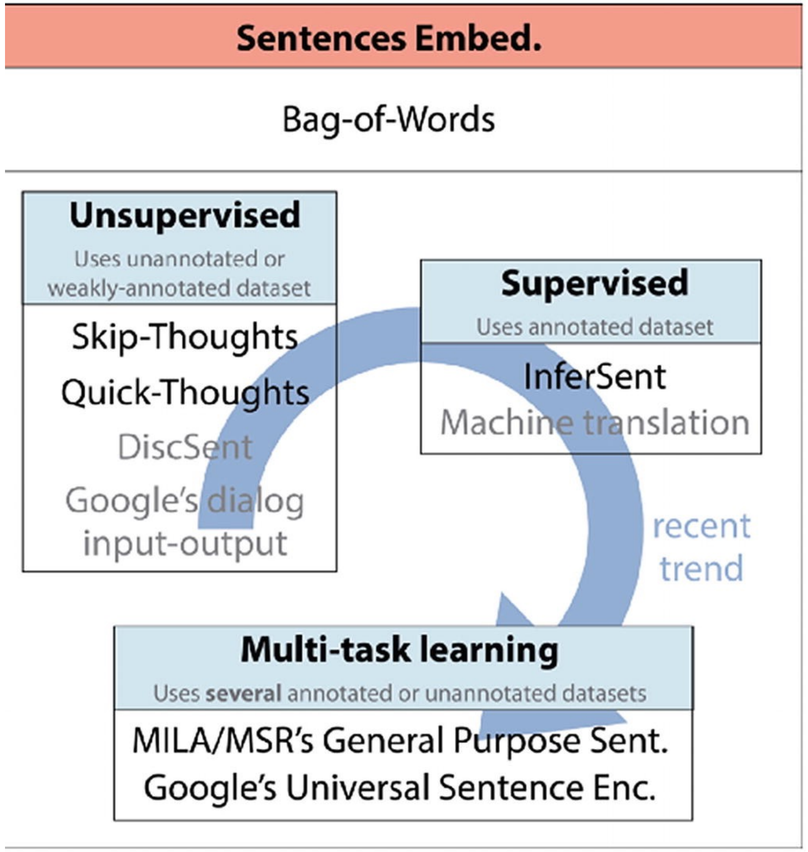


*Figure 1.* A framework for learning word vectors. Context of three words ("the," "cat," and "sat") is used to predict the fourth word ("on"). The input words are mapped to columns of the matrix $W$ to predict the output word.

*Figure 2.* A framework for learning paragraph vector. This framework is similar to the framework presented in Figure 1; the only change is the additional paragraph token that is mapped to a vector via matrix $D$. In this model, the concatenation or average of this vector with a context of three words is used to predict the fourth word. The paragraph vector represents the missing information from the current context and can act as a memory of the topic of the paragraph.



**Sentences Embed.**

Bag-of-Words

**Unsupervised**
Uses unannotated or weakly-annotated dataset

Skip-Thoughts
Quick-Thoughts
DiscSent
Google's dialog input-output

**Supervised**
Uses annotated dataset

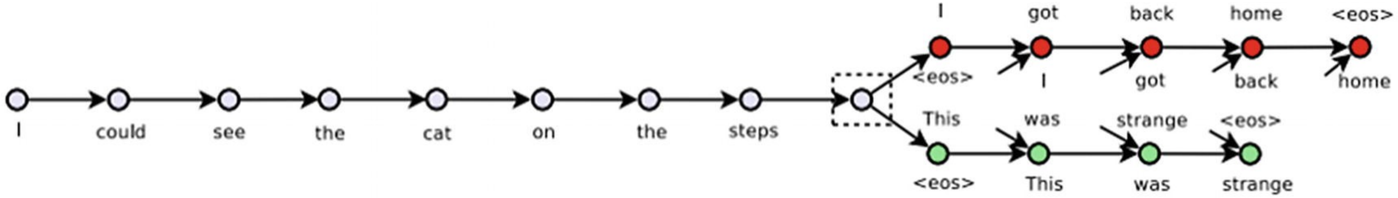
InferSent
Machine translation

recent trend

**Multi-task learning**
Uses **several** annotated or unannotated datasets

MILA/MSR's General Purpose Sent.
Google's Universal Sentence Enc.

**Skip-thoughts (unsupervised)**
(similar to skip-gram model)



**InferSent (supervised)**

Word2Vec vs. Doc2Vec (source: [https://arxiv.org/abs/1405.4053](https://arxiv.org/abs/1405.4053) )

Ch.10 Text Analytics with Python. Dipanjan Sankar. 2019. Apress

Thomas Wolf. 2018. The Current Best of Universal Word Embeddings and Sentence Embeddings