

# MSiA414 SEC01

## Text Analytics

### Lab 2 - Word2Vec

Timo Wang

Northwestern University

October 1st, 2020

Some slides of this document is built based on the content provided on  
<https://github.com/tmikolov/word2vec>.

What is Word2Vec?

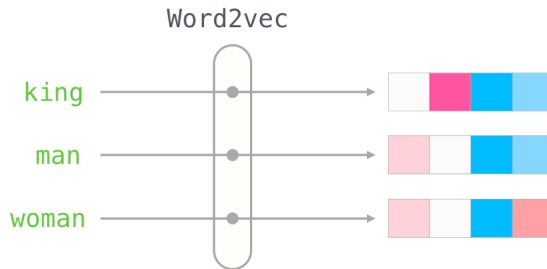
Why do we care?

How does it work?

How well does it perform?

Tools & libraries

# What is Word2Vec?



**Figure:** It encodes a word into a vector, in comparison to one-hot encoder, where each word is represented by an integer.

What is Word2Vec?

Why do we care?

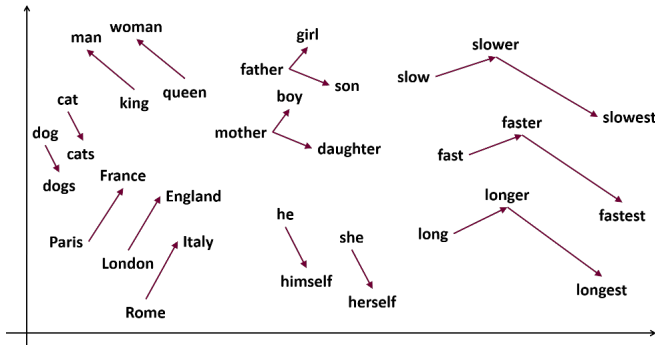
How does it work?

How well does it perform?

Tools & libraries

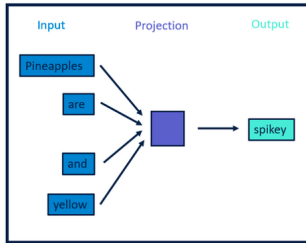
# Why do we care?

Capture of word meaning and relations

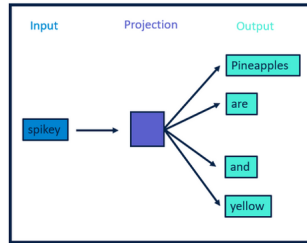


What is Word2Vec?  
Why do we care?  
**How does it work?**  
How well does it perform?  
Tools & libraries

# How does it work?



CBOW



Skip-gram

What is Word2Vec?  
Why do we care?  
How does it work?  
How well does it perform?  
Tools & libraries

Strength  
Weakness

# How well does it perform?

Strength

```
~$ ./word2vec-master (distance)
Last login: Thu Oct 30 11:18:08 on tty000
~/E/ta-fair-text-analytics$ ./word2vec-master -distance -topk 10 -input /data/ta-fair-text-analytics/word2vec/vectors.txt
Input file found
Enter word or sentence (EXIT to break): pear
word: pear Position in vocabulary: 19769

-----
Word      Cosine distance
-----
kumquat   0.588867
plum      0.587566
pineapple 0.583902
drupe     0.559355
fruity    0.553488
juniper   0.549459
peony     0.538453
peach     0.538459
japonica  0.527448
apricot   0.526989
tamarind  0.521487
butyrate  0.521388
flashy    0.522059
coriander 0.518283
berries   0.517893
sweet     0.515998
grape     0.508633
prickly   0.507783
fruit     0.506325
raspberry 0.506352
fortunella 0.504774
stalk     0.502588
citrus    0.502315
carabiner 0.502011
clement 0.499654
chiling   0.499507
quince    0.499433
cabbage   0.499353
almond    0.499176
walnut    0.499062
grayish   0.498743
petals    0.498485
net       0.498192
juices    0.497665
plum      0.497385
raisins   0.497039
cultivars 0.496877
acetal    0.496148
cultivar  0.494975
unfermented 0.489712

Enter word or sentence (EXIT to break):
```

What is Word2Vec?  
Why do we care?  
How does it work?  
How well does it perform?  
Tools & libraries

Strength  
Weakness

# How well does it perform?

Weakness

```
~$ ./word2vec-master (distance)
Enter word or sentence (EXIT to break): apple
word: apple Position in vocabulary: 1221

----- Word Cosine distance -----
macintosh 0.682317
liga 0.683916
lisa 0.559909
appleworks 0.589189
book 0.588229
performance 0.565671
quickdraw 0.543338
amiga 0.538145
amiga 0.526469
javelin 0.524408
hypercard 0.522909
lra 0.515485
lisa 0.511895
compaq 0.509782
lrc 0.509215
mexc 0.506404
microcomputer 0.505918
macintosh 0.503702
mac 0.503243
visiomatic 0.503209
wozniak 0.502743
ipod 0.494812
os 0.494728
openstep 0.488405
jer 0.488075
macbook 0.487304
macintoshes 0.486822
rakitin 0.484608
microsoft 0.483773
atarl 0.475369
imovie 0.469714
commodore 0.468627
javel 0.460134
amigone 0.462212
prodos 0.459668
amigaos 0.458595
pda 0.458483
ceme 0.458265
geos 0.457607
msx 0.457265

Enter word or sentence (EXIT to break):
```

What is Word2Vec?

Why do we care?

How does it work?

How well does it perform?

Tools & libraries

Demo

The original library

Gensim wrapper for Python

Troubleshooting tips for the original library (macOS)

More resources on the Gensim library

# Tools & libraries

## Demo

### Shell

```
# Download and decompress a sample text corpus.
wget http://mattmahoney.net/dc/text8.zip -O text8.gz
tar xvf text8.gz

# Train a model named vectors.bin with the downloaded
  corpus text8
./word2vec -train text8 -output vectors.bin -cbow 1 -
  size 200 -window 8 -negative 25 -hs 0 -sample 1e-4 -
  threads 20 -binary 1 -iter 15
# Examine the embedding
./distance vectors.bin
```

What is Word2Vec?  
Why do we care?  
How does it work?  
How well does it perform?  
**Tools & libraries**

Demo

**The original library**

Gensim wrapper for Python

Troubleshooting tips for the original library (macOS)

More resources on the Gensim library

## Tools & libraries

### The original library

Step 1 Download the project from

<https://github.com/tmikolov/word2vec>.

Step 2 Read `demo-word.sh`.

Step 3 Use `demo-word.sh` as a guideline and train the model with your own corpus.



What is Word2Vec?

Why do we care?

How does it work?

How well does it perform?

**Tools & libraries**

Demo

The original library

**Gensim wrapper for Python**

Troubleshooting tips for the original library (macOS)

More resources on the Gensim library

# Tools & libraries

## Gensim wrapper for Python

Step 1 Install Gensim with `pip install gensim`.

Step 2 Depending on the content of your corpus, you may need to process the text corpus into this format:  
`List[List[str]]`.

Step 3 Train and save the model.

What is Word2Vec?

Why do we care?

How does it work?

How well does it perform?

Tools & libraries

Demo

The original library

Gensim wrapper for Python

Troubleshooting tips for the original library (macOS)

More resources on the Gensim library

## Tools & libraries

### Troubleshooting tips for the original library (macOS)

- 1 `./distance` results in a segmentation fault
  - remove `-march=native` from `makefile`
- 2 Undefined symbols for architecture `x86_64`:  
“`_fgetc_unlocked`”, referenced from:
  - replace `fgetc_unlocked` with `getc_unlocked` and  
`fputc_unlocked` with `putc_unlocked`

What is Word2Vec?

Why do we care?

How does it work?

How well does it perform?

Tools & libraries

Demo

The original library

Gensim wrapper for Python

Troubleshooting tips for the original library (macOS)

More resources on the Gensim library

## Tools & libraries

More resources on the Gensim library

- <https://radimrehurek.com/gensim/models/word2vec.html>
- <https://towardsdatascience.com/a-beginners-guide-to-word-embedding-with-gensim-word2vec-model5970fa56cc92>