MSiA490 SEC20/28
Text Analytics
Lab 2 - Word2Vec

Timo Wang

Northwestern University

September 24th, 2020

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

What is Word2Vec?

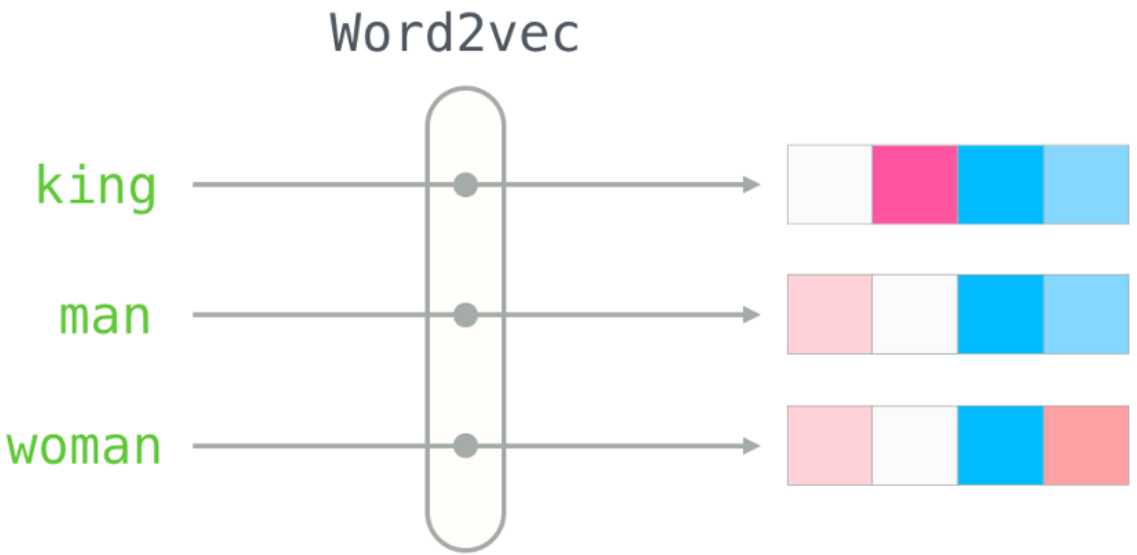
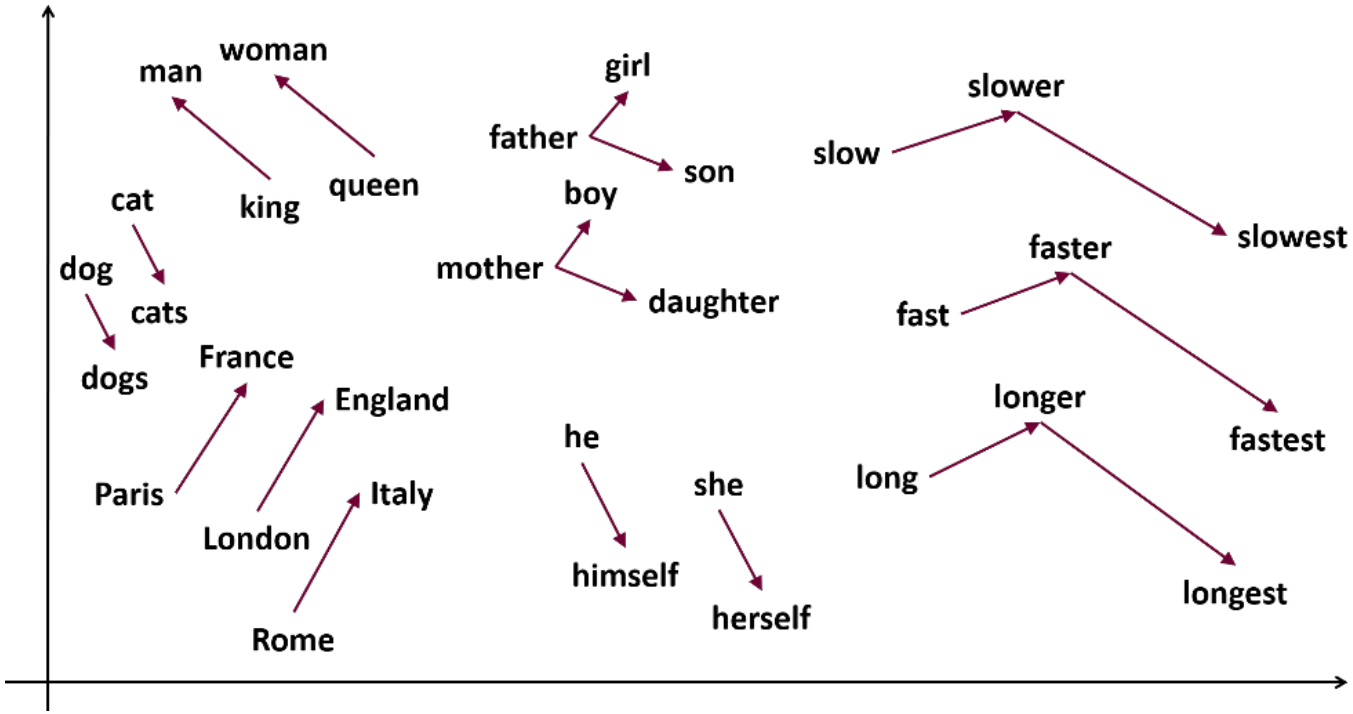


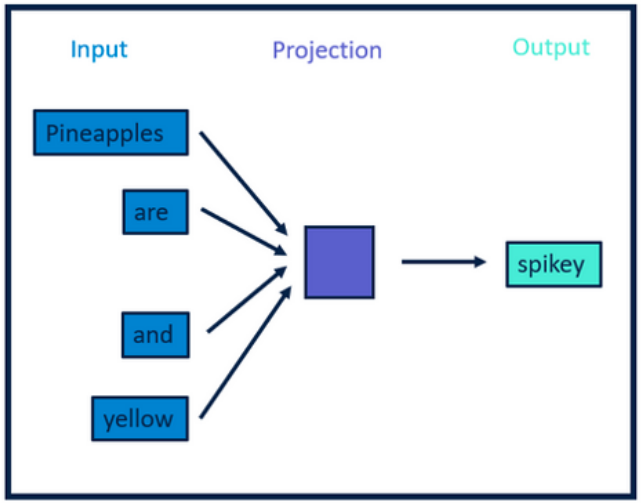
Figure: It encodes a word into a vector, in comparison to one-hot encoder, where each word is represented by an integer.

Why do we care?

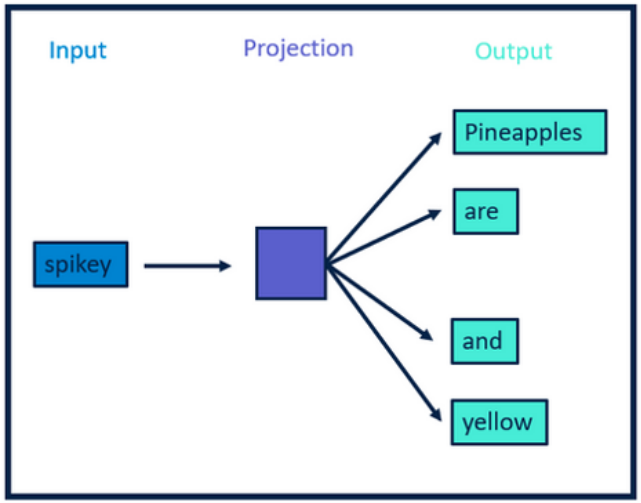
Capture of word meaning and relations



How does it work?



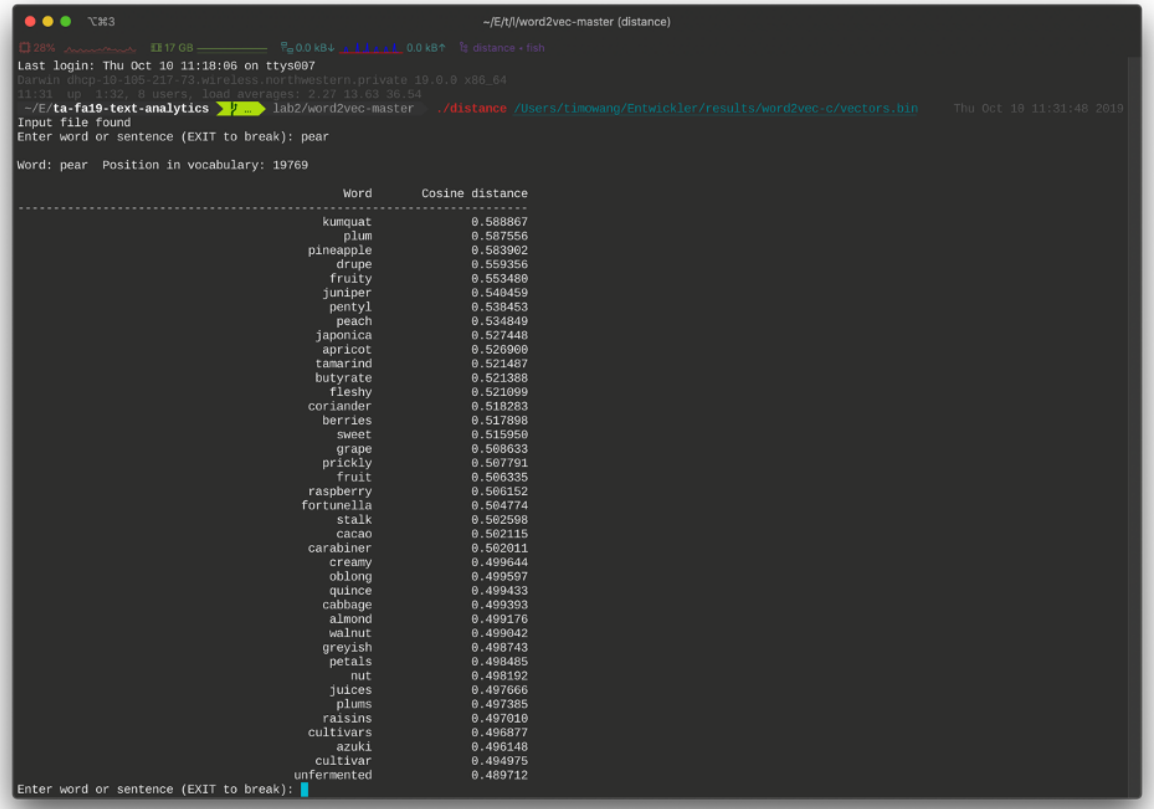
CBOW



Skip-gram

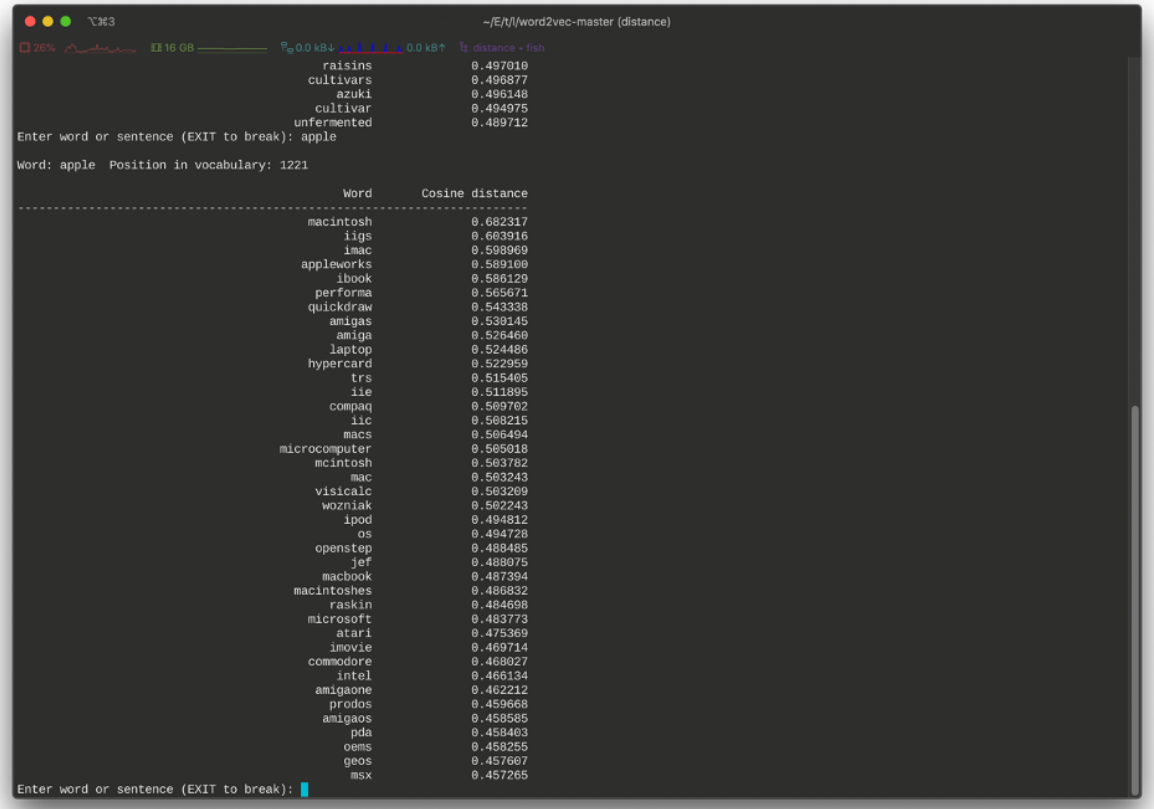
How well does it perform?

Strength



How well does it perform?

Weakness



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
```

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.

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.

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`

Tools & libraries

More resources on the Gensim library

- https://radimrehurek.com/gensim/models/word2vec.html
- https://towardsdatascience.com/a-beginners-guide-toword-embedding-with-gensim-word2vec-model5970fa56cc92

Quiz

Task 1

Which of the following is/are the reason/reasons why the embeddings produced by a trained Word2Vec model results in poor text processing performance?

A The text corpus for training is too small

B The content of the corpus is irrelevant to the text that needs to be processed

C The model is overfitted

D All of the above

Quiz

Task 2

In which language was the original Word2Vec library developed by Mikolov implemented?

A C++

B Java

C C

D Python

Thoughts & feedbacks