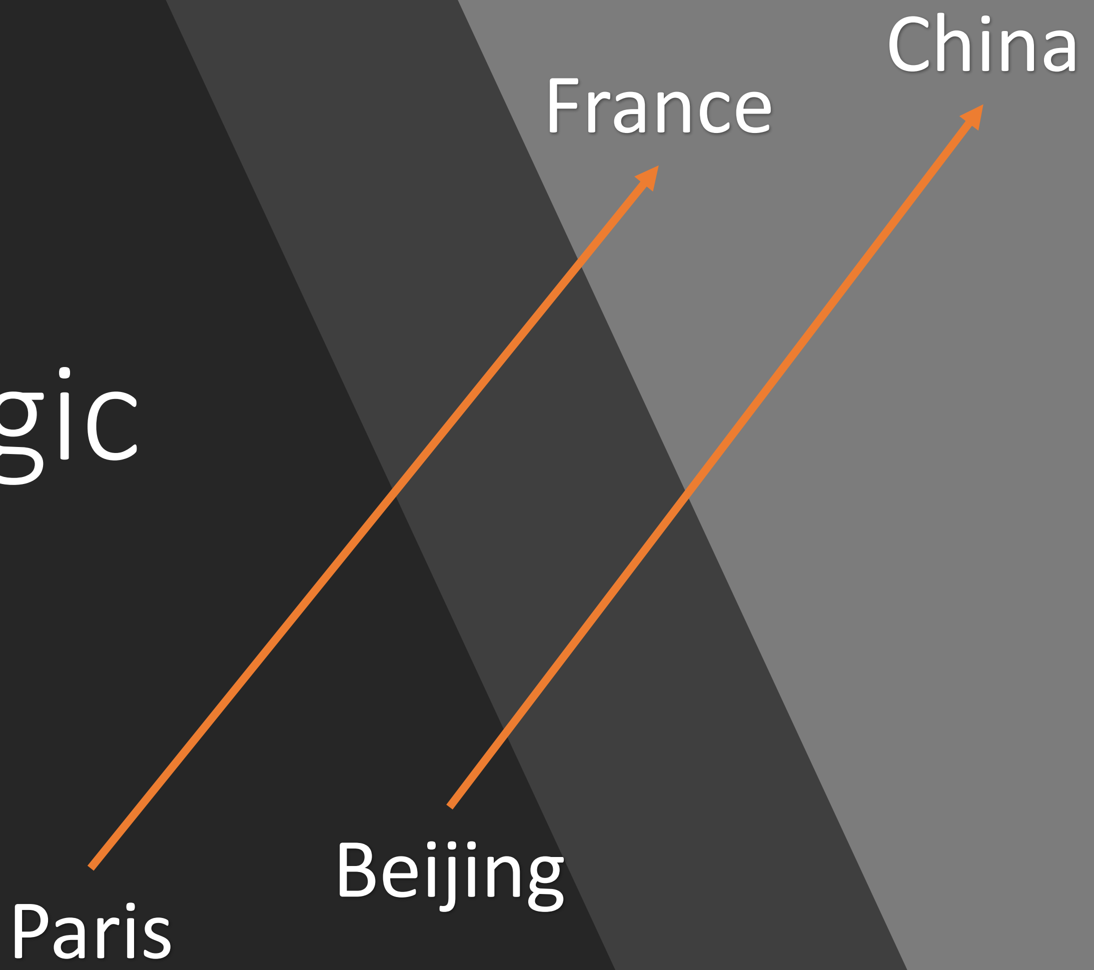
Shubhanshu Mishra
http://Shubhanshu.com

Word Embedding
Unravelling the magic



What are word vectors?

- Think of representing a sentence for a given task such as text classification:

The cat sat on the mat

=

Bag of words assumption

cat mat on sat The the

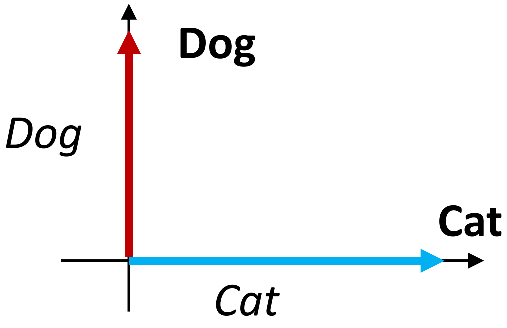
Very simple word vector?

Vocab →	The	Cat	Sat	On	mat	Dog
The	1	0	0	0	0	0
Cat	0	1	0	0	0	0
Sat	0	0	1	0	0	0
On	0	0	0	1	0	0
The	1	0	0	0	0	0
mat	0	0	0	0	1	0

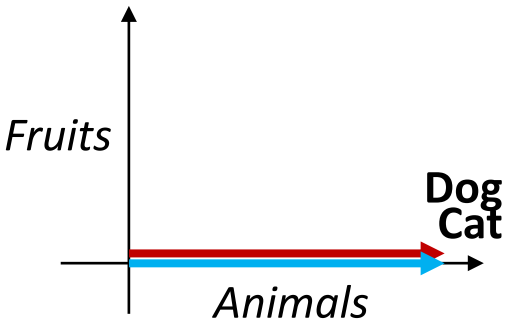
Vocab size = |V| = 5

- Matrix grows column wise for each new word
- What if new sentence was – ***The dog sat on the mat*** – are they similar?
- This is commonly referred to as ***One-hot representation***

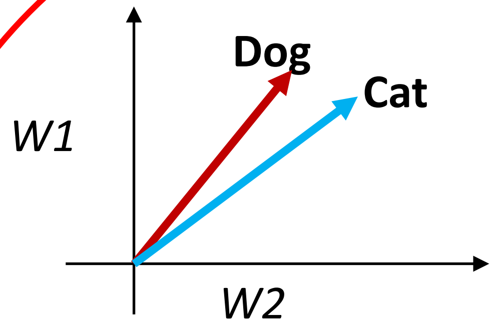
Similarity(word1, word2) = $V_1 \cdot V_2 = \sum_i v_1^i * v_2^i$



One hot
- No similarity
- Too sparse



Dictionary based
- Perfect similarity
- Needs dictionaries
- High error on unseen words



Task based
- Optimal representation
- Learn from data
- Can infer representation of new words

Approaches

You shall know a word by the company it keeps

(Firth, J. R. 1957:11)

Distributional hypothesis

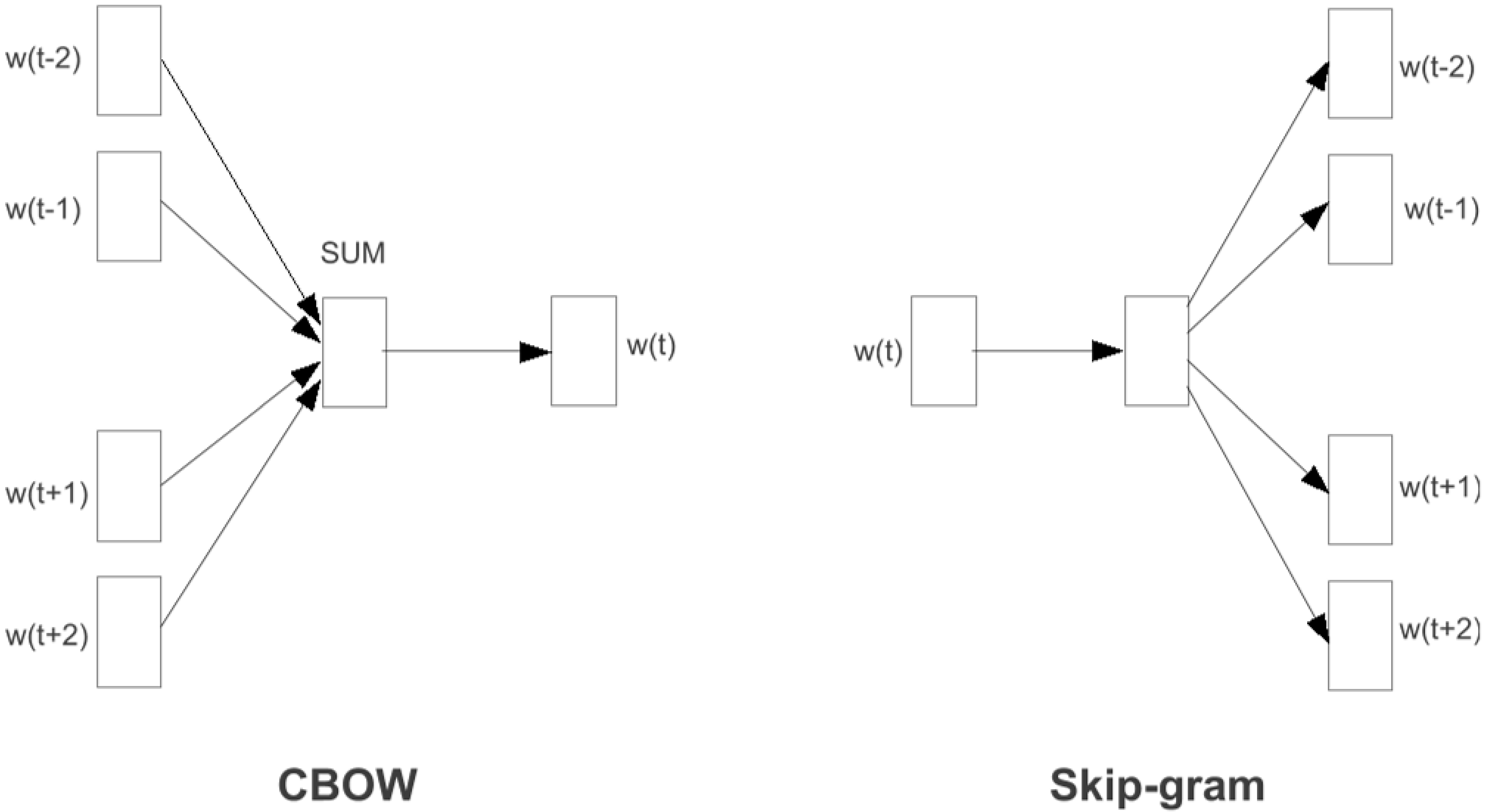
Harris, Z. (1954). "Distributional structure". Word. ***10*** *(23): 146–162.*

- Any way which defines the concept of "company" (usually referred to as context in many models) in a more useful manner.
- Count co-occurrence based word representations (context = other words in a window)
- Topic model based representations (LSI and LDA) (context = document and latent topics)
- Dictionary based representation (context = various dictionaries)

```
>>> text1.concordance("monstrous")
Displaying 11 of 11 matches:
ong the former , one was of a most monstrous size . ... This came towards us ,
ON OF THE PSALMS . " Touching that monstrous bulk of the whale or ork we have r
ll over with a heathenish array of monstrous clubs and spears . Some were thick
d as you gazed , and wondered what monstrous cannibal and savage could ever hav
that has survived the flood ; most monstrous and most mountainous ! That Himmal
they might scout at Moby Dick as a monstrous fable , or still worse and more de
th of Radney .'" CHAPTER 55 Of the monstrous Pictures of Whales . I shall ere l
ing Scenes . In connexion with the monstrous pictures of whales , I am strongly
ere to enter upon those still more monstrous stories of them which are to be fo
ght have been rummaged out of this monstrous cabinet there is no telling . But
of Whale - Bones ; for Whales of a monstrous size are oftentimes cast up dead u
>>>
```

Word concordance
Source: http://www.nltk.org/book/ch01.html

Neural word representations (embedding)



The CBOW architecture predicts the current word based on the context, and the Skip-gram predicts surrounding words given the current word.

Image source: Mikolov, Tomas, Kai Chen, Gregory S. Corrado and Jeffrey Dean. "Efficient Estimation of Word Representations in Vector Space." *CoRR* abs/1301.3781 (2013): n. pag.

Fun with word vectors

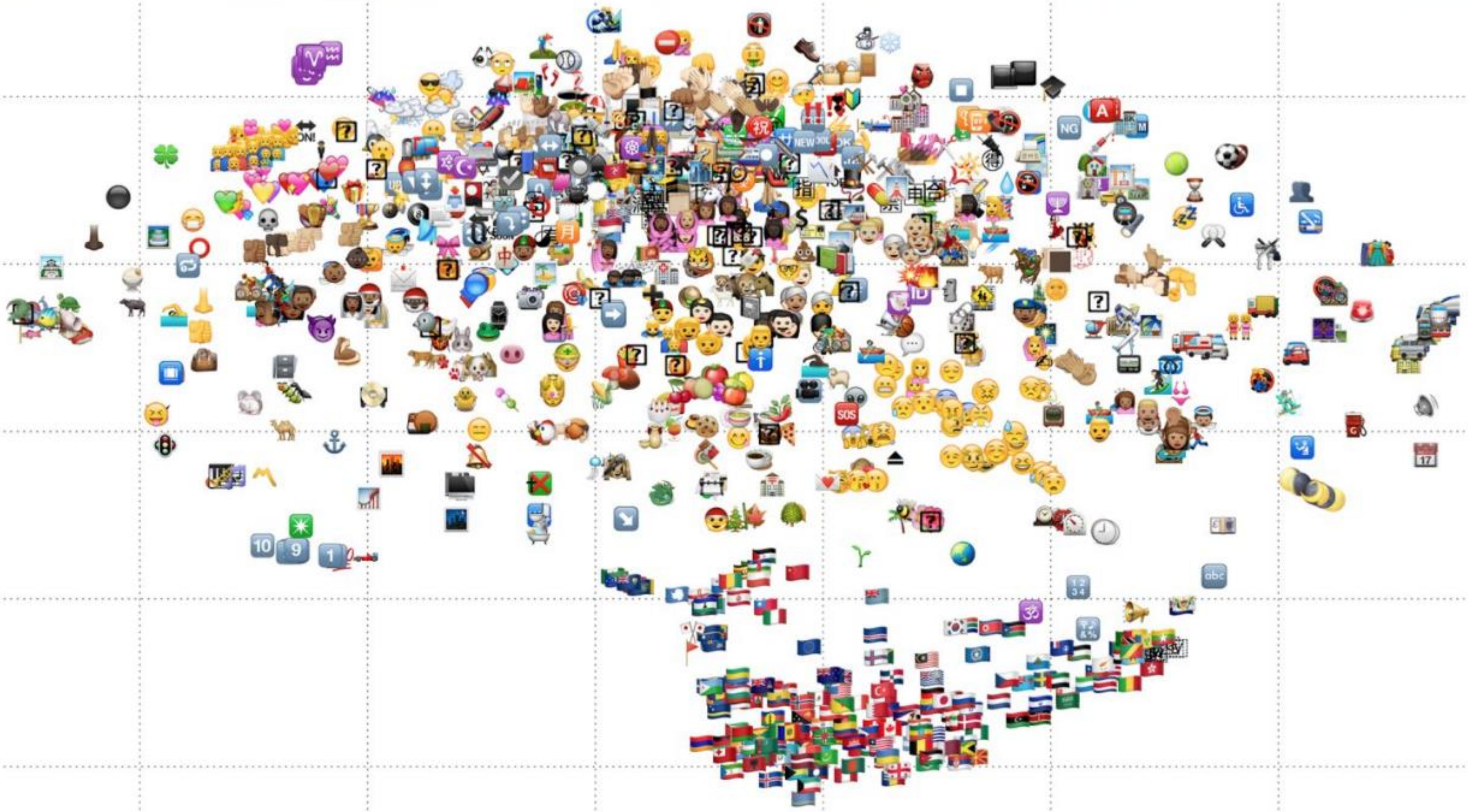


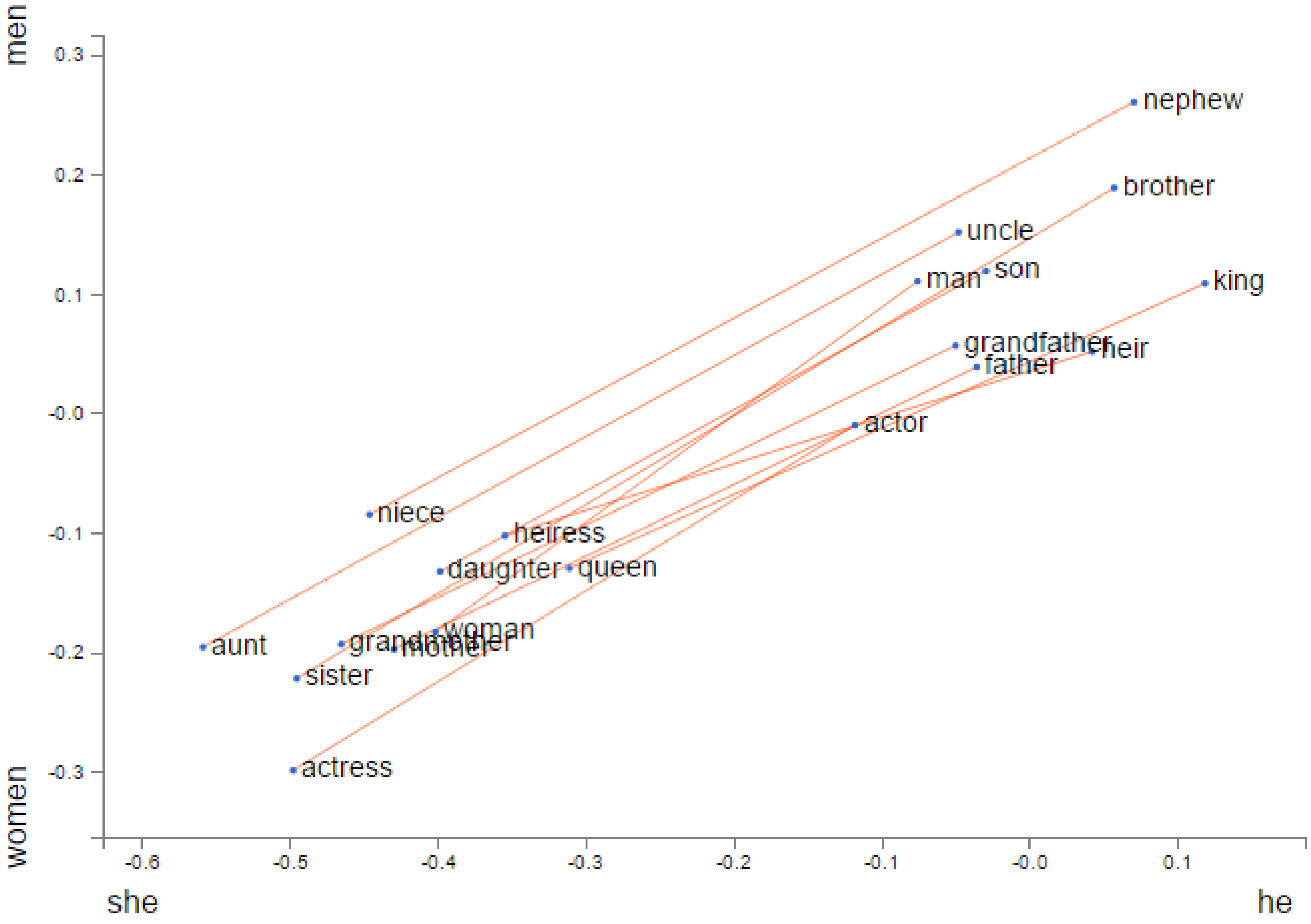
Figure 3: Emoji vector embeddings, projected down into a 2-dimensional space using the t-SNE technique. Note the clusters of

Shubhanshu Mishra @TheShubhanshu · Jan 25
Of-course someone did emoji2vec. That moment you think of a X2vec for some data X & there exists a paper on it in @arxiv_org. Rule 34 of ML.

2

Words aligned along gender dimensions.

In 50 dimensional glove embedding space.



Source: https://lamyiowce.github.io/word2viz/

Interesting uses

- Rejecting the gender binary. - http://bookworm.benschmidt.org//posts/2015-10-30-rejecting-the-gender-binary.html
  - Simple math - similar words should have similar dot products in the same context. v.(u1-u2) should be close to 0.
  - Further (u1 - u2) denotes a dimension of difference between vectors so (man - woman) is similar dimension of gender, of course we can take (mean(male words) - mean(female words)) to be a more robust dimension of gender.
  - In fact go further and use any function to compose the f(words) to weight words based on importance.

Applications

- Going to our original problem - representing documents as bag of words efficiently
  - document - f(words), f(words) can be composed in different ways:
  - mean(word embedding) - Bag of word assumption but encodes related word enformation - works quite well
  - weighted mean (word embedding) - use your favorite, entropy, TF-IDF etc
  - max(word embedding)
  - use some deep neural network to approximate the composition of word embeddings for the task. (Most used in Deep Learning community)
- Directly learn the representation of the document – ”Distributed Representations of Sentences and Documents” Quoc Le, 2015.
- “Retrofitting Word Vectors to Semantic Lexicons” – Faruqui et al, NAACL 2015
- C-BOW is better for smaller corpus and Skipgram is better for larger corpora.

Improved sentence representation?

Vocab →	*W1*	*W2*	*W3*	*W4*	*W5*	*W6*
The	0.5	-0.03	0.33	1.5	-0.5	-0.4
Cat	0.6	-1.03	1.93	-0.9	0.4	-0.5
Sat	0.4	-0.63	0.37	1.5	0.2	-0.1
On	-0.7	-0.07	0.43	1.5	0.2	1.1
The	-0.6	-0.03	0.33	1.6	-1.2	5.1
mat	0.3	-0.04	0.33	1.9	1.2	6.1

Vocab size = |V| = 5

- **Fixed** number of columns = number of embedding dimensions d.
- Take average of each column to get d dimensional document representation.
- Can handle unseen words which are present in unlabeled corpora e.g. – ***The dog sat on the mat*** – should be similar?
- This is commonly referred to as ***Average word embedding***

Resources

- Pre-trained word embedding:
  - Google word2vec: https://code.google.com/archive/p/word2vec/
  - Glove: http://nlp.stanford.edu/projects/glove/
  - PubMed: http://bio.nlplab.org/
- Word embedding software:
  - Google code for word to vec: https://opensource.googleblog.com/2013/08/learning-meaning-behind-words.html
  - Gensim for python: https://radimrehurek.com/gensim/
  - GloVe: http://nlp.stanford.edu/projects/glove/
  - Retrofitting code: https://github.com/mfaruqui/retrofitting

Ideas and discussion

- What applications can you think of?
- Questions?