What is a *wampos*?

...some believe that	**wampos**	scales have medicinal qualities...
...approach to fighting	**wampos**	(and general wildlife) trafficking...
Even though	**wampos**	scales are made of exactly the...

Utrecht University

Word Embeddings

Introduction to Text Mining with R

Images by Dong Nguyen

2023-07-13

Utrecht University

What is a *wampos*?

...some believe that	**wampos**	scales have medicinal qualities...
...approach to fighting	**wampos**	(and general wildlife) trafficking...
Even though	**wampos**	scales are made of exactly the...

What is a **wampos**?

What is a *wampos*?

Figure: Photo by Piekfrosch;
CC-BY-SA-3.0

...some believe that	**wampos**	scales have medicinal qualities...
...approach to fighting	**wampos**	(and general wildlife) trafficking...
Even though	**wampos**	scales are made of exactly the...

wampos = pangolin

You shall know a word by
the company it keeps
(Firth, J. R. 1957:11)

What is a *wampos*?

Figure: Photo by Piekfrosch;

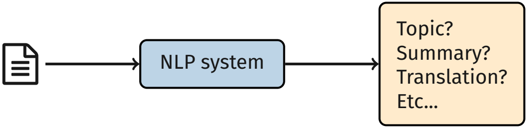
...some believe that	**wampos**	scales have medicinal qualities...
...approach to fighting	**wampos**	(and general wildlife) trafficking...
Even though	**wampos**	scales are made of exactly the...

wampos = *pangolin*

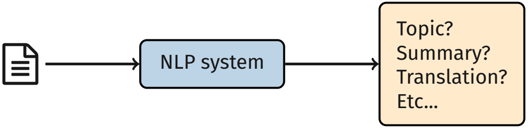
You shall know a word by the company it keeps (Firth, J. R. 1957:11)

The distributional hypothesis: Words that occur in similar contexts tend to have similar meanings

Recap: NLP

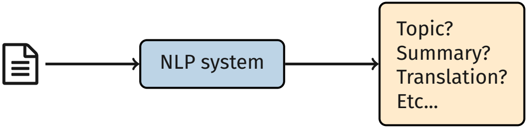


Recap: NLP



- Rule-based
- Machine Learning-based

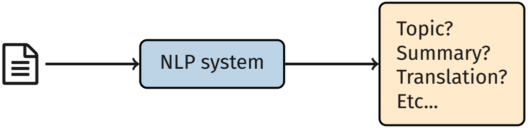
Recap: NLP



- Rule-based
- Machine Learning-based

How can we convert texts into numbers?
Collect answers on blackboard

Recap: NLP



- Rule-based
- Machine Learning-based

Bag-of-words, TF-IDF, ...

Recap: One hot encoding

Map each word to a unique identifier
e.g. *cat (3)* and *dog (5)*.
$\rightarrow$ Vector representation: all zeros, except 1 at the ID

cat	0	0	1	0	0	0	0
dog	0	0	0	0	1	0	0
car	0	0	0	0	0	0	1

Recap: One hot encoding

Map each word to a unique identifier
e.g. *cat (3)* and *dog (5)*.
$\rightarrow$ Vector representation: all zeros, except 1 at the ID

cat	0	0	1	0	0	0	0
dog	0	0	0	0	1	0	0
car	0	0	0	0	0	0	1

Even related words have distinct vectors!

High number of dimensions

😔

Recap: Topic Modeling

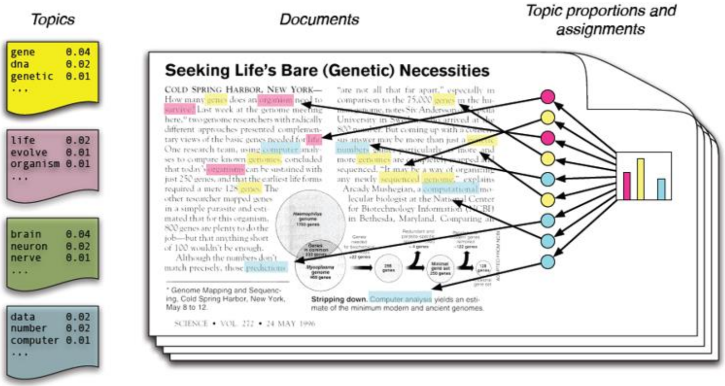


Figure: Text Clustering handout (Ayoub Bagheri)

Why do we need word embeddings?

Word representations

How can we represent the *meaning* of words?

Word representations

How can we represent the *meaning* of words?

So we can ask:

- How similar is *cat* to *dog*, or *Paris* to *London*?
- How similar is *document A* to *document B*?

Word representations

How can we represent the *meaning* of words?

So we can ask:

- How similar is *cat* to *dog*, or *Paris* to *London*?
- How similar is *document A* to *document B*?

And use such representations for:

- various NLP tasks: translation, classification, etc.
- studying linguistic questions

Words as vectors

The vector representations should:

- capture semantics
  - similar words should be close to each other in the vector space
  - relation between vectors should reflect the relationship between words
- be efficient (vectors with fewer dimensions are easier to work with)
- be interpretable

Words as vectors

The vector representations should:

- capture semantics
  - similar words should be close to each other in the vector space
  - relation between vectors should reflect the relationship between words
- be efficient (vectors with fewer dimensions are easier to work with)
- be interpretable

How similar are *smart* and *intelligent?* (not similar 0–10 very similar):
How similar are *easy* and *big* (not similar 0–10 very similar):

Words as vectors

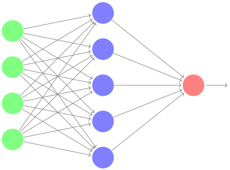
The vector representations should:

- capture semantics
  - similar words should be close to each other in the vector space
  - relation between vectors should reflect the relationship between words
- be efficient (vectors with fewer dimensions are easier to work with)
- be interpretable

How similar are *smart* and *intelligent?* (not similar 0–10 very similar): 9.2
How similar are *easy* and *big* (not similar 0–10 very similar): 1.12
(SimLex-999 dataset)

How are they used?

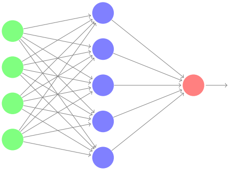
How are they used?



In neural networks (text classification, sequence tagging, etc..)

How are they used?

How are they used?



In neural networks (text classification, sequence tagging, etc..)

cat	0.52	0.48	-0.01	...	0.28
dog	0.32	0.42	-0.09	...	0.78

As research objects

Word embeddings (*vs* One-hot encoding)

Word embeddings:

- Vectors are short; typically 50-1024 dimensions ☺
- Very effective for many NLP tasks ☺
- Vectors are dense (mostly non-zero values)
- Individual dimensions are less interpretable ☹

cat	0.52	0.48	-0.01	...	0.28
dog	0.32	0.42	-0.09	...	0.78

Agenda

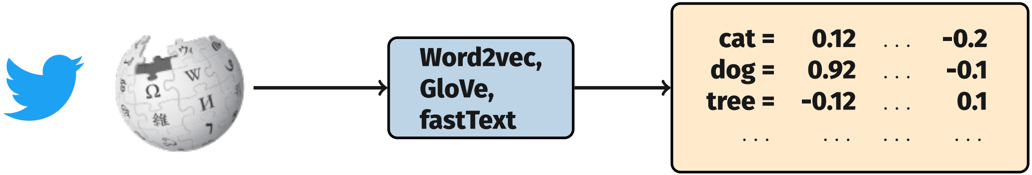
- ~~What are word embeddings?~~
- How do we learn word embeddings?
- How do we use word embeddings?
- How do we evaluate word embeddings?

Agenda

- ~~What are word embeddings?~~
- How do we learn word embeddings?
- How do we use word embeddings?
- How do we evaluate word embeddings?

Learning word embeddings

Learning word embeddings



Training data

How can we train a model to learn the meaning of words?
Which data can we use for supervised learning?

Training data

How can we train a model to learn the meaning of words?
Which data can we use for supervised learning?

Key idea:

Use text itself as training data for the model!

A form of *self-supervision*.

Training data

How can we train a model to learn the meaning of words?
Which data can we use for supervised learning?

Key idea:

Use text itself as training data for the model!

A form of *self-supervision*.

Example: Train a neural network to predict the next word given previous words.

A neural probabilistic language model. Bengio et al. (2003), JMLR [url]

Natural language processing (almost) from scratch, Collobert et al. (2011), JMLR, [url]

Exercise: Word prediction task

`yesterday I went to the` **?**

`A new study has highlighted the positive` **?**

Which word comes next?

Common Models

- Word2Vec
- fastText
- GloVe
- Bert

Common Models

- Word2Vec
- fastText
- GloVe
- Bert

Word2Vec

The domestic **cat** is a small, typically furry carnivorous mammal

w_{-2} w_{-1} w_0 w_1 w_2 w_3 w_4 w_5

We have **target** words (*cat*) and **context** words (here: window=5).

Remember: distributional hypothesis

Word2Vec

Two different tasks (context):

- Continuous Bag-Of-Words (CBOW)
- Skipgram

Two training regimes

- Hierachical softmax
- Negative sampling

https://code.google.com/archive/p/word2vec/

Tomas Mikolov, Kai Chen, Greg Corrado, and Jeffrey Dean. Efficient Estimation of Word Representations in Vector Space. In Proceedings of Workshop at ICLR, 2013 [url]

Tomas Mikolov, Ilya Sutskever, Kai Chen, Greg Corrado, and Jeffrey Dean. Distributed Representations of Words and Phrases and their Compositionality. In Proceedings of NIPS, 2013 [url]

Word2Vec

Two different tasks (context):

- Continuous Bag-Of-Words (CBOW)
- Skipgram

Two training regimes

- Hierachical softmax
- Negative sampling

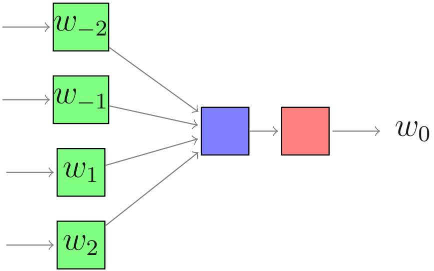
https://code.google.com/archive/p/word2vec/

Tomas Mikolov, Kai Chen, Greg Corrado, and Jeffrey Dean. Efficient Estimation of Word Representations in Vector Space. In Proceedings of Workshop at ICLR, 2013 [url]

Tomas Mikolov, Ilya Sutskever, Kai Chen, Greg Corrado, and Jeffrey Dean. Distributed Representations of Words and Phrases and their Compositionality. In Proceedings of NIPS, 2013 [url]

Word2Vec

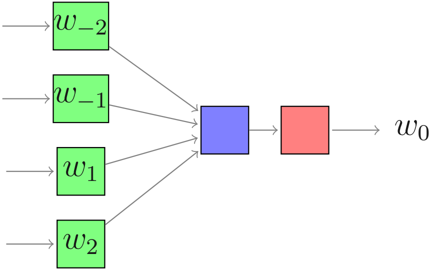
Continuous Bag-Of-Words (CBOW)



`one snowy ? she went`

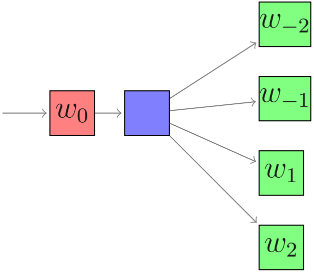
Word2Vec

Continuous Bag-Of-Words (CBOW)



`one snowy` **?** `she went`

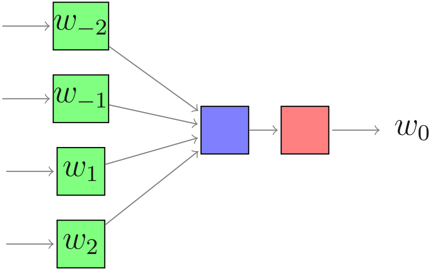
skipgram



? **?** `day` **?** **?**

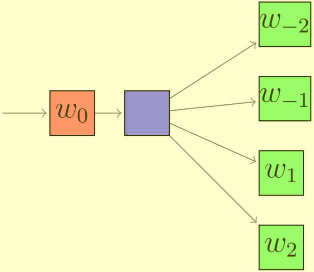
Word2Vec

Continuous Bag-Of-Words (CBOW)



`one snowy ? she went`

skipgram



`? ? day ? ?`

Word2Vec

Two different tasks (context):

- Continuous Bag-Of-Words (CBOW)
- Skipgram

Two training regimes

- Hierachical softmax
- Negative sampling

https://code.google.com/archive/p/word2vec/

Tomas Mikolov, Kai Chen, Greg Corrado, and Jeffrey Dean. Efficient Estimation of Word Representations in Vector Space. In Proceedings of Workshop at ICLR, 2013 [url]

Tomas Mikolov, Ilya Sutskever, Kai Chen, Greg Corrado, and Jeffrey Dean. Distributed Representations of Words and Phrases and their Compositionality. In Proceedings of NIPS, 2013 [url]

Word2Vec: skipgram overview

The domestic **cat** is a small, typically furry carnivorous mammal

word (w)	**context (c)**	**label**
cat	small	1
cat	furry	1
cat	car	0
...	...	...

Word2Vec: skipgram overview

The domestic **cat** is a small, typically furry carnivorous mammal

word (w)	**context (c)**	**label**
cat	small	1
cat	furry	1
cat	car	0
...	...	...

1. **Create examples**
   - Positive examples: Target word and neighboring context
   - Negative examples: Target word and randomly sampled words from the lexicon (*negative sampling*)
2. Train a **logistic regression** model to distinguish between the positive and negative examples
3. The resulting **weights** are the embeddings!

Word2Vec: skipgram overview

The domestic **cat** is a small, typically furry carnivorous mammal

word (w)	context (c)	label
cat	small	1
cat	furry	1
cat	car	0
...	...	...

Embedding vectors are essentially a byproduct!

1. **Create examples**
    - Positive examples: Target word and neighboring context
    - Negative examples: Target word and randomly sampled words from the lexicon (*negative sampling*)
2. Train a **logistic regression** model to distinguish between the positive and negative examples
3. The resulting **weights** are the embeddings!

Word2Vec: skipgram

The domestic **cat** is a small, typically furry carnivorous mammal

$c1 \quad c2 \quad w \quad c3 \; c4 \quad c5 \quad c6 \quad c7$

We have **target** words (*cat*) and **context** words (here: window=5).

The probability that *c* is a real context word:

$$P(+|w, c)$$

The probability that *c* is not a real context word:

$$P(-|w, c)$$

Word2Vec: skipgram

Intuition: A word c is likely to occur near the target if its embedding is similar to the target embedding.

$$\text{similarity} \approx \mathbf{w} \cdot \mathbf{c}$$

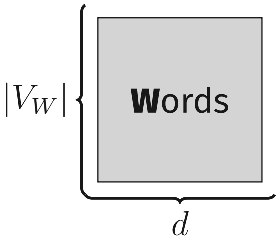
Turn this into a probability using the sigmoid function

$$P(+|w,c) = \frac{1}{1 + e^{-\mathbf{w} \cdot \mathbf{c}}}$$

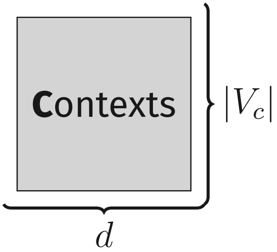
See also: 6.8 of Speech and Language Processing (3rd ed. draft), Dan Jurafsky and James H. Martin
https://web.stanford.edu/~jurafsky/slp3/

Word2Vec

Words:
Each word w is represented as a d-dimensional vector.



Contexts:
Each word c is represented as a d-dimensional vector.



All vectors are initialized with random weights.

Word2vec: skipgram (learning) 1

We **start** with random embedding vectors.

Word2vec: skipgram (learning) 2

We **start** with random embedding vectors.

During training:

- *Maximize* the similarity between the embeddings of the target word and context words from the positive examples
- *Minimize* the similarity between the embeddings of the target word and context words from the negative examples

Word2vec: skipgram (learning) 3

We **start** with random embedding vectors.

During training:

- *Maximize* the similarity between the embeddings of the target word and context words from the positive examples
- *Minimize* the similarity between the embeddings of the target word and context words from the negative examples

After training:

- frequent word-context pairs in data: $\mathbf{w} \cdot \mathbf{c}$ high
- not word-context pairs in data: $\mathbf{w} \cdot \mathbf{c}$ low

Exercise (5 min)

- Go to `https://projector.tensorflow.org/`. The site should load 'Word2Vec 10K' vectors by default (see left panel).
- What are the 5 nearest words to '*cat*'?
- What are the 5 nearest words to '*computer*'?

Limitations

- What are limitations of Word2vec?

Limitations

- What are limitations of Word2vec?
- What is the embedding vector for “wampos”?

fastText

Limitation of word2vec: Can't handle unknown words :(

fastText is very similar to word2vec, but each word is **represented as a bag of character n-grams** (+ the word itself). $\leq$ and $\geq$ mark word boundaries.

Example: *where* with $n = 3$: <wh, whe, her, ere, re> and <where>

Representation of a word: The sum of the vector representations of its n-grams.

Enriching Word Vectors with Subword Information, Bojanowski et al., TACL 2017, [url], software: https://fasttext.cc/

GloVe

- First create a *global word-word co-occurrence matrix* (how frequent pairs of words occur with each other). Requires a pass through the entire corpus at the start!
- Training objective: learn word embeddings so that their dot products equals the log of the words' co-occurrence probability.

GloVe: Global Vectors for Word Representation, Pennington et al., EMNLP 2015 [url], software
`https://nlp.stanford.edu/projects/glove/`

Pre-trained embeddings

- I want to build a system to solve a task (e.g. sentiment analysis)
  - Use pre-trained embedddings. Should I fine-tune?
    - Lots of data: yes
    - Just a small dataset: no
- Analysis (e.g. bias, semantic change)
  - Train embeddings from scratch

Agenda

- ~~What are word embeddings?~~
- ~~How do we learn word embeddings?~~
- How do we use word embeddings?
- How do we evaluate word embeddings?

Agenda

- ~~What are word embeddings?~~
- ~~How do we learn word embeddings?~~
- How do we use word embeddings?
- How do we evaluate word embeddings?

Using word embeddings

Downstream Tasks

Word2vec, GloVe, fastText

cat =	**0.12**	. . .	**-0.2**
dog =	**0.92**	. . .	**-0.1**
tree =	**-0.12**	. . .	**0.1**
. . .	. . .	. . .	. . .

cat =	**0.12**	. . .	**-0.2**
dog =	**0.92**	. . .	**-0.1**
tree =	**-0.12**	. . .	**0.1**
. . .	. . .	. . .	. . .

ML model

Topic?
Summary?
Translation?
Etc...

Downstream Task Performance

Model	Dev	Test	ACE	MUC7
Discrete	91.0	85.4	77.4	73.4
SVD	90.8	85.7	77.3	73.7
SVD-S	91.0	85.5	77.6	74.3
SVD-L	90.5	84.8	73.6	71.5
HPCA	92.6	**88.7**	81.7	80.7
HSMN	90.5	85.7	78.7	74.7
CW	92.2	87.4	81.7	80.2
CBOW	93.1	88.2	82.2	81.1
GloVe	**93.2**	88.3	**82.9**	**82.2**

Figure: *GloVe: Global Vectors for Word Representation*, J. Pennington, R. Socher and C.D. Manning (2014)

Properties of word embeddings

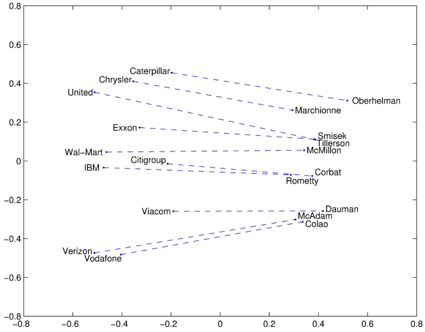


Figure: company - ceo

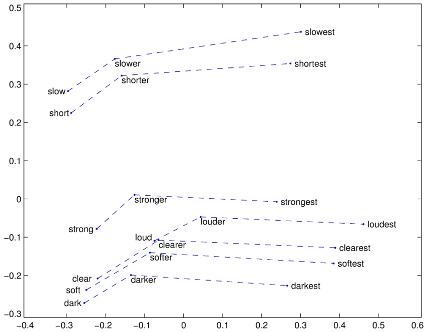


Figure: comparative - superlative

Source: https://nlp.stanford.edu/projects/glove/

Utrecht University

Applications: Semantic change

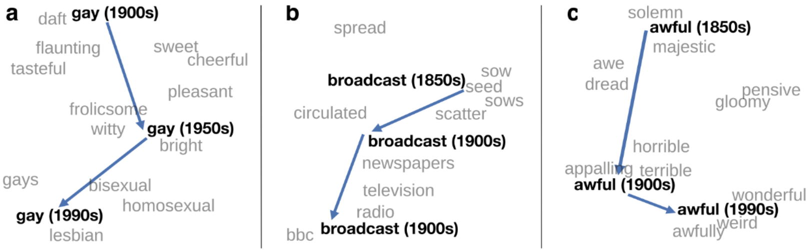
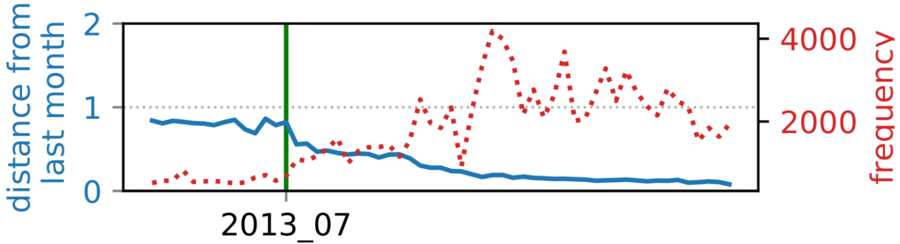


Figure 1. from Diachronic Word Embeddings Reveal Statistical Laws of Semantic Change, Hamilton et al., ACL 2016 [url]

Semantic change: *glo*

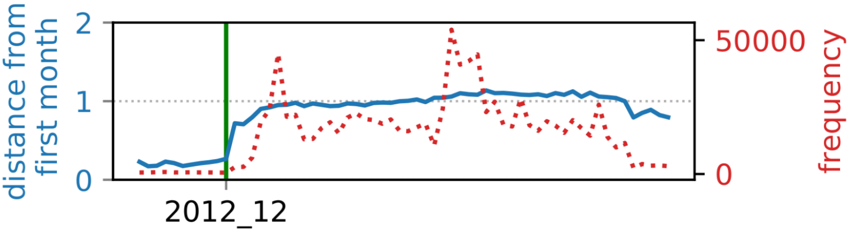
August 2013 Chief Keef “Gotta Glo Up One Day”



P. Shoemark*, F. F. Liza*, D. Nguyen, S. A. Hale, B. McGillivray. Room to glo: A systematic comparison of semantic change detection approaches with word embeddings, EMNLP 2019 [url]

Semantic change: *vine*

Launched January 2013



P. Shoemark*, F. F. Liza*, D. Nguyen, S. A. Hale, B. McGillivray. Room to glo: A systematic comparison of semantic change detection approaches with word embeddings, EMNLP 2019 [url]

Agenda

- ~~What are word embeddings?~~
- ~~How do we learn word embeddings?~~
- ~~How do we use word embeddings?~~
- How do we evaluate word embeddings?

Agenda

- ~~What are word embeddings?~~
- ~~How do we learn word embeddings?~~
- ~~How do we use word embeddings?~~
- How do we evaluate word embeddings?

Evaluation

Evaluation 1

Evaluation 2

Types of evaluation

1. Extrinstic evaluation
2. Intrinsic evaluation

Evaluation 3

Types of evaluation

1. Extrinstic evaluation
2. Intrinsic evaluation

0.12	...	-0.2

Intrinsic evaluation

- Similarity
- Analogies
- Clustering
- Coherence
- Probing classifiers

Similarity

- Similarity
- Analogies
- Clustering
- Coherence
- Probing classifiers

Input: Dataset with relatedness or similarity scores for pairs of words.

Goal: High (pearson or spearman) correlation between scores and the cosine similarity of the embeddings for the two words.

Example from *WordSim353*:
wood and *forest*: 7.73
money and *cash*: 9.15
month and *hotel*: 1.81

Analogies

- Similarity
- Analogies
- Clustering
- Coherence
- Probing classifiers

```
Base/3rd Person Singular Present
see:sees return: ?

Singular/Plural
year:years law: ?

Meronyms
player:team fish: ?

UK city county
york:yorkshire Exeter: ?
```

(Mikolov et al. 2013 [url]; Gladkova et al. 2016 [url])

Analogies: 3COSADD

- Similarity
- Analogies
- Clustering
- Coherence
- Probing classifiers

This method is referred to by Levy and Goldberg (2014) as **3COSADD**
$\mathbf{a} - \mathbf{a}^* \approx \mathbf{b} - \mathbf{b}^*$. We can find $\mathbf{b}^*$ as follows:

$$\underset{\mathbf{b}^* \in V}{\text{argmax}}\, cos(\mathbf{b}^*, \mathbf{b} - \mathbf{a} + \mathbf{a}^*)$$

Clustering

- Similarity
- Analogies
- Clustering
- Coherence
- Probing classifiers

Cluster the words based on their embeddings and compare them against a known categorization.

Evaluation methods for unsupervised word embeddings, Schnabel et al. EMNLP 2015 [url]

Coherence

- Similarity
- Analogies
- Clustering
- Coherence
- Probing classifiers

Are words in the neighborhood of the *query* word mutually related? Present four words (query word + two close neighbors + intruder). Task: identify the intruder (e.g. Turkers).

Example: (a) *finally*; (b) *eventually*; (c) *immediately*; (d) *put*

Which word is the intruder?

Evaluation methods for unsupervised word embeddings, Schnabel et al. EMNLP 2015 [url]

Coherence: Intruder

- Similarity
- Analogies
- Clustering
- Coherence
- Probing classifiers

Are words in the neighborhood of the *query* word mutually related? Present four words (query word + two close neighbors + intruder). Task: identify the intruder (e.g. Turkers).

Example: (a) *finally*; (b) *eventually*; (c) *immediately*; (d) *put*

Which word is the intruder?

Evaluation methods for unsupervised word embeddings, Schnabel et al. EMNLP 2015 [url]

Probing classifiers

- Similarity
- Analogies
- Clustering
- Coherence
- Probing classifiers

Also called *diagnostic classifiers*

Mostly used to evaluate sentence embeddings, but sometimes also used for analyzing word embeddings.

But, be careful! Performance might seem high, but classifier might learn other signals (e.g. word frequency, part of speech classes) than what you focus on.

What you can cram into a single $&!# vector: Probing sentence embeddings for linguistic properties, Conneau et al., ACL 2018 [url]

Resources

Resources

Readings:

- *Contextual Word Representations: Putting Words into Computers*, Noah A. Smith, 2020 https://cacm.acm.org/magazines/2020/6/245162-contextual-word-representations/fulltext
- *Vector Semantics and Embeddings (Chapter 6)*, Speech and Language Processing (3rd ed. draft), Dan Jurafsky and James H. Martin, 2020 https://web.stanford.edu/~jurafsky/slp3/
- *SimLex-999: Evaluating Semantic Models with (Genuine) Similarity Estimation*, Felix Hill, Roi Reichart, and Anna Korhonen, 2014 https://arxiv.org/abs/1408.3456v1

Videos:

- *Stanford CS224N: NLP with Deep Learning | Winter 2019 | Lecture 1 – Introduction and Word Vectors* (and lecture 2): https://www.youtube.com/watch?v=8rXD5-xhemo
- video's by Jordan Boyd-Graber, e.g. *Understanding Word2Vec* https://www.youtube.com/watch?v=QyrUentbkvw and others

Resources: blogposts

- *The Illustrated BERT, ELMo, and co. (How NLP Cracked Transfer Learning)* by Jay Alammar
  `http://jalammar.github.io/illustrated-bert/` (2018)
- *The Illustrated Word2vec* by Jay Alammar
  `http://jalammar.github.io/illustrated-word2vec/` (2019)
- *Generalized Language Models* by Lilian Weng
  `https://lilianweng.github.io/lil-log/2019/01/31/generalized-language-models.html`

Software

- **word2vec**: gensim (https://radimrehurek.com/gensim/) and official implementation (https://code.google.com/archive/p/word2vec/).
- **fasttext**: official implementation (https://fasttext.cc/)
- **GloVe**: official implementation (https://nlp.stanford.edu/projects/glove/)
- **Hugging Face**: for BERT and other transformer models (https://huggingface.co/) [for Python, not R]

Addendums

Contextual word embeddings

Tokens versus types

The hut is located near the bank of the river

Tokens	**Types**
The	the
hut	hut
is	is
located	located
near	near
the	bank
bank	of
of	river
the	
river	

Contextualized word representations 1

So far: an embedding for **each word (type)**.

*Today, I went to the **bank** to deposit a check.*

*The hut is located near the **bank** of the river.*

bank	0.52	0.48	-0.01	⋯	0.28
bank	-0.27	0.28	-0.07	⋯	0.82

Contextualized word representations 2

So far: an embedding for **each word (type)**.

*Today, I went to the **bank** to deposit a check.*

bank	0.52	0.48	-0.01	$\cdots$	0.28

*The hut is located near the **bank** of the river.*

bank	-0.27	0.28	-0.07	$\cdots$	0.82

Key idea in NLP:

Can we have an embedding for each **word token?**

Contextualized word representations

Key idea: Have embeddings for each **word token**

Previously:

- One embedding for each word **type**
- A table where each word is mapped to a vector.

Now:

- One embedding for each work **token**
- Embeddings for a token are created based on the context
- There is *no single* embedding for a word anymore.

BERT

Two tasks:

- Masked LM
- Next sentence prediction

BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding, Devlin et al. NAACL 2019 [url]

BERT

Two tasks:

- Masked LM
- Next sentence prediction

BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding, Devlin et al. NAACL 2019 [url]

`my dog is hairy`

- mask word:
  `my dog is [MASK]`

(some details are omitted.)

BERT

Two tasks:

- Masked LM
- Next sentence prediction

BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding, Devlin et al. NAACL 2019 [url]

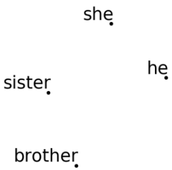
Input = `[CLS] the man went to [MASK] store [SEP] he bought a gallon [MASK] milk [SEP]`
Label = IsNext

Input = `[CLS] the man [MASK] to the store [SEP] penguin [MASK] are flight ## less birds [SEP]`
Label=NotNext

Utrecht University

Biases in word embeddings

Biases in word embeddings



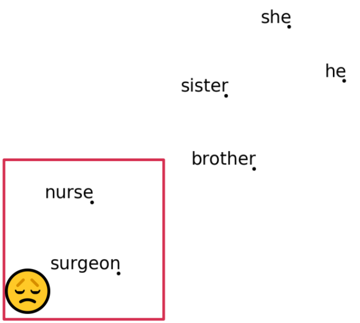
Measuring gender bias:

- To assess NLP models and investigate the impact of ‘bias mitigation’ techniques
- To study societal trends

Man is to Computer Programmer as Woman is to Homemaker? Debiasing Word Embeddings, Bolukbasi, et al. NIPS 2016 [url]

Semantics derived automatically from language corpora contain human-like biases, Caliskan, Bryson, Narayanan, Science 2017 [url]

Biases in word embeddings



Pre-trained GloVe model on Twitter

Measuring gender bias:

- To assess NLP models and investigate the impact of ‘bias mitigation’ techniques
- To study societal trends

Man is to Computer Programmer as Woman is to Homemaker? Debiasing Word Embeddings, Bolukbasi, et al. NIPS 2016 [url]

Semantics derived automatically from language corpora contain human-like biases, Caliskan, Bryson, Narayanan, Science 2017 [url]

Biases reflected in analogy tasks

Biases reflected in analogy tasks:

man is to *computer programmer* as *woman* is to ? : x = homemaker
father is to *doctor* as *mother* is to ? : x = nurse

Note: Input words are excluded as possible answers! (see also Nissim et al. 2020 [url])

Compare: gender-specific words (e.g., *brother, businesswoman*) vs. *gender-neutral* words (e.g. *nurse*, *teacher*).

Man is to Computer Programmer as Woman is to Homemaker? Debiasing Word Embeddings, Bolukbasi, et al. NIPS 2016 [url]

Biases in word embeddings

$$S_{(a,b)}(\mathrm{x,y}) = cos(\underbrace{\mathrm{a}}_{\text{embedding}_{she}} - \underbrace{\mathrm{b}}_{\text{embedding}_{he}}, x-y) \quad if \quad \underbrace{\|x-y\|_2}_{L_2 \text{ distance}} \leq \delta$$

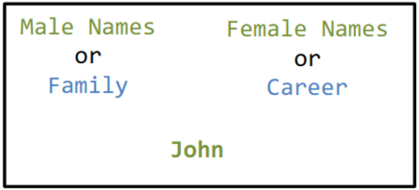
Gender appropriate she-he analogies
queen–king
sister–brother
ovarian cancer–prostate cancer
mother–father
convent–monastery

Gender stereotype she-he analogies
nurse–surgeon
sassy–snappy
cupcakes–pizzas
lovely–brilliant
vocalist–guitarist

Bolukbasi et al. look at 300-dimensional embeddings from w2vec Google news corpus.

Word-Embedding Association Test

- The Implicit Association Test (IAT) is based on response times and has been widely used.



Semantics derived automatically from language corpora contain human-like biases, Caliskan, Bryson, Narayanan, Science 2017 [url]

Word-Embedding Association Test

- The Implicit Association Test (IAT) is based on response times and has been widely used.
- Word-Embedding Association Test (WEAT) by Caliskan et al: use the cosine similarity between pairs of vectors as analogous to reaction time in the IAT

Were able to replicate well-known IAT findings!

Semantics derived automatically from language corpora contain human-like biases, Caliskan, Bryson, Narayanan, Science 2017 [url]

Perpetuation of bias in sentiment analysis

> *"I had tried building an algorithm for sentiment analysis based on word embeddings [..]. When I applied it to restaurant reviews, I found it was ranking Mexican restaurants lower. The reason was not reflected in the star ratings or actual text of the reviews. It's not that people don't like Mexican food.* ***The reason was that the system had learned the word "Mexican" from reading the Web****."*

(emphasis by Dr. Dong Nguyen)

http://blog.conceptnet.io/posts/2017/conceptnet-numberbatch-17-04-better-less-stereotyped-word-vectors/

Word Analogies: Warning

Word analogies

We can look at analogies in the vector space, for example:
king - man + woman ≈ queen

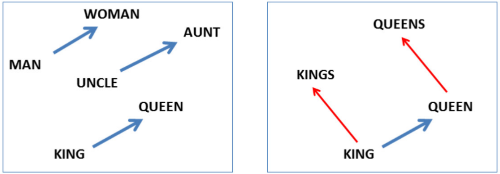
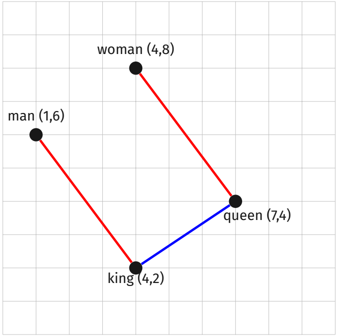


Figure: Figure 2 from Linguistic Regularities in Continuous Space Word Representations, Mikolov et al. NAACL 2013 [url]

Word analogies: math

We can look at analogies in the vector space, for example:
king - man + woman ≈ *queen*




```
king-man = [4,2] - [1,6] = [3,-4]
king-man + woman = [3,-4] + [4,8] = [7,4]
```

Word analogies: warning

https://blog.esciencecenter.nl/king-man-woman-king-9a7fd2935a85
These analogies only work with cheating!