Quantitative Text Analysis

University College Dublin

Instructor: Yen-Chieh Liao & Stefan Müller

Week 11 (15 April 2024)

Outline

From Onehot and BoW to Word Embedding

Bag of Words (BoW)

```
## Document-feature matrix of: 3 documents, 11 features (36.36% sparse) and 0 docvars.
##     features
## docs shipment of gold damaged in a fire delivery silver arrived
##   d1        1  1    1       1  1 1    1        0      0       0
##   d2        0  1    0       0  1 1    0        1      2       1
##   d3        1  1    1       0  1 1    0        0      0       1
## [ reached max_nfeat ... 1 more feature ]
```

One-Hot Encoding to Word Embeddings

1-of-N Encoding

apple = [1 0 0 0 0]

bag = [0 1 0 0 0]

cat = [0 0 1 0 0]

dog = [0 0 0 1 0]

elephant = [0 0 0 0 1]

Word Embedding

Word Class

1-of-N Encoding

apple = [1 0 0 0 0]

bag = [0 1 0 0 0]

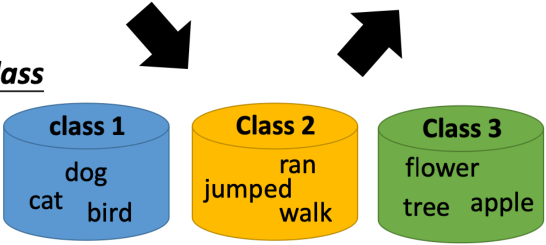
cat = [0 0 1 0 0]

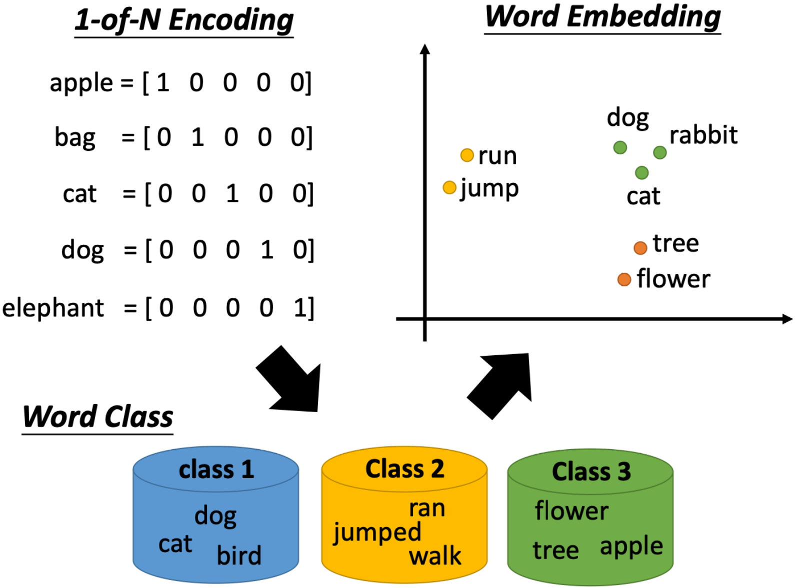
dog = [0 0 0 1 0]

elephant = [0 0 0 0 1]

Word Embedding

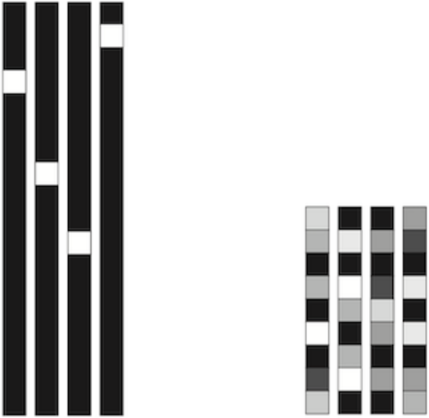
Word Class





Source: Hung-yi Lee

One-Hot Encoding vs. Vector Representation



Source: François Chollet (2017)

Applications in Natural Language Processing

- Machine Translation
- Text Classification: NER (sequence labeling), Part-of-Speech Tagging, Sentiment Analysis.
- Information Retrieval: Keyword Extraction, Document Summarization, Search Relevance Tuning.

Word Embeddings for Dictionary Expansion

No Longer Conforming to Stereotypes? Gender, Political Style and Parliamentary Debate in the UK

Lotte Hargrave* and Jack Blumenau

University College London, London, UK
*Corresponding author. Email: lotte.hargrave.16@ucl.ac.uk

(Received 24 September 2020; revised 18 June 2021; accepted 28 October 2021)

Abstract
Research on political style suggests that where women make arguments that are more emotional, empathetic and positive, men use language that is more analytical, aggressive and complex. However, existing work does not consider how gendered patterns of style vary over time. Focusing on the UK, we argue that pressures for female politicians to conform to stereotypically 'feminine' styles have diminished in recent years. To test this argument, we describe novel quantitative text-analysis approaches for measuring a diverse set of styles at scale in political speech data. Analysing UK parliamentary debates between 1997 and 2019, we show that the debating styles of female MPs have changed substantially over time, as women in Parliament have increasingly adopted stylistic traits that are typically associated with 'masculine' stereotypes of communication. Our findings imply that prominent gender-based stereotypes of politicians' behaviour are significantly worse descriptors of empirical reality now than they were in the past.

Word Embeddings for Dictionary Expansion

Negative Emotion			Aggression		
Top	*Added*	*Removed*	*Top*	*Added*	*Removed*
upset	upset	painstaking	disgraceful	utterly	inferior
suffering	terrible	painting	shameful	cynical	offenders
terrible	hurt	alarms	outrageous	frankly	assaulted
distressing	deeply	paint	scaremongering	embarrassing	annoyance
hurt	unfortunate	paints	utterly	incompetence	fiddle
distress	angry	terrific	cynical	misguided	fiddled
frightening	felt	disappointingly	frankly	irresponsible	steal
unhappy	feeling	terrorists	scandalous	pathetic	assault
worry	caused	avoidance	dishonest	dreadful	offend
deeply	horrendous	cowardly	embarrassing	bizarre	furious
dreadful	appalling	grievance	absurd	complacency	fail
unfortunate	shocked	hopelessly	ridiculous	illogical	deceived
worried	frustrating	lone	ludicrous	incompetent	predictable

- Source: Supporting Information of Hargrave and Blumenau (2022)
- Blumenau, Jack E; Hargrave, Lotte, 2022, *Replication Data for: No Longer Conforming to Stereotypes? Gender, Political Style, and Parliamentary Debate in the UK*

Word Embeddings and Biases

Identify Biases:

- Gender, Race, or Class
- Package: `sweater` (Speedy Word Embedding Association Test & Extras using R)

The Journal of Open Source Software

sweater: Speedy Word Embedding Association Test and Extras Using R

Chung-hong Chan[1]

1 Mannheimer Zentrum für Europäische Sozialforschung, Universität Mannheim

Recommendations for Political Science Research

American Political Science Review (2023) 117, 4, 1255–1274
doi:10.1017/S0003055422001228

Embedding Regression: Models for Context-Specific Description and Inference

PEDRO L. RODRIGUEZ *New York University, United States*
ARTHUR SPIRLING *New York University, United States*
BRANDON M. STEWART *Princeton University, United States*

Social scientists commonly seek to make statements about how word use varies over circumstances—including time, partisan identity, or some other document-level covariate. For example, researchers might wish to know how Republicans and Democrats diverge in their understanding of the term "immigration." Building on the success of pretrained language models, we introduce the à la carte on text (conText) embedding regression model for this purpose. This fast and simple method produces valid vector representations of how words are used—and thus what words "mean"—in different contexts. We show that it outperforms slower, more complicated alternatives and works well even with very few documents. The model also allows for hypothesis testing and statements about statistical significance. We demonstrate that it can be used for a broad range of important tasks, including understanding US polarization, historical legislative development, and sentiment detection. We provide open-source software for fitting the model.

Word Embeddings

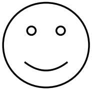
General Concepts

- Learning from extensive textual information, with an emphasis on the relationships and contextual similarities between words.

General Concepts

Joe Biden attends the inaugural speech.

Donald Trump attends the inaugural speech.



- Machine learn the meaning of words from reading a lot of documents without supervision
- A word can be understood by its context.
- Generating Word Vector is unsupervised.

How to Exploit the Context?

Count-based:

- If two words wi (dog) and wj (cat) frequently co-occur, $V(wi)$ and $V(wj)$ would be close to each other.
- Word embeddings models take an entire corpus (either an existing one, or our corpus) and encode relation between words into a multidimensional space (100 or 300).
- Utilize matrix factorization techniques on matrices that represent word co-occurrences.
- Applicaitom: Glove (Global Vectors for Word Representation) (Pennington, Socher Manning 2014)

How to Exploit the Context?

Problem for Count-based Models?

- Fixed Word Representations
- Contextual Ambiguity

How to Exploit the Context?

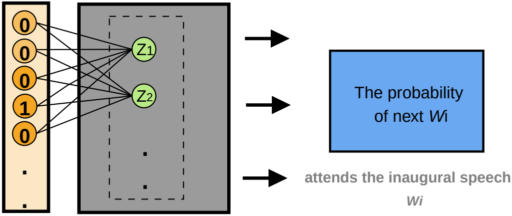
Prediction-based Models

Trump attends the inaugural speech
Wi-1 *Wi*

Biden attends the inaugural speech
Wi-1 *Wi*

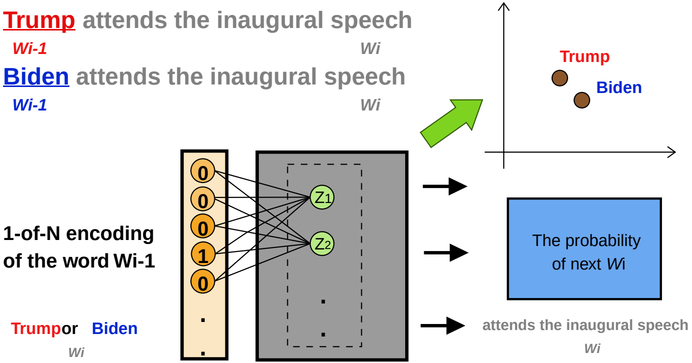
1-of-N encoding of the word Wi-1

Trump or Biden
Wi



How to Exploit the Context?

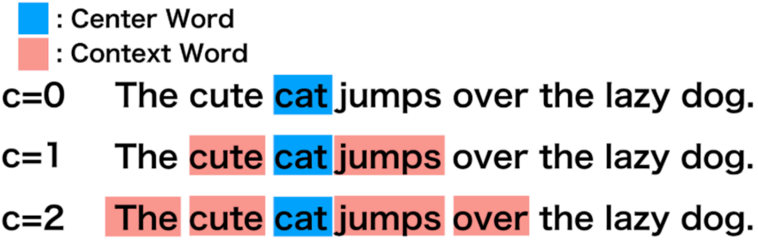
Prediction-based Models



Source: Hung-yi Lee

How to Exploit the Context?

Prediction-based Models



Source:
https://cbail.github.io/textasdata/word2vec/rmarkdown/word2vec.html

How to Exploit the Context?

Applications in Prediction-based Models

- Word2Vec: Skip-gram CBOW (Continuous Bag of Words)
- FastText, Flair embedding (contextual word embeddings)

How to Exploit the Context?

Problem in Prediction-based Model

- Biases in the Training Data?
- Out-of-Vocabulary (OOV) Words Still
- Interpretability

Word Embeddings: Application in FlaiR

Flair NLP

Variety of Embeddings

- Supports multiple language embeddings and contextual Flair embeddings.
- Easy to use, we have an flaiR package for handling same task in R.

ID	Language	Embedding
‘en-glove’ (or ‘glove’)	English	GloVe embeddings
‘en-extvec’ (or ‘extvec’)	English	Komninos embeddings
‘en-crawl’ (or ‘crawl’)	English	FastText embeddings over Web crawls
‘en-twitter’ (or ‘twitter’)	English	Twitter embeddings
‘en-turian’ (or ‘turian’)	English	Turian embeddings (small)
‘en’ (or ‘en-news’ or ‘news’)	English	FastText embeddings over wikipedia data

Import Sentence and WordEmbeddings

The Classes of Sentence and WordEmbeddings

Download the Model

```
word2vec <- WordEmbeddings('en')
print(word2vec)
```

```
## WordEmbeddings(
##   'en'
##   (embedding): Embedding(1000001, 300)
## )
```

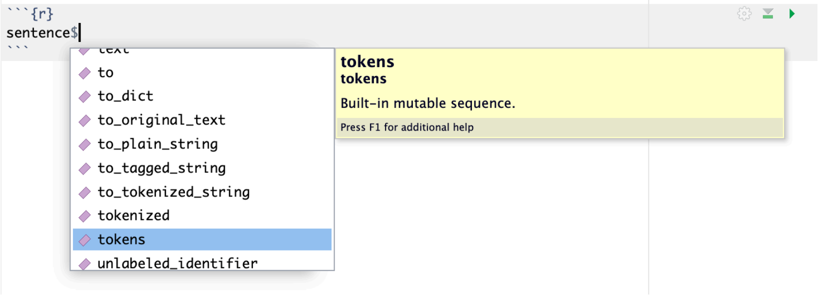
Tokenize the Text

```
string <- "king queen man woman Paris London apple orange Taiwan
sentence <- Sentence(string)
```

Embed Words in the Sentence

```r
word2vec$embed(sentence)
```

```
## [[1]]
## Sentence[11]: "king queen man woman Paris London apple orange Taiwan Dublin Bamberg"
```



Check Tokened Object

```r
sentence$tokens[1:7]
```

```
## [[1]]
## Token[0]: "king"
##
## [[2]]
## Token[1]: "queen"
##
## [[3]]
## Token[2]: "man"
##
## [[4]]
## Token[3]: "woman"
##
## [[5]]
## Token[4]: "Paris"
##
## [[6]]
## Token[5]: "London"
##
## [[7]]
## Token[6]: "apple"
```

Check the Vector of Embedding

PyTorch Tensor

```r
head(sentence$tokens[[1]]$embedding, 30)
```

```
## tensor([ 0.0445, -0.0384,  0.0011, -0.0888,  0.0713, -0.0696, -0.0477,  0.0071,
##         -0.0408, -0.0707, -0.0266,  0.0500, -0.0824,  0.0848, -0.1627, -0.0851,
##         -0.0295,  0.1534, -0.1828, -0.2208,  0.0243, -0.0921, -0.1089, -0.1009,
##         -0.0119,  0.0377,  0.2038,  0.0720,  0.0202,  0.2798])
```

Convert the PyTorch Tensor to Vector

```r
head(sentence$tokens[[1]]$embedding$numpy(), 30)
```

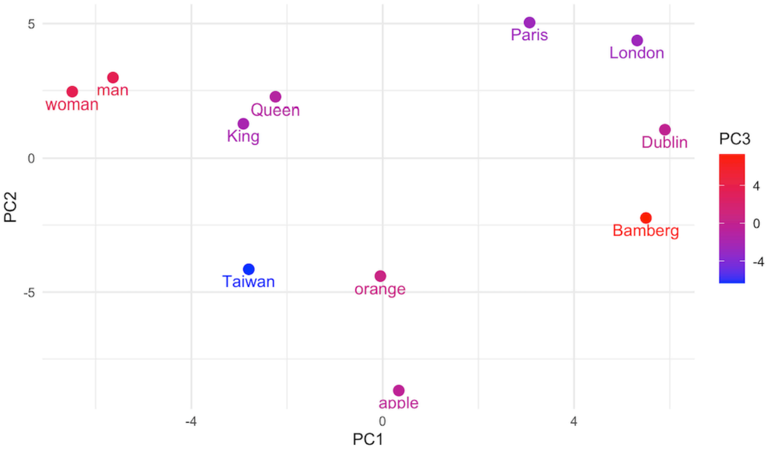
```
##  [1]  0.1082  0.0445 -0.0384  0.0011 -0.0888  0.0713 -0.0696 -0.0477  0.0071
## [10] -0.0408 -0.0707 -0.0266  0.0500 -0.0824  0.0848 -0.1627 -0.0851 -0.0295
## [19]  0.1534 -0.1828 -0.2208  0.0243 -0.0921 -0.1089 -0.1009 -0.0119  0.0377
## [28]  0.2038  0.0720  0.0202
```

Dimensional Reduction

Principal Component Analysis (PCA)

	PC1	PC2	PC3
king	-5.0552052	4.2170838	-3.2342508
queen	-5.2168270	3.4342244	-5.7479557
man	-3.7214261	3.9001685	-6.4519920
woman	-3.9774293	4.6973181	-8.7606991
Paris	5.3177064	-6.6571556	-0.9188498
London	6.8836924	-7.1938026	-2.0385105
apple	-10.0459044	0.0496133	11.5337245
orange	-7.0710917	-2.2111903	9.9470497
Taiwan	0.0183298	-5.1547824	-0.6699573
Dublin	7.6230017	-8.8329227	-0.4285303
Bamberg	15.2451533	13.7514454	6.7699713

Visualize Each Word Vector on the 2D dimension



Wrap-Up

Reflection:

- Do you think word embedding is outdated? If yes, can you share your thoughts?
- If not, what can we do with word embedding for social science?

Resource:

- Flair NLP (Python): https://flairnlp.github.io/flair/v0.13.1/
- FlaiR (R Wrapper): https://davidycliao.github.io/flaiR/

Lab Session

Lab Session: Tasks

- Read the instructions and install the flaiR documents for R user.
- Explore Flair NLP Official documents.
- Run the code and replace it with your own text.
- Answer the final question.