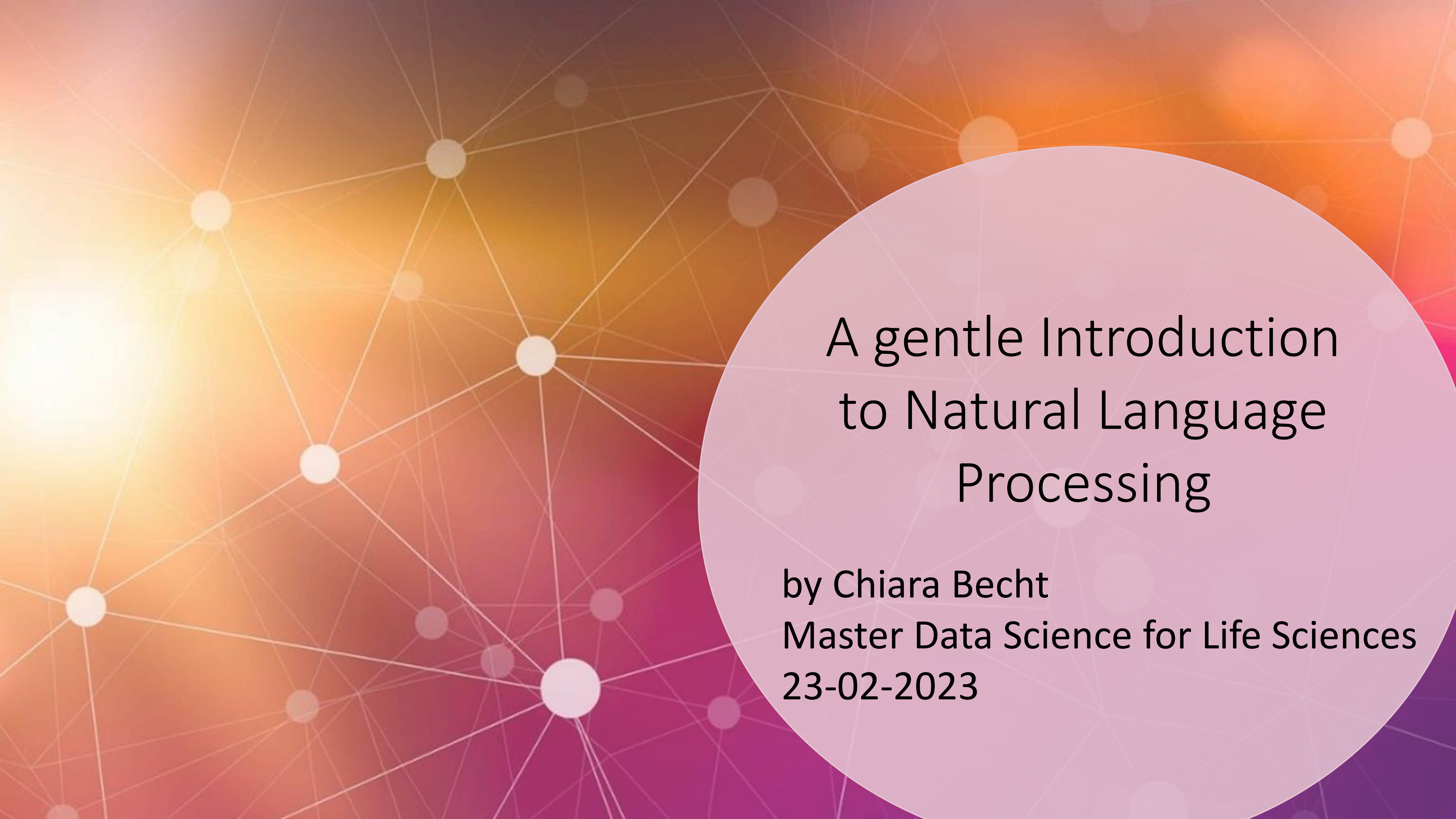


A gentle Introduction to Natural Language Processing

by Chiara Becht
Master Data Science for Life Sciences
23-02-2023

Natural Language Processing in daily business

Translation → Multiple tasks

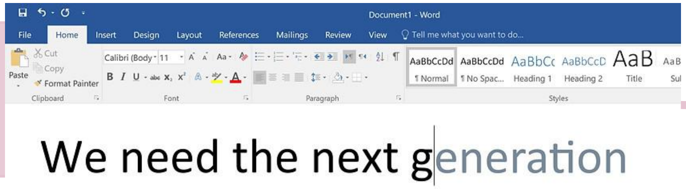
Spam filtering

[1]

→ Text classification

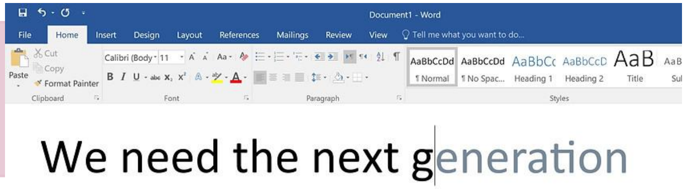
[1]

Word prediction



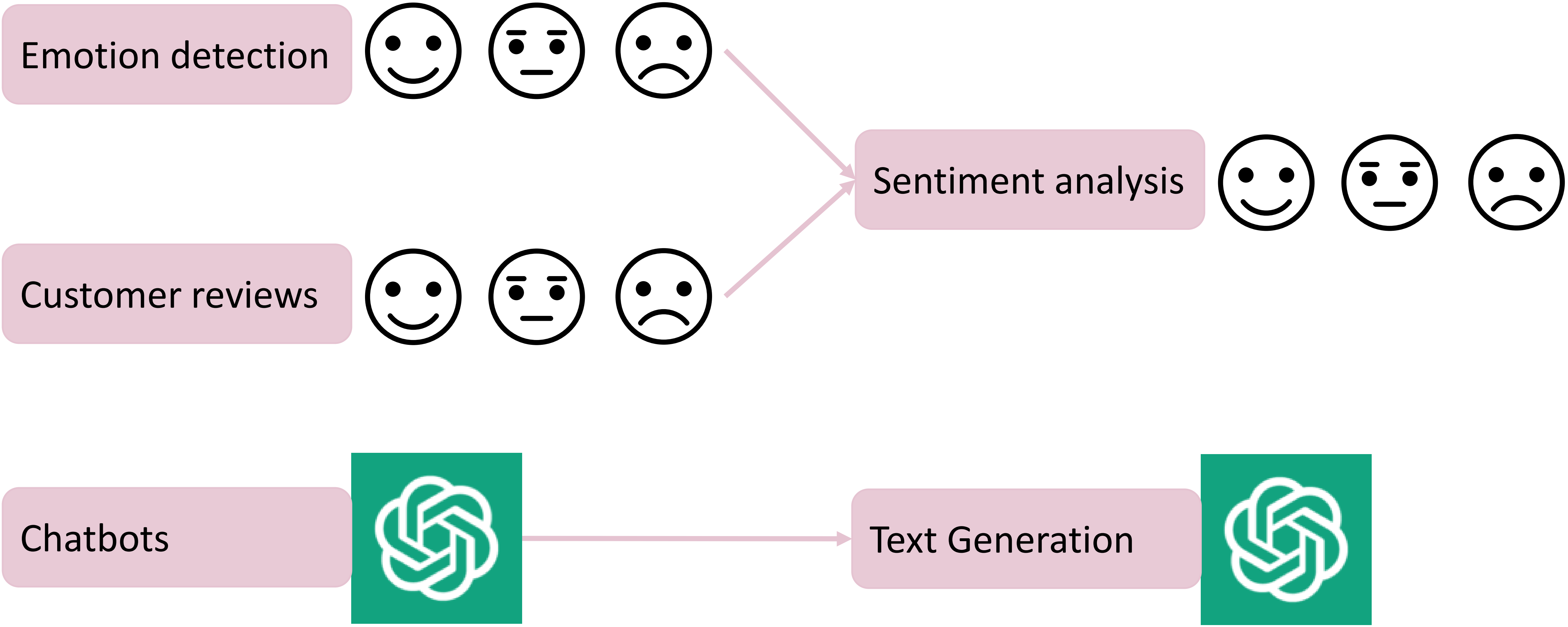
We need the next generation

→ Word prediction



We need the next generation

NLP in daily business



Emotion detection

Customer reviews

Sentiment analysis

Chatbots

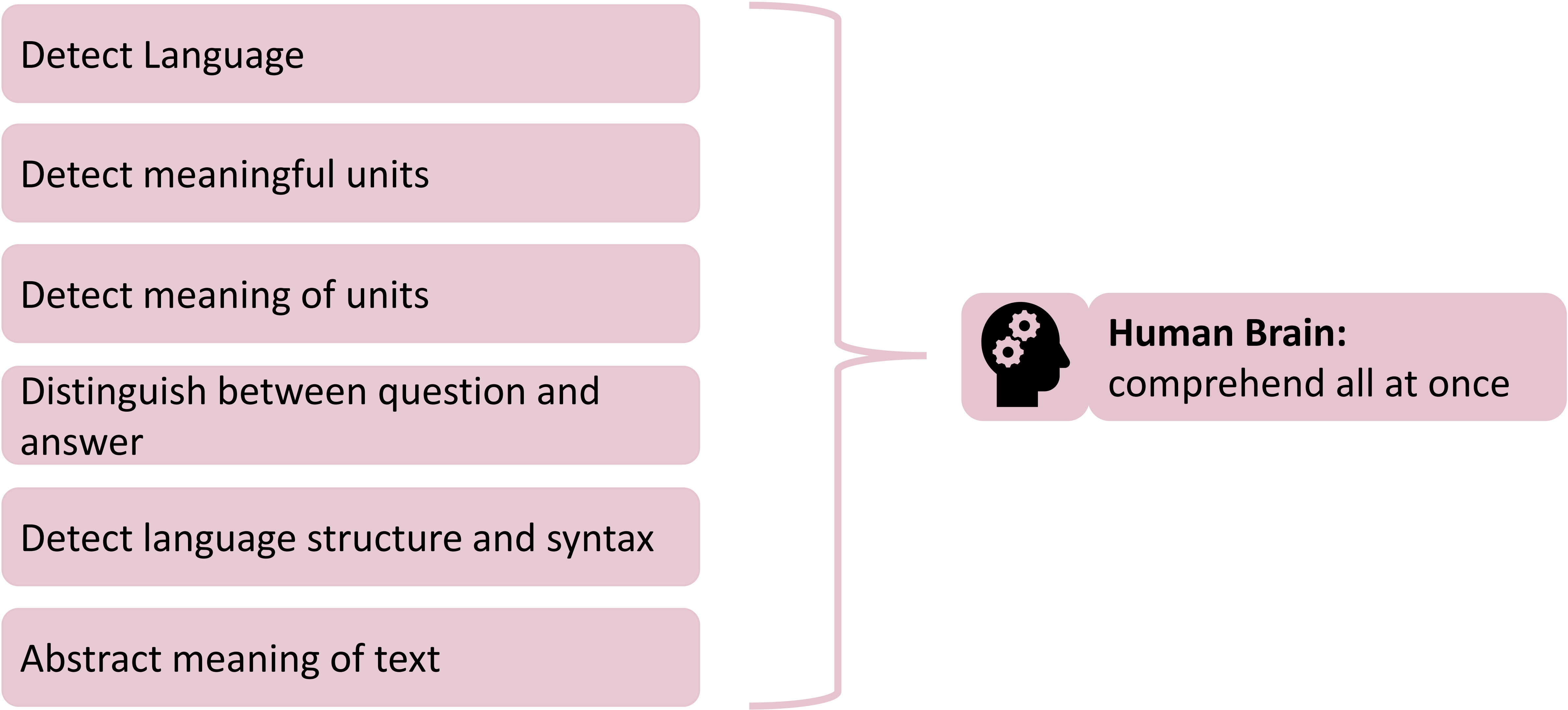
Text Generation

„Look closely.
Because the closer you look the
less you see“

- now you see me –

How does the magic of NLP comes to life?

- Text vectorization
- Powerful NLP models
- Application for my project



Detect Language

Detect meaningful units

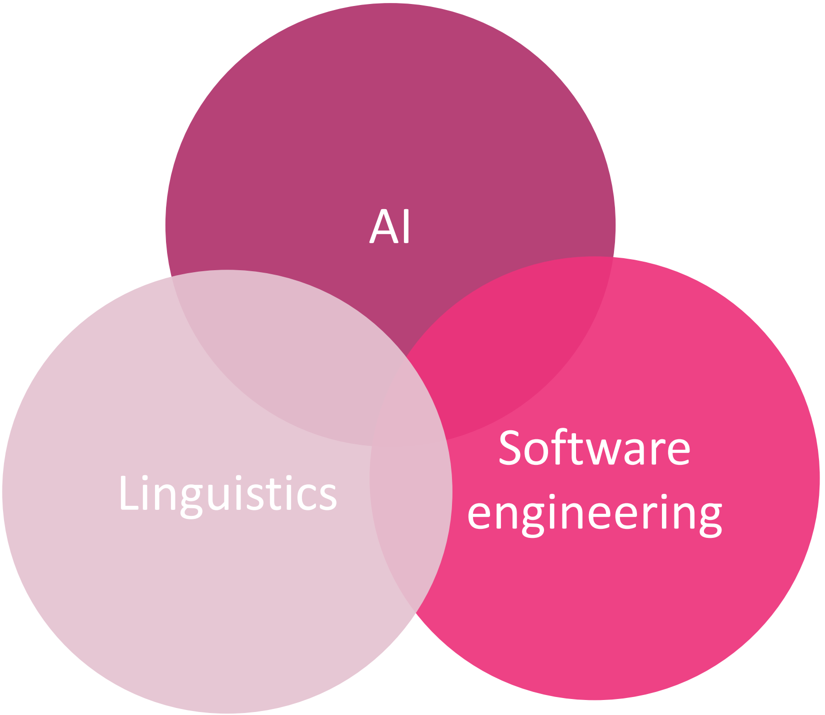
Detect meaning of units

Distinguish between question and answer

Detect language structure and syntax

Abstract meaning of text

Human Brain:
comprehend all at once



AI

Linguistics

Software engineering

Example text

What more could you ask for?

Almost all degree programmes on campus, good bus connections,

smoke-free entire campus and very friendly staff ;)

What more could you ask for?

Almost all degree programmes on campus, good bus connections,

smoke-free entire campus and very friendly staff ;)

Detect Language

Conversational English

What more could you ask for?

Almost all degree programmes on campus, good bus connections,

smoke-free entire campus and very friendly staff ;)

Detect Language

Conversational English

Tokenization

2 sentences,
24 words

What more could you ask for?

Almost all degree programmes on campus, good bus connections,

smoke-free entire campus and very friendly staff ;)

Detect Language	Tokenization	Text Vectorization
Conversational English	2 sentences, 24 words	Numerical representation: • Words • Sentences • paragraph

What more could you ask for?

Almost all degree programmes on campus, good bus connections, smoke-free entire campus and very friendly staff ;)

Detect Language

Conversational English

Tokenization

2 sentences, 24 words

Text Vectorization

Numerical representation:
- Words
- Sentences
- paragraph

Basic Model building blocks

What more could you ask for?

Almost all degree programmes on campus, good bus connections, smoke-free entire campus and very friendly staff ;)

Named Entity Recognition (NER)

What more could you ask for?

Almost all degree programmes on campus, good bus connections, smoke-free entire campus and very friendly staff ;)

Named Entity Recognition (NER)

Part of speech (POS) tagging

What more could you ask for?

Almost all degree programmes on campus, good bus connections, smoke-free entire campus and very friendly staff ;)

Named Entity Recognition (NER)

Dependency tagging

Part of speech (POS) tagging

What more could you ask for?

Almost all degree programmes on campus, good bus connections, smoke-free entire campus and very friendly staff ;)

Named Entity Recognition (NER)

Dependency tagging

Part of speech (POS) tagging

Stemming / Lemmatization

What more could you ask for?

Almost all degree programmes on campus, good bus connections, smoke-free entire campus and very friendly staff ;)

Named Entity Recognition (NER)

Dependency tagging

Part of speech (POS) tagging

Stemming / Lemmatization

Sequence Tagging Tasks

What more could you ask for?

Almost all degree programmes on campus, good bus connections, smoke-free entire campus and very friendly staff ;)

DON'T STUDY HERE!

Awful way of teaching. Horrible teachers. Do not waste your money on this hilarious institution.

NLP Downstream Tasks

What more could you ask for?
Almost all degree programmes on campus, good bus connections, smoke-free entire campus and very friendly staff ;)

DON'T STUDY HERE!
Awful way of teaching. Horrible teachers. Do not waste your money on this hilarious institution.

Text Classification

E.g. Sentiment analysis

NLP Downstream Tasks

What more could you ask for?

Almost all degree programmes on campus, good bus connections, smoke-free entire campus and very friendly staff ;)

DON'T STUDY HERE!

Awful way of teaching. Horrible teachers. Do not waste your money on this hilarious institution.

Text Classification

E.g. Sentiment analysis

Question Answering

E.g. should I study at the Hanze?

NLP Downstream Tasks

How does the magic of NLP comes to life?

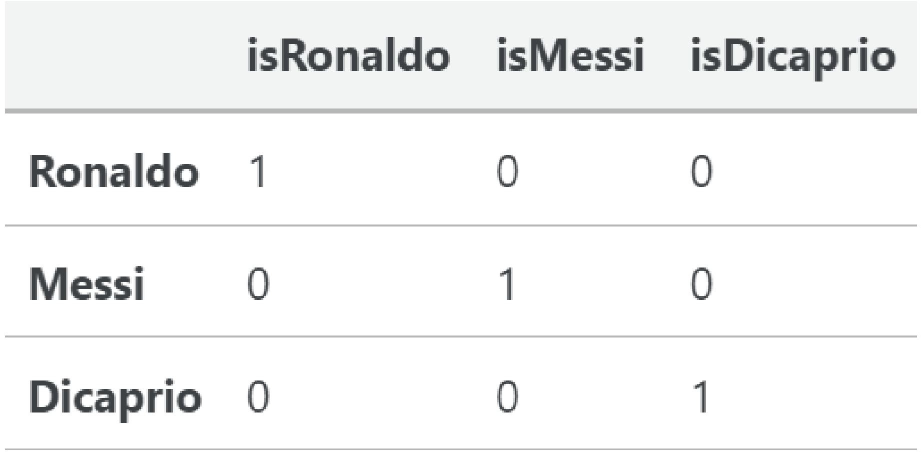
- **Text vectorization**
- Powerful NLP models
- Application for my project

Ronaldo, Messi, Dicaprio

How can we define this words numerically?

[2]

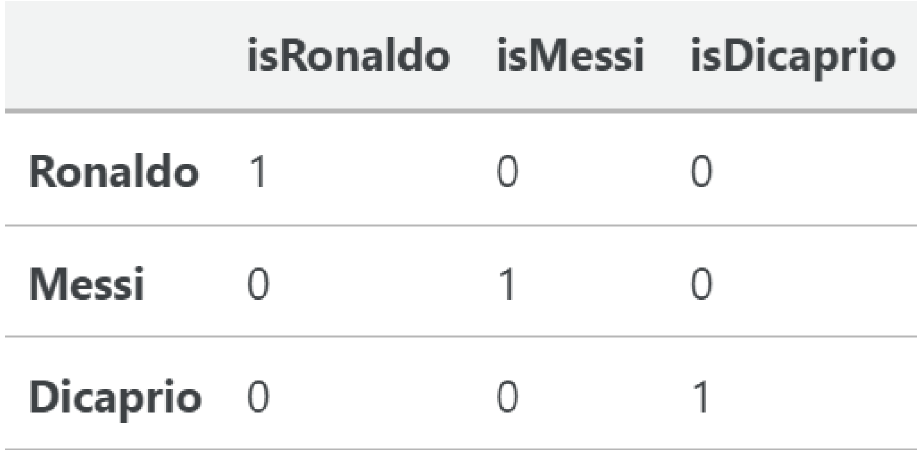
Ronaldo, Messi, Dicaprio



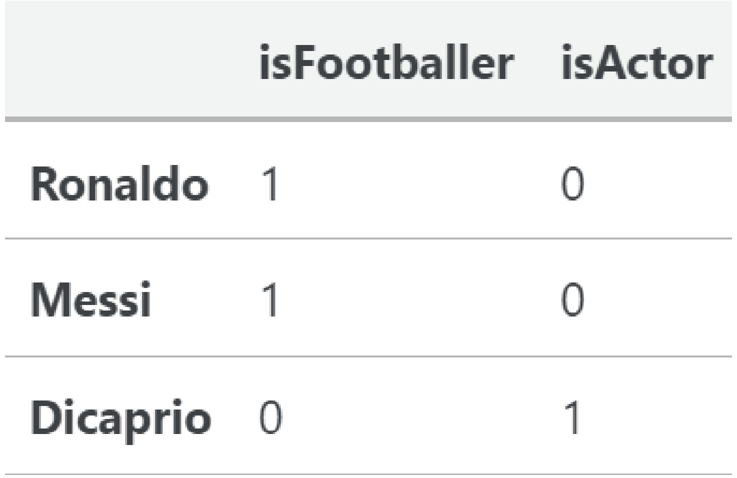
	isRonaldo	isMessi	isDicaprio
Ronaldo	1	0	0
Messi	0	1	0
Dicaprio	0	0	1

One hot encoding

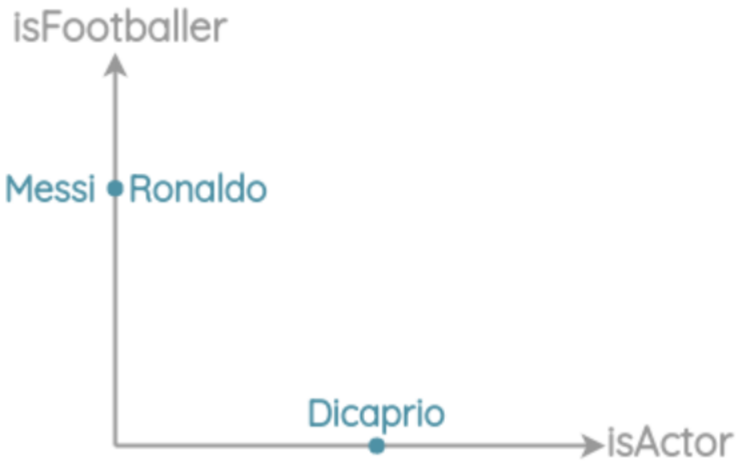
Ronaldo, Messi, Dicaprio



	isRonaldo	isMessi	isDicaprio
Ronaldo	1	0	0
Messi	0	1	0
Dicaprio	0	0	1



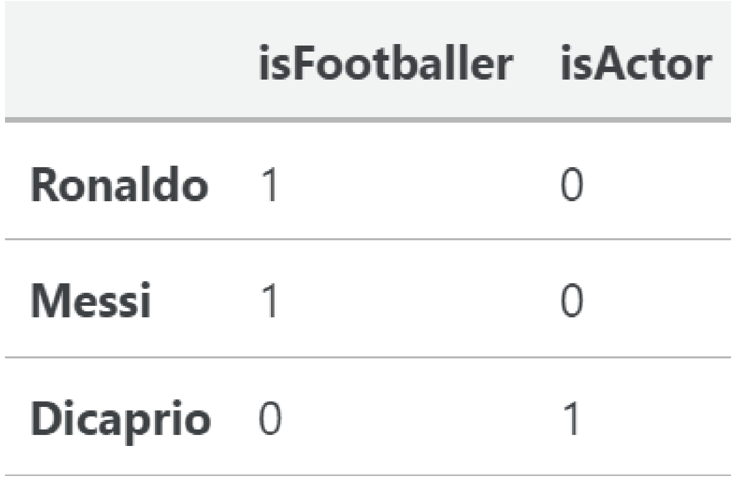
	isFootballer	isActor
Ronaldo	1	0
Messi	1	0
Dicaprio	0	1



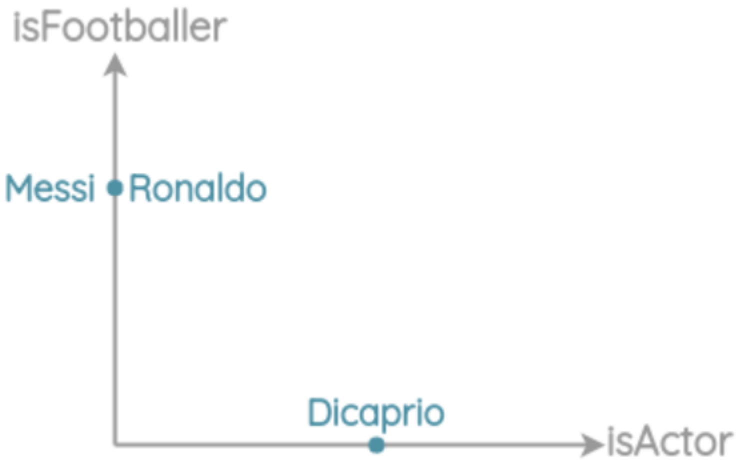
Embedding

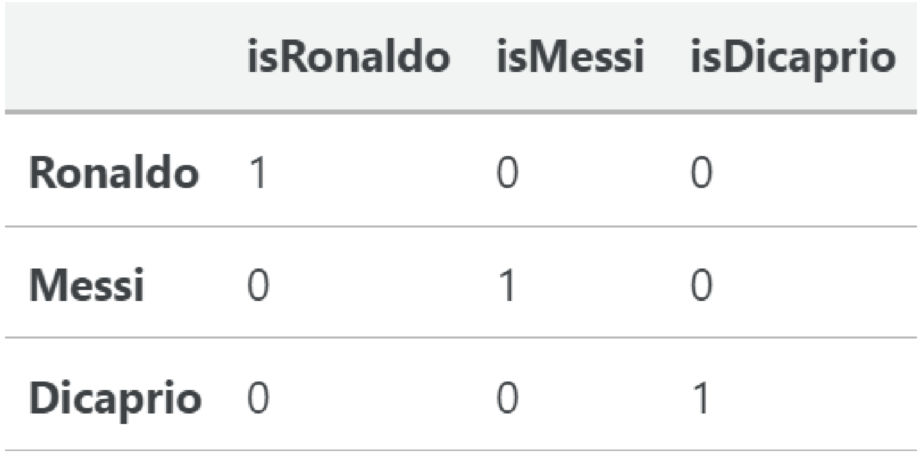
[2]

Ronaldo, Messi, Dicaprio

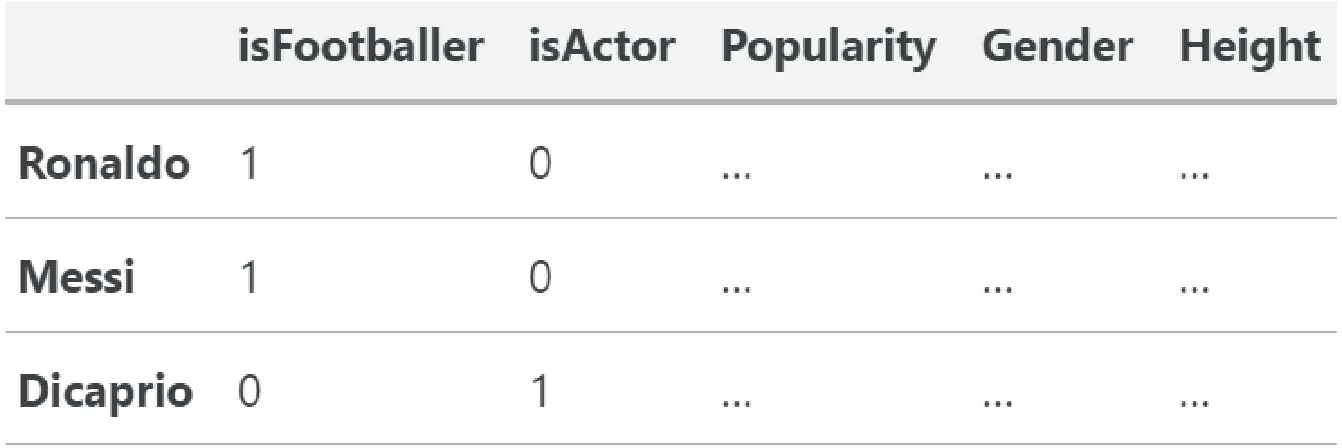


	isFootballer	isActor
Ronaldo	1	0
Messi	1	0
Dicaprio	0	1





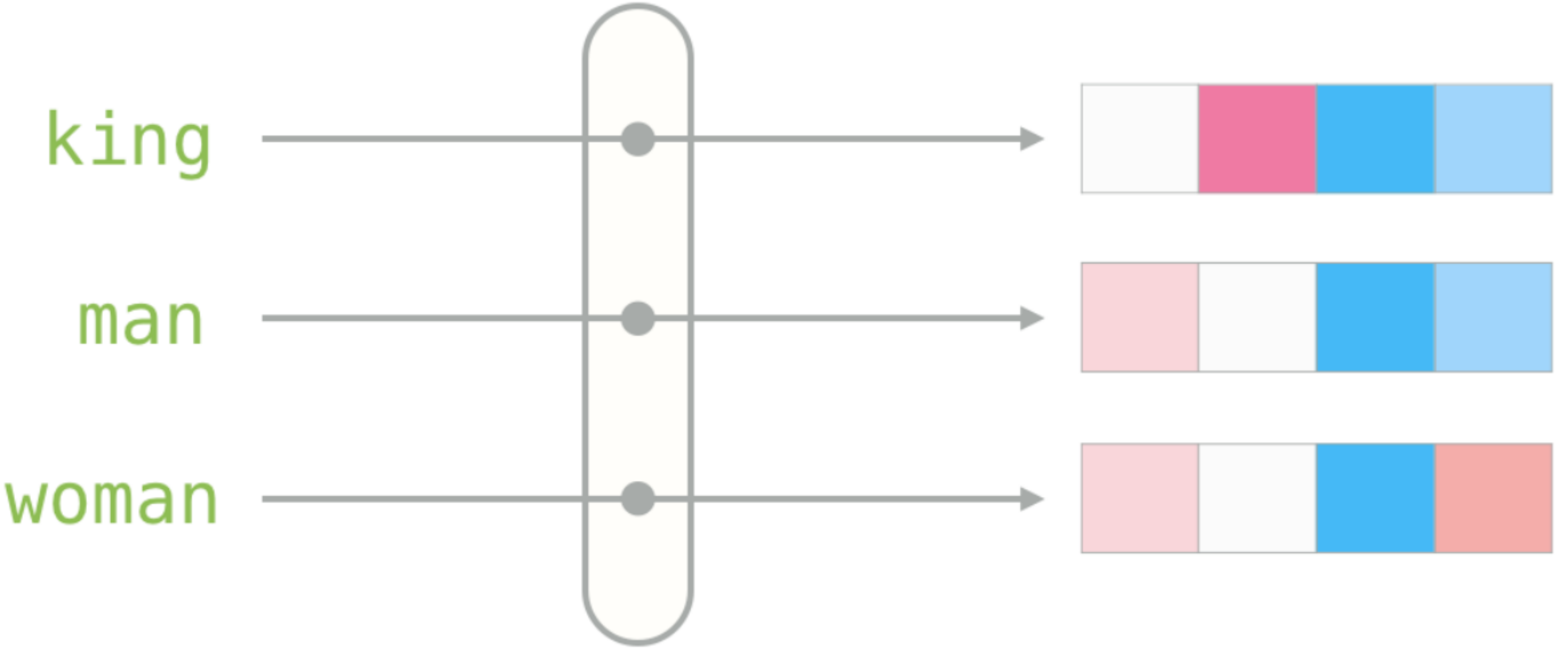
	isRonaldo	isMessi	isDicaprio
Ronaldo	1	0	0
Messi	0	1	0
Dicaprio	0	0	1



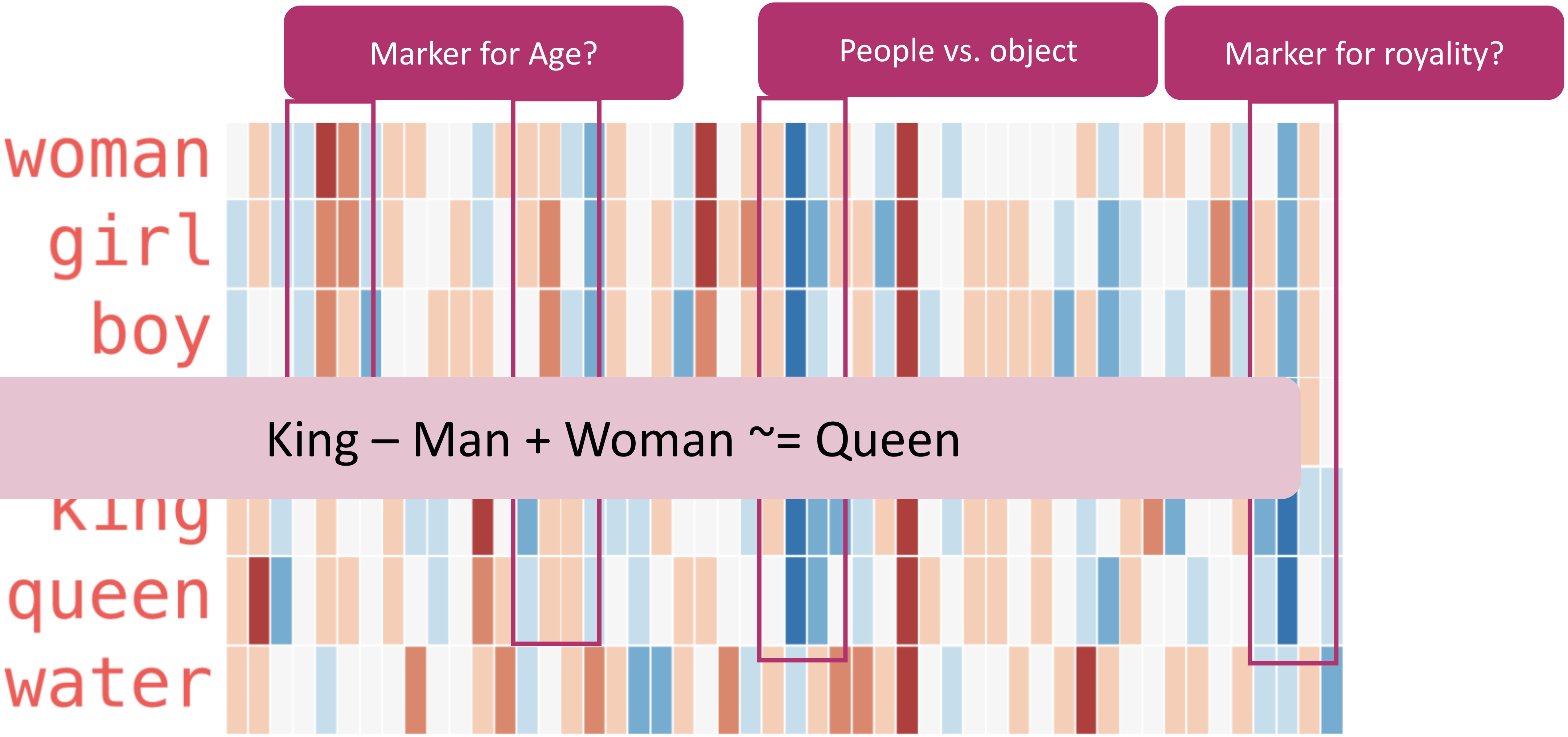
	isFootballer	isActor	Popularity	Gender	Height
Ronaldo	1	0	...	...	...
Messi	1	0	...	...	...
Dicaprio	0	1	...	...	...

Can a Neural Network do this for us?

[2]



[3]



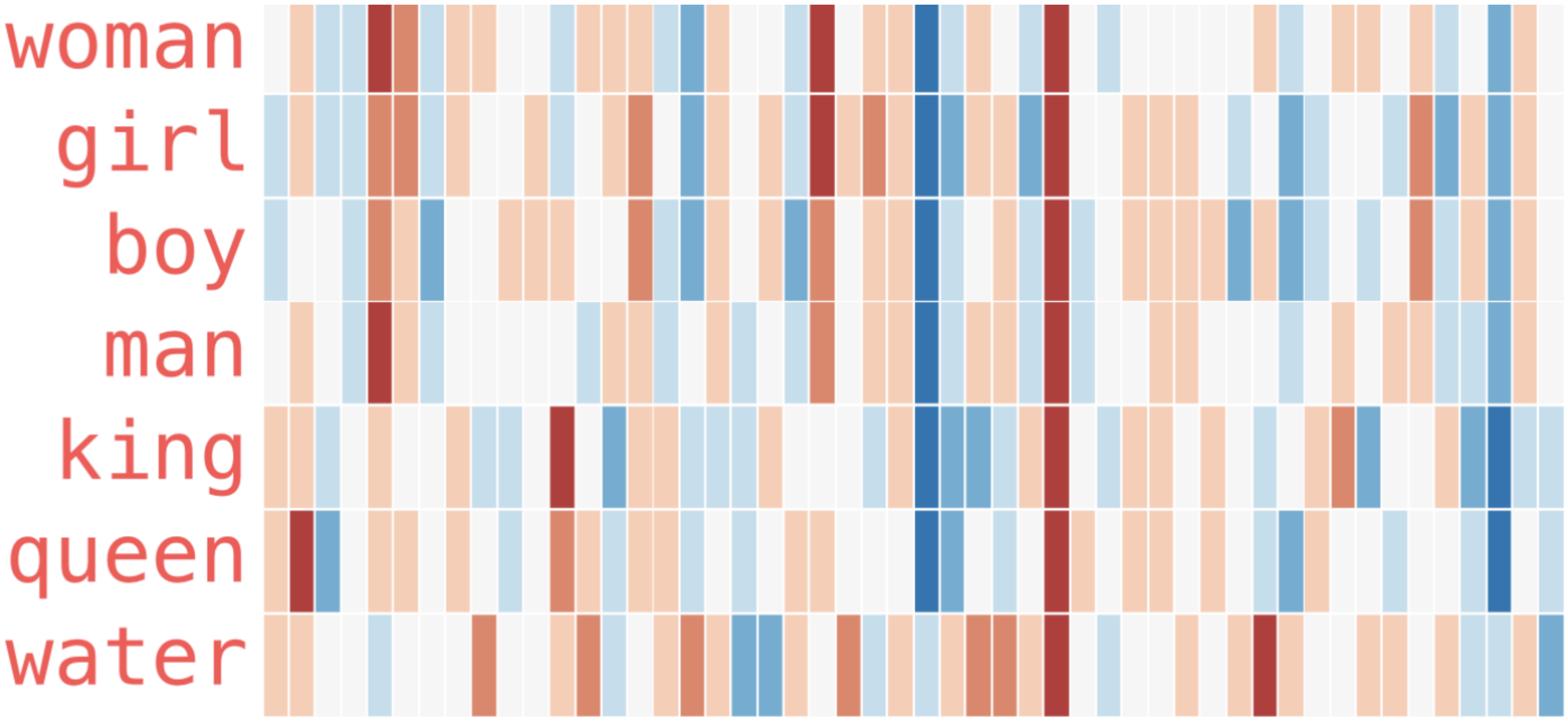
Marker for Age?

People vs. object

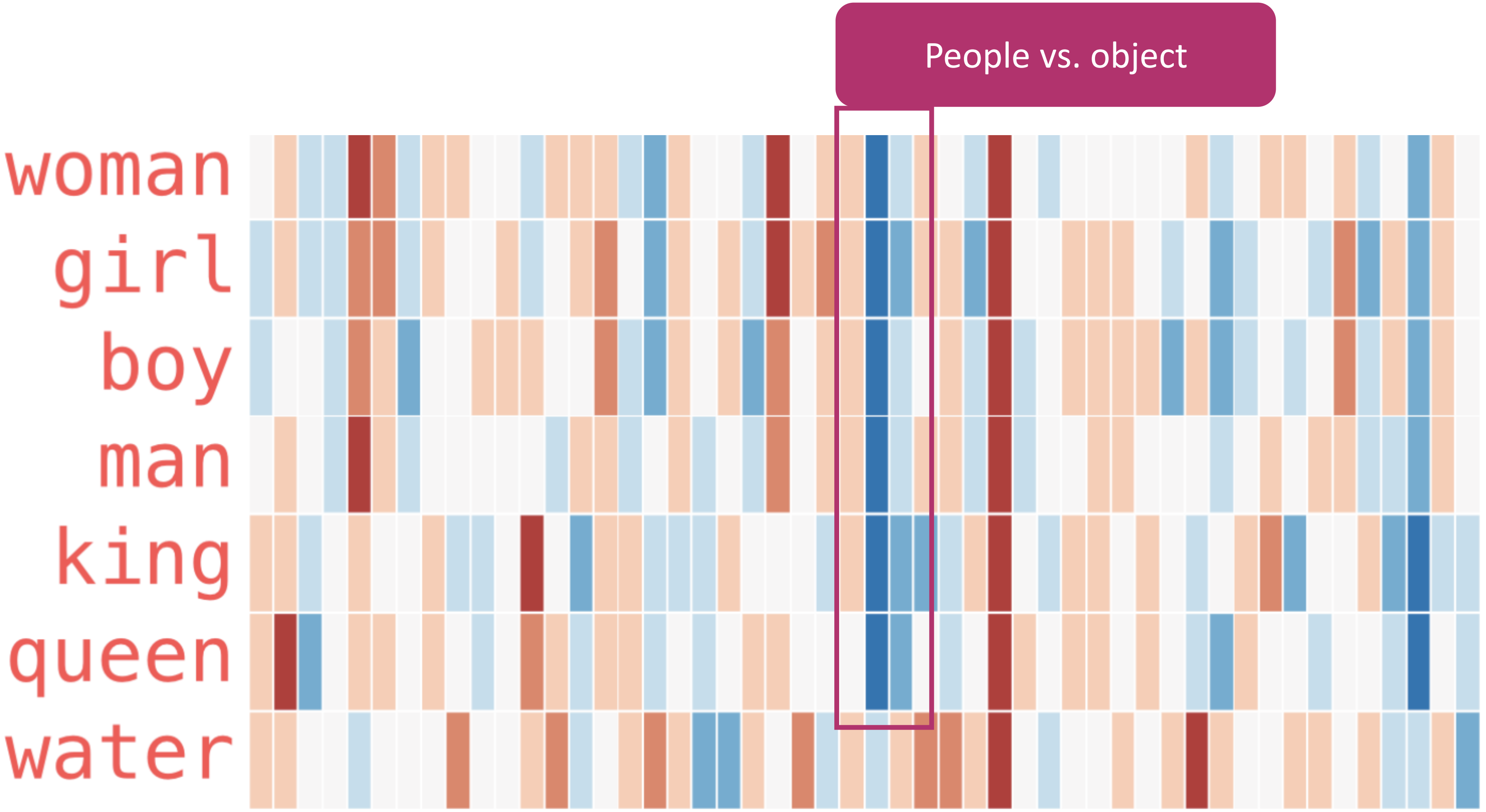
Marker for royality?

King – Man + Woman ~= Queen

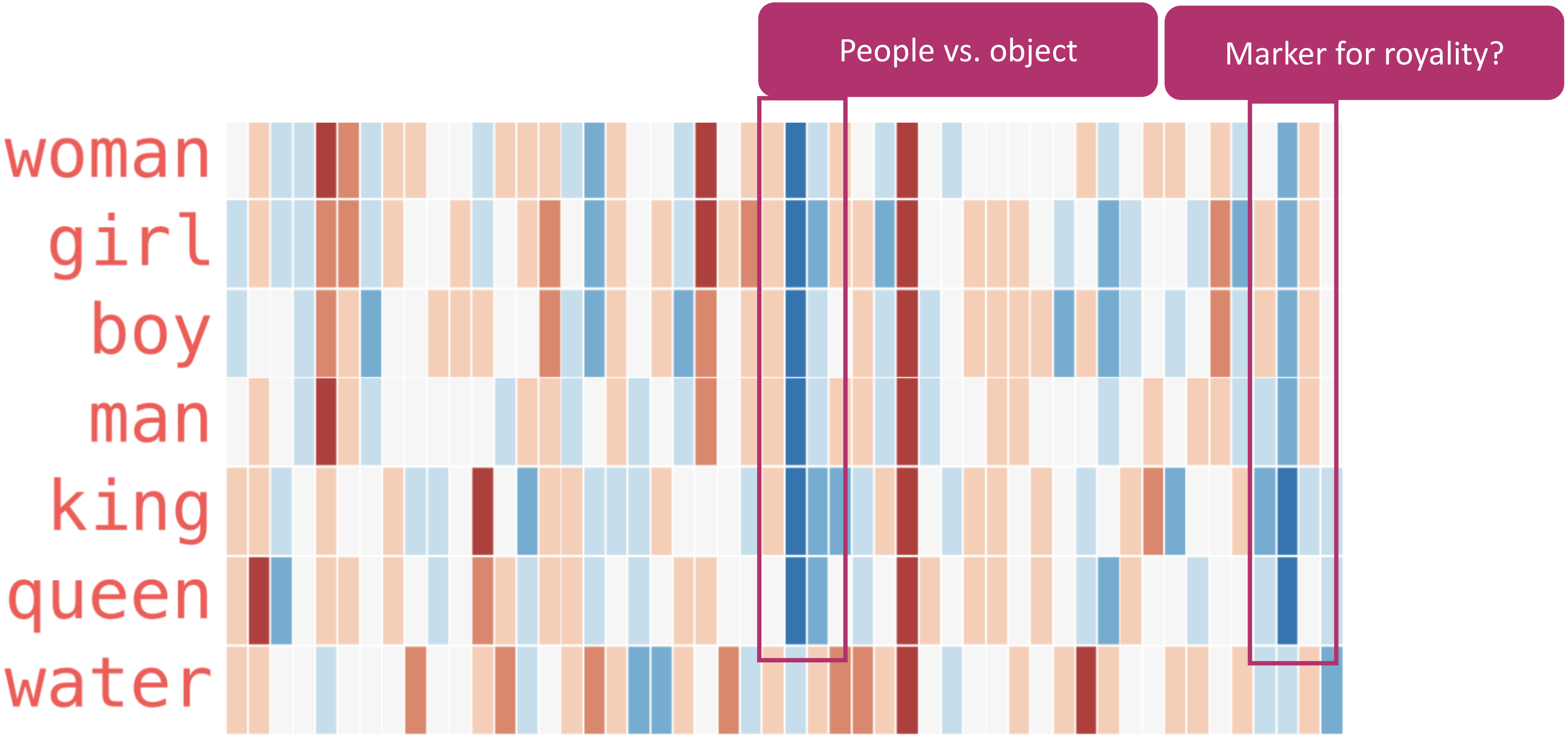
[3]



[3]

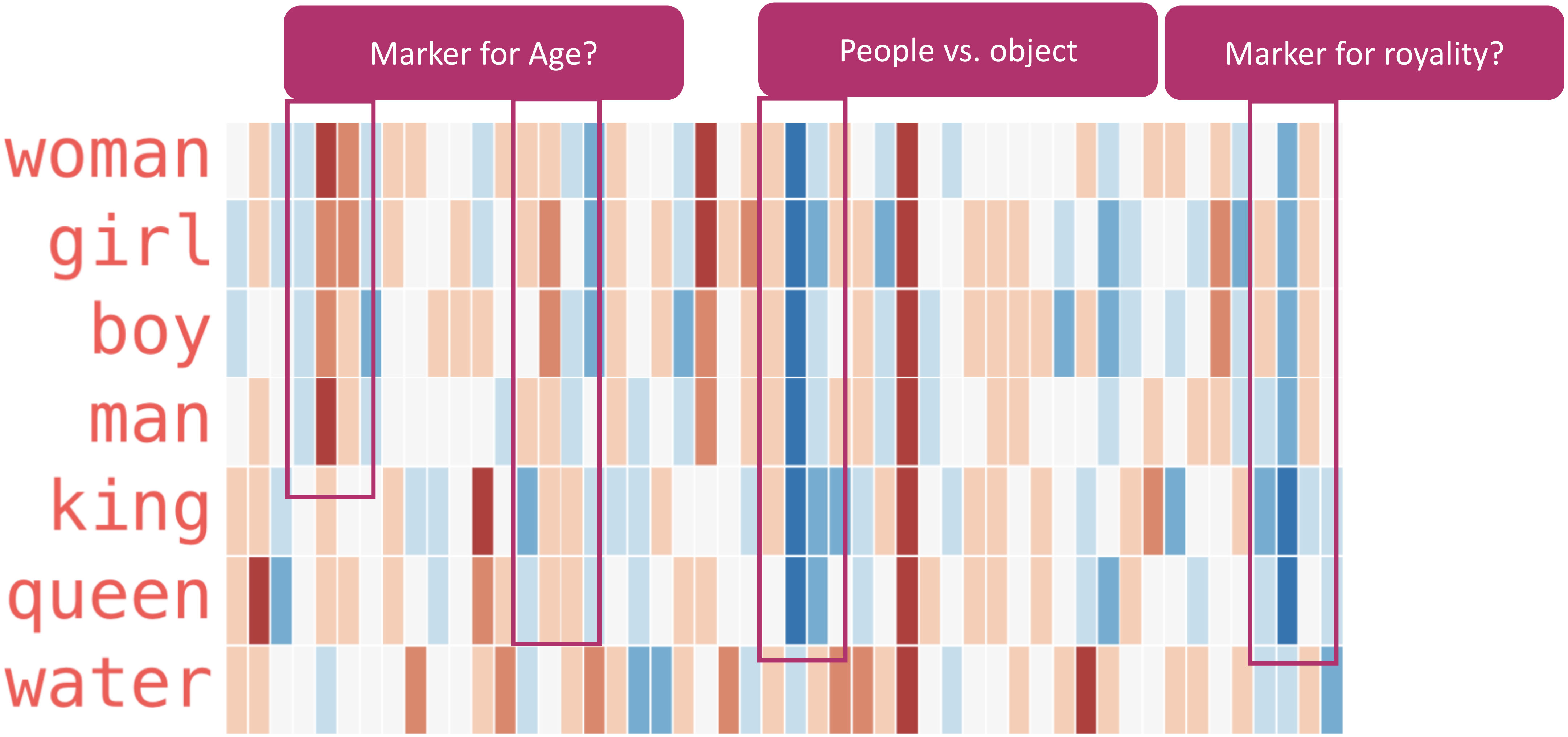


[3]



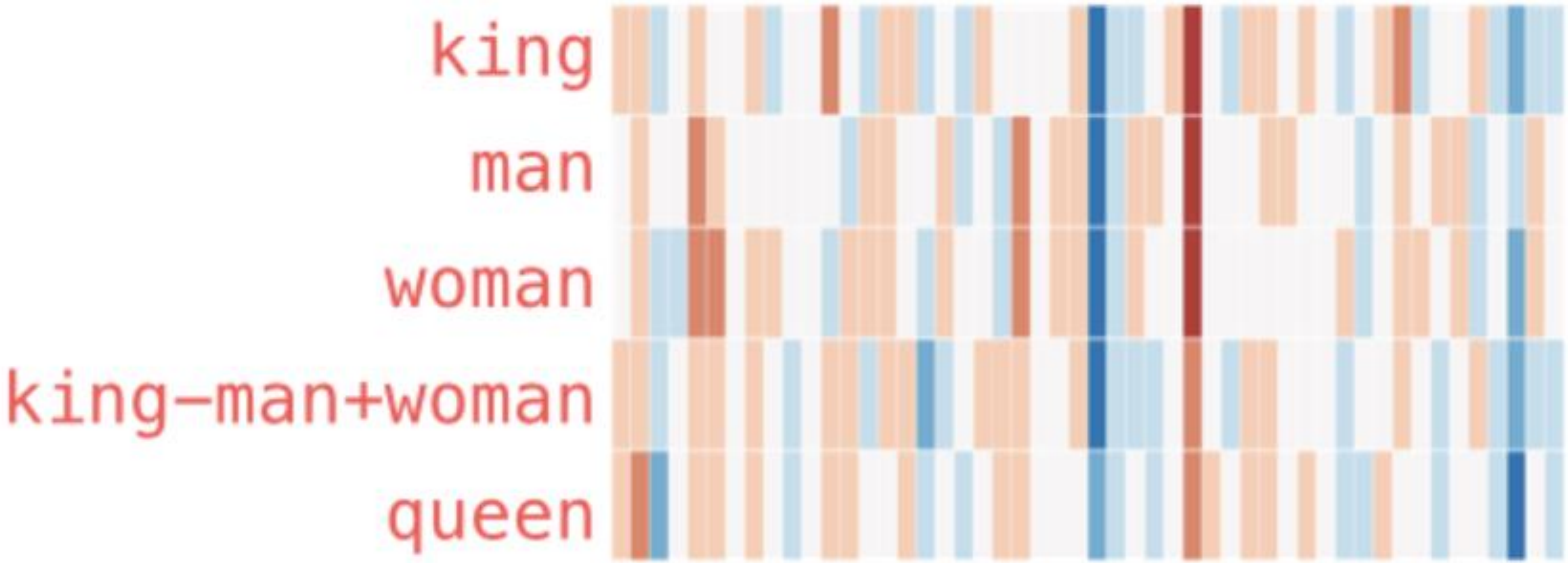
People vs. object
Marker for royality?
woman
girl
boy
man
king
queen
water

[3]

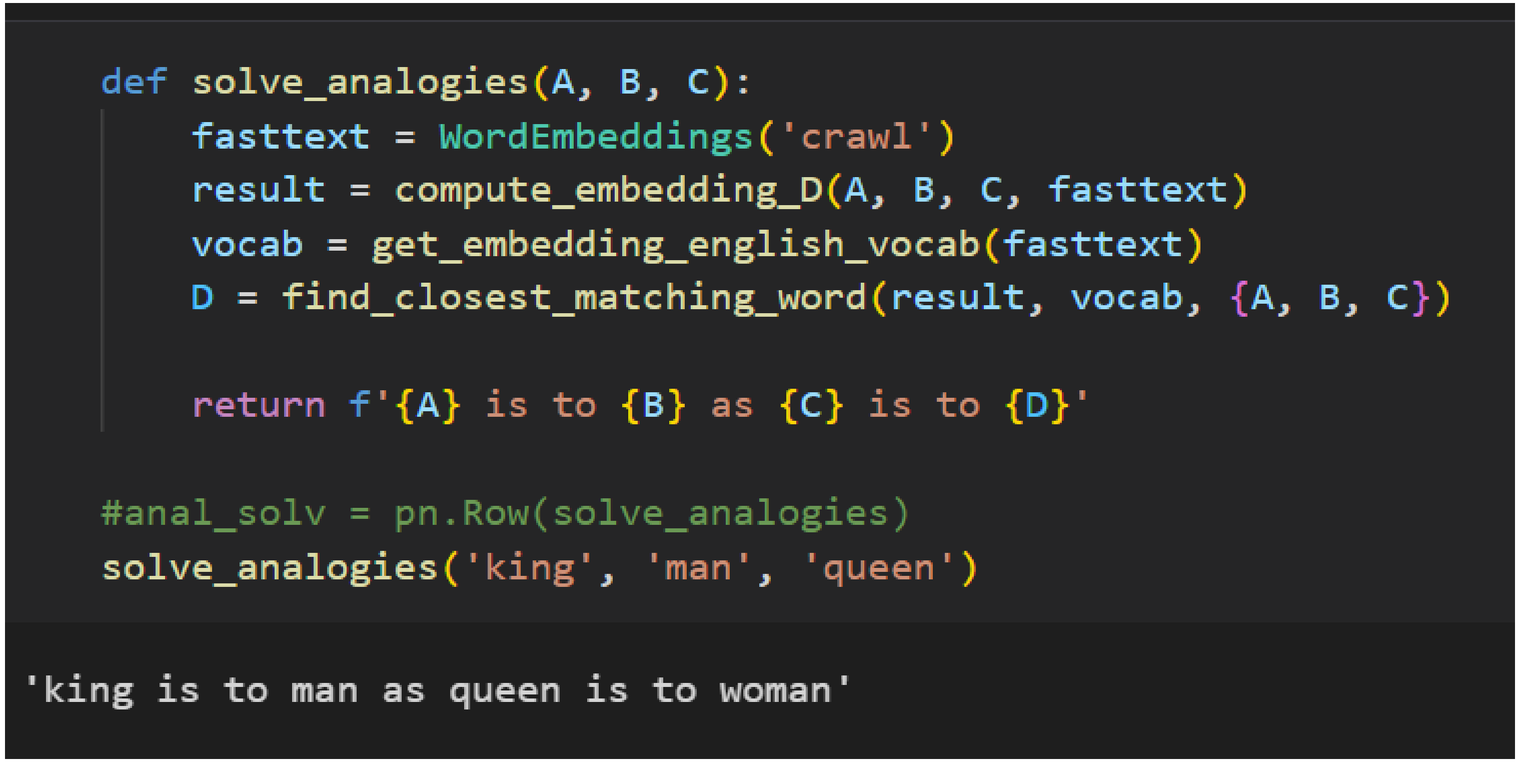

Marker for Age?
People vs. object
Marker for royality?
woman
girl
boy
man
king
queen
water

[3]

king – man + woman ~= queen



[3]



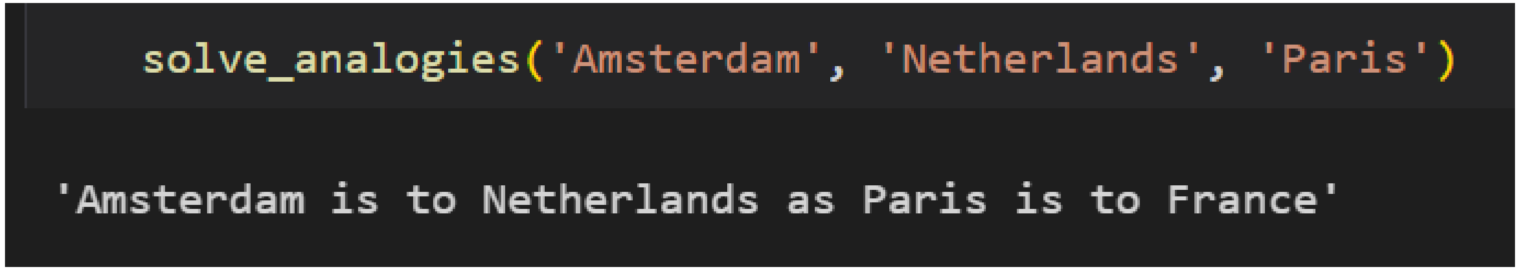
```
def solve_analogies(A, B, C):
    fasttext = WordEmbeddings('crawl')
    result = compute_embedding_D(A, B, C, fasttext)
    vocab = get_embedding_english_vocab(fasttext)
    D = find_closest_matching_word(result, vocab, {A, B, C})

    return f'{A} is to {B} as {C} is to {D}'

#anal_solv = pn.Row(solve_analogies)
solve_analogies('king', 'man', 'queen')
```

```
'king is to man as queen is to woman'
```

Word A is to Word B
As
Word C is to Word D



```
solve_analogies('Amsterdam', 'Netherlands', 'Paris')
```

```
'Amsterdam is to Netherlands as Paris is to France'
```

Different Embedding methods

Word2Vec (Mikolov et al.)

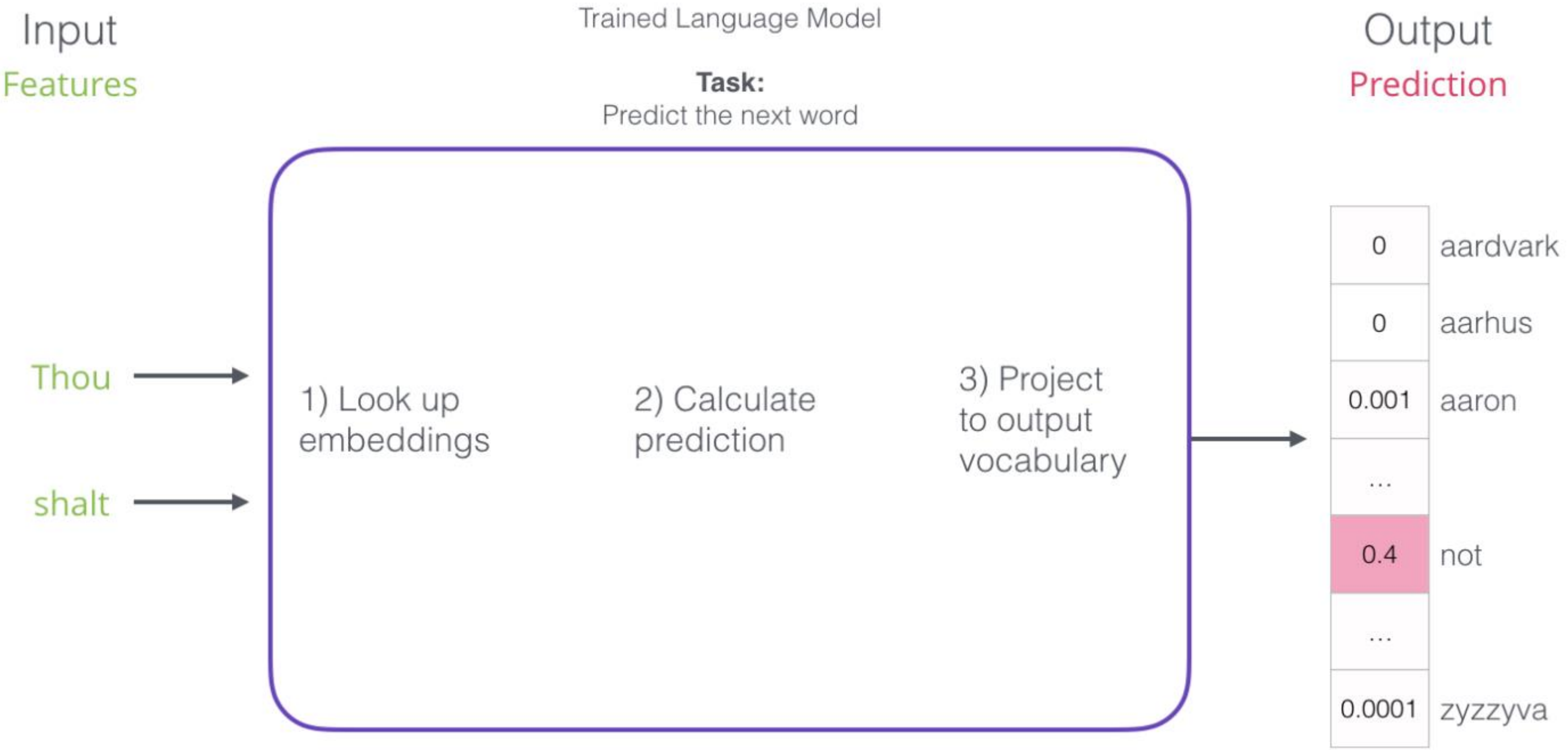
- Embedding for every word in corpus
- Sematics: consider direct neighbors
- Out of vocabulary words

FastText (Bojanowski et al.)

- Embedding for every word in corpus extended by subwords
- Sematics: consider direct neighbors

BERT

- Contextual embeddings
- Sematics: consider pairs of sentences



Input

Features

Trained Language Model

Task:
Predict the next word

Output

Prediction

Thou

shalt

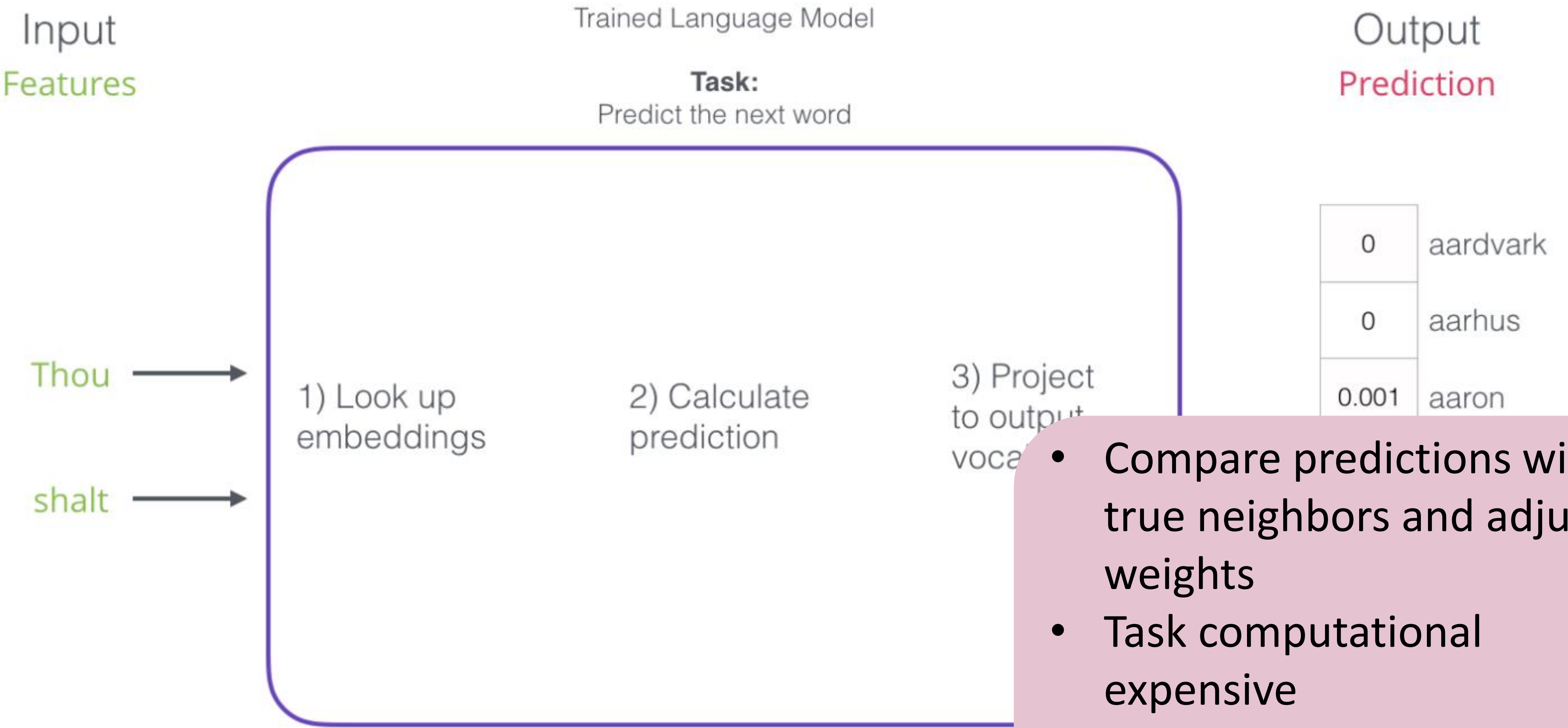
1) Look up embeddings

2) Calculate prediction

3) Project to output vocabulary

0	aardvark
0	aarhus
0.001	aaron
...	
0.4	not
...	
0.0001	zyzzyva

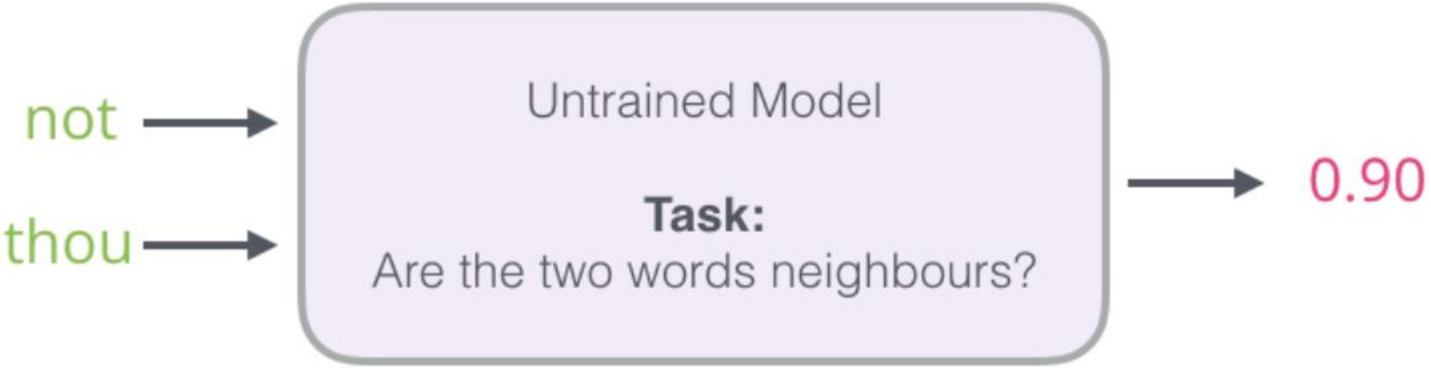
[2]



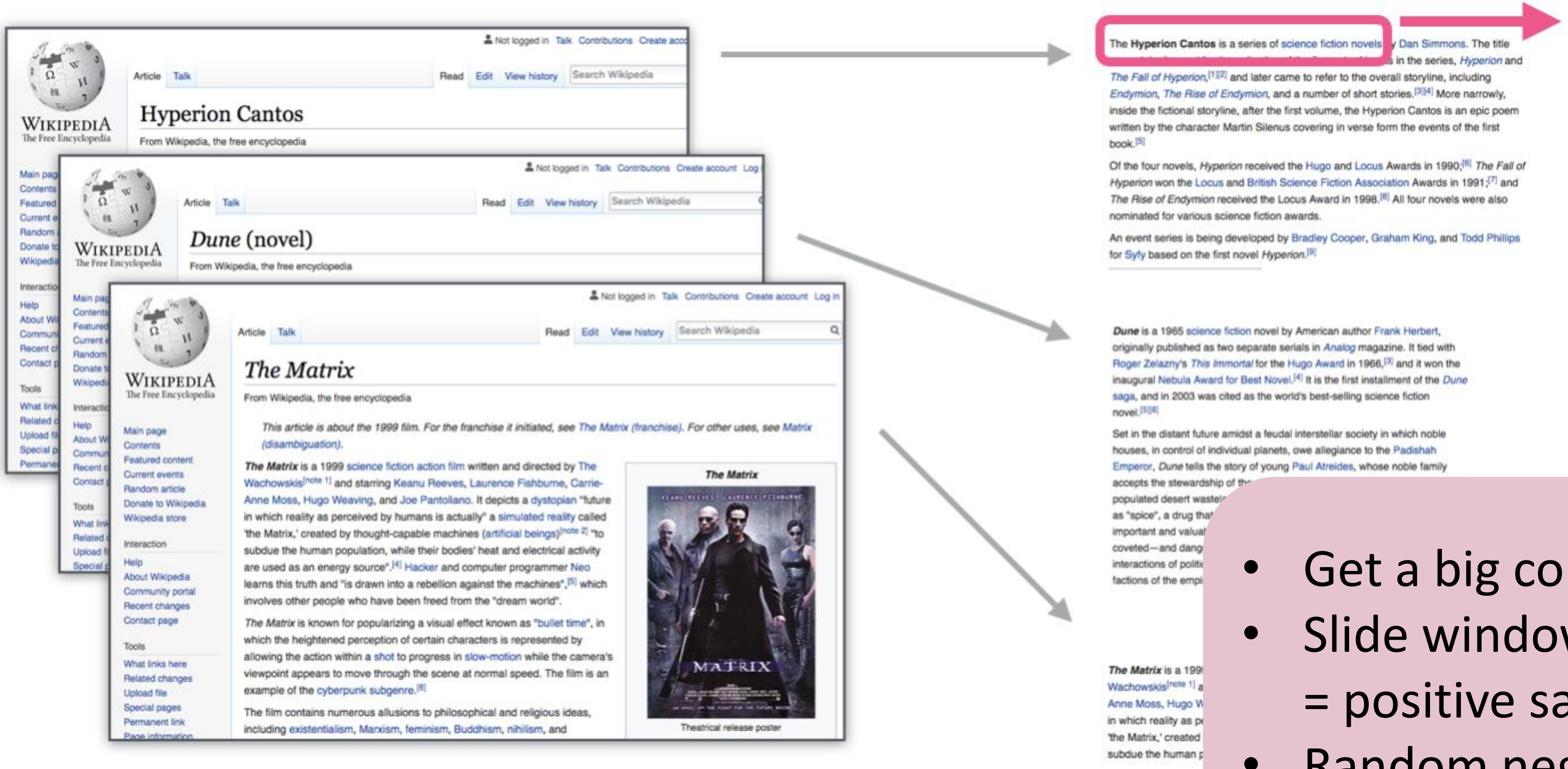
0	aardvark
0	aarhus
0.001	aaron

- Compare predictions with true neighbors and adjust weights
- Task computational expensive

➔ We want this to be a binary classification task


not
thou
Untrained Model
Task:
Are the two words neighbours?
0.90

[2]



Hyperion Cantos

From Wikipedia, the free encyclopedia

Dune (novel)

From Wikipedia, the free encyclopedia

The Matrix

From Wikipedia, the free encyclopedia

This article is about the 1999 film. For the franchise it initiated, see The Matrix (franchise). For other uses, see Matrix (disambiguation).

The Matrix is a 1999 science fiction action film written and directed by The Wachowskis[note 1] and starring Keanu Reeves, Laurence Fishburne, Carrie-Anne Moss, Hugo Weaving, and Joe Pantoliano. It depicts a dystopian "future in which reality as perceived by humans is actually" a simulated reality called 'the Matrix,' created by thought-capable machines (artificial beings)[note 2] "to subdue the human population, while their bodies' heat and electrical activity are used as an energy source".[4] Hacker and computer programmer Neo learns this truth and "is drawn into a rebellion against the machines",[5] which involves other people who have been freed from the "dream world".

The Matrix is known for popularizing a visual effect known as "bullet time", in which the heightened perception of certain characters is represented by allowing the action within a shot to progress in slow-motion while the camera's viewpoint appears to move through the scene at normal speed. The film is an example of the cyberpunk subgenre.[6]

The film contains numerous allusions to philosophical and religious ideas, including existentialism, Marxism, feminism, Buddhism, nihilism, and

The Matrix

Theatrical release poster

The **Hyperion Cantos** is a series of science fiction novels by Dan Simmons. The title s in the series, *Hyperion* and *The Fall of Hyperion*,[1][2] and later came to refer to the overall storyline, including *Endymion*, *The Rise of Endymion*, and a number of short stories.[3][4] More narrowly, inside the fictional storyline, after the first volume, the Hyperion Cantos is an epic poem written by the character Martin Silenus covering in verse form the events of the first book.[5]

Of the four novels, *Hyperion* received the Hugo and Locus Awards in 1990;[6] *The Fall of Hyperion* won the Locus and British Science Fiction Association Awards in 1991;[7] and *The Rise of Endymion* received the Locus Award in 1998.[8] All four novels were also nominated for various science fiction awards.

An event series is being developed by Bradley Cooper, Graham King, and Todd Phillips for Syfy based on the first novel *Hyperion*.[9]

Dune is a 1965 science fiction novel by American author Frank Herbert, originally published as two separate serials in *Analog* magazine. It tied with Roger Zelazny's *This Immortal* for the Hugo Award in 1966,[3] and it won the inaugural Nebula Award for Best Novel.[4] It is the first installment of the *Dune* saga, and in 2003 was cited as the world's best-selling science fiction novel.[5][6]

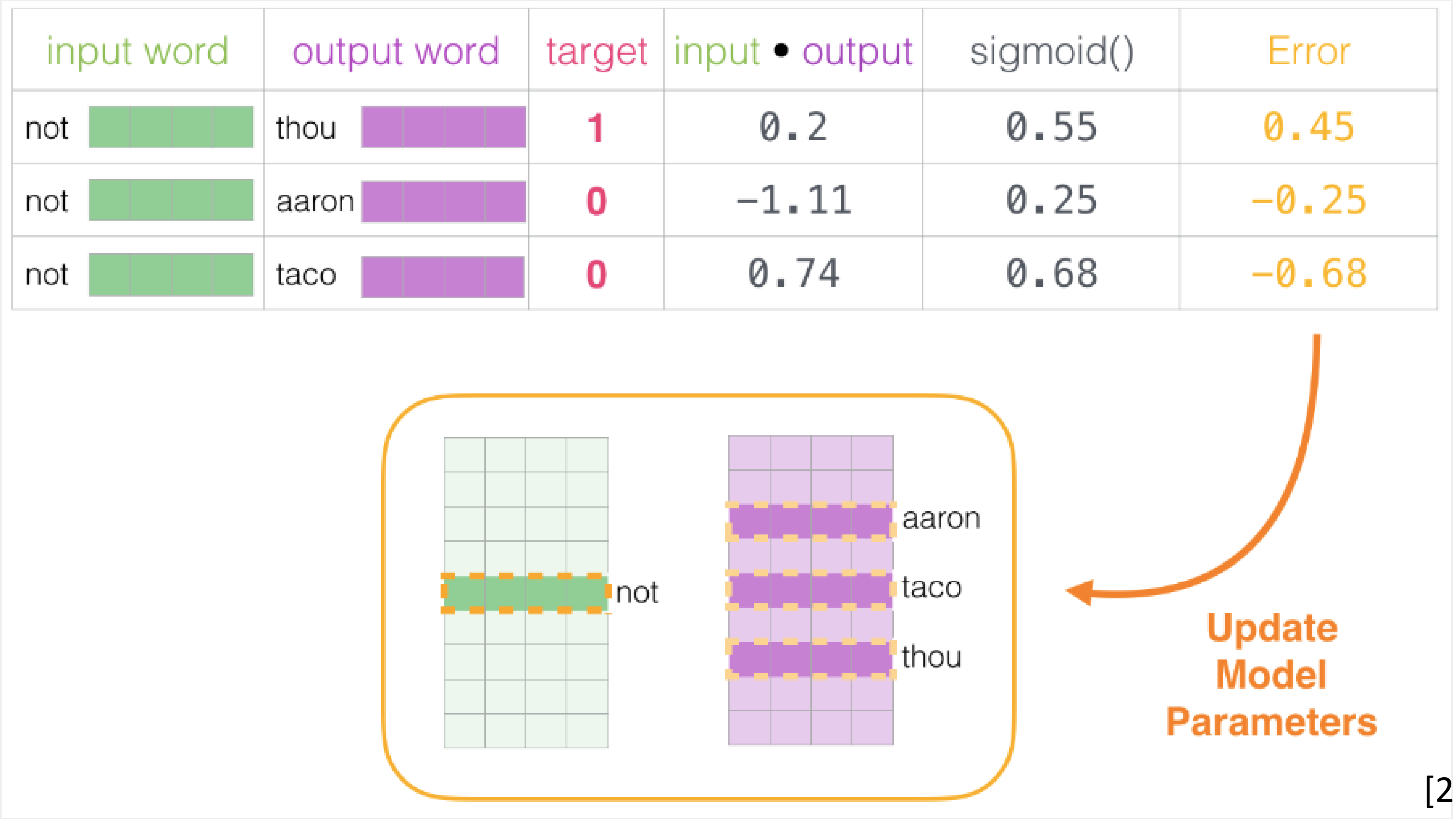
Set in the distant future amidst a feudal interstellar society in which noble houses, in control of individual planets, owe allegiance to the Padishah Emperor, *Dune* tells the story of young Paul Atreides, whose noble family accepts the stewardship of the
populated desert wastel
as "spice", a drug that
important and valua
coveted—and dang
interactions of politi
factions of the empi

The Matrix is a 199
Wachowskis[note 1] a
Anne Moss, Hugo W
in which reality as p
'the Matrix,' created
subdue the human p

[2]

- Get a big corpus
- Slide window over corpus = positive samples
- Random negative samples

➔ Training examples

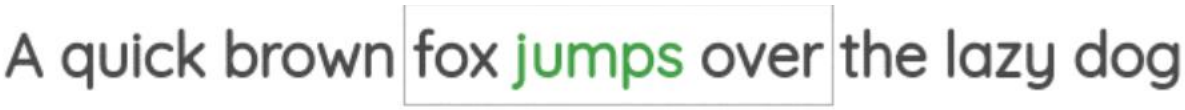


input word	output word	target	input • output	sigmoid()	Error
not 🟩🟩🟩🟩	thou 🟪🟪🟪🟪	1	0.2	0.55	0.45
not 🟩🟩🟩🟩	aaron 🟪🟪🟪🟪	0	-1.11	0.25	-0.25
not 🟩🟩🟩🟩	taco 🟪🟪🟪🟪	0	0.74	0.68	-0.68

Update Model Parameters

[2]

Continuous Bag of words (CBOW)

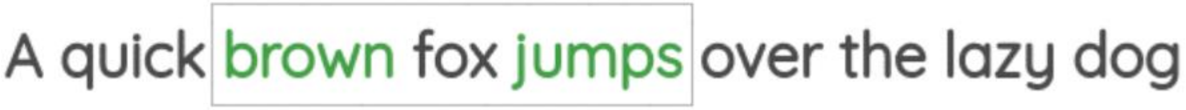


A quick brown fox jumps over the lazy dog

[3]

Predict central word based on neighbors

Skip-gram



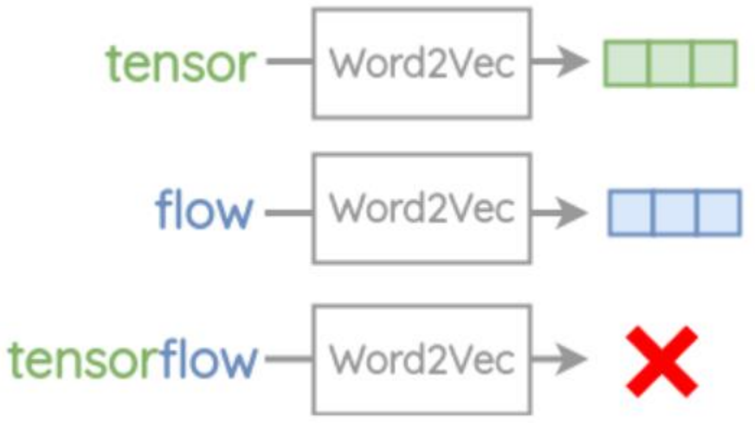
A quick brown fox jumps over the lazy dog

[3]

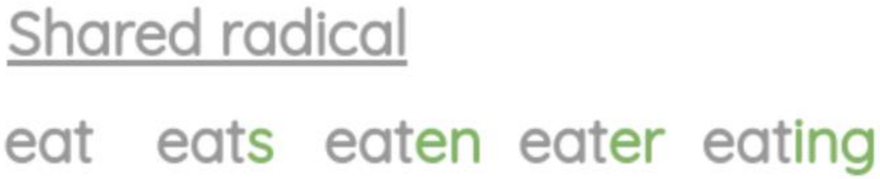
Predict neighbor words from central word

Limitation: Out of Vocabulary Words cause problems

Limitation: Morphology, no Parameter sharing



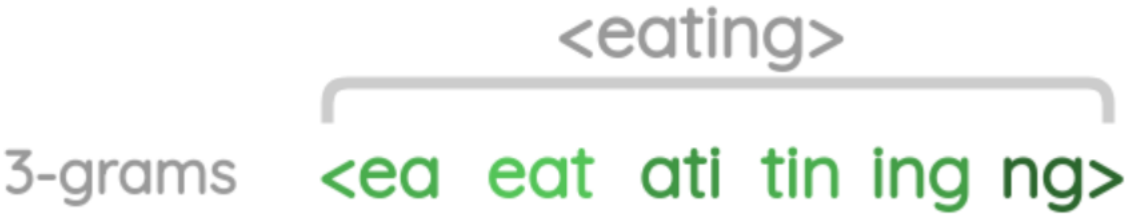
[2]



[2]

Use internal structure to improve embeddings

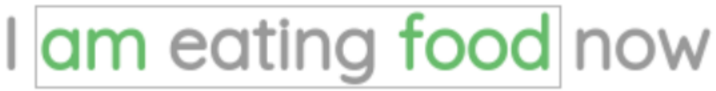
Do skip-gram embeddings and obtain subwords for central word



[2]

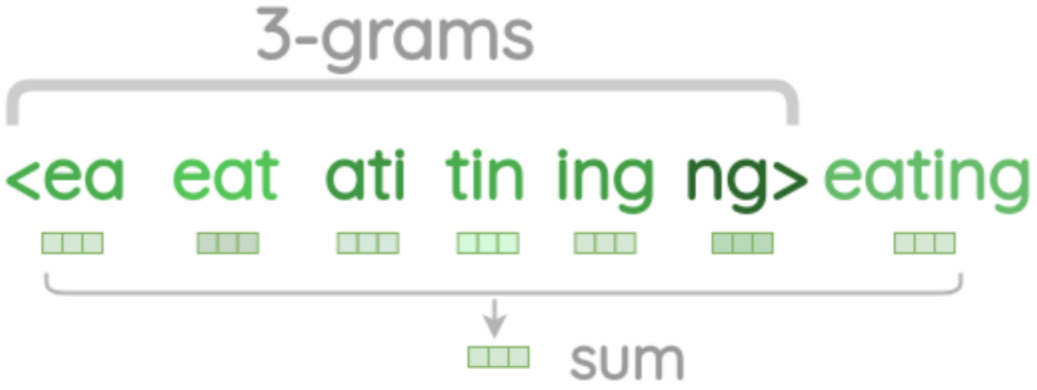
1.) Embed central word

Example Sentence



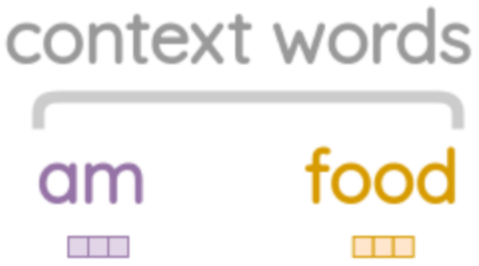
[2]

Central word embedding

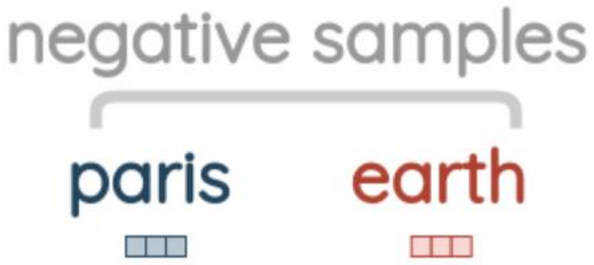


[2]

2.) Sampling

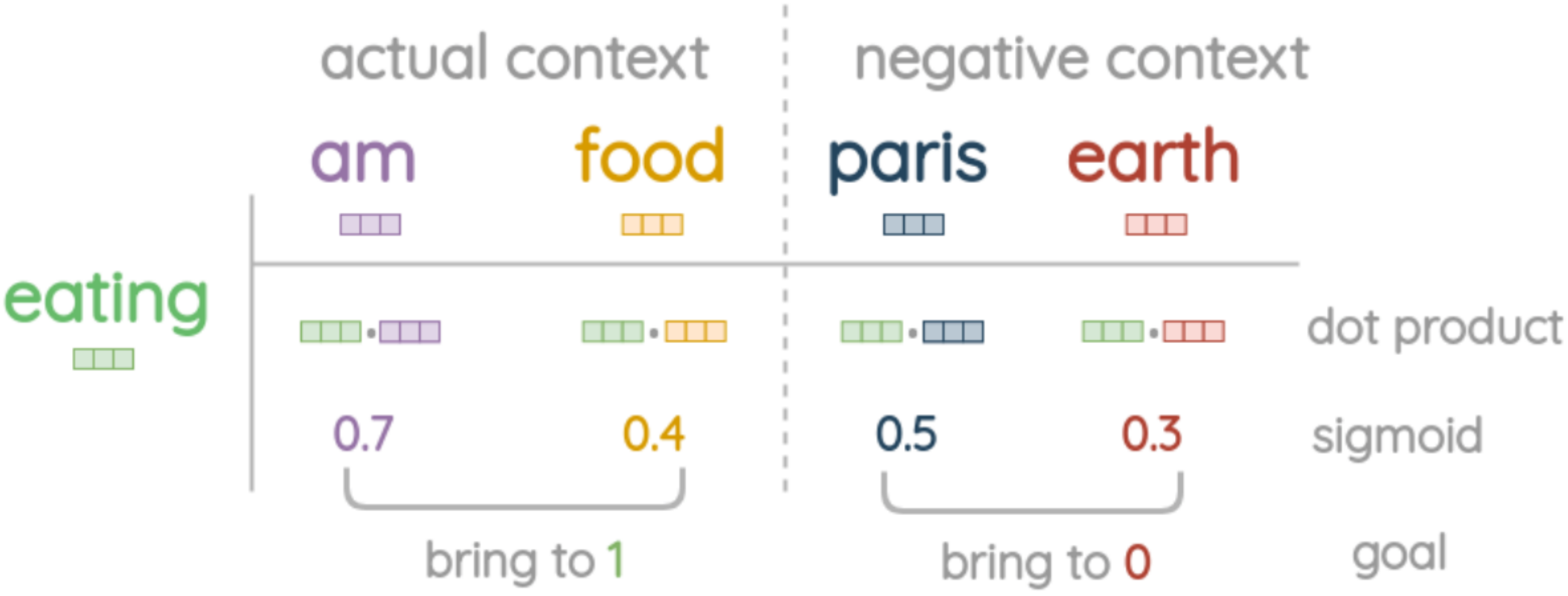


[2]



[2]

2.) Train the model



[2]

BERT embeddings: Bidirectional Encoder Representations from Transformer

BERT embeddings: Bidirectional Encoder Representations from Transformer

Contextual embeddings: the same word can have different embeddings based on context

BERT embeddings: Bidirectional Encoder Representations from Transformer

Contextual embeddings: the same word can have different embeddings based on context

Embeddings based on whole sentences

BERT embeddings: Bidirectional Encoder Representations from Transformer

Contextual embeddings: the same word can have different embeddings based on context

Embeddings based on whole sentences

Whole word embedding, subword embedding, character embedding

Multi-layer model → how to define the final embedding?

BERT embeddings: Bidirectional Encoder Representations from Transformer

- Word embedding:
  - Concatenate last four layers (3.072 dimensions)
  - Sum last for layers (768 dimensions)

BERT embeddings: Bidirectional Encoder Representations from Transformer

- Word embedding:
  - Concatenate last four layers (3.072 dimensions)
  - Sum last for layers (768 dimensions)

- Sentence embedding:
  - Average second – last hidden layer (768 dimensions)

BERT embeddings: Bidirectional Encoder Representations from Transformer

- Word embedding:
  - Concatenate last four layers (3.072 dimensions)
  - Sum last for layers (768 dimensions)

- Sentence embedding:
  - Average second – last hidden layer (768 dimensions)

Purpose:

- Information retrieval without keyword or phrase overlap
- High-quality input features for downstream NLP tasks

Different Embedding methods, different performance

Word2Vec (Mikolov et al.)

- Good performance on semantic analogy

FastText (Bojanowski et al.)

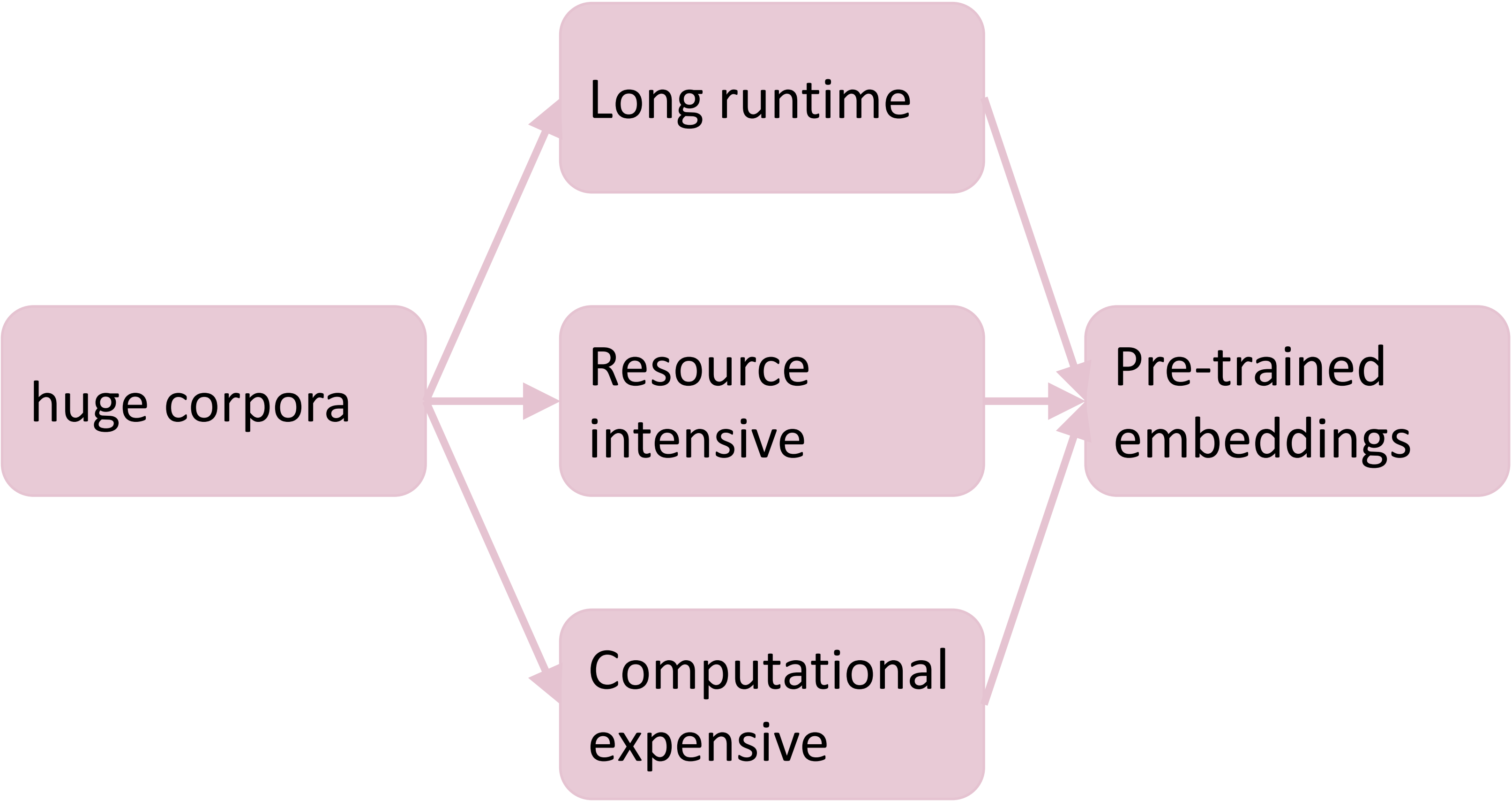
- Improved performance on syntactic analogy
- Worse performance on semantic analogy

BERT

- similarity comparison for words invalid
- Similarity comparison for sentences valid

state-of-the-art performance:

- No out-of-vocabulary words
- Capture syntax and semantics



huge corpora

Long runtime

Resource intensive

Computational expensive

Pre-trained embeddings

How does the magic of NLP comes to life?

- Text vectorization
- **Powerful NLP models**
- Application for my project

ELMO

BERT

Open-GPT

Allen AI

Google

OpenAI

[4]

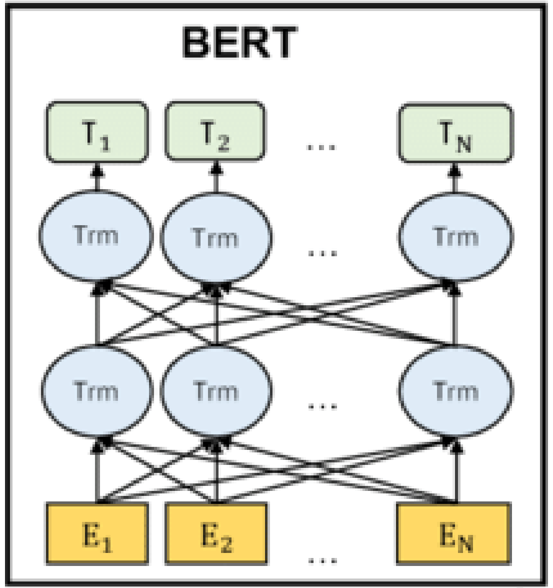
[5]

[6]

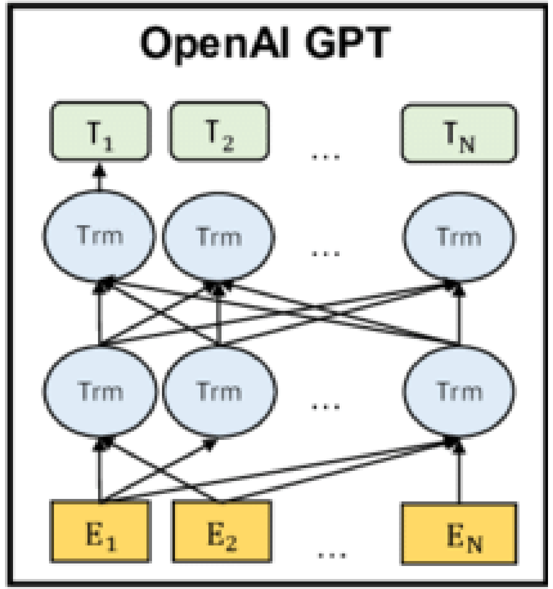
Powerful NLP models

[5]

	BERT	Open-GPT [6]
Model Architecture	• Transformers • bidirectional	• Transformers • Unidirectional (left)



[7]



[7]

[5]

	BERT	Open-GPT [6]
Model Architecture	• Transformers • bidirectional	• Transformers • Unidirectional (left)
Task Type	Supervised e.g. text classification	Unsupervised e.g. text generation

[5]

[6]

	BERT	Open-GPT
Model Architecture	• Transformers • bidirectional	• Transformers • Unidirectional (left)
Task Type	Supervised e.g. text classification	Unsupervised e.g. text generation
Training data	masked language modelling, next sentence prediction	Language modelling

[5]

[6]

	BERT	Open-GPT
Model Architecture	- Transformers - bidirectional	- Transformers - Unidirectional (left)
Task Type	Supervised e.g. text classification	Unsupervised e.g. text generation
Training data	masked language modelling, next sentence prediction	Language modelling
Output	Fixed length embeddings for downstream NLP tasks	Sequence of tokens (variable length)

How does the magic of NLP comes to life?

- Text vectorization
- Powerful NLP models
- **Application for my project**

Concluding Remarks

- **Limitations**
- My favourite NLP tools

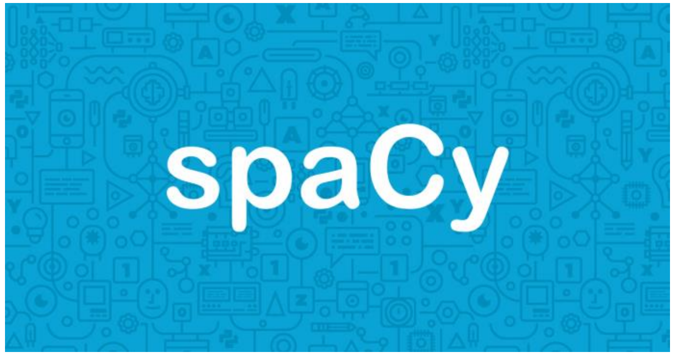
Custom solutions too expensive and
no state-of-the-art performance

Work with the Resources given:
Domain transfer challenging, suboptimal results

Language is not precise: exceptions to be handled

Concluding Remarks

- Limitations
- **My favourite NLP tools**



[11]

[12]

Outstanding performance for Lemmatization
Biomedical sequence tagging with scientific models

[13]

Own embedding method
Interface for third-party model use especiall for text classification

[14]

NLTK

[15]

Do you see me now?

References:
[1] spam filter image: https://i0.wp.com/www.metronetworksllc.com/wp-content/uploads/2018/08/iStock-538057636.jpg?fit=2510%2C1194&ssl=1
[2] A visual Guide to FastText Embeddings: https://amitness.com/2020/06/fasttext-embeddings/
[3] The illustrated Word2vec: https://jalammar.github.io/illustrated-word2vec/
[4] ELMO image: https://static.smalljoys.me/2020/04/img_5e8f13ed41e91.png
[5] BERT image: https://i1.wp.com/jacobiem.org/wp-content/uploads/2020/10/Bert.jpg
[6] GPT image: https://mixed.de/wp-content/uploads/2019/03/open_ai_lp_logo.jpg
[7] BERT vs GPT image: https://www.researchgate.net/publication/340797092_Recent_Trends_in_Deep_Learning_Based_Open-Domain_Textual_Question_Answering_Systems/figures?lo=1
[8] PubMed: http://gomerpedia.org/images/thumb/1/10/PubMed_Logo.jpg/600px-PubMed_Logo.jpg
[9] BioBERT: https://academic.oup.com/view-large/figure/394146824/BIOINFORMATICS_36_4_1234_f1.png
[10] Sentiment analysis: https://www.expressanalytics.com/wp-content/uploads/2021/06/sentimentanalysishotelgeneric-2048x803-1.jpg
[11] spaCy: https://external-content.duckduckgo.com/iu/?u=https%3A%2F%2Fcdn.analyticsvidhya.com%2Fwp-content%2Fuploads%2F2020%2F03%2Flogo.jpg&f=1&nofb=1&ipt=e302d95cf8cf666fd7b986920eb64ad92db863a3c7e15207b864f217a3ceeca4&ipo=images
[12] ScispaCy: https://external-content.duckduckgo.com/iu/?u=https%3A%2F%2Fraw.githubusercontent.com%2Fallenai%2Fscispacy%2Fmaster%2Fdocs%2Fscispacy-logo.png&f=1&nofb=1&ipt=74c2da01bc4c1d01842d993131ef45b3309c5bb97269d4067336c536d1738a37&ipo=images
[13] flair: https://i.pinimg.com/originals/b3/76/fa/b376fa02b4699f22b4f9ec2d314a4f13.png
[14] textblob: https://unipython.com/wp-content/uploads/2018/03/An%C3%A1lisis-de-sentimientos-con-Python-min-1316x547.png
[15] GENSIM: https://tech.clickdo.co.uk/wp-content/uploads/2021/07/Gensim.jpg