

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



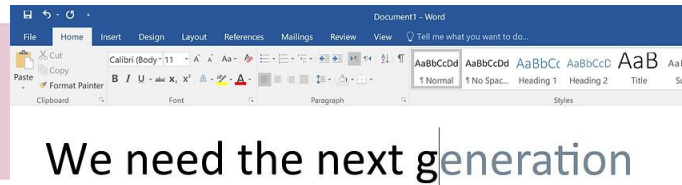
Spam filtering



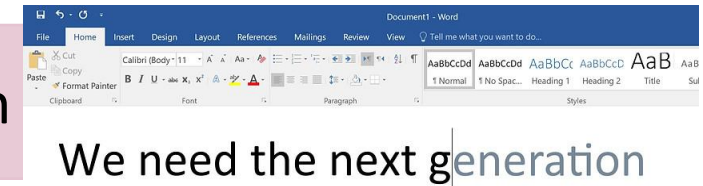
Text classification



Word prediction



Word prediction



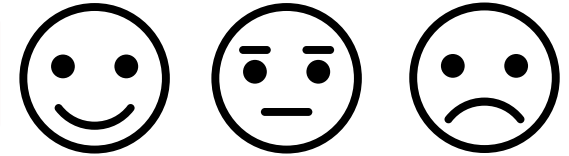
Emotion detection



Customer reviews



Sentiment analysis

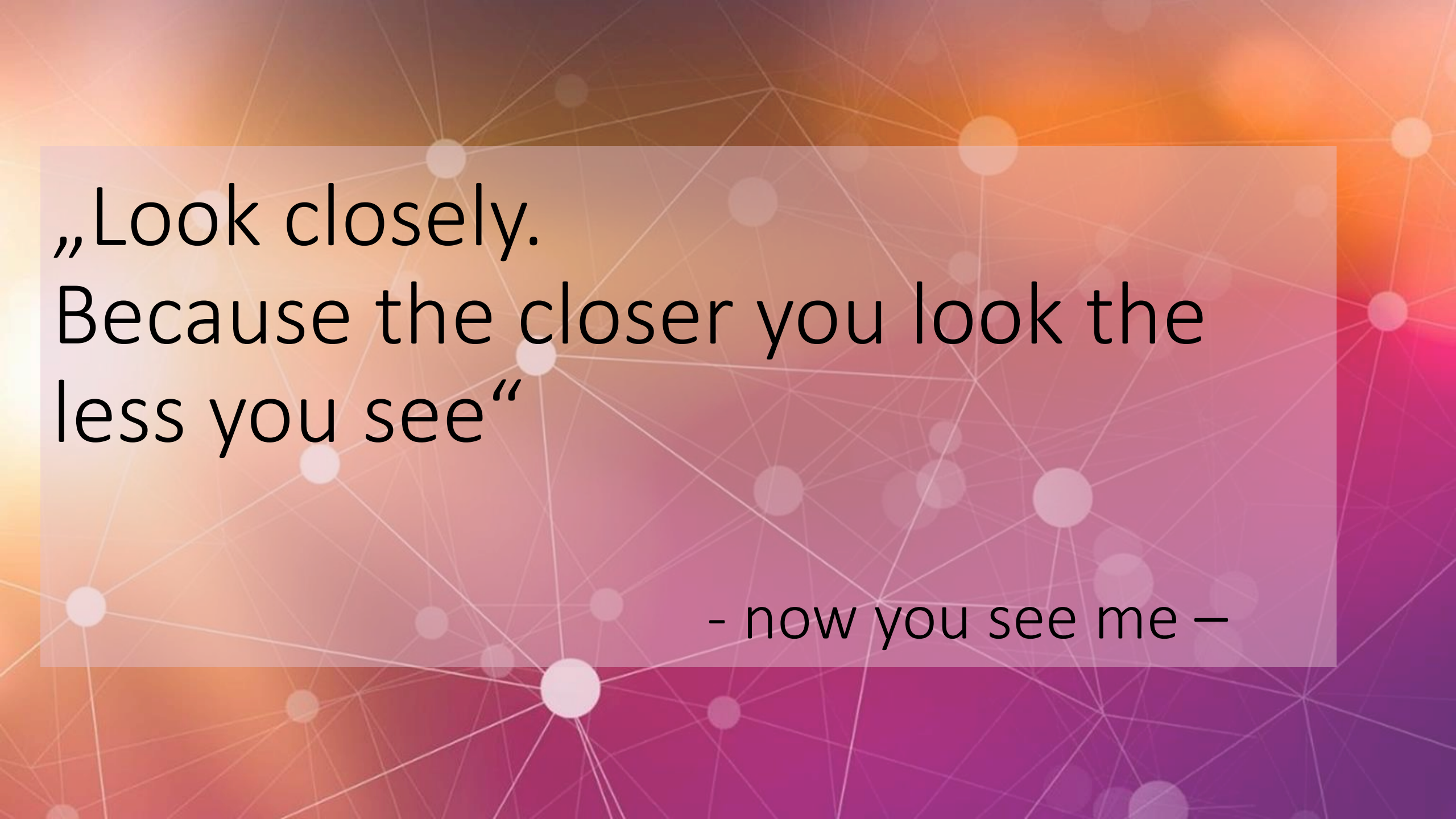


Chatbots



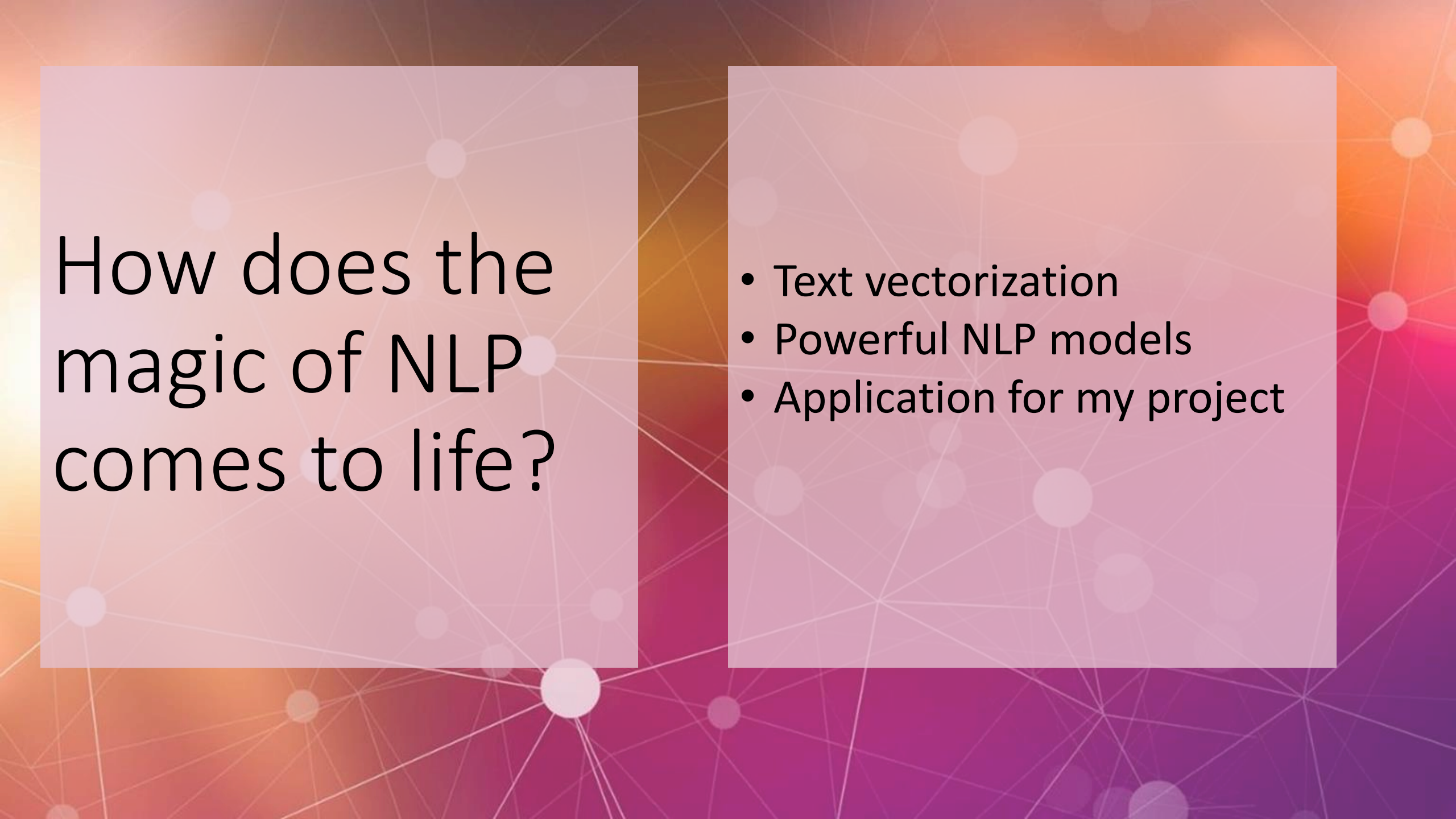
Text Generation





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

- now you see me –

The background of the slide features a network of white dots connected by thin white lines, set against a gradient background transitioning from orange at the top to purple at the bottom. Two semi-transparent rectangular boxes are overlaid on this background.

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

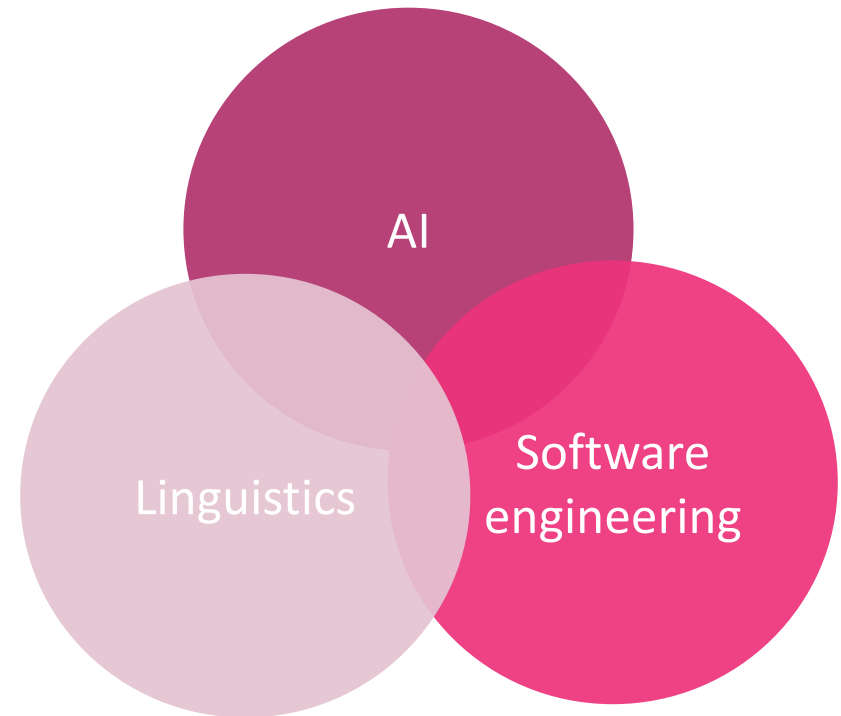
Distinguish between question and answer

Detect language structure and syntax

Abstract meaning of text



Human Brain:
comprehend all at once



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

Tokenization

Conversational
English

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

Tokenization

Text Vectorization

Conversational
English

2 sentences,
24 words

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 ;)

Text Classification

E.g. Sentiment
analysis

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 ;)

Text Classification

E.g. Sentiment
analysis

Question
Answering

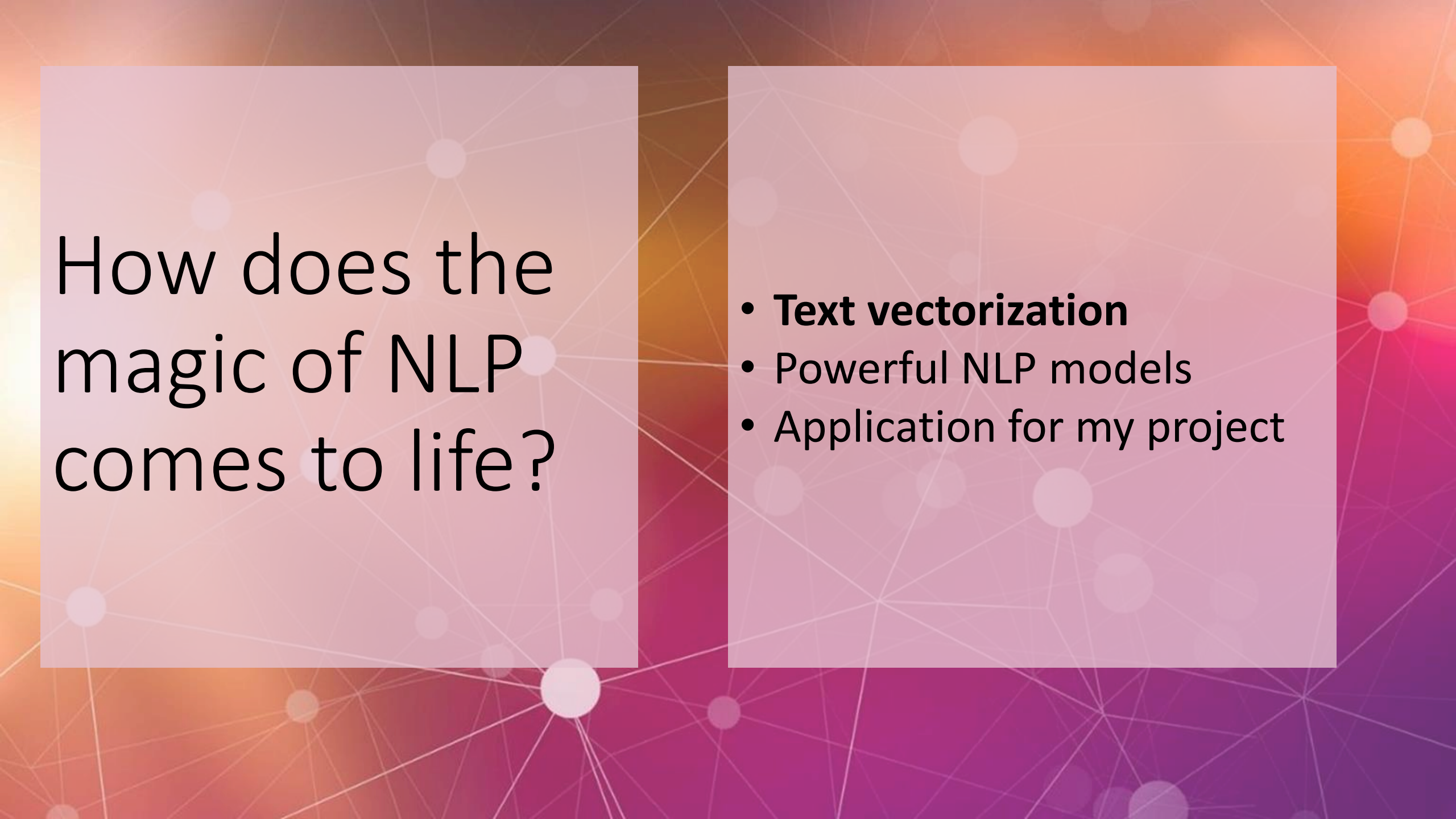
E.g. should I study
at the Hanze?

DON'T STUDY HERE!

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



NLP Downstream Tasks

The background of the slide features a network of white dots connected by thin white lines, set against a gradient background transitioning from orange at the top to purple at the bottom. The dots vary in size, and the lines form a complex web-like structure.

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

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

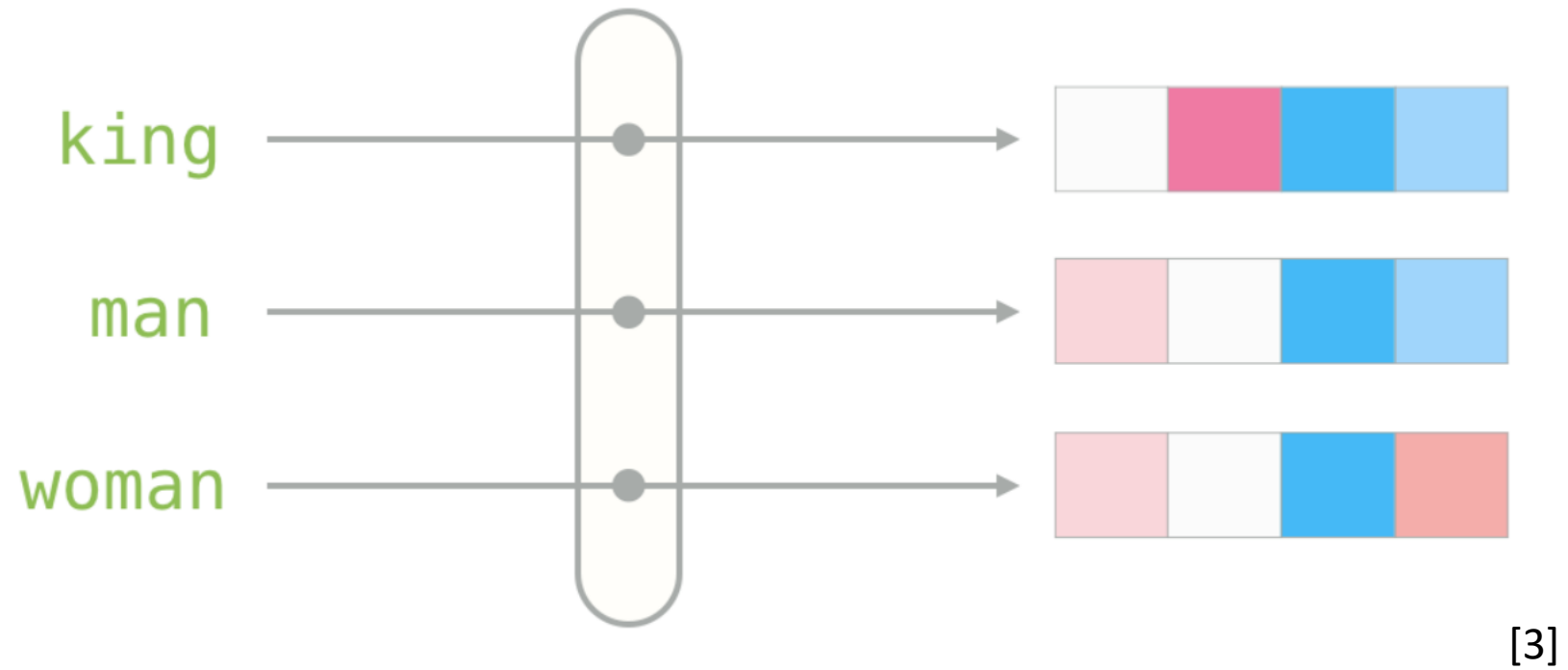
Can a Neural Network
do this for us?

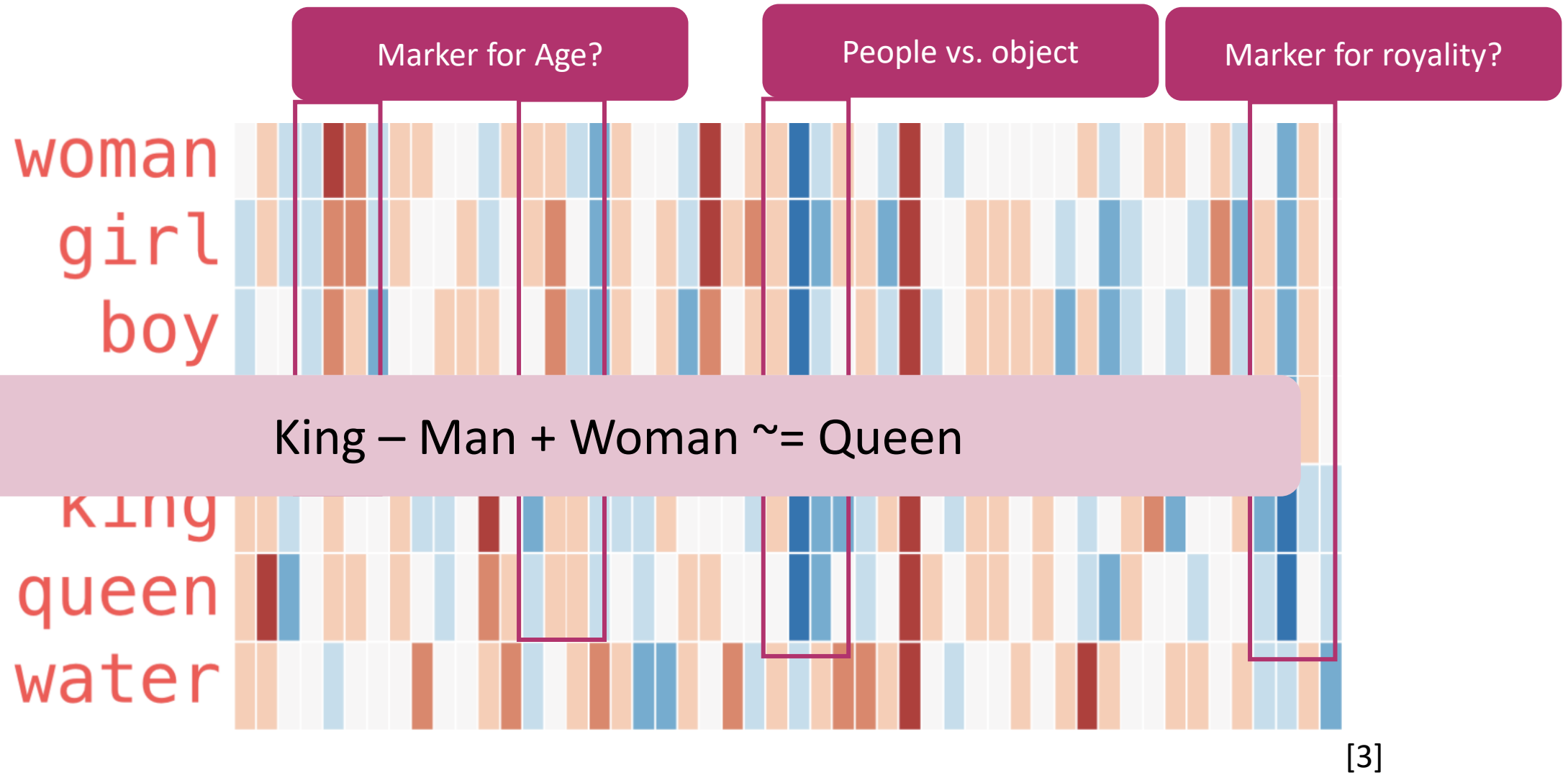
	isFootballer	isActor
Ronaldo	1	0
Messi	1	0
Dicaprio	0	1

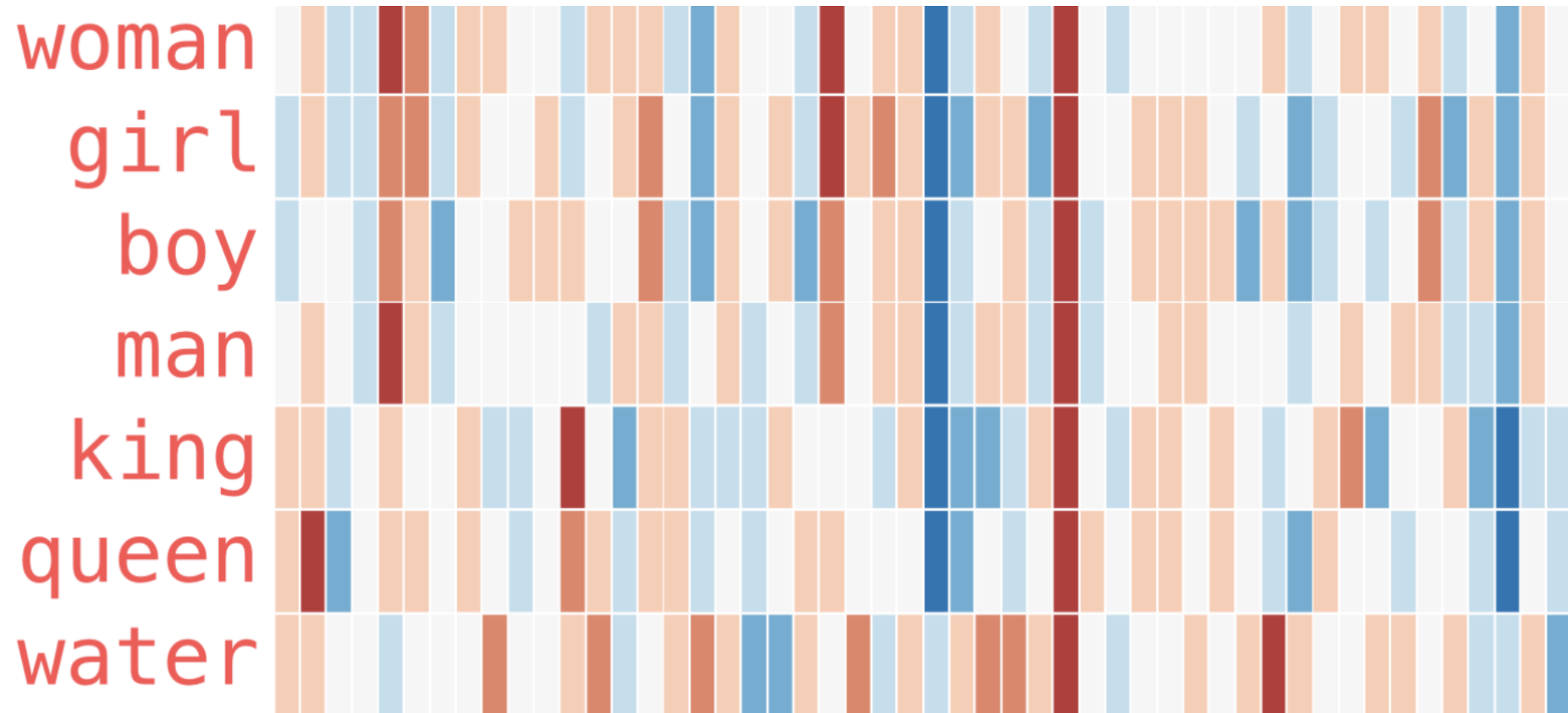


	isFootballer	isActor	Popularity	Gender	Height
Ronaldo	1	0
Messi	1	0
Dicaprio	0	1

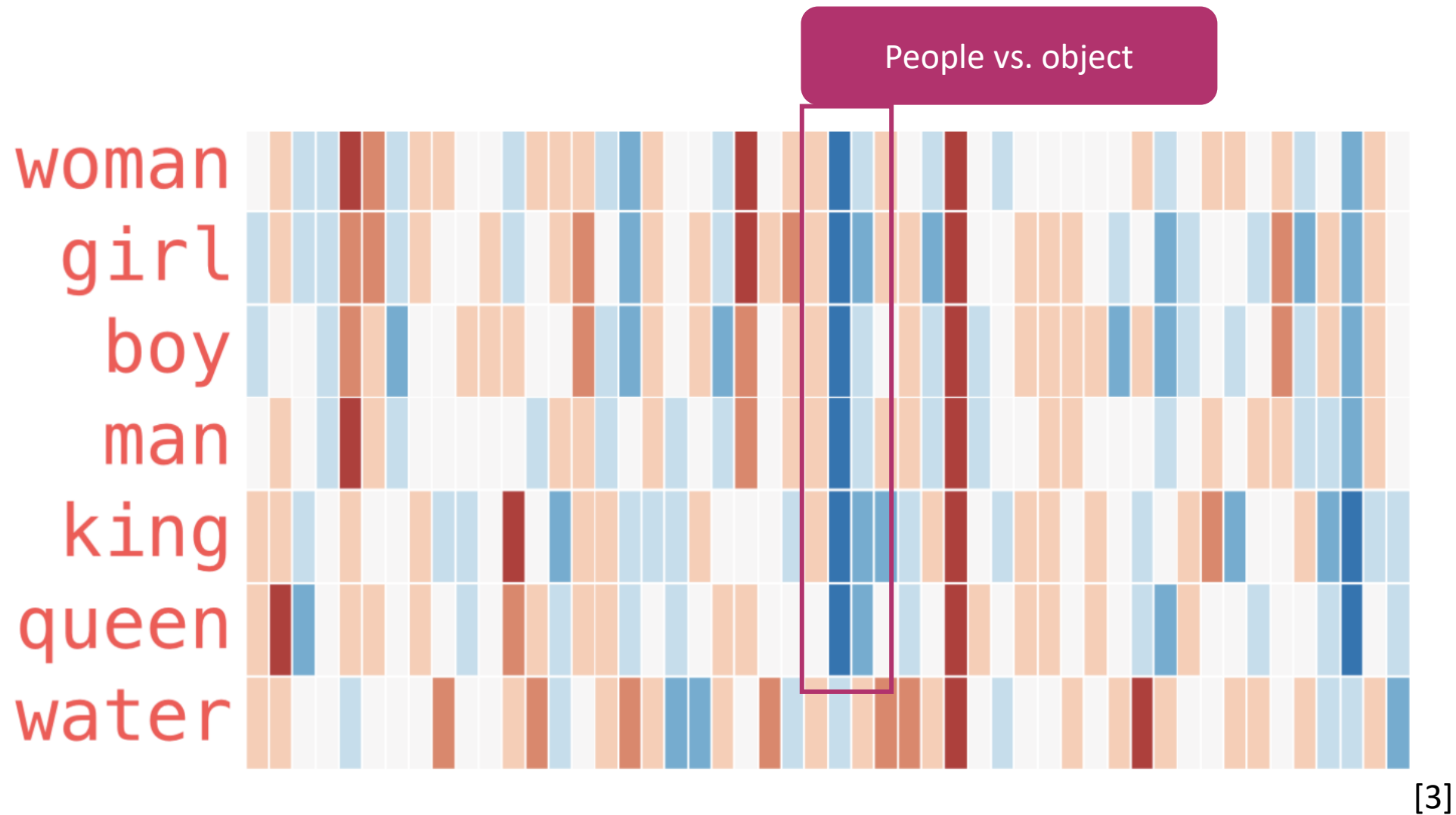
[2]

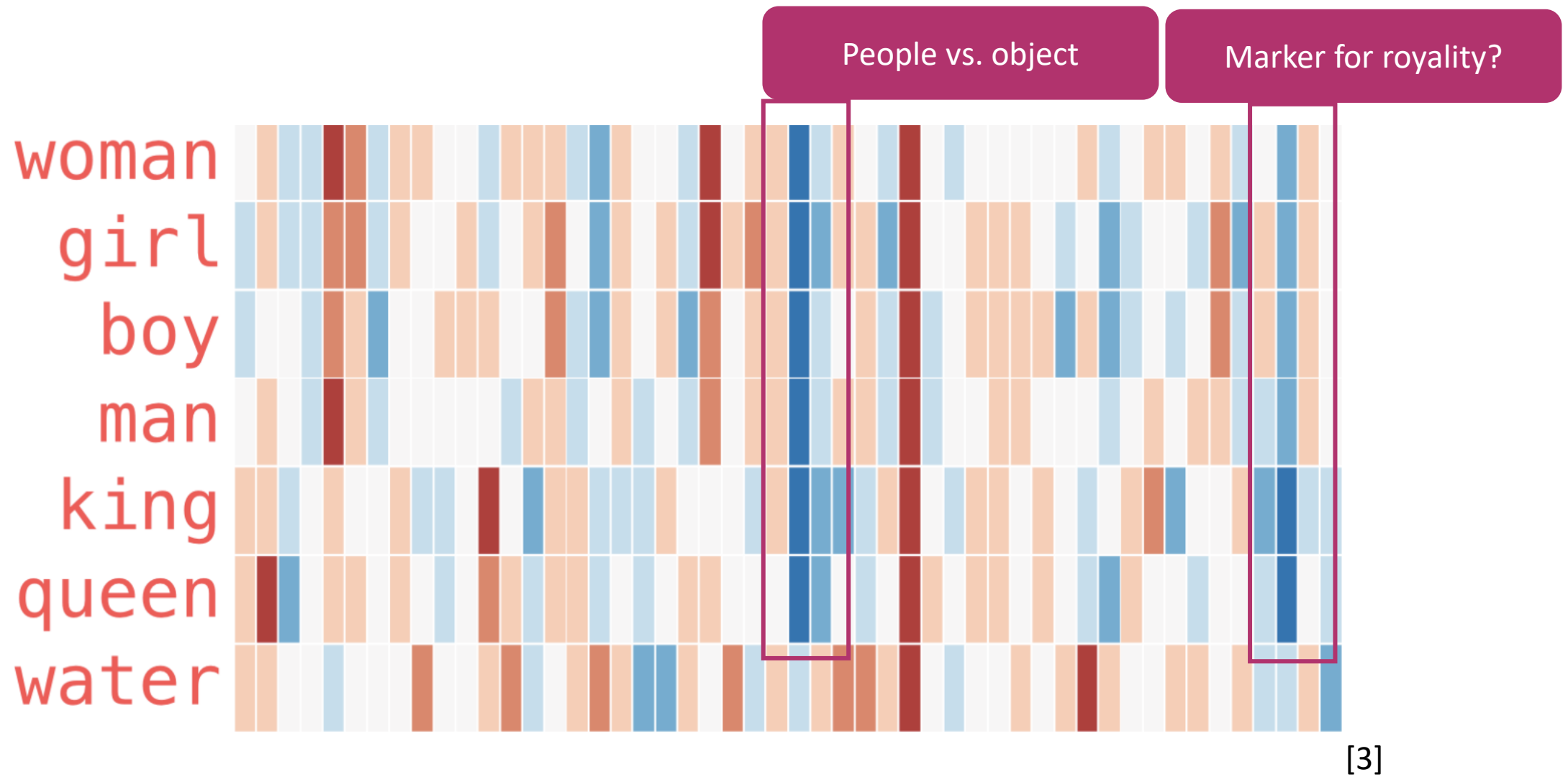


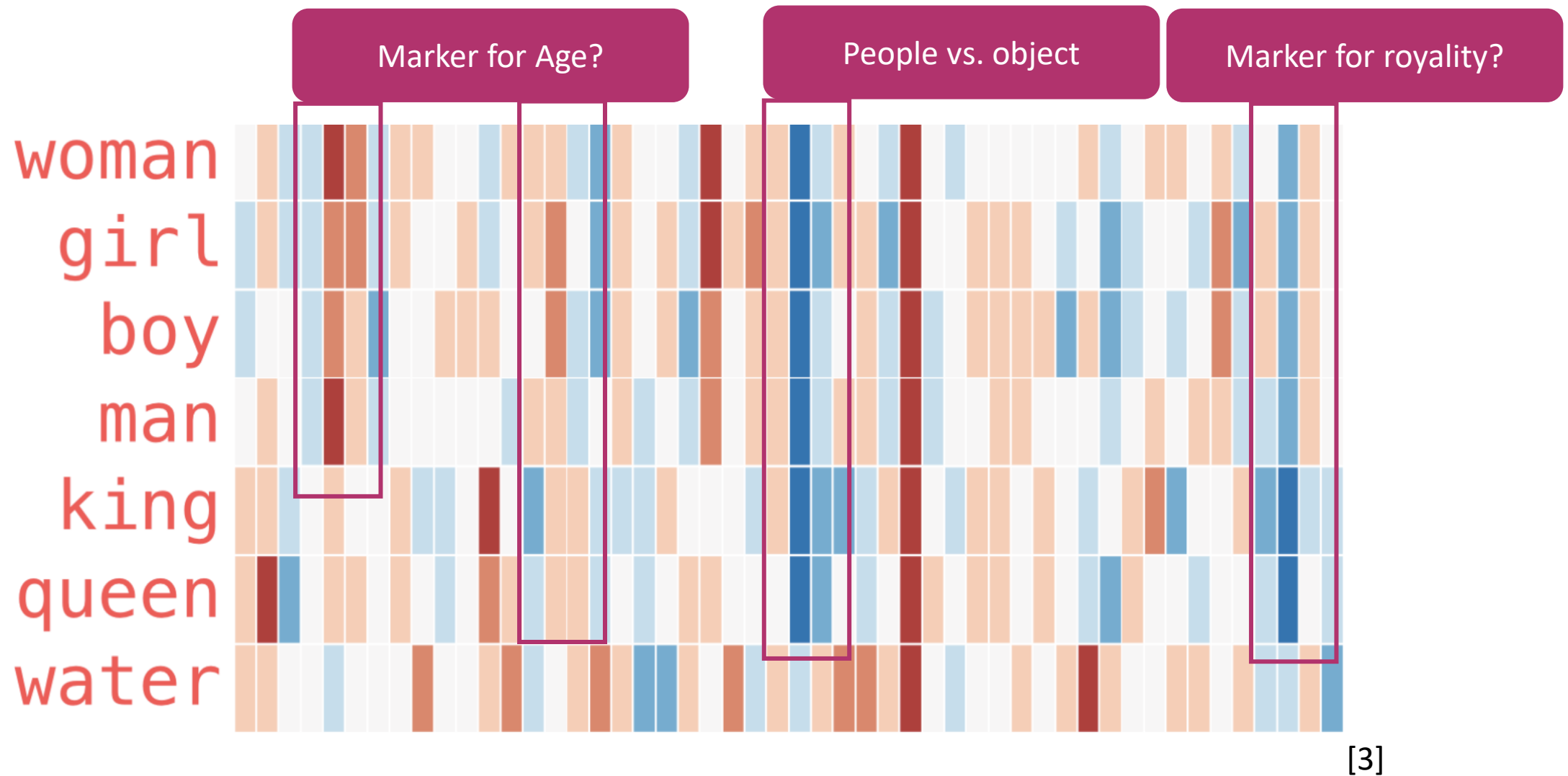




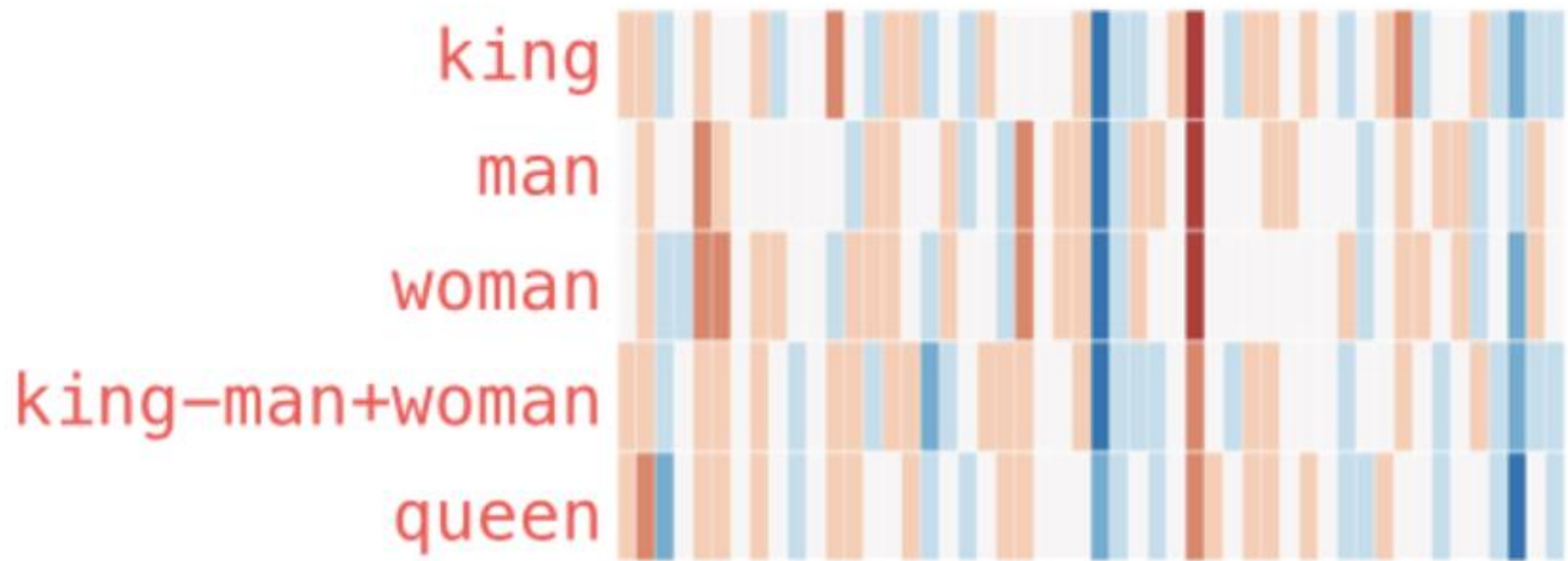
[3]







king - man + woman \approx 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
- Semantics: consider direct neighbors
- Out of vocabulary words

FastText (Bojanowski et al.)

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

BERT

- Contextual embeddings
- Semantics: consider pairs of sentences

Input
Features

Trained Language Model

Output
Prediction

Task:

Predict the next word

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]

Input
Features

Trained Language Model

Output
Prediction

Task:

Predict the next word

Thou →

shalt →

1) Look up
embeddings

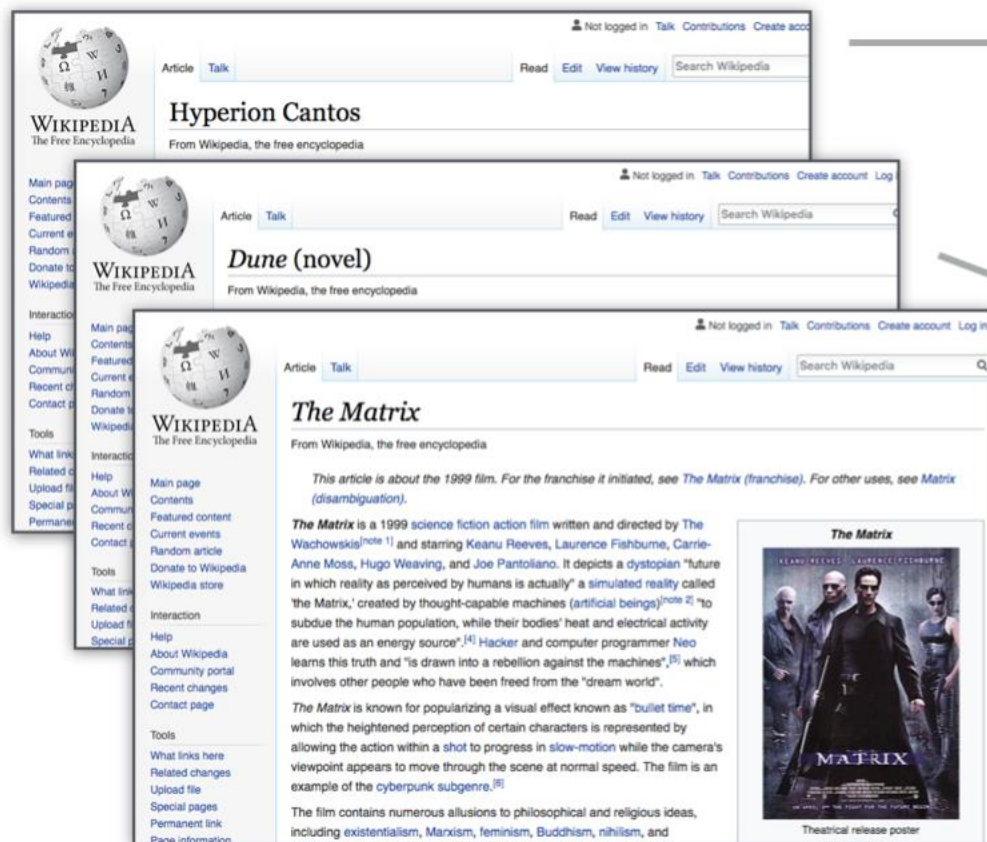
2) Calculate
prediction

3) Project
to output
vocabulary

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





The **Hyperion Cantos** is a series of science fiction novels by Dan Simmons. The title refers to the fictional planet of Hyperion, which is the setting for 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]}

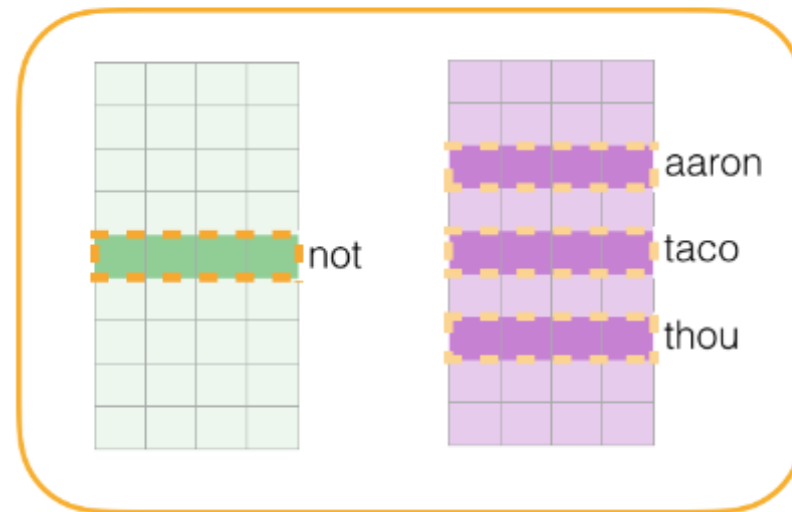
Set in the distant future amidst a feudal interstellar society in which noble houses, in control of individual planets, owe allegiance to the *Padishah Emperor*, *Dune* tells the story of young Paul Atreides, whose noble family accepts the stewardship of the populated desert wasteland of Arrakis as "spice", a drug that is an important and valuable commodity—coveted—and dangerous interactions of political factions of the empire.

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".

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

[2]

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

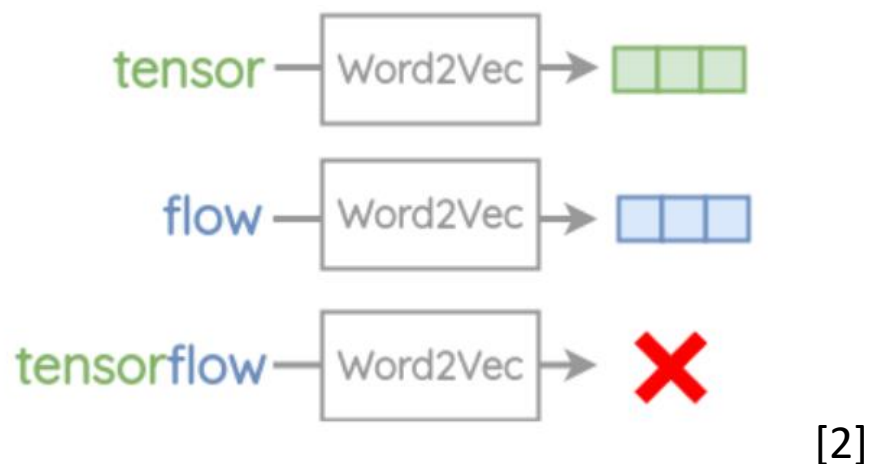
A quick brown fox jumps over the lazy dog

[3]

Predict neighbor words from central word



Limitation: Out of Vocabulary
Words cause problems



Limitation: Morphology, no
Parameter sharing

Shared radical

eat eats eaten eater eating

[2]

Use internal structure to improve embeddings

Do skip-gram embeddings and obtain subwords for central word

3-grams <eating>
 ┌───────────┐
<ea eat ati tin ing ng>
[2]

1.) Embed central word

Example Sentence

I am eating food now

[2]

Central word embedding



[2]

2.) Sampling

context words



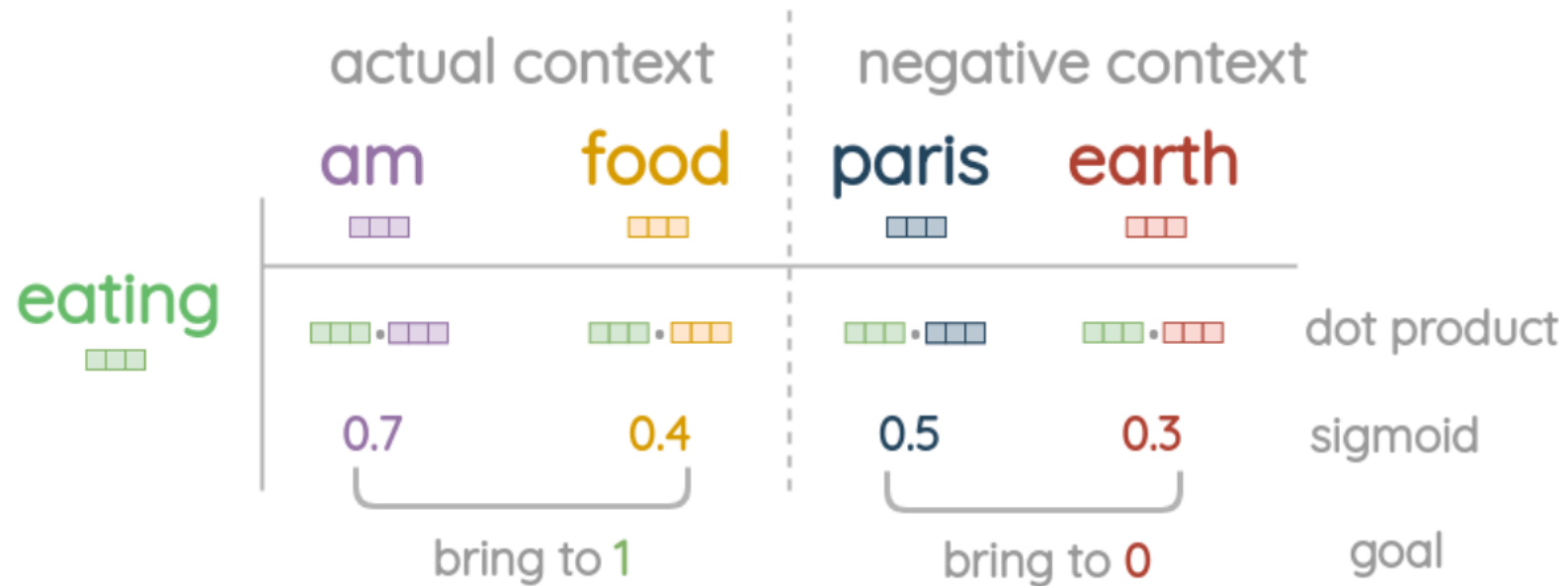
[2]

negative samples



[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 four layers (768 dimensions)

BERT embeddings: Bidirectional Encoder Representations from Transformer

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

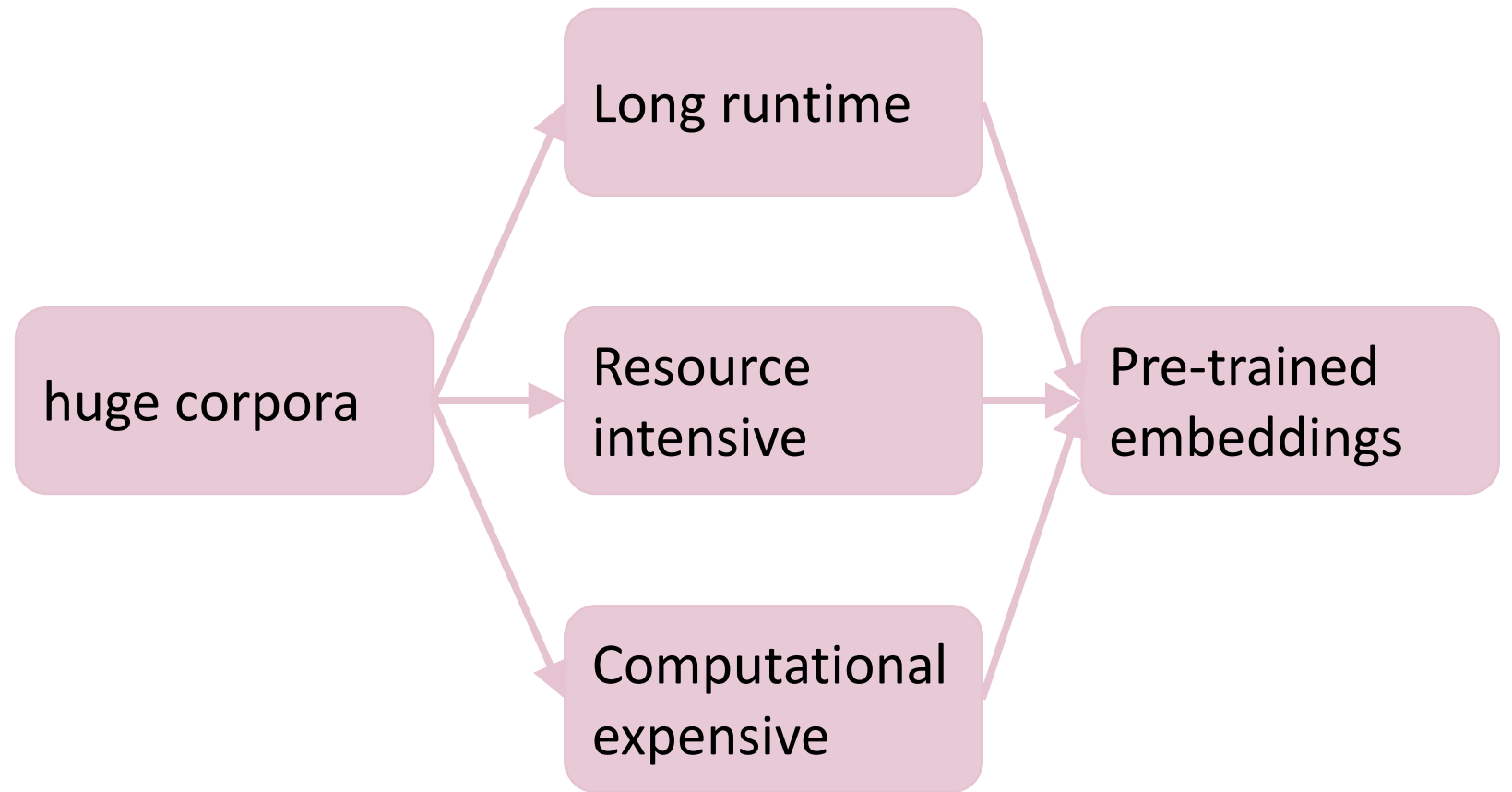
- Improved performance on syntactic analogy
- Worse performance on semantic analogy

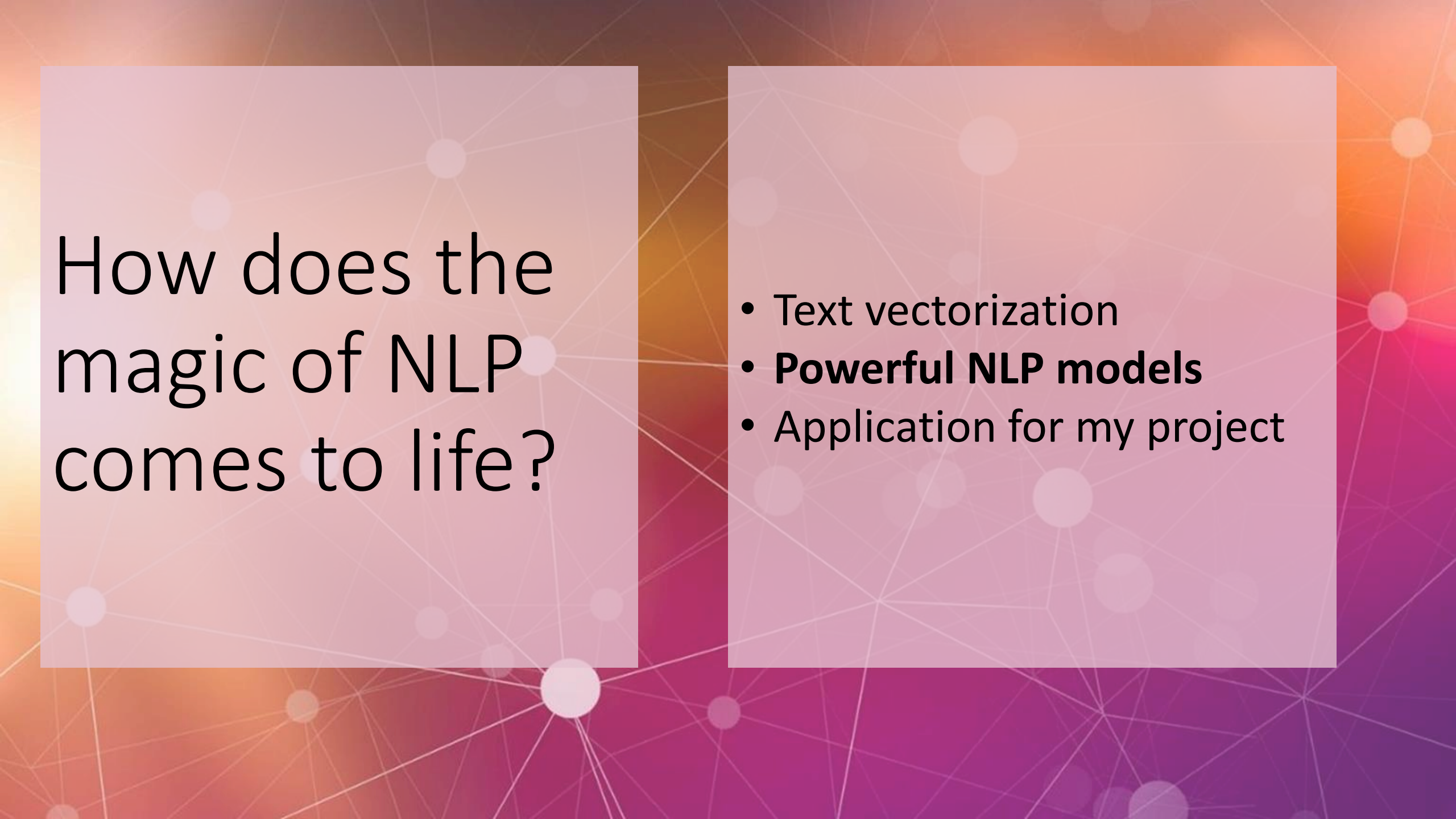
BERT

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

state-of-the-art
performance:

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



The background of the slide features a network of white dots connected by thin white lines, set against a gradient background transitioning from orange at the top to purple at the bottom. Two semi-transparent rectangular boxes are overlaid on this background.

How does the magic of NLP comes to life?

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

ELMO

Allen AI



[4]

BERT

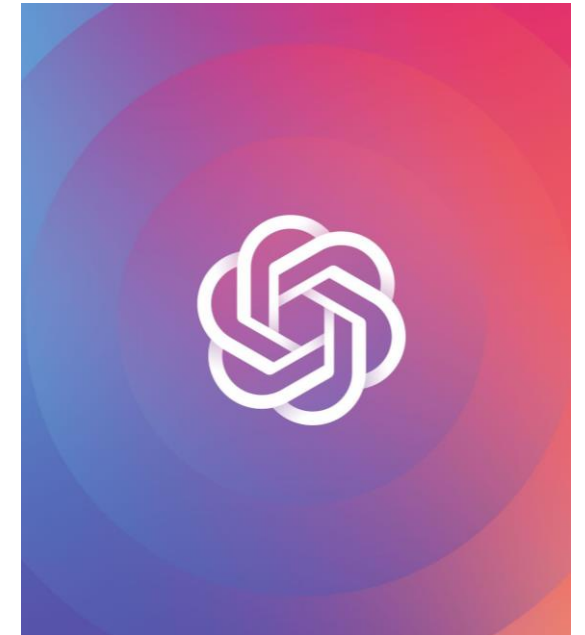
Google



[5]

Open-GPT

OpenAI



[6]



[5]

BERT



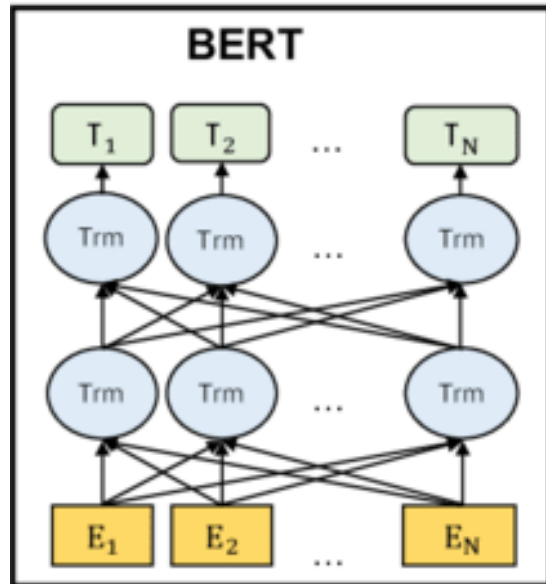
[6]

Open-GPT

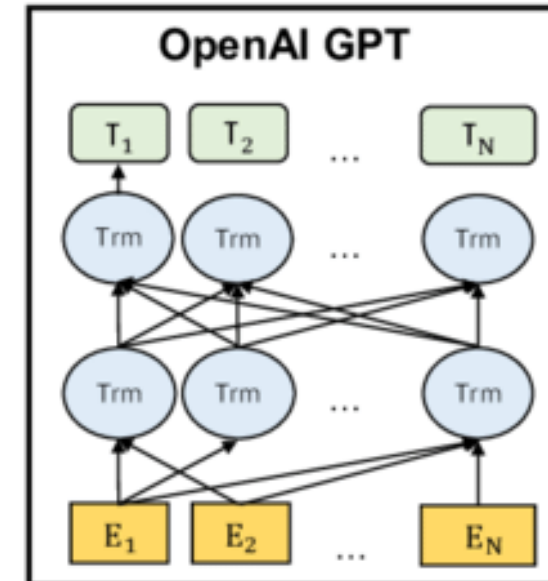
Model Architecture

- Transformers
- bidirectional

- Transformers
- Unidirectional (left)



[7]



[7]



[5]

BERT



[6]

Open-GPT

Model Architecture

- Transformers
- bidirectional

- Transformers
- Unidirectional (left)

Task Type

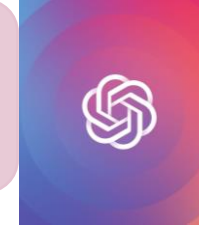
Supervised e.g. text classification

Unsupervised e.g. text generation



[5]

BERT



[6]

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]

BERT



[6]

Open-GPT

Model Architecture

- Transformers
- bidirectional

- Transformers
- Unidirectional (left)

Task Type

Supervised e.g. text classification

Unsupervised e.g. text generation

Training data

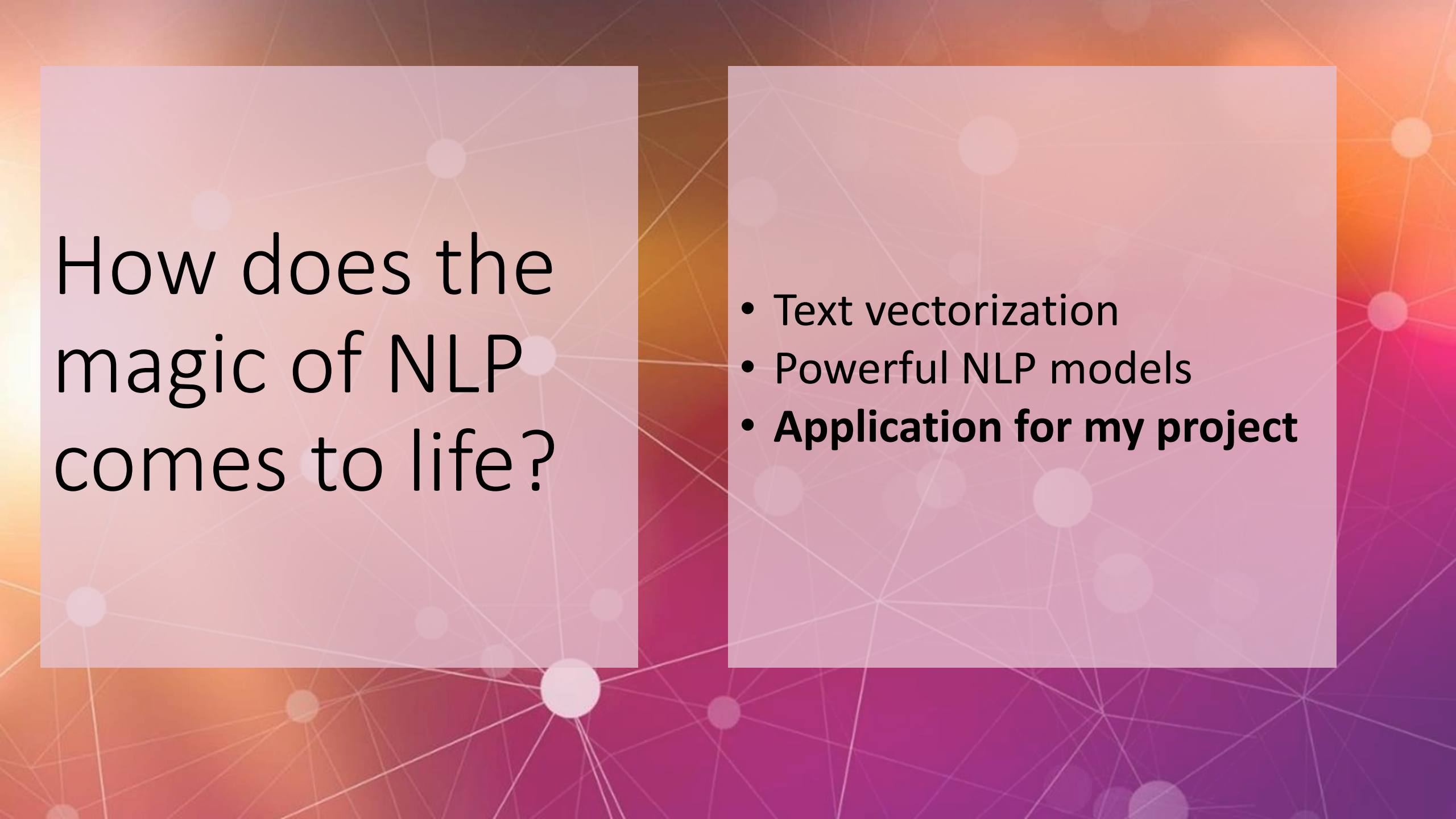
masked language modelling, next sentence prediction

Language modelling

Output

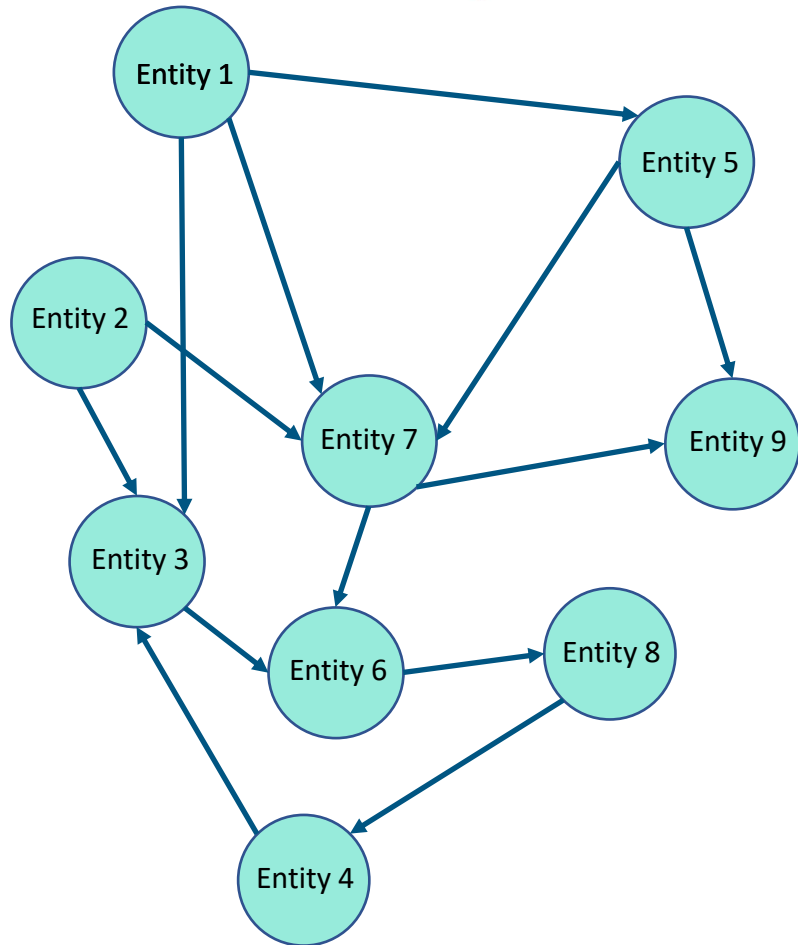
Fixed length embeddings for downstream NLP tasks

Sequence of tokens (variable length)

The background of the slide features a network diagram with white nodes and connecting lines on a gradient background transitioning from orange at the top to purple at the bottom. Two semi-transparent white rectangular boxes are overlaid on the image.

How does the magic of NLP comes to life?

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



Biomedical Knowledge Graph

Entity 1 (Subject) – Verb → Entity 2 (Object)

Knowledge Filtering and Priorization tools:

- Certainty an Author expresses
- Polarity of the verb

Previous studies have **shown** that entity 1 inhibits entity 2

Entity 1 **may** inhibit entity 2.

Certainty:

- Experimentally proven
- Proven by previous studies
- Always observed in setting

Uncertainty:

- Speculative terms
- Suggestive terms
- Usually observed but not always

TEXT CLASSIFICATION TASK:

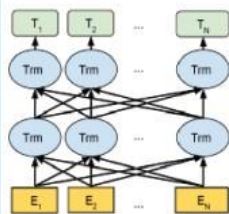
- High-quality Sentence embeddings for biomedical text
- Downstream task: Classification
- Tagged dataset for classifier training

Pre-training of BioBERT

Pre-training Corpora

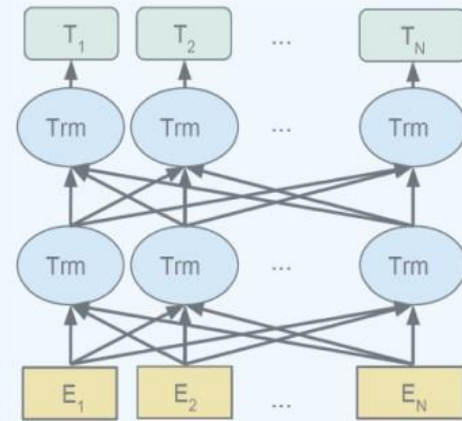
PubMed 4.5B words
PMC 13.5B words

Weight Initialization



BERT
from Devlin et al.

BioBERT Pre-training



**Pre-trained BioBERT with
biomedical domain corpora**

[9]

Document Embeddings

Application for my project

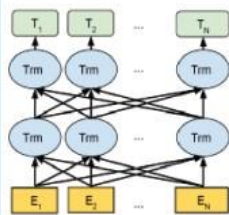
31/42

Pre-training of BioBERT

Pre-training Corpora

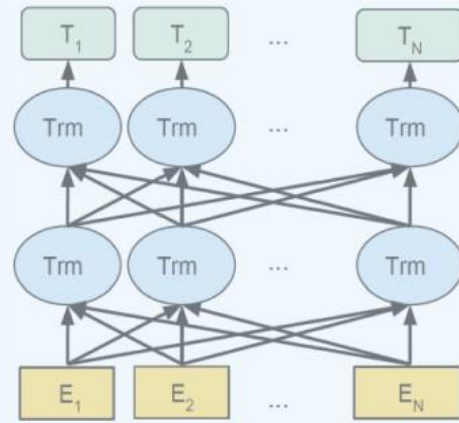
PubMed 4.5B words
PMC 13.5B words

Weight Initialization



BERT
from Devlin et al.

BioBERT Pre-training



**Pre-trained BioBERT with
biomedical domain corpora**

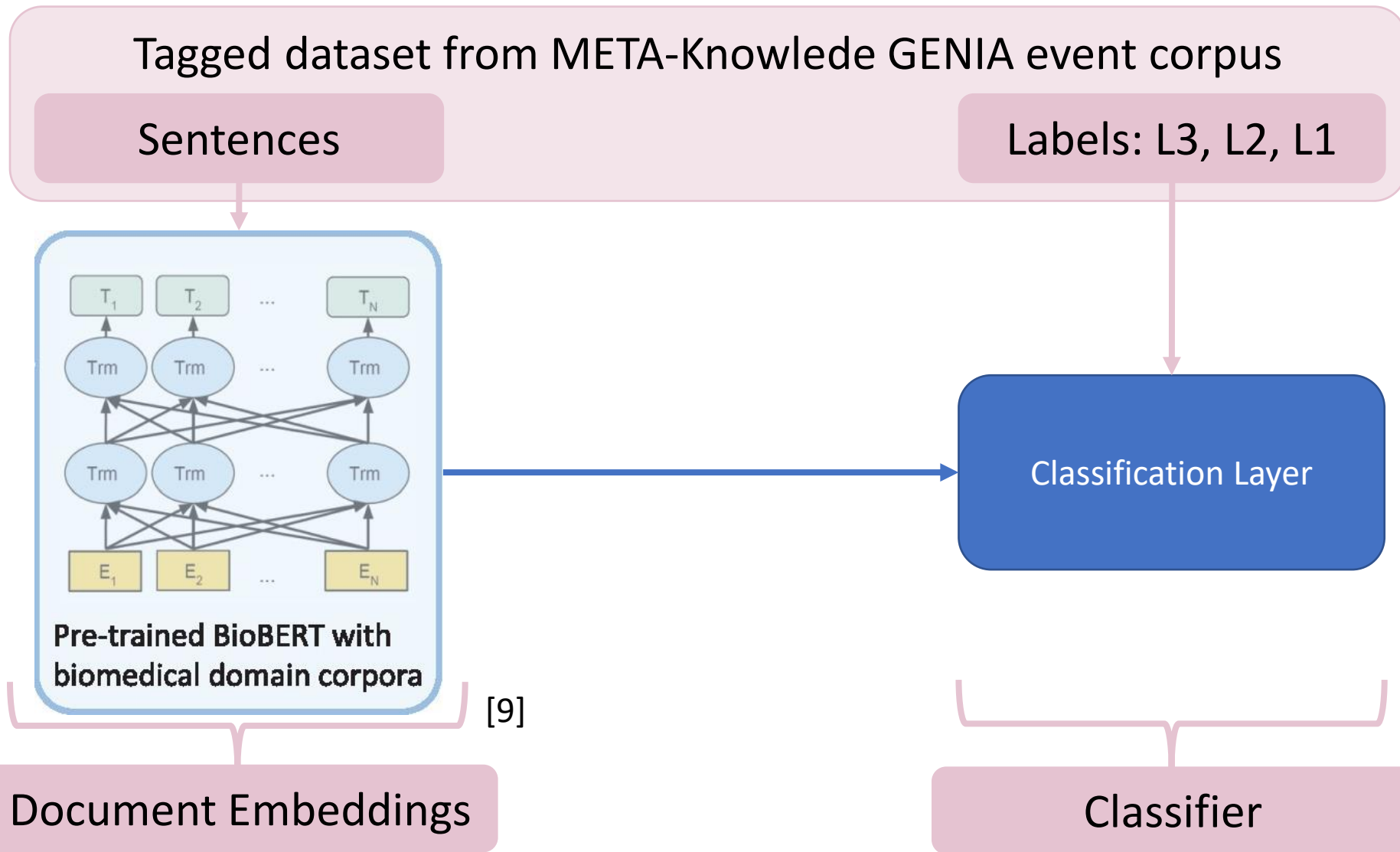
[9]

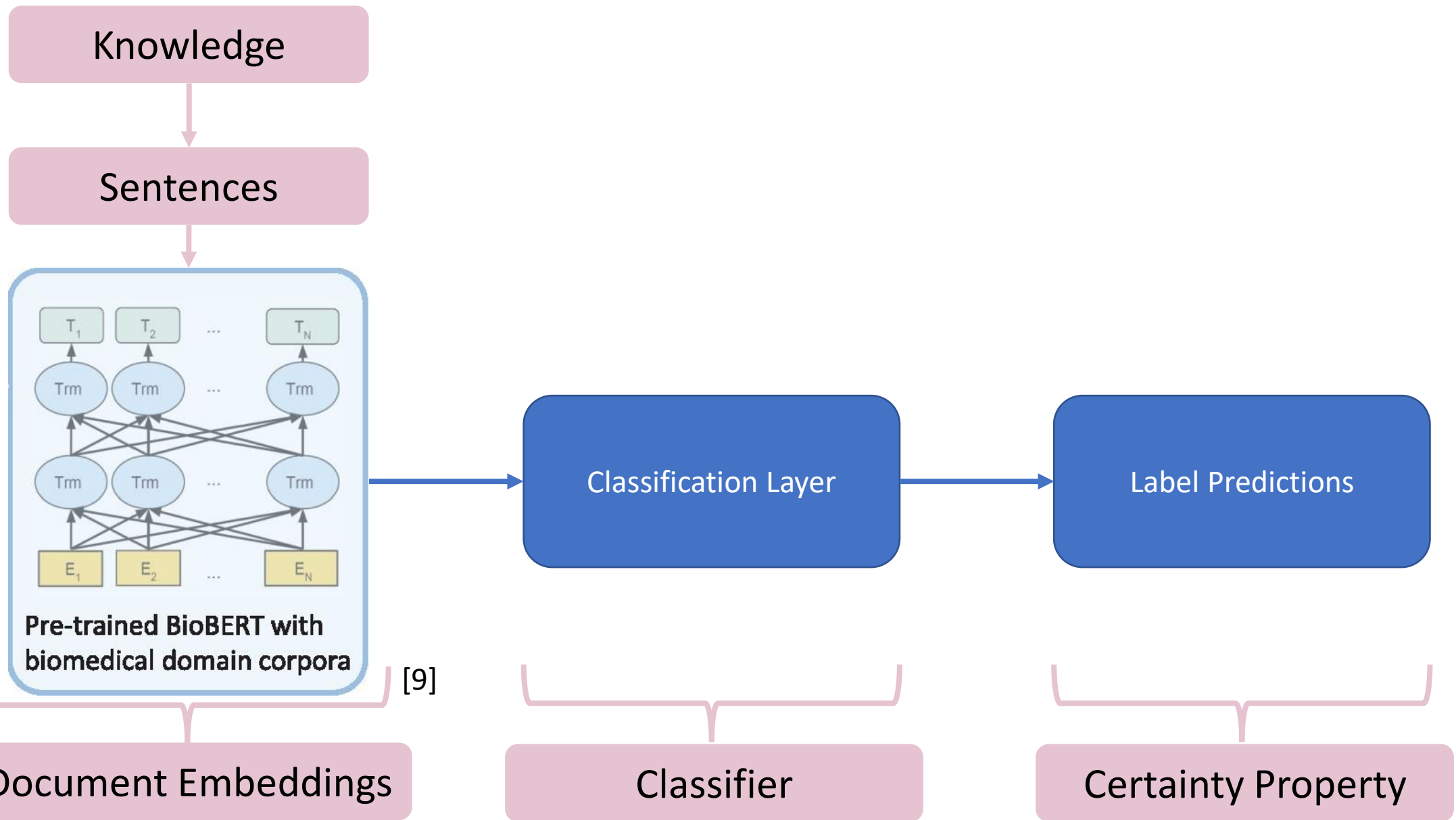
Linear Neural
Network Layer

Class Label

Document Embeddings

Classifier





- Total number of Sentences: 3.588
- Label Distribution:
 - L3: 37,21 %
 - L2: 52,15 %
 - L1: 10,65 %
- Overall Accuracy: 80,89 %

Performance Certainty Classifier trained on sampled datasets

	predicted L1	predicted L2	predicted L3	
	24	10	4	label L1
	5	80	49	label L2
	3	45	387	label L3



Bad Performance on L2



Improve Dataset:

- Merge L1 and L2
- More restrictive with L3 sampling
- Different Corpus

Improve Model:

- fine-tuning method instead of training
- Hyperparameter tuning



Main limitations: domain and language specific, state-of-the-art results difficult to achieve

SENTIMENT ANALYSIS



NEUTRAL



POSITIVE

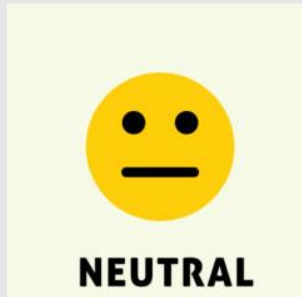


NEGATIVE

[10]

- Sentiment carrier:
 - Adjectives
 - Adverbs
- Granularity:
 - Document
 - Sentence
 - Aspects
- Domains of application
 - Opinion mining from social media posts
 - Feedback from customer reviews
- Algorithmic approach:
 - Rule-based
 - ML
 - hybrid

SENTIMENT ANALYSIS



NEUTRAL



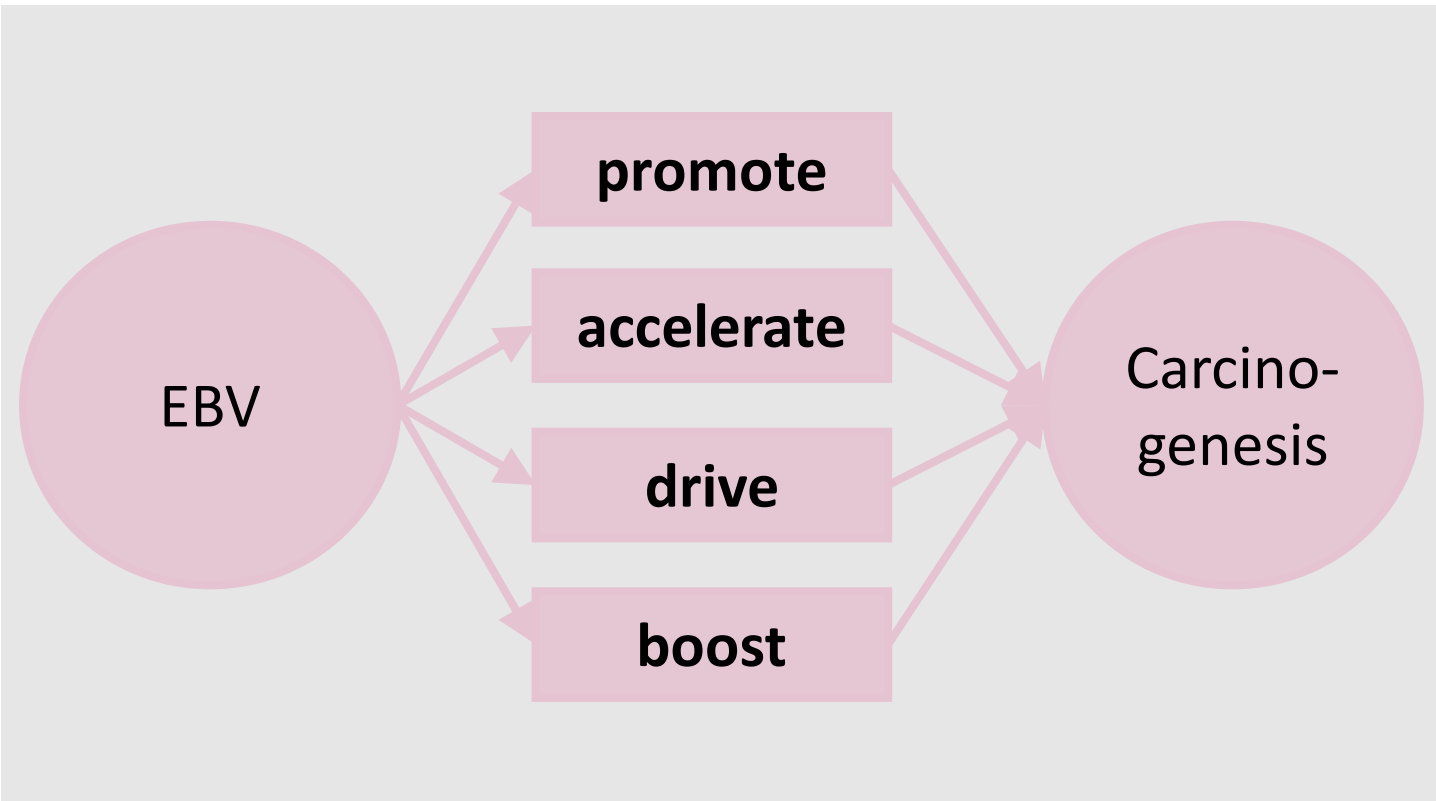
POSITIVE



NEGATIVE

- Adjective driven
- Tweets, Customer Reviews

[10]



- Verb based
- Scientific Vocabulary



Custom Polarity
tagging Algorithm



Verb Categorisation



Verb tagging Rate: 99%



Misclassifications

Application for my project



KNOWLEDGE

PubMedID	source	verb	target	polarity
31316604	foxp3 silencing	inhibit	cell growth	-

PubMedID	source	verb	target	polarity
35532158	actin assembly sites	promotes	er autophagy	+

PubMedID	source	verb	target	polarity
35498032	coenzyme q	decreased	myocardial infarct size	-

PubMedID	source	verb	target	polarity
35534864	glycolysis	accelerated	apoptosis cells	-

38/42



Improve Polarity Resource:

- ➔ Ruleset for manual Annotations
- ➔ Restrict synonym antonym Resource

Quality of Resource is key for correct tagging

Domain Transfer is challenging

Concluding Remarks

- **Limitations**
- My favourite NLP tools



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



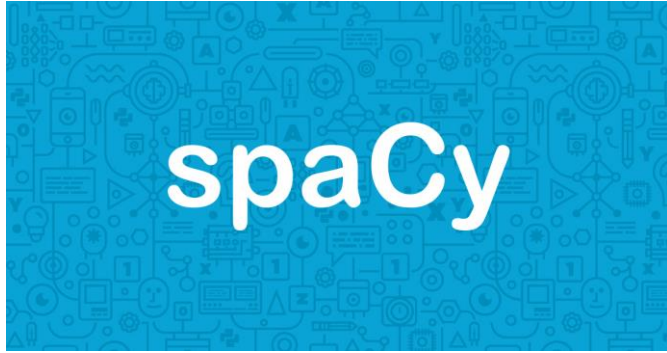
Work with the Resources given:
Domain transfer challenging, suboptimal results



Language is not precise: exceptions to be handled

Concluding Remarks

- Limitations
- **My favourite NLP tools**



[11]



spaCy

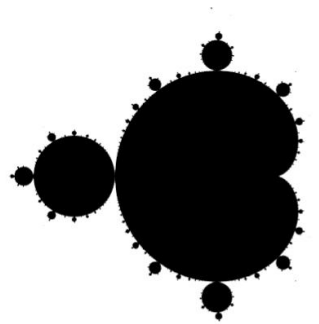
[12]

Outstanding performance for Lemmatization
Biomedical sequence tagging with scientific models

flair

[13]

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



TextBlob



[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://amitnness.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>