1 2 9 0

Universidade de Coimbra
Faculdade de Ciências e Tecnologia
Departamento de Engenharia Informática

Human-Centered Artificial Intelligence

Master in Data Science and Engineering

Human-AI Communication

Text Mining

Hugo Gonçalo Oliveira

hroliv@dei.uc.pt

Overview

1. Introduction

2. Text Representation

3. Text Mining Tasks

4. Language Modelling

Introduction

- Human language is a natural way for **humans to communicate**
  - Used for **encoding** knowledge and **reasoning**
  - Large digital repositories of documents **written in human language**
    - Digital libraries
    - The Web
    - Newswire services

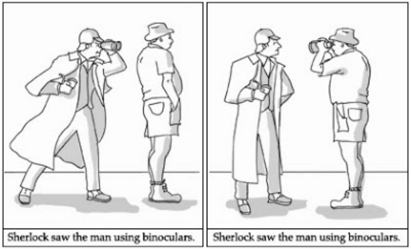
Text Mining

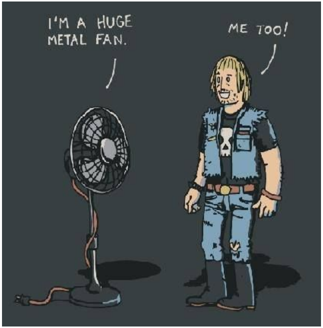
- Applying Data Mining techniques to textual data!

Natural Language

Processing Human Language offers several challenges...

The GW4-40 is a smartwatch with an elegant design, extremely comfortable to use. It has 1.5GB of RAM and 16GB of internal memory. It weights only 26 grams, has a 1.1" screen and a 247mAh battery, with an estimated autonomy of 40 hours. On the other hand, GW4C-46 has the same memory as the GW4-40, but a 1.4" screen, with twice the weight . Despite the 361mAh battery, autonomy is similar . Both devices come with Wear OS operative system preinstalled.

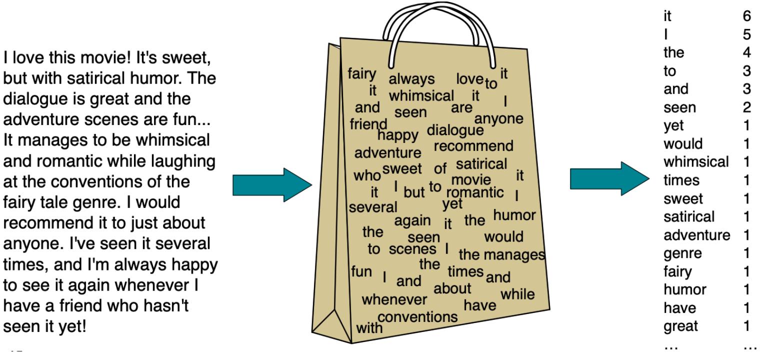

Sherlock saw the man using binoculars.
Sherlock saw the man using binoculars.



Text Representation

- How to apply Data Mining to Text?

- *Bag-of-Words*: documents can be represented by their **terms**
  - Each new **term** = a **feature**
  - **Frequency** of the term = **value** of the feature



I love this movie! It's sweet, but with satirical humor. The dialogue is great and the adventure scenes are fun... It manages to be whimsical and romantic while laughing at the conventions of the fairy tale genre. I would recommend it to just about anyone. I've seen it several times, and I'm always happy to see it again whenever I have a friend who hasn't seen it yet!

it	6
I	5
the	4
to	3
and	3
seen	2
yet	1
would	1
whimsical	1
times	1
sweet	1
satirical	1
adventure	1
genre	1
fairy	1
humor	1
have	1
great	1
...	...

From: [Jurafsky and Martin, 2024]

Text Representation

Traditional Preprocessing

Sentence

User warning: a new threat is affecting millions of users.

Tokenization: split text into smaller meaningful units

User warning : a new threat is affecting millions of users .

Stopword removal: ignore tokens that contribute little to meaning

User warning new threat affecting millions users

Stemming: consolidate variations of the same word

user warning new threat affect million user

Text Representation

Vector-space model

Corpus

D	Content
d1	`social web technologies`
d2	`web programming`
d3	`languages and technologies for the web`
d4	`web applications`
d5	`web applications development`
d6	`databases and web applications`
d7	`semantic web and social web`
d8	`web design`

Term-document matrix

Vocabulary	d1	d2	d3	d4	d5	d6	d7	d8
technologies	1	0	1	0	0	0	0	0
web	1	1	1	1	1	1	2	1
social	1	0	0	0	0	0	1	0
programming	0	1	0	0	0	0	0	0
languages	0	0	1	0	0	0	0	0
applications	0	0	0	1	1	1	0	0
development	0	0	0	0	1	0	0	0
databases	0	0	0	0	0	1	0	0
semantic	0	0	0	0	0	0	1	0
design	0	0	0	0	0	0	0	1

- $\vec{d_1} = [1, 1, 1, 0, 0, 0, 0, 0]$
- $\vec{d_2} = [0, 1, 0, 1, 0, 0, 0, 0]$

...

- $similarity(d_1, d_2) = cos(\vec{d_1}, \vec{d_2}) = \frac{1}{\sqrt{6}} = \frac{\sqrt{6}}{6}$

Text Representation

TF-IDF

- Are **all** the features **equally** important?
- Some tokens are more **frequent** than the others...
  - Frequent in document → more relevant for the document!
  - Frequent in corpus → less relevant for any document! (e.g., stopwords)

Term Frequency (TF)

$$tf_{i,d} = \frac{f_{i,d}}{\sum_{j=0}^{|d|} f_{j,d}}$$

tf_i frequency of term i in the document

$f_{i,d}$ number of times term i occurs in document d

$|d|$ number of distinct terms in document d

Inverse Document Frequency (IDF)

$$idf_i = log \frac{N}{n_i}$$

idf_i inverse document frequency of term i in the corpus

N total number of documents in the collection

n_i number of documents where term i occurs

$$weight(i, d) = tf_{i,d} \times idf_i$$

Text Representation

TF-IDF

D	Content
d1	`social web technologies`
d2	`web programming`
d3	`languages and technologies for the web`
d4	`web applications`
d5	`web applications development`
d6	`databases and web applications`
d7	`semantic web and social web`
d8	`web design`

$$df('web')^* = \frac{8}{8} = 1$$
$$df('semantic')^* = \frac{1}{8}$$
$$df('applications')^* = \frac{3}{8}$$

* ignoring log in idf

	d1	d2	d3	d4	d5	d6	d7	d8
technologies	4/3	0	4/3	0	0	0	0	0
web	1/3	0.5	1/3	0.5	1/3	1/3	0.5	0.5
social	4/3	0	0	0	0	0	2	0
programming	0	4	0	0	0	0	0	0
languages	0	0	8/3	0	0	0	0	0
applications	0	0	0	4/3	8/9	8/9	0	0
development	0	0	0	0	8/3	0	0	0
databases	0	0	0	0	0	8/3	0	0
semantic	0	0	0	0	0	0	2	0
design	0	0	0	0	0	0	0	4

d5

$$tf('web') = \frac{1}{3}$$
$$tf('applications') = \frac{1}{3}$$
$$tf.idf('web') = \frac{1}{3} \times 1 = \frac{1}{3}$$
$$tf.idf('applications') = \frac{1}{3} \times \frac{8}{3} = \frac{8}{9}$$

d7

$$tf('web') = \frac{2}{4} = 0.5$$
$$tf('semantic') = \frac{1}{4}$$
$$tf.idf('web') = \frac{2}{4} \times 1 = 0.5$$
$$tf.idf('semantic') = \frac{1}{4} \times \frac{8}{1} = 2$$

Word Vectors

Distributional hypothesis [Harris, 1970]

You shall know a word by the company it keeps [Firth, 1957]

→ *Words that occur in the same contexts tend to convey similar meanings.*

What is a 'steltnah'?

- *tasty steltnah*
- *steltnah sprinkled with cinnamon*
- *crispy steltnah*
- *coffee and a steltnah*
- *a steltnah with a smoother filling*
- *best steltnah in Portugal*

Word Vectors

Contexts – 5-word window

… dig a [hole. The	**car**	drove away] leaving behind …
… to directly [drive the	**car**	wheel angle] 3. Force …
… celebrity status, [drove fast	**cars**	and partied] with some …
… but there [are police	**cars**	that chase] you. Each …
… world of [money, fast	**cars**	and excitement] and, under …
… to pet [the family's	**cat**	and dog,] who tended …
… and then[wanted a	**cat**	to eat] the many …
… murmur is [detectable. The	**cat**	often eats] and drinks …
… behaviour of [a domestic	**cat**	playing with] a caught …
… have never[seen a	**cat**	eat so] little and …
… bank, children [playing with	**dogs**	and a] man leading …
… sure you [encourage your	**dog**	to play] appropriate chase …
… Truth, Lord: [yet the	**dogs**	eat of] the crumbs …
… vegetable material [and enzymes.	**Dogs**	also eat] fruit, berries …
… hubby once [ate the	**dog**	food and] asked for …
… were back [at the	**van**	and drove] down to …
… go down [as the	**van**	droveoff.] Ashe …
… heavy objects, [driving transit	**vans**	, wiring plugs] and talking …
… of the [fast food	**van**	being located] outside their …
… each of [the six	**van**	wheels , and] also under …

Example from http://lope.linguistics.ntu.edu.tw/courses/nlp/slides/Lenci.tutorial.gwc2014

Word Vectors

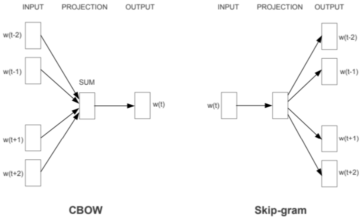
Term-term matrix

	dog	drive	eat	fast	play	...	the	wheel
car	0	3	0	2	0	...	2	1
cat	1	0	3	0	1	...	2	0
dog	0	0	3	0	2	...	2	0
van	0	3	0	1	0	...	3	1

- Co-occurrence matrices can be very **large** ...
  - $|V| \times |V|$, where $|V|$ is the vocabulary size
  - **Sparse** vectors → many cells are zero!
- Dimensionality-reduction techniques, towards
  - **Smaller, dense** vectors
  - **Fixed-dimension** vectors, regardless of vocabulary size

Word Embeddings

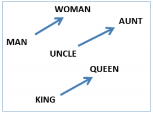
- **Word embedding**: efficient methods for learning **dense, lower-dimension word vectors** from large corpus
  - Word2Vec [Mikolov et al., 2013]
  - GloVe [Pennington et al., 2014]
- Two methods for learning Word2Vec models

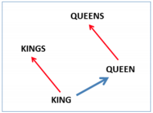


CBOW

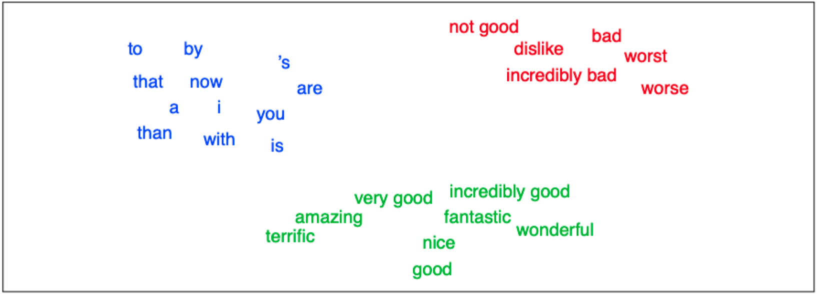
Skip-gram

- Known for keeping **linguistic regularities**
  - $\vec{king} - \vec{man} + \vec{woman} \approx \vec{queen}$





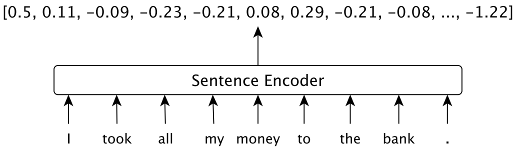
Words in a Vector Space



[Jurafsky and Martin, 2024]

Document / Sentence Embeddings

- Word embeddings represent words...
  - One vector for each word, capturing all of its possible senses
  - How to embed longer sequences?
- Document / sentence embedding
  - doc2vec [Le and Mikolov, 2014]: documents represented by the **average word2vec vectors of their words**
  - Models trained specifically for sentence embedding:
    - Sentence Transformers[1] [Reimers and Gurevych, 2019]



[1] https://sbert.net

Sentence-Level tasks

- Tasks where sentence embeddings are typically trained

Semantic Textual Similarity

Sim	Interpretation	Example
5	Completely equivalent (same meaning)	The bird is bathing in the sink. Birdie is washing itself in the water basin.
4	Mostly equivalent, but some unimportant details differ	Two boys on a couch are playing video games. Two boys are playing a video game.
3	Roughly equivalent, but some important information differs/missing	John said he is considered a witness but not a suspect. "He is not a suspect anymore." John said.
2	Not equivalent, but share some details	They flew out of the nest in groups. They flew into the nest together.
1	Not equivalent, but are on the same topic	The woman is playing the violin. The young lady enjoys listening to the guitar.
0	Completely dissimilar	The black dog is running through the snow. A race car driver is driving his car through the mud.

`http://alt.qcri.org/semeval2015/task2/index.php?id=semantic-textual-similarity-for-english`

Natural Language Inference

Label	Premise	Hypothesis
Neutral	The Old One always comforted Ca'daan, except today.	Ca'daan knew the Old One very well.
Entailment	At the other end of Pennsylvania Avenue, people began to line up for a White House tour.	People formed a line at the end of Pennsylvania Avenue.
Contradiction	Vrenna and I both fought him and he nearly took us.	Neither Vrenna nor myself have ever fought him.

`https://cims.nyu.edu/~sbowman/multinli/`

Bag-of-Words *vs* Embeddings

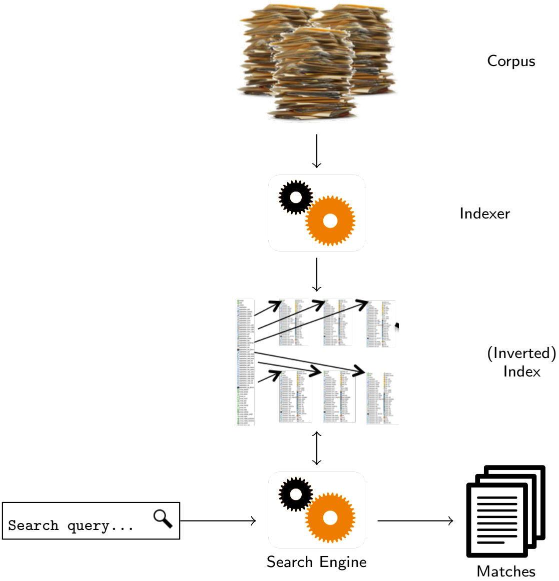
Bag-of-Words

- Simple and easy to implement
- Human-interpretable

Embeddings

- More efficient representations
  - Fixed-size dense vectors
- Better generalisation
  - incl. similarity and synonymy
- Generally better performance

Information Retrieval



- **Query:**

  `nice web applications`

- **Vector space model...**
  - $\vec{q} = [0, 0.5, 0, 0, 0, 1.5, 0, 0, 0, 0]$
  - $sim(d_i, q) = cos(\vec{d_i}, \vec{q})$

Topic Modelling

Uncovering abstract themes in a collection of documents...

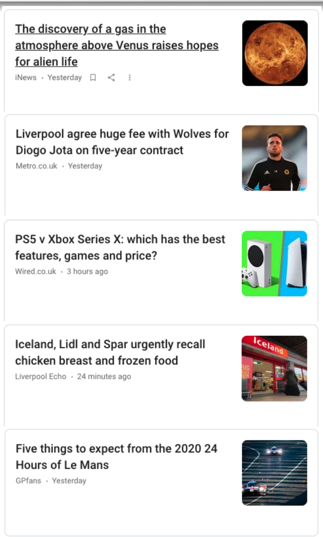
Song lyrics dataset...

- What topics are present in the lyrics of each artist?
- What artists cover similar topics?

Topic	Signature
1	time, back, home, night, feel, gonna, face, life, wait, miss, things, make, ...
2	bitch, fuck, back, ass, shit, make, man, love, bad, pussy, girls, big, ...
3	love, baby, yeah, time, girl, make, gonna, hey, wanna, world, good, heart...
4	man, night, day, sun, lie, heart, sweet, love, sea, light, true, sky, ...
5	baby, yeah, girl, wanna, feel, work, boy, make, gimme, ooh, night, music...
..	...

Topic									...
1	0.115	0.777	0.111	0.218	0.309	0.123	0.069	0.143	...
2	0.011	0.012	0.012	0.012	0.357	0.005	0.004	0.011	...
3	0.714	0.123	0.281	0.167	0.077	0.720	0.832	0.301	...
4	0.008	0.041	0.011	0.490	0.050	0.068	0.072	0.008	...
5	0.095	0.025	0.561	0.007	0.055	0.023	0.007	0.517	...
..	...	...	...	...	...	...	...	...	...

Text Classification



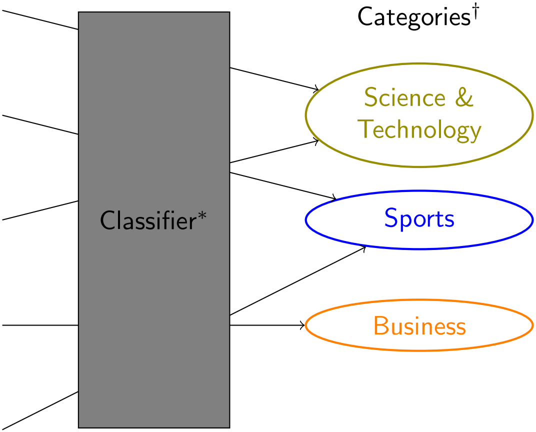
The discovery of a gas in the atmosphere above Venus raises hopes for alien life
iNews · Yesterday

Liverpool agree huge fee with Wolves for Diogo Jota on five-year contract
Metro.co.uk · Yesterday

PS5 v Xbox Series X: which has the best features, games and price?
Wired.co.uk · 3 hours ago

Iceland, Lidl and Spar urgently recall chicken breast and frozen food
Liverpool Echo · 24 minutes ago

Five things to expect from the 2020 24 Hours of Le Mans
GPfans · Yesterday



Classifier*

Categories†

Science & Technology

Sports

Business

* Generally a supervised model (e.g., Naive Bayes, SVM, Transformer)

† Any human-defined classes, e.g.: Positive, Negative, Neutral

Probabilistic Language Models

- What is the **probability** of ...
  - ... a sentence?

    $$P(S = \text{“I study artificial intelligence}'') = P(I, study, artificial, intelligence)$$

  - ... an upcoming word?

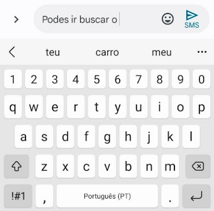
    $$P(intelligence|I, study, artificial)$$

- **Markov assumption**
  - Unigram model: $P(w_1, w_2, w_3, ..., w_n) = \prod_i P(w_i)$
  - Bigram model: $P(w_1, w_2, w_3, ..., w_n) = \prod_i P(w_i|w_{i-1})$
  - Trigram model: $P(w_1, w_2, w_3, ..., w_n) = \prod_i P(w_i|w_{i-1}, w_{i-2})$
  - ...
- Maximum Likelihood Estimate: $P(w_i|w_{i-1}) = \frac{count(w_{i-1}, w_i)}{count(w_{i-1})}$

Probabilistic Language Models

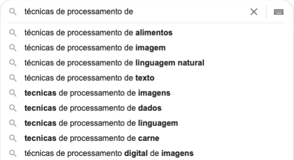
Applications

- Machine Translation
  - PT: *com grande prazer*
  - EN: $P(\text{with great pleasure}) > P(\text{with big pleasure})$
- Spell Correction
  - $P(\text{about fifteen minutes from}) > P(\text{about fifteen minuets from})$
- Speech Recognition
  - $P(\text{I saw a van}) >> P(\text{eyes awe of an})$
- Text Prediction



Podes ir buscar o

Google



técnicas de processamento de

- técnicas de processamento de **alimentos**
- técnicas de processamento de **imagem**
- técnicas de processamento de **linguagem natural**
- técnicas de processamento de **texto**
- **tecnicas** de processamento de **imagens**
- **tecnicas** de processamento de **dados**
- **tecnicas** de processamento de **linguagem**
- **tecnicas** de processamento de **carne**
- técnicas de processamento **digital** de **imagens**

Probabilistic Language Models

Limitations

- Towards more fluency, N needs to increase!
  - Need to deal with sequences not appearing in the training data
  - More and more data required
  - Much memory necessary for storing the probabilities
  - May still not be enough due to **long range dependencies**

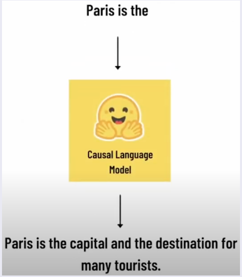
- *The* **man** *seating in the shadow of the big orange tree near the city centre lost* **his** *hat.*
- *The* **woman** *seating in the shadow of the big orange tree near the city centre lost* **her** *hat.*

- For some applications, still a good approximation!

Large Language Models

- Today, (Large) Language Models are used in a broader range of tasks
  - Based on the Transformer architecture [Vaswani et al., 2017]
  - May have hundreds of billions of parameters
  - Pretrained on large quantities of text
  - Can be fine-tuned on specific tasks, such as following instructions
- Two types of language modelling

Causal Language Modelling

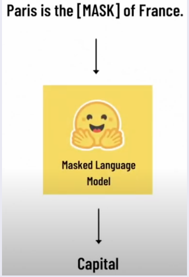


Paris is the

Paris is the capital and the destination for many tourists.

https://www.youtube.com/watch?v=Vpjb1lu0MDk

Masked Language Modelling



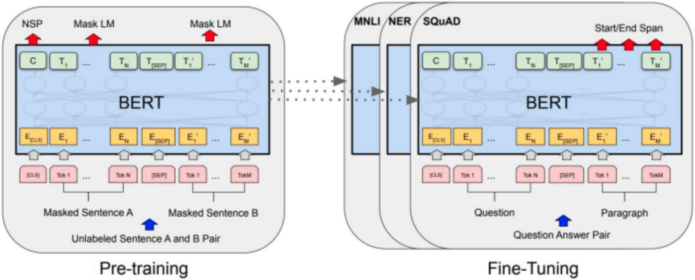
Paris is the [MASK] of France.

Capital

https://www.youtube.com/watch?v=mqElG5QJWUg&t=1s

BERT [Devlin et al., 2019]

- **B**idirectional **E**ncoder **R**epresentations from **T**ransformers
- Pre-trained in two tasks:
  - Masked Language Modelling, Next Sentence Prediction
- **Fine-tunable** in other tasks...
  - Sentence Classification, Natural Language Inference, Semantic Textual Similarity, Question Answering, ...



GPT

Generative **P**re-**T**raining

2018 GPT, 117M parameters, trained in 8M web pages

- Fluent text generation (causal language model)

2019 GPT-2 [Radford et al., 2019], 1.5B params, trained in 40GB of text

- *Unsupervised learner*, i.e., generates answers for questions in the topics it was trained on
- Opened to the community ≈1 year after release

2020 GPT-3 [Brown et al., 2020], 175B params, trained in 570GB of text

- *Few-shot learner*, i.e., can learn (virtually) any task given a few examples
- Accessible through an API (paid after certain usage)

2023 GPT-4 [OpenAI, 2023], ?1.76T? params (many unknown details...)

- Multimodal processing (text, image)
- Accessible through a paid API

* ChatGPT

- GPT model trained for following instructions / chatting

Prompt Engineering

- Larger instruction-tuned models may perform several tasks
- Many Text Mining tasks can be cast as a text generation task!

https://www.promptingguide.ai/

Prompt Engineering is a relatively new discipline for developing and optimizing prompts to efficiently use language models for a wide variety of applications and research topics.

Zero-shot prompting

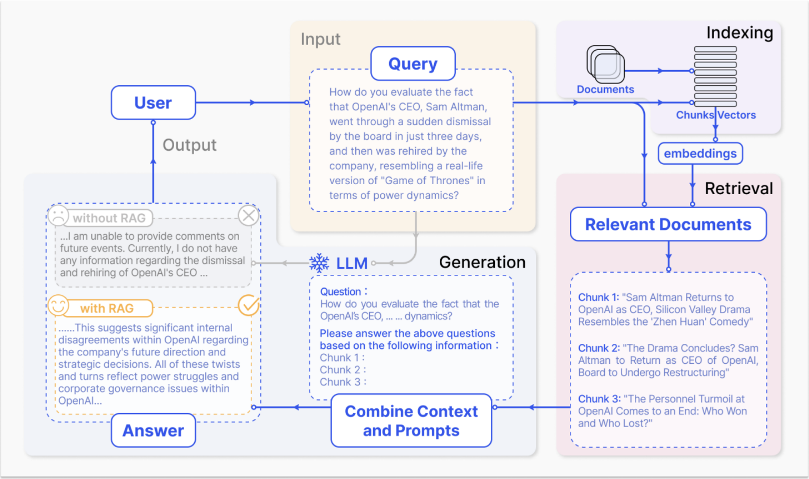
```
Classify the sentiment transmitted by the following
text as:  Positive, Negative, or Neutral.
Text:  I like chocolate.
Sentiment:
```

Few-shot prompting

```
Classify the sentiment transmitted by the following
text as:  Positive, Negative, or Neutral.
Text:  I saw a squirrel.
Sentiment:  Neutral
Text:  She doesn’t like soup.
Sentiment:  Negative
Text:  The meal that did not go well.
Sentiment:  Negative
Text:  The service was excellent!
Sentiment:  Positive
Text:  They opted for a classic decoration.
Sentiment:  Neutral
Text:  I like chocolate.
Sentiment:
```

Retrieval Augmented Generation

- Retrieval-Augmented Generation (**RAG**)
  - Information Retrieval + Large Language Model



How do you evaluate the fact that OpenAI's CEO, Sam Altman, went through a sudden dismissal by the board in just three days, and then was rehired by the company, resembling a real-life version of "Game of Thrones" in terms of power dynamics?

...I am unable to provide comments on future events. Currently, I do not have any information regarding the dismissal and rehiring of OpenAI's CEO ...

......This suggests significant internal disagreements within OpenAI regarding the company's future direction and strategic decisions. All of these twists and turns reflect power struggles and corporate governance issues within OpenAI...

Question：
How do you evaluate the fact that the OpenAI's CEO, ... dynamics?

Please answer the above questions based on the following information：
Chunk 1 :
Chunk 2 :
Chunk 3 :

Chunk 1: "Sam Altman Returns to OpenAI as CEO, Silicon Valley Drama Resembles the 'Zhen Huan' Comedy"

Chunk 2: "The Drama Concludes? Sam Altman to Return as CEO of OpenAI, Board to Undergo Restructuring"

Chunk 3: "The Personnel Turmoil at OpenAI Comes to an End: Who Won and Who Lost?"

[Gao et al., 2024]

Bibliography I

Aggarwal, C. C. (2015).
Data mining: the textbook.
Springer.

Brown, T., Mann, B., Ryder, N., Subbiah, M., Kaplan, J. D., Dhariwal, P., Neelakantan, A., Shyam, P., Sastry, G., Askell, A., Agarwal, S., Herbert-Voss, A., Krueger, G., Henighan, T., Child, R., Ramesh, A., Ziegler, D., Wu, J., Winter, C., Hesse, C., Chen, M., Sigler, E., Litwin, M., Gray, S., Chess, B., Clark, J., Berner, C., McCandlish, S., Radford, A., Sutskever, I., and Amodei, D. (2020).
Language models are few-shot learners.
In *Advances in Neural Information Processing Systems*, volume 33, pages 1877–1901. Curran Associates, Inc.

Devlin, J., Chang, M.-W., Lee, K., and Toutanova, K. (2019).
BERT: Pre-training of deep bidirectional transformers for language understanding.
In *Procs. of 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)*, pages 4171–4186, Minneapolis, Minnesota. ACL Press.

Firth, J. R. (1957).
Studies in linguistic analysis.
Wiley-Blackwell.

Gao, Y., Xiong, Y., Gao, X., Jia, K., Pan, J., Bi, Y., Dai, Y., Sun, J., Wang, M., and Wang, H. (2024).
Retrieval-augmented generation for large language models: A survey.

Harris, Z. (1970).
Distributional structure.
In *Papers in Structural and Transformational Linguistics*, pages 775–794. D. Reidel Publishing Company, Dordrecht, Holland.

Bibliography II

Jurafsky, D. and Martin, J. H. (2024).
Speech and Language Processing: An Introduction to Natural Language Processing, Computational Linguistics and Speech Recognition.
Prentice Hall, Pearson Education International, 3rd edition.
Online manuscript released August 20, 2024: `https://web.stanford.edu/~jurafsky/slp3/`.

Le, Q. and Mikolov, T. (2014).
Distributed representations of sentences and documents.
In *Proceedings of the 31st International Conference on International Conference on Machine Learning - Volume 32*, ICML'14, pages II–1188–II–1196. JMLR.org.

Mikolov, T., Chen, K., Corrado, G., and Dean, J. (2013).
Efficient estimation of word representations in vector space.
In *Proceedings of the Workshop track of the International Conference on Learning Representations (ICLR)*, Scottsdale, Arizona.

OpenAI (2023).
GPT-4 Technical Report.
arXiv:2303.08774 [cs].

Pennington, J., Socher, R., and Manning, C. D. (2014).
Glove: Global vectors for word representation.
In *Empirical Methods in Natural Language Processing (EMNLP)*, pages 1532–1543.

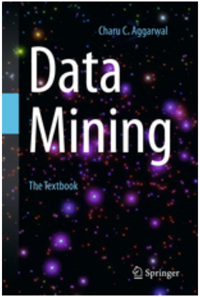
Radford, A., Wu, J., Child, R., Luan, D., Amodei, D., and Sutskever, I. (2019).
Language models are unsupervised multitask learners.
OpenAI Blog, 1(8):9.

Bibliography III

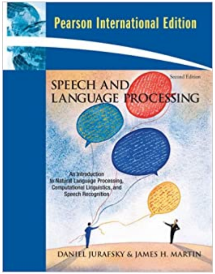
Reimers, N. and Gurevych, I. (2019).
Sentence-BERT: Sentence embeddings using siamese BERT-networks.
In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing*. Association for Computational Linguistics.

Vaswani, A., Shazeer, N., Parmar, N., Uszkoreit, J., Jones, L., Gomez, A. N., Kaiser, Ł., and Polosukhin, I. (2017).
Attention is all you need.
In *Advances in neural information processing systems*, pages 5998–6008.

Bibliography (books)



[Aggarwal, 2015]
(chapter 13)


Pearson International Edition
Second Edition
SPEECH AND LANGUAGE PROCESSING
An Introduction to Natural Language Processing, Computational Linguistics, and Speech Recognition
DANIEL JURAFSKY & JAMES H. MARTIN

[Jurafsky and Martin, 2024]

Questions?