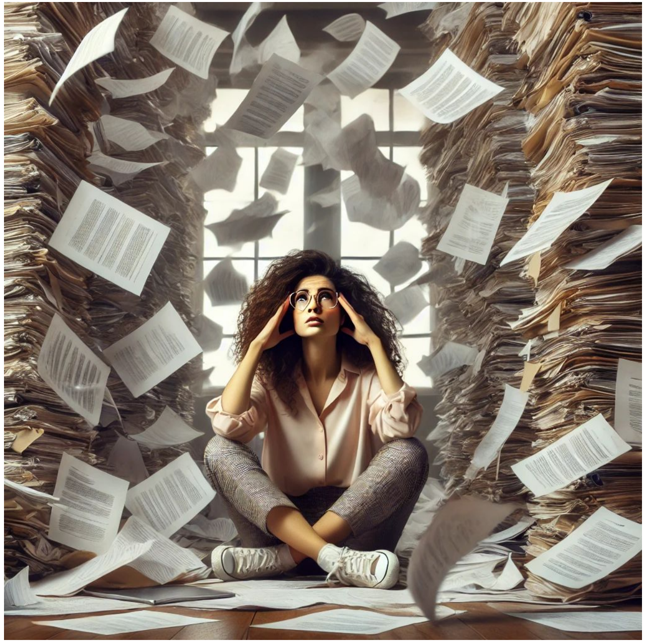
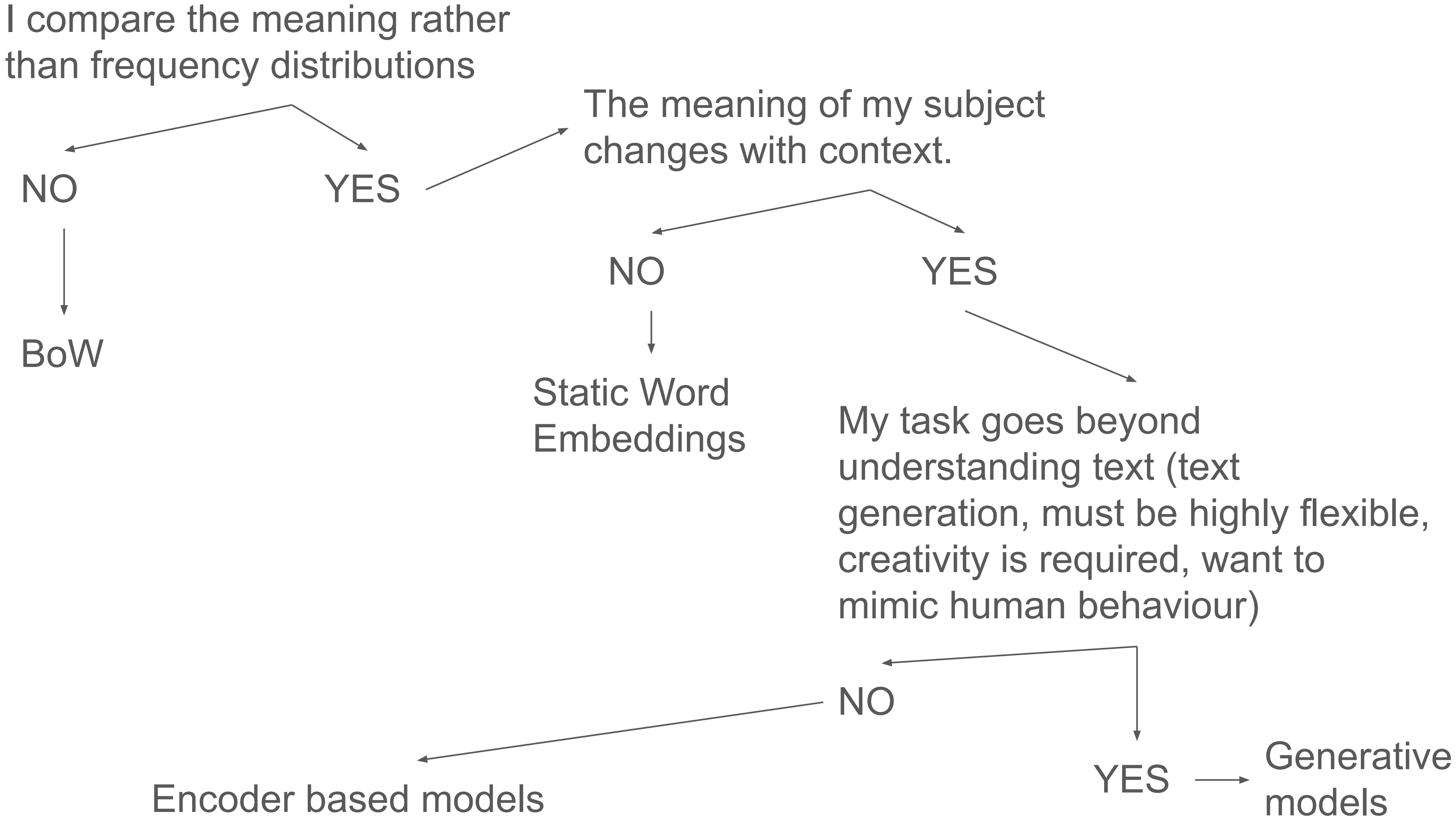
THE END

= THE BEGINNING

Do you have YOUR plan?





I compare the meaning rather than frequency distributions

NO

BoW

YES

The meaning of my subject changes with context.

NO

Static Word Embeddings

YES

My task goes beyond understanding text (text generation, must be highly flexible, creativity is required, want to mimic human behaviour)

NO

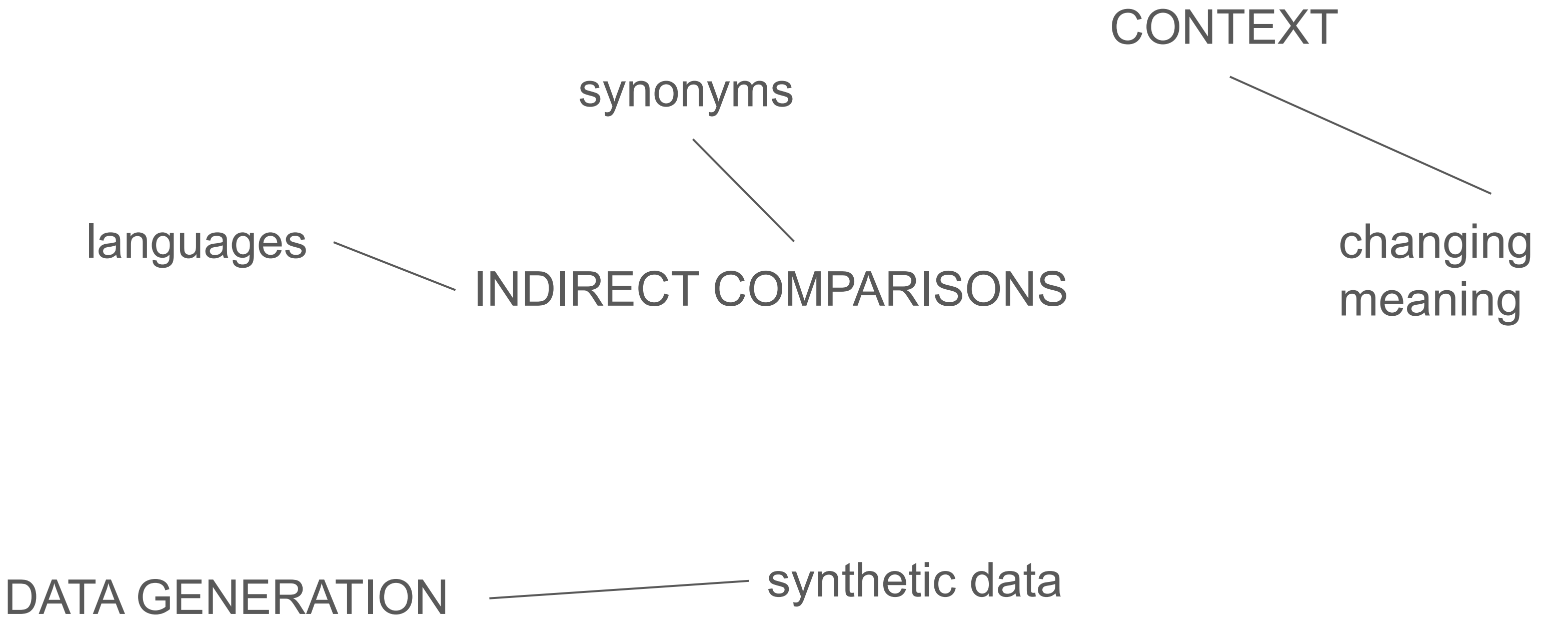
Encoder based models

YES

Generative models

How can LLMs help us think out of the box?

WHAT IS HERE THAT WE HAD NOT HAD BEFORE



CONTEXT

changing meaning

synonyms

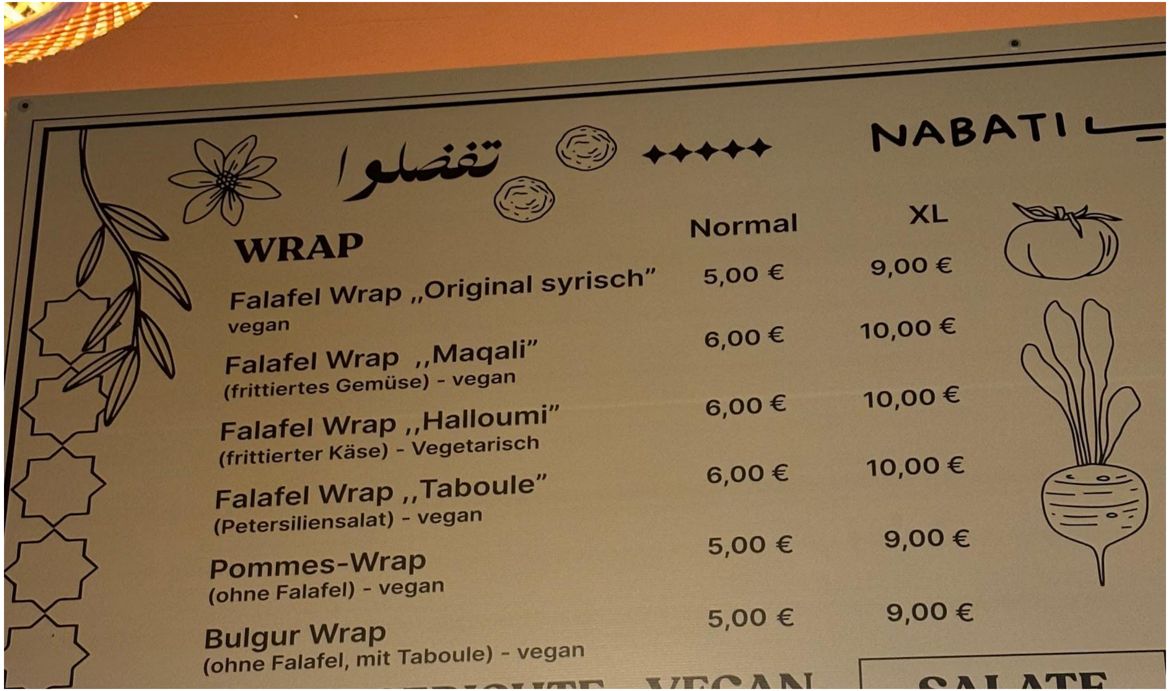
languages

INDIRECT COMPARISONS

DATA GENERATION

synthetic data

- How do we use a concept/word differently in different contexts
- Analysis across languages and countries
- Experiments: Synthetic data generation after a trigger (for instance, how does the politician argue once being put in situation XY)
- …



تفضلوا

NABATI

WRAP

	Normal	XL
Falafel Wrap „Original syrisch" vegan	5,00 €	9,00 €
Falafel Wrap „Maqali" (frittiertes Gemüse) - vegan	6,00 €	10,00 €
Falafel Wrap „Halloumi" (frittierter Käse) - Vegetarisch	6,00 €	10,00 €
Falafel Wrap „Taboule" (Petersiliensalat) - vegan	6,00 €	10,00 €
Pommes-Wrap (ohne Falafel) - vegan	5,00 €	9,00 €
Bulgur Wrap (ohne Falafel, mit Taboule) - vegan	5,00 €	9,00 €

VEGAN

SALATE

Embedding question

HIGHER LEVEL EMBEDDINGS

HOW TO GET DOCUMENT EMBEDDINGS?

- POOLING
  - ONE STEP
  - MULTIPLE STEP
- LARGER CONTEXT

Word vs. sequence embedding with transformers

- obtaining contextualized embeddings for words is nice (thanks BERT!)
- but often our unit of measurement (or even unit of analysis) are **sequences of text** like sentences or paragraphs
- enter stage **sentence BERT**
  - re-uses pre-trained BERT
  - but fine-tunes it on labeled data sets
  - the goal is to get similar embeddings for pairs of texts that have been assigned into similar categories , marked as similar, or etc. by human annotators
- BUT on thursday we'll use generative (decoder) LLM to generate embeddings for longer seqs

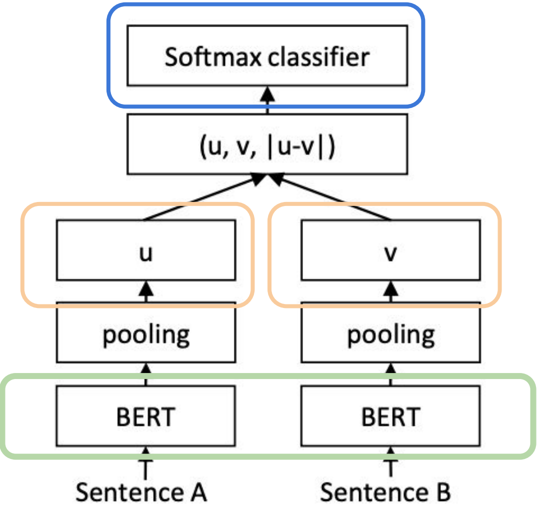


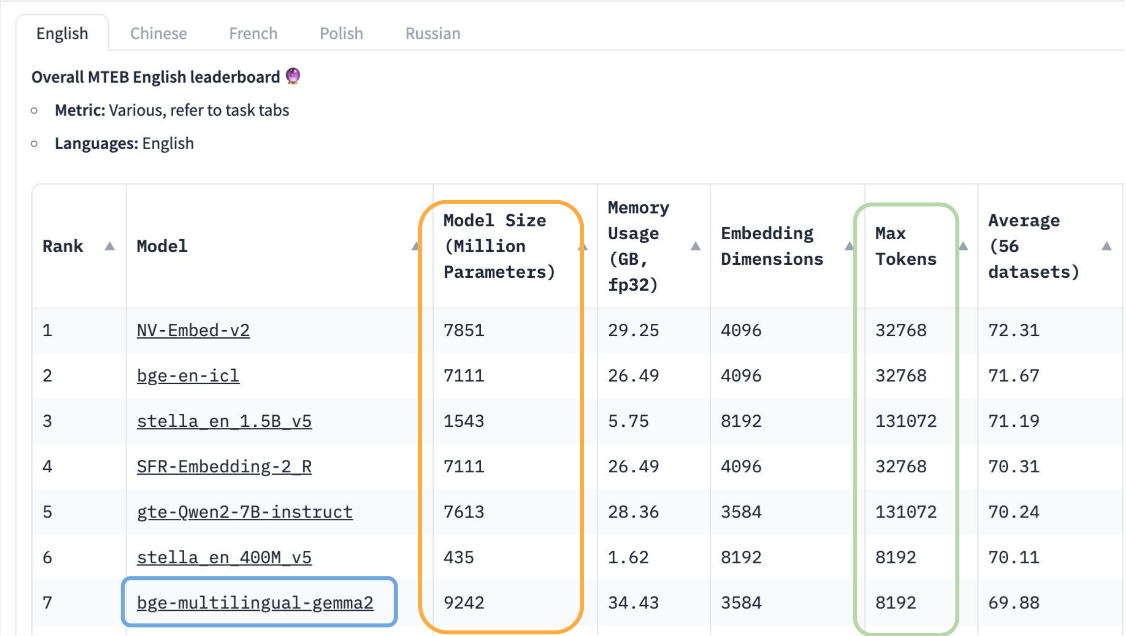
Figure 1: SBERT architecture with classification objective function, e.g., for fine-tuning on SNLI dataset. The two BERT networks have tied weights (siamese network structure).

There are *many* pre-trained embedding models!

Check models available via Hugging Face hub https://huggingface.co/spaces/mteb/leaderboard and there results on the ***Massive Text Embedding Benchmark*** (paper)

different models with

- large and small context width
- few and many parameters
- single and multiple languages



Overall MTEB English leaderboard 🔮

- **Metric:** Various, refer to task tabs
- **Languages:** English

Rank	Model	Model Size (Million Parameters)	Memory Usage (GB, fp32)	Embedding Dimensions	Max Tokens	Average (56 datasets)
1	NV-Embed-v2	7851	29.25	4096	32768	72.31
2	bge-en-icl	7111	26.49	4096	32768	71.67
3	stella_en_1.5B_v5	1543	5.75	8192	131072	71.19
4	SFR-Embedding-2_R	7111	26.49	4096	32768	70.31
5	gte-Qwen2-7B-instruct	7613	28.36	3584	131072	70.24
6	stella_en_400M_v5	435	1.62	8192	8192	70.11
7	bge-multilingual-gemma2	9242	34.43	3584	8192	69.88

Code tutorial on github

https://github.com/haukelicht/advanced_text_analysis/blob/main/notebooks/sentence_embedding_illustration.ipynb

BERTopic

Application of sentence/document embeddings

BERTopic

- (first?) Transformer-based topic model
- *not* a statistical model (like LDA), but a pipeline of data science techniques

Github: https://maartengr.github.io/BERTopic/api/bertopic.html

Documentation: https://maartengr.github.io/BERTopic/api/bertopic.html

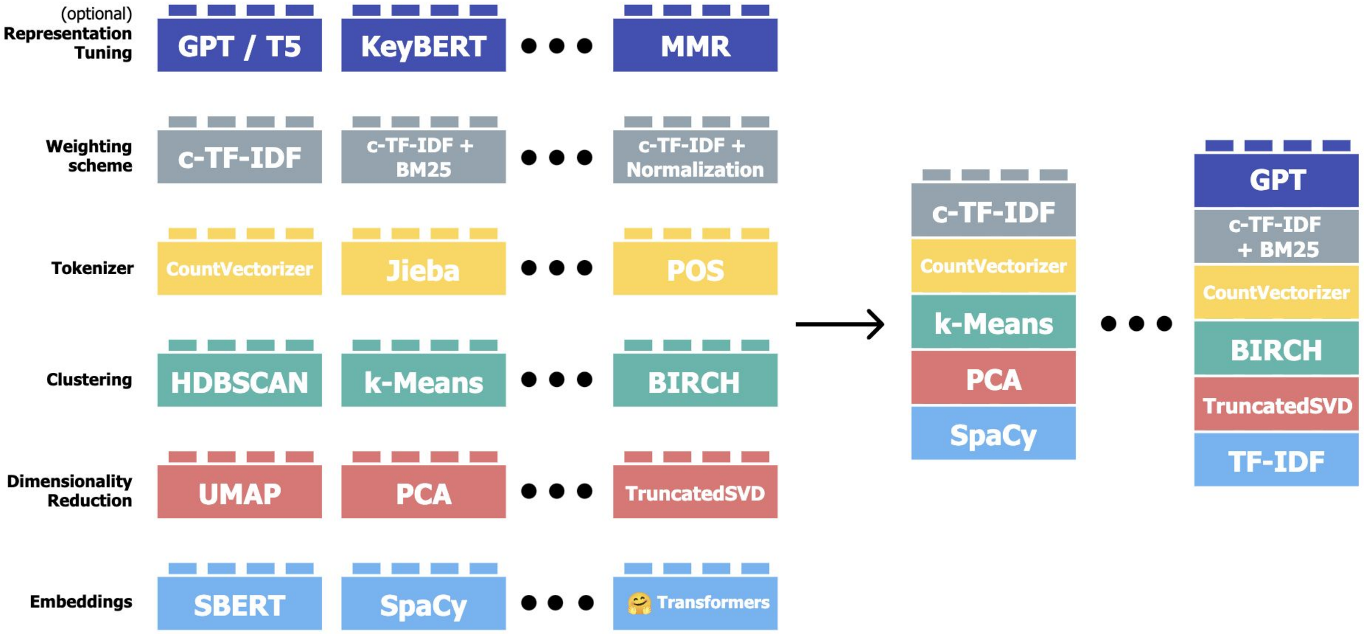
BERTopic

a modular **pipeline** of data science techniques

1. document/sentence embedding ⇒ from text to numeric vectors
2. dimensionality reduction ⇒ lower-dimensional doc. representation
3. un-/semi-supervised clustering ⇒ topic assignment
4. bag-of-words–based topic representation ⇒ returns topic-word scores

BERTopic

a **modular** pipeline



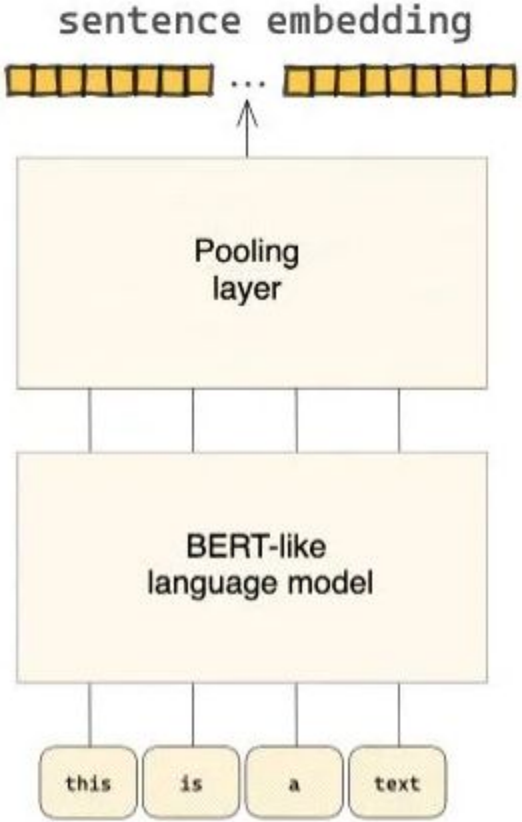
Code tutorial on github

https://github.com/haukelicht/advanced_text_analysis/blob/main/notebooks/topicmodel_bertopic.ipynb

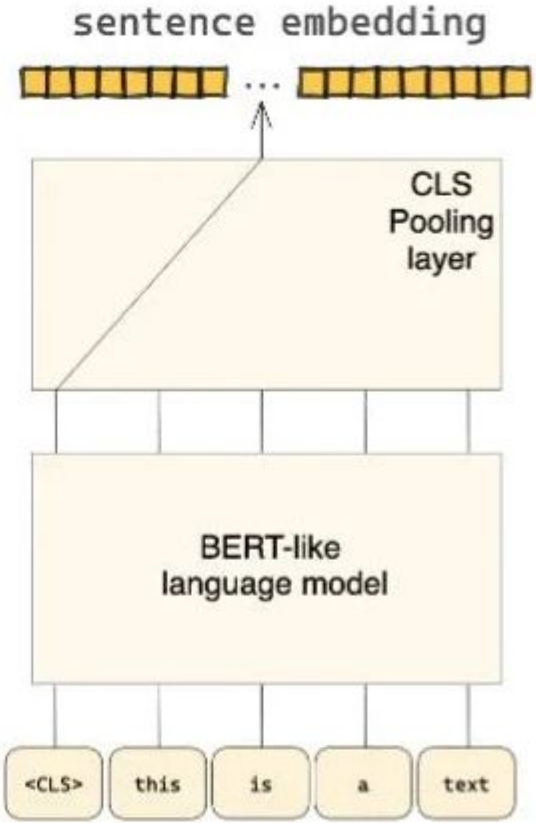
Further transformer-based approaches to topic modeling

papers to read

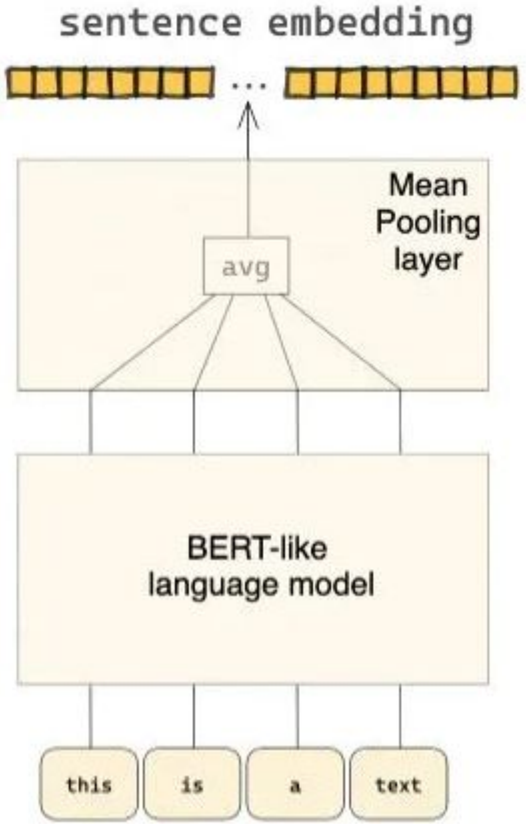
- Stammbach et al. 2023. “Revisiting Automated Topic Model Evaluation with Large Language Models” https://arxiv.org/abs/2305.12152
- *Lam et al. 2024. “*Concept Induction: Analyzing Unstructured Text with High-Level Concepts Using LLooM.” https://github.com/michelle123lam/lloom?tab=readme-ov-file
- Pham et al. 2023. “TopicGPT” https://arxiv.org/abs/2311.01449



A pooling layer aggregates token-level embeddings into one sentence-level embedding



CLS pooling aggregates by taking the token embedding of a special CLS token



Mean pooling aggregates by taking the element-wise arithmetic mean

Also possible:

- taking the max
- taking the mean/sqrt(n)

GOOD PRACTICE: Use embedding models with large context window and calculate one embedding per document

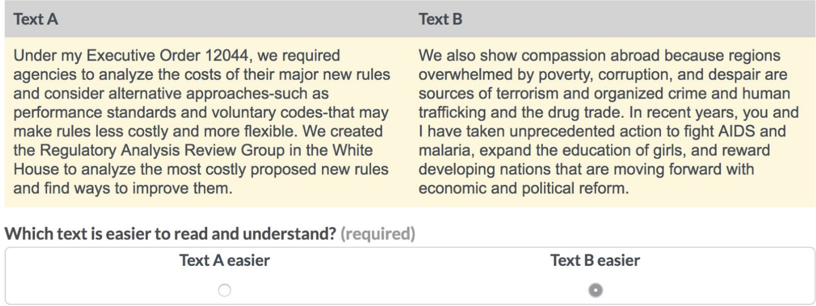
Alternative supervised classification task

Coding tasks

Pairwise comparison

rank alternatives on a (latent) conceptual continuum in terms of their intensity

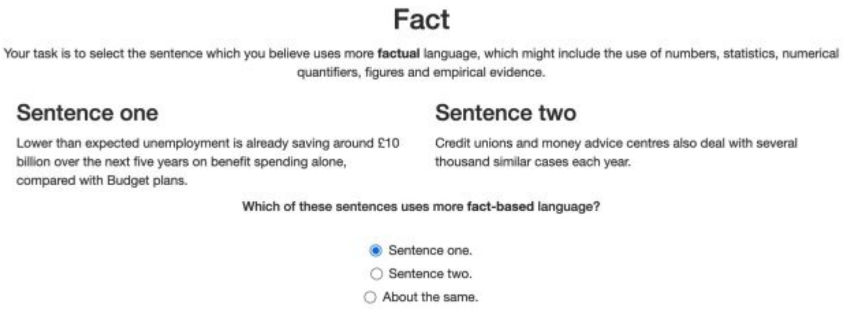
- *examples*
  - political sophistication (Benoit et al., 2019)
  - emotionality, factuality, human narrative, complexity, etc. (Hargrave & Blumenau, 2022)



Text A	Text B
Under my Executive Order 12044, we required agencies to analyze the costs of their major new rules and consider alternative approaches-such as performance standards and voluntary codes-that may make rules less costly and more flexible. We created the Regulatory Analysis Review Group in the White House to analyze the most costly proposed new rules and find ways to improve them.	We also show compassion abroad because regions overwhelmed by poverty, corruption, and despair are sources of terrorism and organized crime and human trafficking and the drug trade. In recent years, you and I have taken unprecedented action to fight AIDS and malaria, expand the education of girls, and reward developing nations that are moving forward with economic and political reform.

Which text is easier to read and understand? (required)

Text A easier ○ Text B easier ●



Fact

Your task is to select the sentence which you believe uses more **factual** language, which might include the use of numbers, statistics, numerical quantifiers, figures and empirical evidence.

Sentence one	Sentence two
Lower than expected unemployment is already saving around £10 billion over the next five years on benefit spending alone, compared with Budget plans.	Credit unions and money advice centres also deal with several thousand similar cases each year.

Which of these sentences uses more fact-based language?

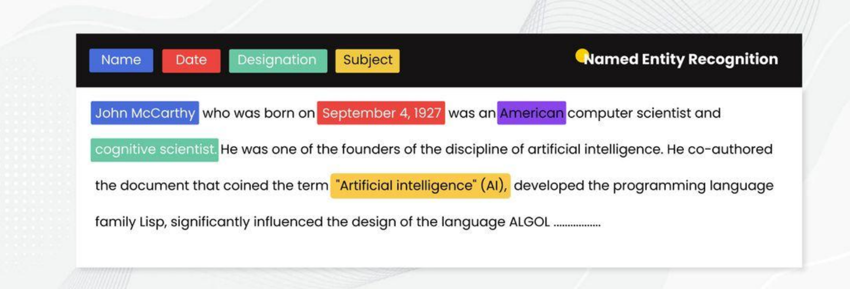
● Sentence one.
○ Sentence two.
○ About the same.

Coding tasks

Word-level classification

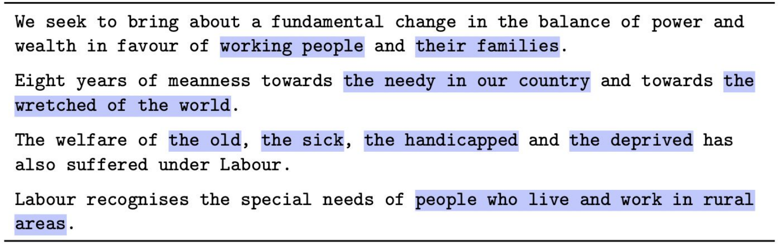
detect (and extract) phrases and multi-word expressions referring to mutually exclusive categories/types

- a.k.a. as token classification
- *examples*
    - *Named Entity Recognition*
    - social group mention detection (Licht & Sczepanski, 2024)



John McCarthy who was born on September 4, 1927 was an American computer scientist and cognitive scientist. He was one of the founders of the discipline of artificial intelligence. He co-authored the document that coined the term "Artificial intelligence" (AI), developed the programming language family Lisp, significantly influenced the design of the language ALGOL ....

Table 1. Examples of group mentions in sentences drawn from British mainstream party manifestos. Highlighted text spans identify groups mentioned in each sentence.



We seek to bring about a fundamental change in the balance of power and wealth in favour of working people and their families.

Eight years of meanness towards the needy in our country and towards the wretched of the world.

The welfare of the old, the sick, the handicapped and the deprived has also suffered under Labour.

Labour recognises the special needs of people who live and work in rural areas.

LLM in-context learning

Intro and bigger-picture papers (in political science)

- Palmer and Spirling. 2023. “Using proprietary language models in academic research requires explicit justification” doi.org/10.1038/s43588-023-00585-1
- Weber & Richardt. 2023. “Evaluation is all you need. Prompting Generative Large Language Models for Annotation Tasks in the Social Sciences. A Primer using Open Models” http://arxiv.org/abs/2401.00284
- Törnberg. 2024. “Best Practices for Text Annotation with Large Language Models” http://arxiv.org/abs/2402.05129
-

Applied papers (in political science)

Text classification

- Gilardi et al. 2023. “ChatGPT outperforms crowd workers for text-annotation tasks” doi.org/10.1073/pnas.2305016120
- Lupo et al. 2023. “How to Use Large Language Models for Text Coding: The Case of Fatherhood Roles in Public Policy Documents” http://arxiv.org/abs/2311.11844
- Reiss. 2023. “Testing the Reliability of ChatGPT for Text Annotation and Classification: A Cautionary Remark” http://arxiv.org/abs/2304.11085

Applied papers (in political science)

Text classification

- Ziems et al. 2023. “Can Large Language Models Transform Computational Social Science?” https://arxiv.org/abs/2305.03514
- Mellon et al. 2024. “Do AIs know what the most important issue is? Using language models to code open-text social survey responses at scale” doi.org/10.1177/20531680241231468
- Umansky et al. 2024. “Enhancing Hate Speech Detection with Fine-Tuned Large Language Models Requires High-Quality Data” doi.org/10.31219/osf.io/7kbqt

Applied papers (in political science)

Text scaling

- Wu et al. 2023. “Large Language Models Can Be Used to Estimate the Latent Positions of Politicians” https://arxiv.org/abs/2303.12057
- Wu et al. 2024. “Concept-Guided Chain-of-Thought Prompting for Pairwise Comparison Scaling of Texts with Large Language Models” http://arxiv.org/abs/2310.12049
- O’Hagan & Schein. 2023. “Measurement in the Age of LLMs: An Application to Ideological Scaling” http://arxiv.org/abs/2312.09203
- Mens & Gallego. 2023. “Scaling Political Texts with Large Language Models: Asking a Chatbot Might Be All You Need” http://arxiv.org/abs/2311.16639

Applied papers (in political science and econ)

Text generation and survey interviewing

- Palmer & Spirling. 2024. “Large Language Models Can Argue in Convincing Ways About Politics, But Humans Dislike AI Authors: Implications for Governance” https://arthurspirling.org/documents/llm.pdf
- Bisbee et al. 2023. ”Synthetic Replacements for Human Survey Data? The Perils of Large Language Models” doi.org/10.31235/osf.io/5ecfa
- Chopra & Haarland. 2024. “Conducting Qualitative Interviews with AI” https://papers.ssrn.com/abstract=4572954

Example selection

The literature

- Example selection for few-shot in-context learning (ICL), for example in text classification applications is a difficult problem.
- lots of research in this area in computer science (search keywords “in-context learning”, “few-shot”, “example selection”, “sample selection”)

Table on the right from paper by Pecher et al. (2024) “Automatic Combination of Sample Selection Strategies for Few-Shot Learning” https://arxiv.org/abs/2402.03038 👉 likely a good starting point

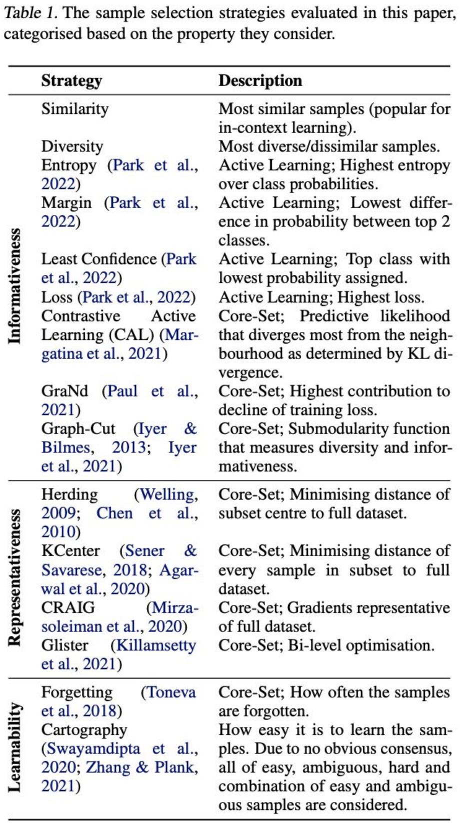


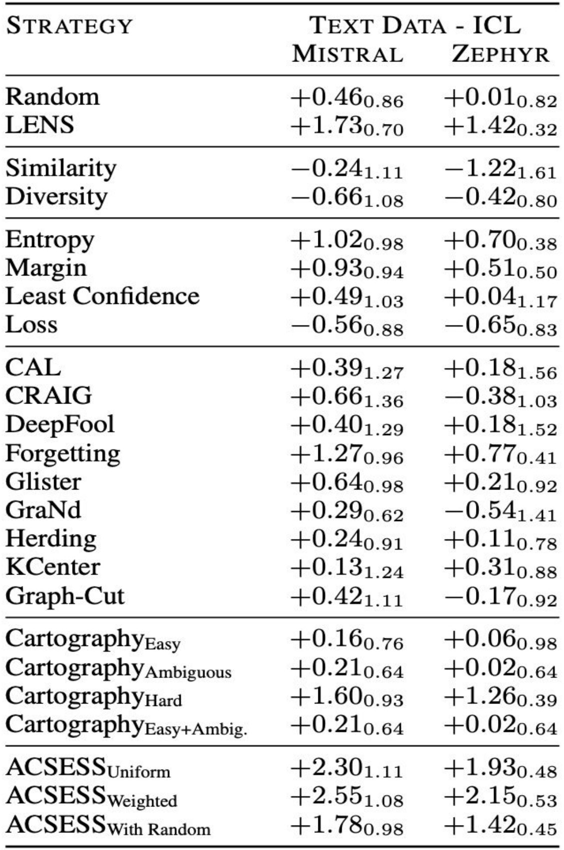
Table 1. The sample selection strategies evaluated in this paper, categorised based on the property they consider.

	Strategy	Description
Informativeness	Similarity	Most similar samples (popular for in-context learning).
	Diversity	Most diverse/dissimilar samples.
	Entropy (Park et al., 2022)	Active Learning; Highest entropy over class probabilities.
	Margin (Park et al., 2022)	Active Learning; Lowest difference in probability between top 2 classes.
	Least Confidence (Park et al., 2022)	Active Learning; Top class with lowest probability assigned.
	Loss (Park et al., 2022)	Active Learning; Highest loss.
	Contrastive Active Learning (CAL) (Margatina et al., 2021)	Core-Set; Predictive likelihood that diverges most from the neighbourhood as determined by KL divergence.
	GraNd (Paul et al., 2021)	Core-Set; Highest contribution to decline of training loss.
	Graph-Cut (Iyer & Bilmes, 2013; Iyer et al., 2021)	Core-Set; Submodularity function that measures diversity and informativeness.
Representativeness	Herding (Welling, 2009; Chen et al., 2010)	Core-Set; Minimising distance of subset centre to full dataset.
	KCenter (Sener & Savarese, 2018; Agarwal et al., 2020)	Core-Set; Minimising distance of every sample in subset to full dataset.
	CRAIG (Mirzasoleiman et al., 2020)	Core-Set; Gradients representative of full dataset.
	Glister (Killamsetty et al., 2021)	Core-Set; Bi-level optimisation.
Learnability	Forgetting (Toneva et al., 2018)	Core-Set; How often the samples are forgotten.
	Cartography (Swayamdipta et al., 2020; Zhang & Plank, 2021)	How easy it is to learn the samples. Due to no obvious consensus, all of easy, ambiguous, hard and combination of easy and ambiguous samples are considered.

The literature

Table on the right from paper by Pecher et al. (2024) “Automatic Combination of Sample Selection Strategies for Few-Shot Learning” https://arxiv.org/abs/2402.03038

- default strategy: ‘a new set of [5] samples is randomly selected for each new task’ (p. 4)
- “Random” (first row): ‘a single set of 5 samples per class is randomly selected at the start and used for every single single task’ (this is what we did)
- a fesimilarity and diversity based strategies actually worse than defualt



STRATEGY	TEXT DATA - ICL	
	MISTRAL	ZEPHYR
Random	$+0.46_{0.86}$	$+0.01_{0.82}$
LENS	$+1.73_{0.70}$	$+1.42_{0.32}$
Similarity	$-0.24_{1.11}$	$-1.22_{1.61}$
Diversity	$-0.66_{1.08}$	$-0.42_{0.80}$
Entropy	$+1.02_{0.98}$	$+0.70_{0.38}$
Margin	$+0.93_{0.94}$	$+0.51_{0.50}$
Least Confidence	$+0.49_{1.03}$	$+0.04_{1.17}$
Loss	$-0.56_{0.88}$	$-0.65_{0.83}$
CAL	$+0.39_{1.27}$	$+0.18_{1.56}$
CRAIG	$+0.66_{1.36}$	$-0.38_{1.03}$
DeepFool	$+0.40_{1.29}$	$+0.18_{1.52}$
Forgetting	$+1.27_{0.96}$	$+0.77_{0.41}$
Glister	$+0.64_{0.98}$	$+0.21_{0.92}$
GraNd	$+0.29_{0.62}$	$-0.54_{1.41}$
Herding	$+0.24_{0.91}$	$+0.11_{0.78}$
KCenter	$+0.13_{1.24}$	$+0.31_{0.88}$
Graph-Cut	$+0.42_{1.11}$	$-0.17_{0.92}$
$\text{Cartography}_{\text{Easy}}$	$+0.16_{0.76}$	$+0.06_{0.98}$
$\text{Cartography}_{\text{Ambiguous}}$	$+0.21_{0.64}$	$+0.02_{0.64}$
$\text{Cartography}_{\text{Hard}}$	$+1.60_{0.93}$	$+1.26_{0.39}$
$\text{Cartography}_{\text{Easy+Ambig.}}$	$+0.21_{0.64}$	$+0.02_{0.64}$
$\text{ACSESS}_{\text{Uniform}}$	$+2.30_{1.11}$	$+1.93_{0.48}$
$\text{ACSESS}_{\text{Weighted}}$	$+2.55_{1.08}$	$+2.15_{0.53}$
$\text{ACSESS}_{\text{With Random}}$	$+1.78_{0.98}$	$+1.42_{0.45}$

Table 2. Benefit of the different selection strategies calculated as the difference in accuracy to the classic few-shot selection strategy, aggregated over the image and text datasets. The subscript represents the standard deviation of the difference over the aggregated datasets.

Other few-shot learning approaches

Natural language inference

NLI: convert classification problems

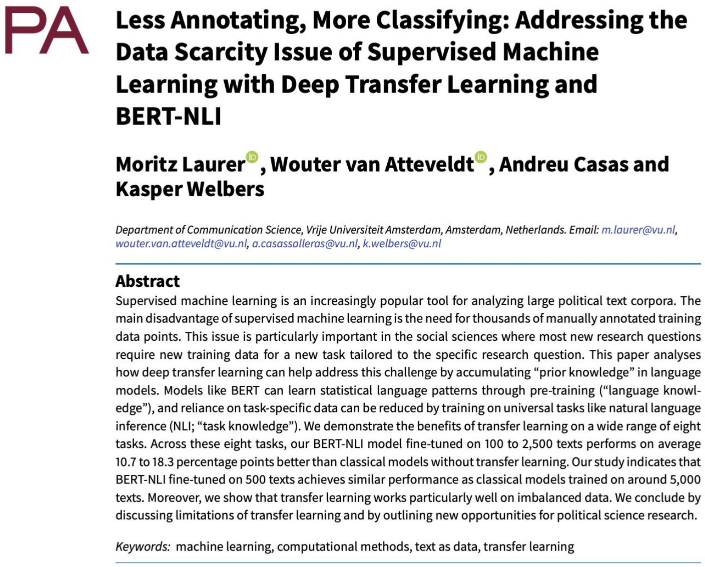
- label class ⇒ label class description (**hypothesis**)
- input text ⇒ input text (**premise**)
- entailment classification: “entails”, “contradicts”, “neutral”

Example

original *text:* “The weather is great.” *label:* “positive”

⇒ NLI format 👇

premise: “The weather is great.”
hypo: “The sentiment of the sentence is positive.”
entailment classification: “entails”



PA

Less Annotating, More Classifying: Addressing the Data Scarcity Issue of Supervised Machine Learning with Deep Transfer Learning and BERT-NLI

Moritz Laurer, Wouter van Atteveldt, Andreu Casas and Kasper Welbers

Department of Communication Science, Vrije Universiteit Amsterdam, Amsterdam, Netherlands. Email: m.laurer@vu.nl, wouter.van.atteveldt@vu.nl, a.casassalleras@vu.nl, k.welbers@vu.nl

Abstract

Supervised machine learning is an increasingly popular tool for analyzing large political text corpora. The main disadvantage of supervised machine learning is the need for thousands of manually annotated training data points. This issue is particularly important in the social sciences where most new research questions require new training data for a new task tailored to the specific research question. This paper analyses how deep transfer learning can help address this challenge by accumulating “prior knowledge” in language models. Models like BERT can learn statistical language patterns through pre-training (“language knowledge”), and reliance on task-specific data can be reduced by training on universal tasks like natural language inference (NLI; “task knowledge”). We demonstrate the benefits of transfer learning on a wide range of eight tasks. Across these eight tasks, our BERT-NLI model fine-tuned on 100 to 2,500 texts performs on average 10.7 to 18.3 percentage points better than classical models without transfer learning. Our study indicates that BERT-NLI fine-tuned on 500 texts achieves similar performance as classical models trained on around 5,000 texts. Moreover, we show that transfer learning works particularly well on imbalanced data. We conclude by discussing limitations of transfer learning and by outlining new opportunities for political science research.

Keywords: machine learning, computational methods, text as data, transfer learning

https://colab.research.google.com/github/MoritzLaurer/less-annotating-with-bert-nli/blob/master/BERT_NLI_demo.ipynb

Natural language inference

NLI: convert classification problems

- label class ⇒ label class description (**hypothesis**)
- input text ⇒ input text (**premise**)
- entailment classification: “entails”, “contradicts”, “neutral”

Example

original *text:* “The weather is great.” *label:* “positive”

⇒ NLI format 👇

premise: “The weather is great.”
hypo: “The sentiment of the sentence is positive.”
entailment classification: “entails”

Pre-trained models on HF model hub

- moritz laurer’s models: https://huggingface.co/collections/MoritzLaurer/zeroshot-classifiers-6548b4ff407bb19ff5c3ad6f (paper)
- Mike Burnham’s models: https://huggingface.co/mlburnham (paper)

setfit: few-shot tuning with SBERT

Fine-tune sentence transformer with *contrastive loss* (see explanation here)

- paper: Tunstall *et al.* (2022) “Efficient Few-Shot Learning Without Prompts” https://arxiv.org/abs/2209.11055
- tutorials: https://huggingface.co/docs/setfit/index
- python library: https://github.com/huggingface/setfit
-

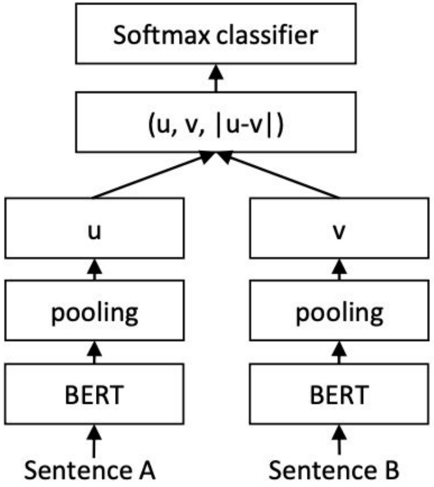


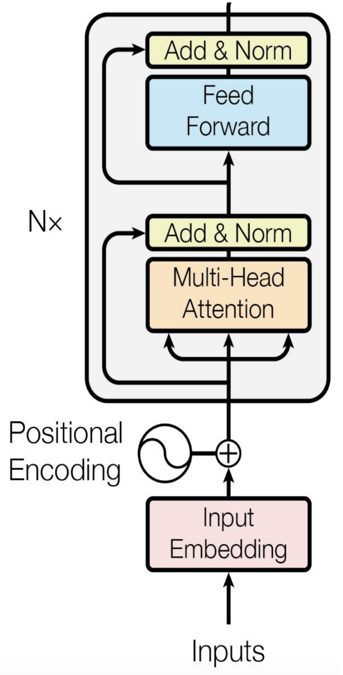
Figure 1: SBERT architecture with classification objective function, e.g., for fine-tuning on SNLI dataset. The two BERT networks have tied weights (siamese network structure).

Parameter-efficient fine-tuning

Saving computation resources

If you have a (very) large model and you want to fine-tune it, you require a lot of GPU memory

It turns out fine-tuning *all* model parameters is not necessary to achieve good performance in the fine-tuning task



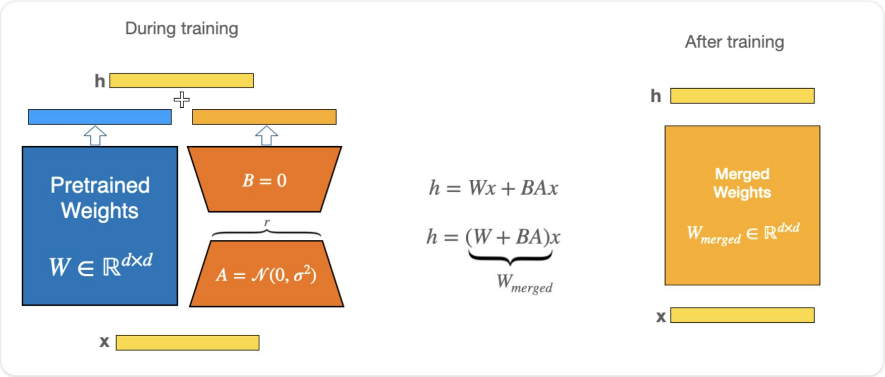
Saving computation resources

If you have a (very) large model and you want to fine-tune it, you require a lot of GPU memory

It turns out fine-tuning *all* model parameters is not necessary to achieve good performance in the fine-tuning task

We can use **Lo**w-**Ra**nk (LORA) adapters to leverage this insight

- only need to fine-tune the adapter parameters
- makes training more efficient (or even possible)



During training
h
Pretrained Weights
$W \in \mathbb{R}^{d \times d}$
$B = 0$
r
$A = \mathcal{N}(0, \sigma^2)$
x
$h = Wx + BAx$
$h = \underbrace{(W + BA)}_{W_{merged}}x$
After training
h
Merged Weights
$W_{merged} \in \mathbb{R}^{d \times d}$
x

see https://huggingface.co/docs/peft/main/en/conceptual_guides/lora

https://github.com/huggingface/peft/tree/main/examples/olora_finetuning