UNIVERSITÄT
LEIPZIG

Toolbox CSS
– Transformers // GPT, BERT, NLI, BERTopic

GWZ H2 1.15 // 09/12/2025

Felix Lennert, M.Sc.

OUTLINE

- Motivation
- Transformers
- How they work and what they can do
  - GPT
  - BERT
  - NLI
  - BERTopic

BOW HYPOTHESIS

Sentence 1: “This is a hell of a movie”

Sentence 2: “This movie is hell”

Preprocessing (stop word removal, word order)

Sentence/Token	movie	hell		
1	1	1	→	Negative
2	1	1	→	Negative

BERT

Sentence 1: “This is hell of a movie”

Sentence 2: “This movie is hell”

bert-base-uncased; fine tuned on ~2,000 labeled IMDb reviews
(~4min on MacBook Pro (2021, 16GB RAM, M1 Pro)

```
>>> predict(model, "this is a hell of a movie", tokenizer)
1
>>> predict(model, "this movie is hell", tokenizer)
0
```

→ Positive

→ Negative

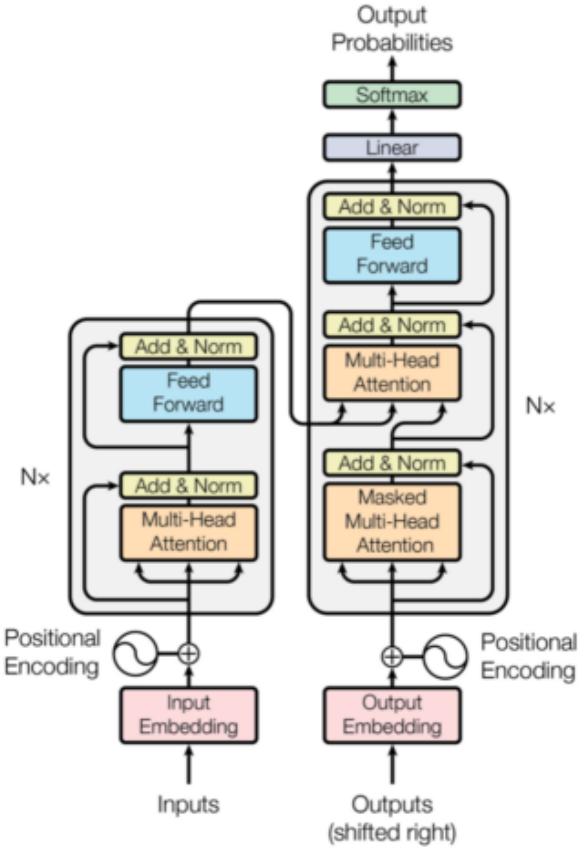
THE SOLUTION – TRANSFER LEARNING

Idea: learn models of language from large corpora (i.e., embeddings)

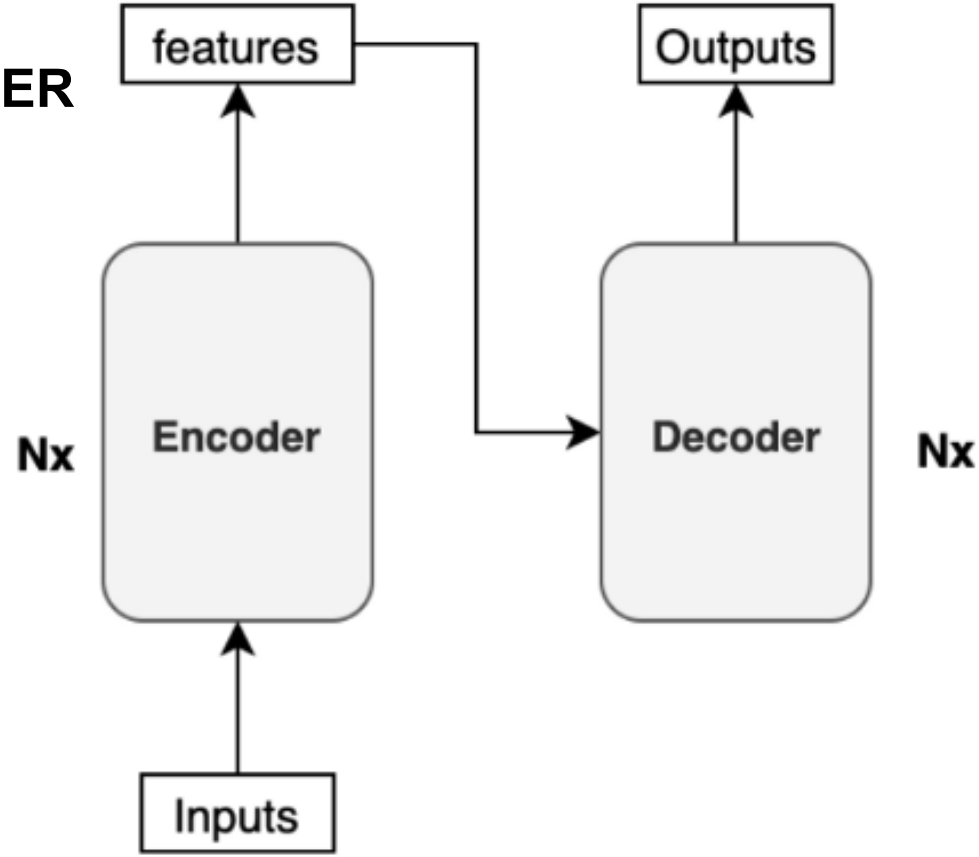
Compared to BoW models, today's models have several advantages:

- They take *all* the text – no preprocessing
- They have "learned" relationships between words from large text data
- They have a way of quantifying word meaning in context – overcomes shortcoming of last week's embeddings
- They incorporate word order

TRANSFORMER



TRANSFORMER

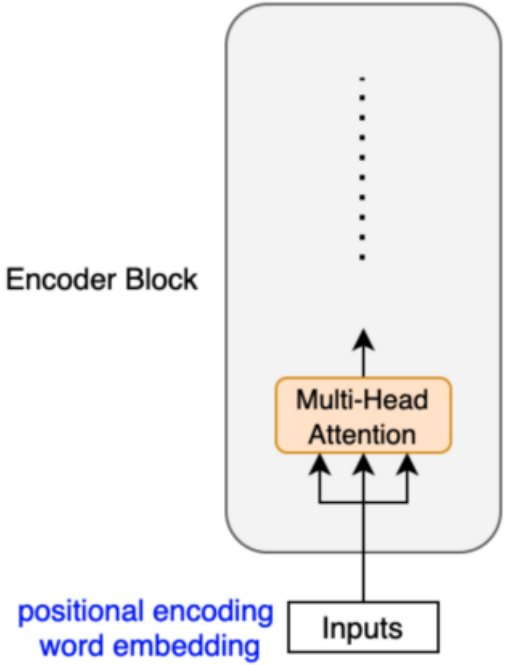


TOKENIZATION

- Here, each text is of fixed length – it needs to get padded (e.g., by including <pad>, <pad>, …, <pad>)
- “.” might become <EOS>
- Also, there are length limits – e.g., BERT takes up to 512 tokens
- tokens also look a bit different, they break up the words a bit
- finally, tokens are replaced by their vectors (including their position)

```
>>> tokenizer = BertTokenizer.from_pretrained('bert-base-uncased')
>>> tokens = tokenizer.tokenize('This is what tokenization in BERT looks like.')
>>> print(tokens)
['this', 'is', 'what', 'token', '##ization', 'in', 'bert', 'looks', 'like', '.']
```

ATTENTION (IS ALL YOU NEED)



Attention Is All You Need

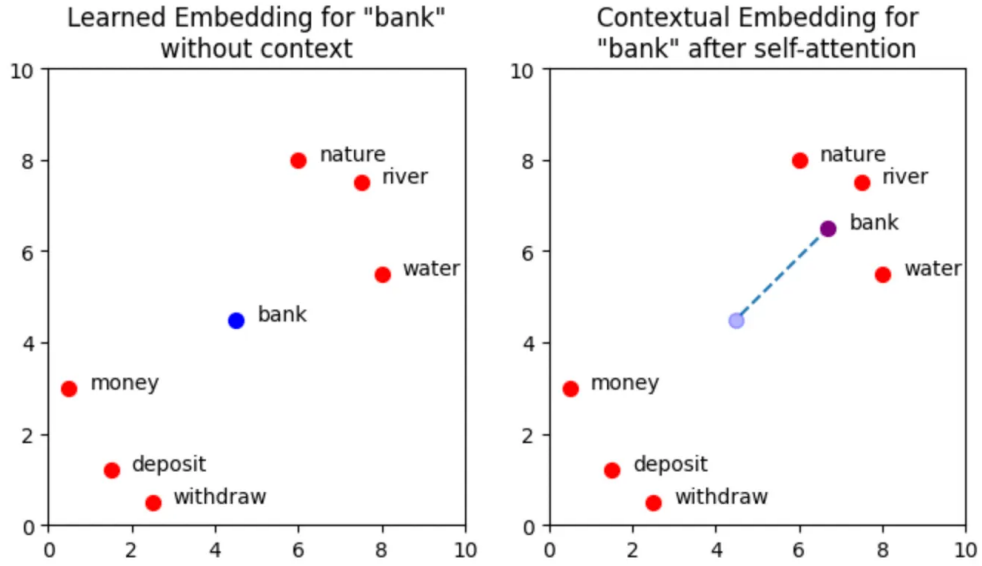
Ashish Vaswani* Google Brain avaswani@google.com
Noam Shazeer* Google Brain noam@google.com
Niki Parmar* Google Research nikip@google.com
Jakob Uszkoreit* Google Research usz@google.com
Llion Jones* Google Research llion@google.com
Aidan N. Gomez* † University of Toronto aidan@cs.toronto.edu
Łukasz Kaiser* Google Brain lukaszkaiser@google.com
Illia Polosukhin* ‡ illia.polosukhin@gmail.com

Cited by 144458

Goal: relate words to each other, impute context

ATTENTION (IS ALL YOU NEED)

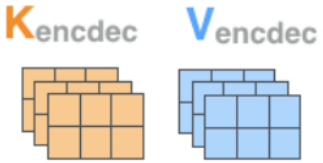
Example sentence: a man is sitting on a river bank

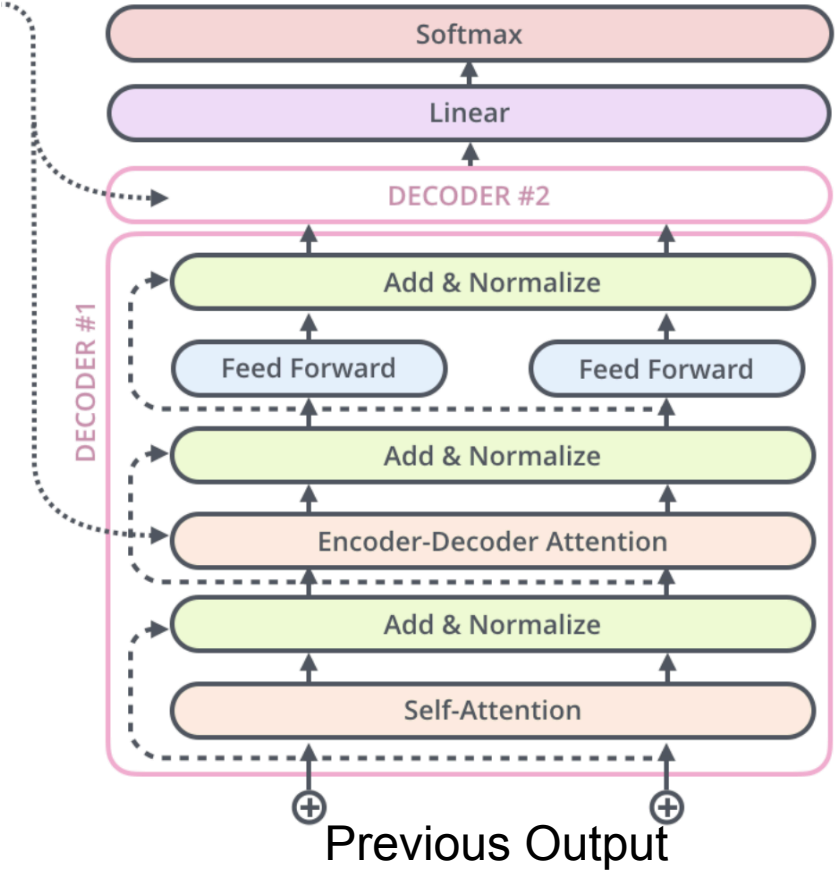


ENCODER OUTPUT

- Embeddings with context – the model has “read” the input text
- This can be used for different tasks:
  - sequence classification head (BERT) => feeds these vectors into a “linear layer” (assigning probability to each class)
  - also: regression head (BERT) => for continuous values
  - token classification head (BERT) => assigns one label to each token (e.g., named-entity recognition)
  - …
  - translation => feed forward to **decoder to generate new text**

DECODER

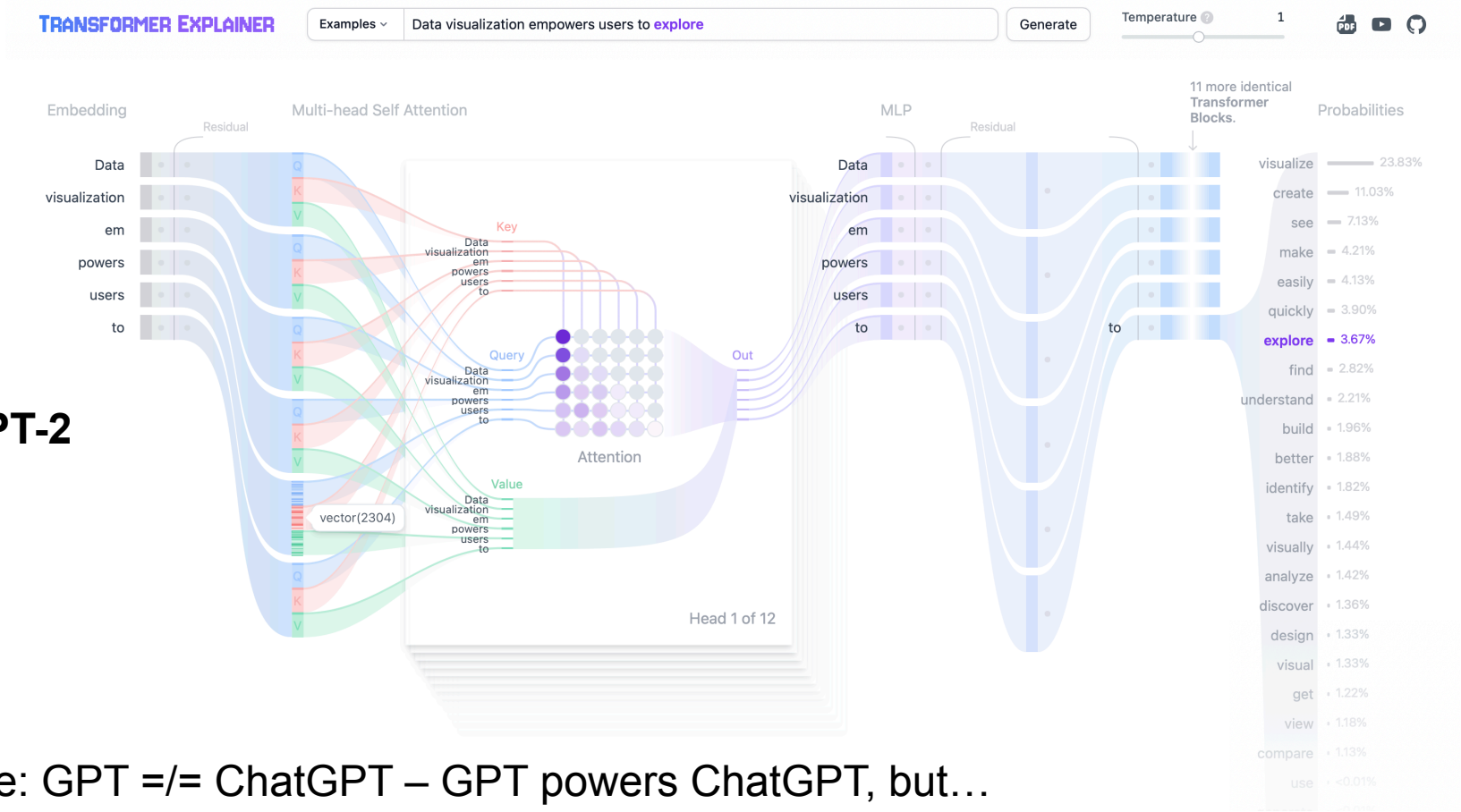




Works similarly to encoder, but:

- **Encoder** sees each word in input – **before and after the focal word**
- **Decoder** only sees the words that have been generated **before the focal word**

=> goal for decoder: predict **next word**



GPT-2

Note: GPT =/= ChatGPT – GPT powers ChatGPT, but…

CHATBOTS

They are powered by GPTs, but modified:

- Data Format: Conversations are formatted as alternating prompts and responses
  - Special tokens mark different speakers/roles
  - Example format:
    <HUMAN>: How do I make pasta?
    <ASSISTANT>: First, boil water...
    <HUMAN>: How long should I boil it?
    <ASSISTANT>: Typically 8-12 minutes...

=> Model learns to predict the next tokens given the conversation history

=> Particular focus on generating **appropriate** responses after human prompts

CHATBOTS

=> Particular focus on generating appropriate responses after human prompts

- Hence, it must learn:
  - Appropriate tone/style
  - Staying in character/role
  - Maintaining conversation context
  - Following instructions
- Furthermore, “raw” models don’t have any filter
  - Racism
  - Sexism
  - Illegal content

HOW IS IT USED IN THE SOCIAL SCIENCES? – GPT

MACHINE BIAS

How do Generative Language Models Answer Opinion Polls?

Julien Boelaert[1], Samuel Coavoux[2], Étienne Ollion[2], Ivaylo Petev[2], and Patrick Präg[2]

"Our results i) confirm that to date, **models cannot replace research subjects for opinion or attitudinal research**; ii) that **they display a strong bias on each question** (reaching only a small region of social space); and iii) that this bias varies randomly from one question to the other (reaching a different region every time)."

Leveraging AI for democratic discourse: Chat interventions can improve online political conversations at scale

Lisa P. Argyle, Christopher A. Bail, Ethan C. Busby, +4, and David Wingate Authors Info & Affiliations

We develop an AI chat assistant that makes real-time, evidence-based suggestions for messages in divisive online political conversations. In a randomized controlled trial, we show that when one participant in a conversation had access to this assistant, **it increased their partner's reported quality of conversation and both participants' willingness to grant political opponents space to express and advocate their views in the public sphere**. Participants had the ability to accept, modify, or ignore the AI chat assistant's recommendations. Notably, participants' policy positions were unchanged by the intervention.

Large language models empowered agent-based modeling and simulation: a survey and perspectives

Chen Gao, Xiaochong Lan, Nian Li, Yuan Yuan, Jingtao Ding, Zhilun Zhou, Fengli Xu & Yong Li

Promises:
- Human-like behavior simulation through natural language understanding
- Rich agent-to-agent and agent-environment interactions
- Potential for more sophisticated economic and social simulations

Shortcomings:
- High computational costs
- Reliability issues: inconsistent responses; hallucination
- Hard to control and validate agent behaviors
- Also: no standardized evaluation methods and benchmarks
- Safety and ethical concerns regarding biased or harmful outputs

Large Language Models Outperform Expert Coders and Supervised Classifiers at Annotating Political Social Media Messages

Petter Törnberg[1,2]

"these models are capable of zero-shot annotation based on instructions written in natural language, they obviate the need of large sets of training data"

"the task used is to identify the political affiliation of politicians based on a single X/ Twitter messages"

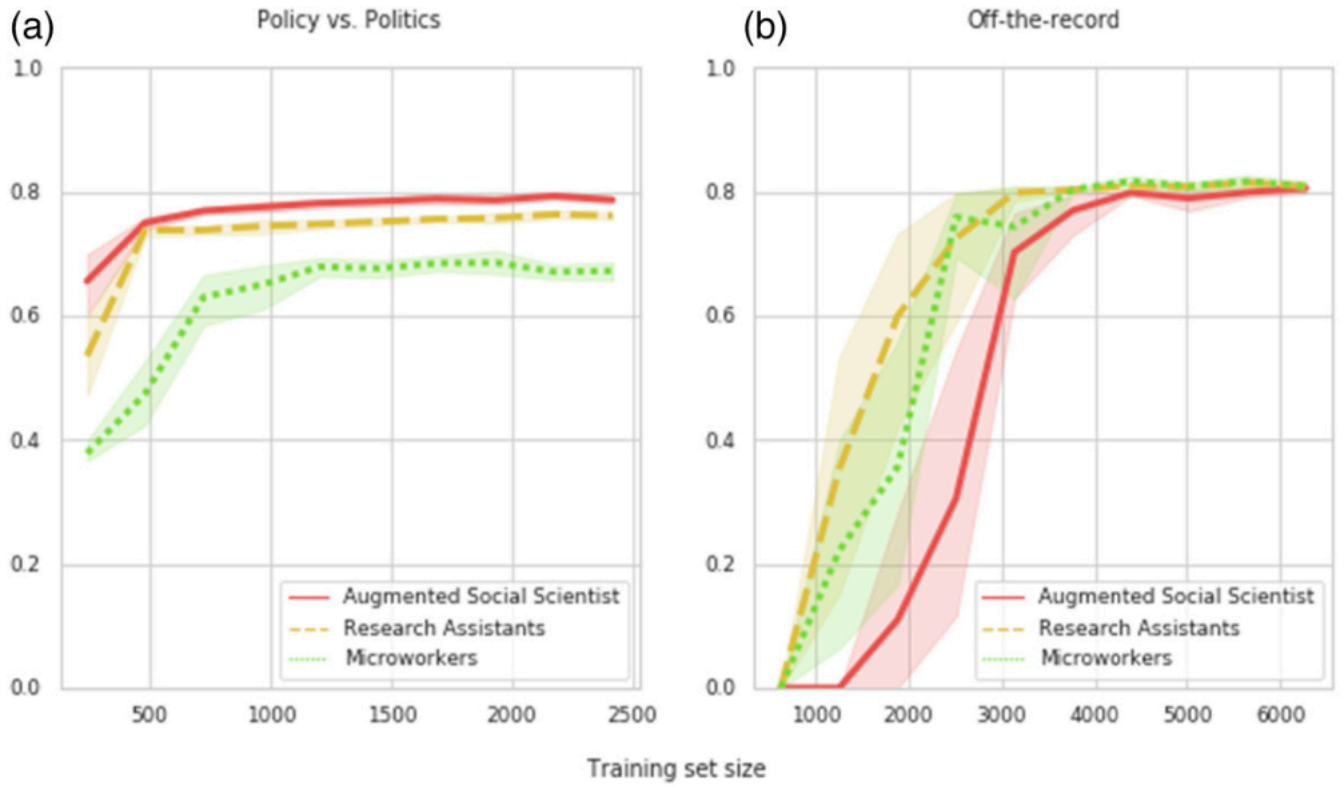
"The paper finds that GPT-4 achieves higher accuracy than both supervised models and human coders across all languages and country contexts. In the US context, it achieves an accuracy of 0.934 and an inter-coder reliability of 0.982."

The Augmented Social Scientist: Using Sequential Transfer Learning to Annotate Millions of Texts with Human-Level Accuracy

Salomé Do[1,2], **Étienne Ollion**[3], **and Rubing Shen**[2,3]

- Claim: LLMs lower the cost of annotation – less training examples required, hence “experts” can annotate small samples
- How do fewer, but better (i.e., more accurate/valid) annotations by experts (the researchers) augmented by LLM hold up against more but potentially biased annotations (all annotations made by research assistants)?
- How do training data generated by researchers hold up against training data generated by research assistants/ microworkers?
- Sequence extraction

	F1 – Policy vs. Politics	F1 – Off the record
Human – Microworkers	0.65	0.70
Human – Research assistants	**0.80**	**0.86**
Model without pre-training	0.67 [0.671, 0.673]	0.41 [0.390, 0.437]
Augmented social scientist (model with pre-training)	0.78 [0.781, 0.792]	0.82 [0.816, 0.834]


(a)
Policy vs. Politics
(b)
Off-the-record
1.0
0.8
0.6
0.4
0.2
0.0
500
1000
1500
2000
2500
3000
4000
5000
6000
Augmented Social Scientist
Research Assistants
Microworkers
Training set size

Politics as Usual? Measuring Populism, Nationalism, and Authoritarianism in U.S. Presidential Campaigns (1952–2020) with Neural Language Models

Bart Bonikowski, Yuchen Luo, and Oscar Stuhler

- Populism: hard to capture and multi-faceted concept
- "The relatively rare, polysemic, and variable frames in our study had previously been difficult to capture at scale because of the inadequacy of traditional machine learning methods and the shortcomings of dictionary-based approaches"

IN A SIMILAR VEIN: NATURAL LANGUAGE INFERENCE (NLI) – HYPOTHESIS TESTING

- NLI: determining the logical relationship between two pieces of text – a premise and a hypothesis
- Does the hypothesis…
  - …entail (logically follow from) the premise,
  - contradict the premise,
  - or is it neutral (neither entails nor contradicts)?

Example:

- Premise: “The cat is sleeping on the couch” | Hypothesis: "There is a cat in the house"
  => Relationship: ENTAILMENT (couches are in houses; if cat on couch, cat in house)
- Premise: “The cat is sleeping on the couch” | Hypothesis: "The cat is playing outside”
  => Relationship: CONTRADICTION (cats can’t play while sleeping)
- Premise: “The cat is sleeping on the couch” | Hypothesis: “The cat is dreaming”
  => Relationship: NEUTRAL (might or might not be dreaming while sleeping)

IN A SIMILAR VEIN: NATURAL LANGUAGE INFERENCE (NLI) – HYPOTHESIS TESTING

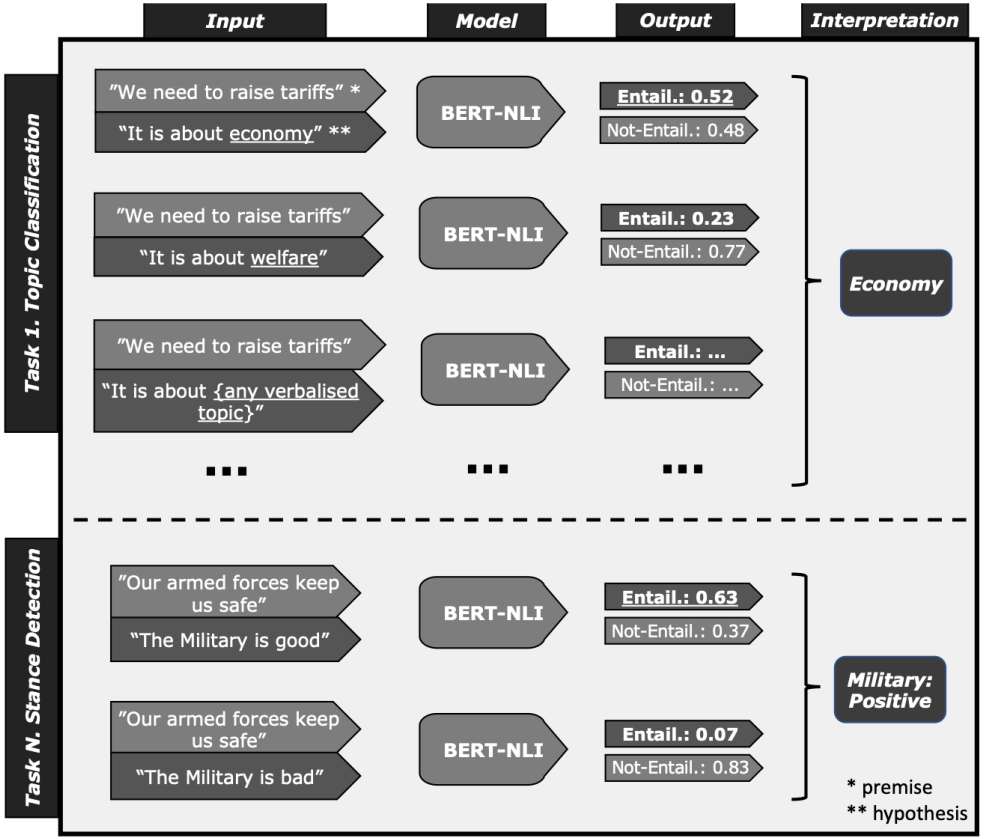
- NLI: determining the logical relationship between two pieces of text – a premise and a hypothesis
- Does the hypothesis…
  - …entail (logically follow from) the premise,
  - contradict the premise,
  - or is it neutral (neither entails nor contradicts)?

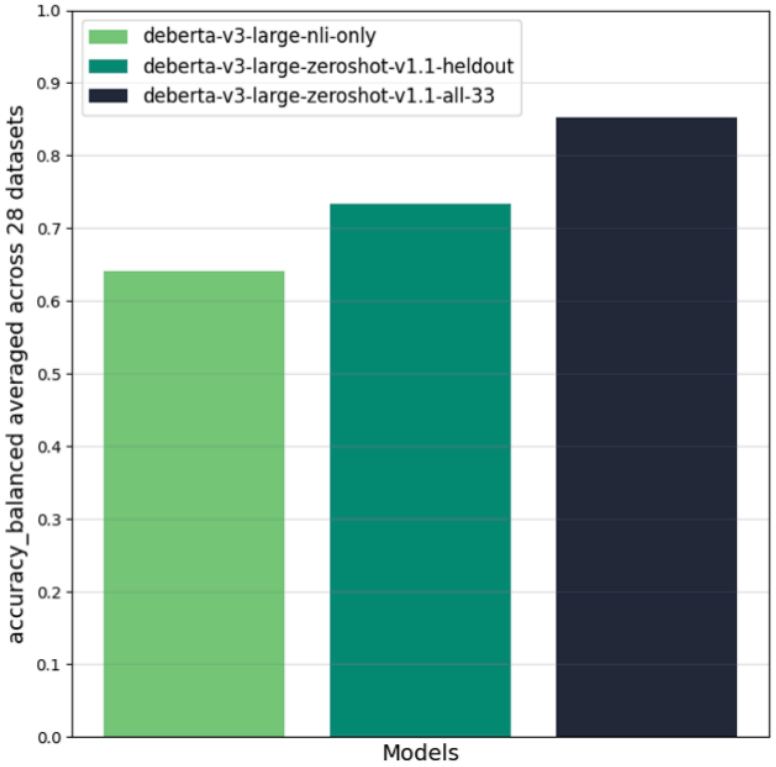
Potential use:

- Use it to classify text
- Example (current BA):
  - Premise: Titles of job ads
  - Hypothesis: The job ads contains gender-neutral language

Building Efficien … age Inference

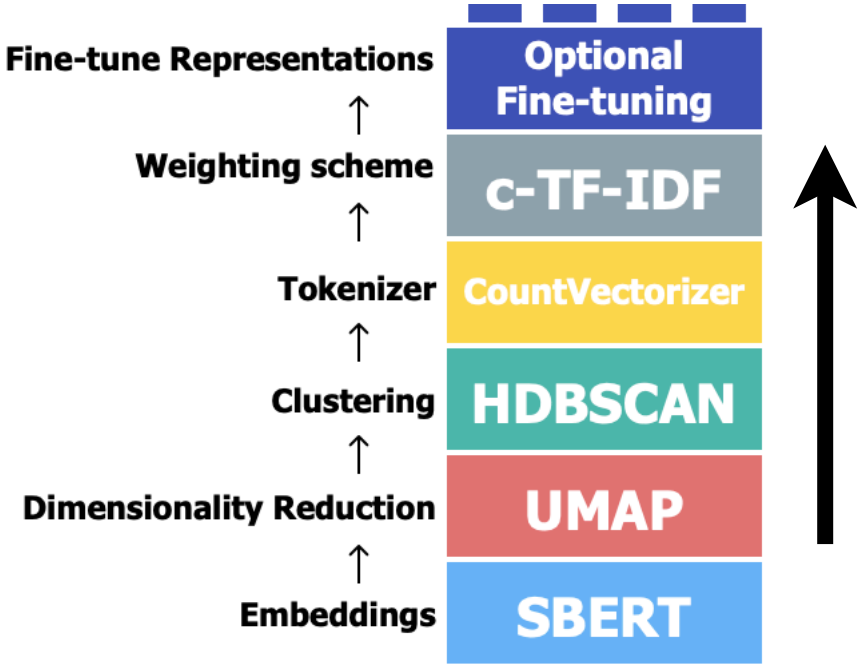
Moritz Laur … Welbers‡





deberta-v3-zeroshot-v1.1-all-33: fine-tuned with up to 500 examples

HOW ABOUT UNSUPERVISED TASKS: BERTOPIC



Promise: flexible framework

- Can use different base models for, e.g., language understanding
- Possibilities:
  - Prime it with topics (seeded topic model)
  - Provide training examples (supervised topic model)
  - Not only use text, but, e.g., images (multi-modal topic modeling)
  - Model topics over time (dynamic topic modeling)

CONCLUSION

- Python!
- Computationally expensive (GPUs!)
  => environmental impact
  “the process of building and testing a final paper-worthy model required training 4,789 models over a six-month period. … it emitted more than 78,000 pounds and is likely representative of typical work in the field.”
  => flight to NYC and back: 5,000 pounds/person // FINETUNING IS LESS COSTLY
- Same principles as with “classic” methods apply (validate etc.)
- Usually: better performance though

THURSDAY

- Python!
- Look at different, transfer learning power models and compare them to their BoW equivalents
  - BERTopic (ft. TayTay)
  - BERT for classification (w/ different base models)
  - LLMs with ollama for classification (in R)

UNIVERSITÄT
LEIPZIG

MERCI

Felix Lennert
Institut für Soziologie

felix.lennert@uni-leipzig.de
www.uni-leipzig.de