# Potentials and pitfalls of transformer-based text classification for the social sciences

Ines Rehbein, Mannheim University

GESIS Fall Seminar on Computational Social Science

Sep 24, 2024

# Outline

# Who am I

- MA in Linguistics, Psychology and Computer Science
- PhD in Computing (NLP) at Dublin City University
- PostDoc at Mannheim University working at the interface of Computational Linguistics, NLP and Political Science

# What's this talk (not) about

- This is *not* an introduction to transformers
- Instead, I'll focus on the pitfalls for using transformers for supervised text classification
- We will talk about:
  1. Bias and spurious correlations
  2. Robustness and generalisability
  3. Reliability

# The evolution of transformers

- First developed by Vaswani et al. (2017): **Attention is all you need**

→ model learns to focus on (attend to) the most relevant parts in the input
  - scaled dot-product attention
  - multi-head attention

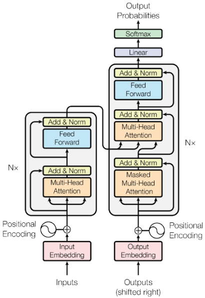


Figure 1: The Transformer - model architecture.

- Variants of the transformer:
  - encoder-only models (BERT, RoBERTa, ...)
  - encoder-decoder models (BART, T5, ...)
  - decoder-only models (GPT-[1-4], ChatGPT, Llama, Bloom, ...)

# The evolution of transformers

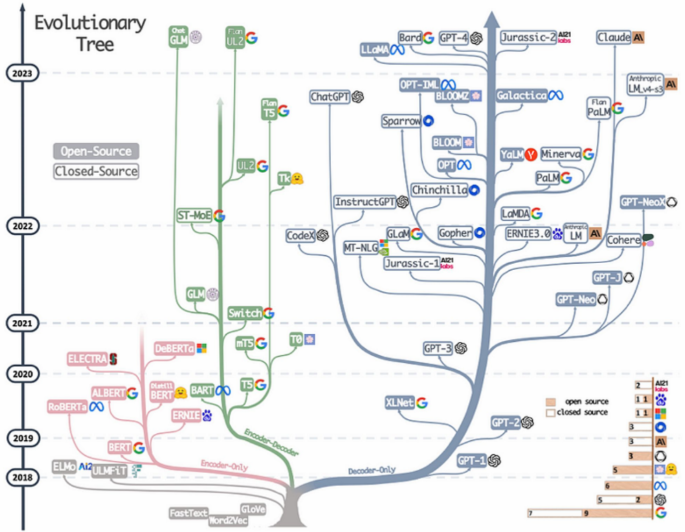


Image from the IJCAI-2023 Tutorial
https://isakzhang.github.io/talks/ijcai23-sentiment-tutorial.html

# Encoder-only transformers for text classification

- This talk focusses on encoder-only transformers for supervised text classification

- How do they work?
  1. Pre-train model on large data (self-supervised)
  2. Fine-tune on specific task data that represents a source population
     $\rightarrow$ learn mapping between features and labels

     The burgers here are **amazing.** — POSITIVE

     However, the appetizers were **just okay,** — NEUTRAL

     and the service was **terrible.** — NEGATIVE

  3. Apply learned mapping to data from unseen target population and predict labels

# Sucess story of transformers for text analysis

- What makes them so successful?
  - Attention mechanism
  - Context-sensitive embeddings
  - Can be pretrained on large data to obtain general knowledge about language and the world
  - Often small task-specific datasets suffice to get decent results

# Sucess story of transformers for text analysis

- What makes them so successful?
  - Attention mechanism
  - Context-sensitive embeddings
  - Can be pretrained on large data to obtain general knowledge about language and the world
  - Often small task-specific datasets suffice to get decent results

- Disadvantages:
  - Black-box → have we learned the right thing?
  - Results often hard to interpret
  - Models usually pick up on biases in the training data

# Outline

# Bias and spurious correlations

- Transformers are prone to learn biases
  - construct-validity bias
  - content-validity bias

# Bias and spurious correlations

- Transformers are prone to learn biases
  - construct-validity bias
  - content-validity bias

- **Example:** Hate speech detection on Twitter

  RQ: Is there a correlation between education and use of hate speech?

# Construct-validity bias

## Construct validity:

- whether a model accurately measures what it was designed to measure (Cronbach and Meehl, 1955)

> A test is valid for measuring an attribute if and only if (a) the attribute exists and (b) variations in the attribute causally produce variations in the outcomes of the measurement procedure. (Boorsboom, 2005)

(also see Riezler & Hagmann, 2024)

## Possible causes of construct-validity bias:

$\Rightarrow$ Bias in the training data can cause a model to learn spurious correlations instead of causal features

# Construct-validity bias: Biased datasets

- On Twitter: ca. 0.1–3% of abusive tweets (Founta et al. 2018)
- Selection bias due to keyword-based sampling:
  - abusive terms + identity groups, e.g. gay, Jew, woman
  
  → strong correlation between identity terms and label

# Construct-validity bias: Biased datasets

- On Twitter: ca. 0.1–3% of abusive tweets (Founta et al. 2018)
- Selection bias due to keyword-based sampling:
  - abusive terms + identity groups, e.g. gay, Jew, woman
  - → strong correlation between identity terms and label
- Author bias:
  - instances from one class are predominantly sampled from the same author (e.g., U.S. presidential speeches)
  - → strong correlation between author style and label
- Topic bias:
  - instances from one class are predominantly sampled from specific topic(s)
  - → strong correlation between topic and label

(see Dixon et al. 2017, Wiegand et al. 2019 for a detailed discussion of dataset biases)

## Content-validity bias

### Content validity:

- How well the model predicts *all* aspects of the construct
  - explicit vs. implicit language

### Possible causes of content-validity bias:

$\Rightarrow$ training data represents only some aspects of the construct

# Content-validity bias

## Content validity:

- How well the model predicts *all* aspects of the construct
  - explicit vs. implicit language

## Possible causes of content-validity bias:

$\Rightarrow$ training data represents only some aspects of the construct

- Implications for analysis:
  - use of explicit/implicit abusive language might have different distribution for different groups (age, education)
  - higher prediction error for implicit abusive language might result in wrong conclusions for our RQ

# Outline

Introduction

Bias and spurious correlations

Robustness and generalisability

Recommendations

References

# Robustness and generalisability

- When is it safe to apply a fine-tuned model to predict labels on new data?
- Example: German Sentiment classifier (Guhr et al, 2020)
  - trained on large datasets
  - covers different domains
  - paper has been peer-reviewed
  - paper reports high F1 score of 97% for BERT model

# Robustness and generalisability

- When is it safe to apply a fine-tuned model to predict labels on new data?
- Example: German Sentiment classifier (Guhr et al, 2020)
  - trained on large datasets
  - covers different domains
  - paper has been peer-reviewed
  - paper reports high F1 score of 97% for BERT model

Sounds good, but...

# Robustness and generalisability

... let's see how well the classifier performs on some test examples.

# Robustness and generalisability

```python
# import the library
from germansentiment import SentimentModel

# initialise the model
model = SentimentModel()
```

# Robustness and generalisability

```
# import the library
from germansentiment import SentimentModel

# initialise the model
model = SentimentModel()
```

```
#          The pizza was very good.     The pizza wasn't very good.
texts = ["Die Pizza war sehr gut.", "Die Pizza war nicht sehr gut."]

print(model.predict_sentiment(texts))
```

```
['positive', 'negative']
```

# Robustness and generalisability

```
# import the library
from germansentiment import SentimentModel

# initialise the model
model = SentimentModel()
```

```
#          I'm not saying that the pizza wasn't good.
texts = ["Ich sage nicht, dass die Pizza nicht gut war."]

print(model.predict_sentiment(texts))
```

```
['negative']
```

# Robustness and generalisability

```python
# import the library
from germansentiment import SentimentModel

# initialise the model
model = SentimentModel()
```

```python
#       If you're looking for a good pizza, go to Mario.
texts = ["Wenn Du eine gute Pizza suchst, geh zu Mario."]

print(model.predict_sentiment(texts))
```

```
['neutral']
```

# Robustness and generalisability

```python
# import the library
from germansentiment import SentimentModel

# initialise the model
model = SentimentModel()
```

```python
#          I thought the pizza was good, Petra didn't like it.
texts = ["Ich fand die Pizza gut, Petra hat sie nicht geschmeckt."]

print(model.predict_sentiment(texts))
```

```
['negative']
```

# Robustness and generalisability

```python
# import the library
from germansentiment import SentimentModel

# initialise the model
model = SentimentModel()
```

```python
#          Was the pizza good?   Wasn't the pizza good?
texts = ["War die Pizza gut?", "War die Pizza nicht gut?"]

print(model.predict_sentiment(texts))
```

```
['positive', 'negative']
```

# Robustness and generalisability

```
# import the library
from germansentiment import SentimentModel

# initialise the model
model = SentimentModel()
```

```
#        The style of the movie fluctuates between pubescent and vulgar.
texts = ["Der Stil des Films schwankt zwischen pubertär und vulgär."]

result = model.predict_sentiment(texts)
print(result)
```

```
['neutral']
```

# Robustness and generalisability

```
#        Pulp Fiction is one of the most successful independent films of its time.
texts = ["Pulp Fiction ist einer der erfolgreichsten Independentfilm seiner Zeit.
 ↪"]

result = model.predict_sentiment(texts)
print(result)
```

```
['neutral']
```

# Robustness and generalisability

Never trust a model that you haven't validated on your data!!!

# Outline

# Take-home message

- Models are biased and learn spurious correlations
  - Does the model really predict what you are interested in?
  - Does the train data capture all aspects of the construct?
  - Does the model have different error rates for different target groups?
- Generalisability:
  - Is the training data representative of the target population?
  - Does the model generalise well to out-of-distribution data?
- Reliability:
  - Check for model consistency $\rightarrow$ train $N$ instantiations of the same model and report stdev

# Take-home message

- Models are biased and learn spurious correlations
  - Does the model really predict what you are interested in?
  - Does the train data capture all aspects of the construct?
  - Does the model have different error rates for different target groups?
- Generalisability:
  - Is the training data representative of the target population?
  - Does the model generalise well to out-of-distribution data?
- Reliability:
  - Check for model consistency $\rightarrow$ train $N$ instantiations of the same model and report stdev

## Recommendations:

- Never rely on reported scores for fine-tuned models
- Always validate the model for your task/data

Thanks for listening!

Questions?

# Outline

# References I

Lucas Dixon, John Li, Jeffrey Sorensen, Nithum Thain, and Lucy Vasserman.
Measuring and mitigating unintended bias in text classification.
In *Proceedings of the National Conference on Artificial Intelligence (AAAI)*, San Francisco, CA, USA, 2017.

Oliver Guhr, Anne-Kathrin Schumann, Frank Bahrmann, and Hans Joachim Böhme.
Training a broad-coverage German sentiment classification model for dialog systems.
In Nicoletta Calzolari, Frédéric Béchet, Philippe Blache, Khalid Choukri, Christopher Cieri, Thierry Declerck, Sara Goggi, Hitoshi Isahara, Bente Maegaard, Joseph Mariani, Hélène Mazo, Asuncion Moreno, Jan Odijk, and Stelios Piperidis, editors, *Proceedings of the Twelfth Language Resources and Evaluation Conference*, pages 1627–1632, Marseille, France, May 2020. European Language Resources Association.

Michael Wiegand, Josef Ruppenhofer, and Thomas Kleinbauer.
Detection of Abusive Language: the Problem of Biased Datasets.
In Jill Burstein, Christy Doran, and Thamar Solorio, editors, *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)*, pages 602–608, Minneapolis, Minnesota, June 2019. Association for Computational Linguistics.

Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin.
Attention is all you need.
In *Advances in neural information processing systems*, pages 5998–6008, 2017.

L. J. Cronbach and P. E. Meehl.
Construct validity in psychological tests.
*Psychological Bulletin*, 4(52):281–302, 1955.

# References II


Zeerak Waseem and Dirk Hovy.
Hateful symbols or hateful people? predictive features for hate speech detection on Twitter.
In Jacob Andreas, Eunsol Choi, and Angeliki Lazaridou, editors, *Proceedings of the NAACL Student Research Workshop*, pages 88–93, San Diego, California, June 2016. Association for Computational Linguistics.

Stefan Riezler and Michael Hagmann.
*Validity, Reliability, and Significance: Empirical Methods for NLP and Data Science - Second Edition*.
Synthesis Lectures on Human Language Technologies. Springer, 2024.

D. Borsboom.
*Measuring the Mind. Conceptual Issues in Contemporary Psychometrics*.
Cambridge University Press, 2005.

Antigoni Founta, Constantinos Djouvas, Despoina Chatzakou, Ilias Leontiadis, Jeremy Blackburn, Gianluca Stringhini, Athena Vakali, Michael Sirivianos, and Nicolas Kourtellis.
Large scale crowdsourcing and characterization of twitter abusive behavior.
*Proceedings of the International AAAI Conference on Web and Social Media*, 12(1), Jun. 2018.