Fine-tuning LLMs for Data Augmentation and Synthesis

Tobias Holtdirk, Anna-Carolina Haensch

SurvAI workshop

Plan for this session

- **Introduction to Fine-tuning LLMs** (25 min - Slides)
  - Overview of key concepts, tools, and techniques
- **Hands-on: Setting up the Pipeline** (30 min - Practical)
  - Code walkthrough and dataset preparation for fine-tuning
- **Discussion and Q&A** (20 min - Interactive)
  - Share experiences and troubleshoot during fine-tuning
- **Results Review and Closing** (5 min - Recap)
  - Inspect results and discuss insights

Synthetic data created with LLMs

Out of One, Many: Using Language Models to Simulate Human Samples

Published online by Cambridge University Press: **21 February 2023**

Lisa P. Argyle, Ethan C. Busby, Nancy Fulda, Joshua R. Gubler, Christopher Rytting and David Wingate

Show author details

Simple “prompting” LLM approaches/Silicon sampling

GLES
German Longitudinal Election Study



I am a women, 50 years old … which party did I vote for? I [INSERT]

I voted for the xy party.

Variable values for ~2000 voting-eligible participants in the post-election cross-section of the GLES

Created ~2000 prompts by inserting the values into our prompt template and prompt the respective LLM

Get back ~2000 filled-in prompts from the LLM

(von der Heyde et al. 2023)

Problems with silicon sample approaches

- Uniformity
  - Difficult to capture the diversity and inconsistency that characterize human individuals and groups
- Temporality
  - LLMs struggle with temporality (datasets they are trained on often lack accurate timestamps, older datasets) making it difficult to model time-sensitive cultural shifts
- Linguistic representation
  - Uneven performance across languages
- Limited sensory representation
  - LLMs trained only on text, limiting their ability to fully capture human experiences

Fine-tuning vs In-context learning

In-context learning
The LLM “learns” to perform a task at inference time, e.g., zero-shot, few-shot

- Best proprietary models are designed to be used in-context
- Less technical knowledge needed
- Less time intensive

Fine-tuning
The LLM “learns” by changing the weights while training on new data

- Best performance for specific tasks
- Not that prompt dependent
- Inference efficiency
- Open-source models are used -> Works with private data

Hernandez et al. (2022)

Fine-tuning vs Training from scratch

Models that are pre-trained need less additional task specific data to have similar performance to models trained from scratch

- ⇨ less training time/compute needed
- ⇨ better for data scarce applications

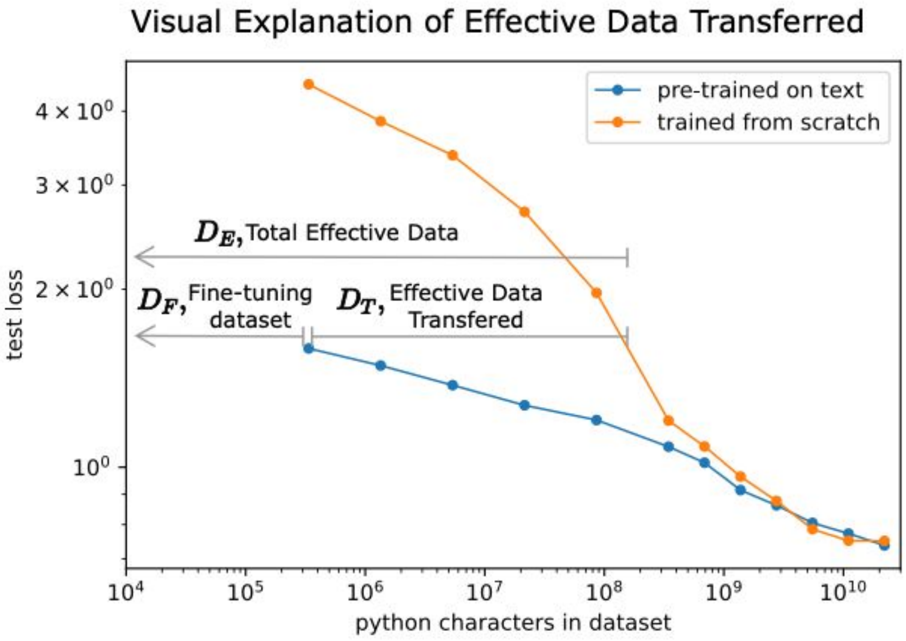


Figure 1 **We display the performance of a 40M parameter transformer model on python, both trained from scratch on python and pre-trained on text then fine-tuned on python.** D_T is the amount of additional python characters that a from-scratch model of the same size would have needed to achieve the same loss on python as a fine-tuned model. In the labeled example, we see that for a 40M parameter transformer fine-tuned on 3e5 characters, D_T is approximately 1000x bigger than D_F. The less fine-tuning data is available, the more pre-training helps.

Quality vs Quantity

Performance scales more with data quality than with data quantity

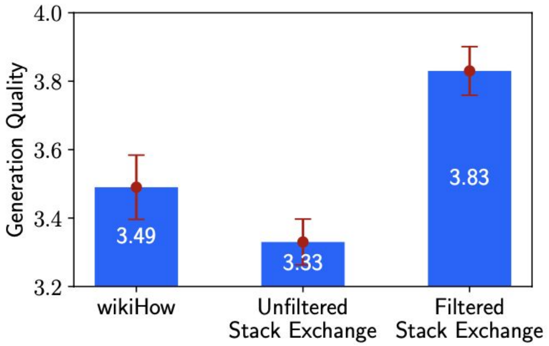


Figure 5: Performance of 7B models trained with 2,000 examples from different sources. **Filtered Stack Exchange** contains diverse prompts and high quality responses; **Unfiltered Stack Exchange** is diverse, but does not have any quality filters; **wikiHow** has high quality responses, but all of its prompts are "how to" questions.

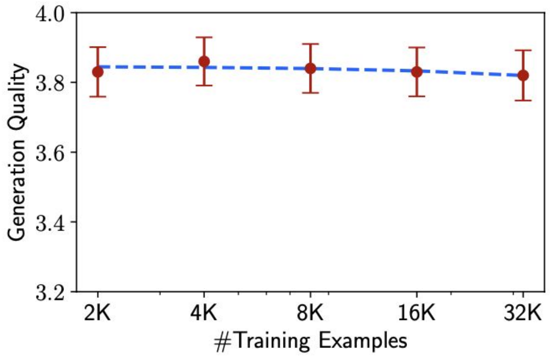


Figure 6: Performance of 7B models trained with exponentially increasing amounts of data, sampled from (quality-filtered) Stack Exchange. Despite an up to 16-fold increase in data size, performance as measured by ChatGPT plateaus.

Zhou et al. (2023)

Fine-Tuning Large Language Models to Simulate German Voting Behaviour

Motivation

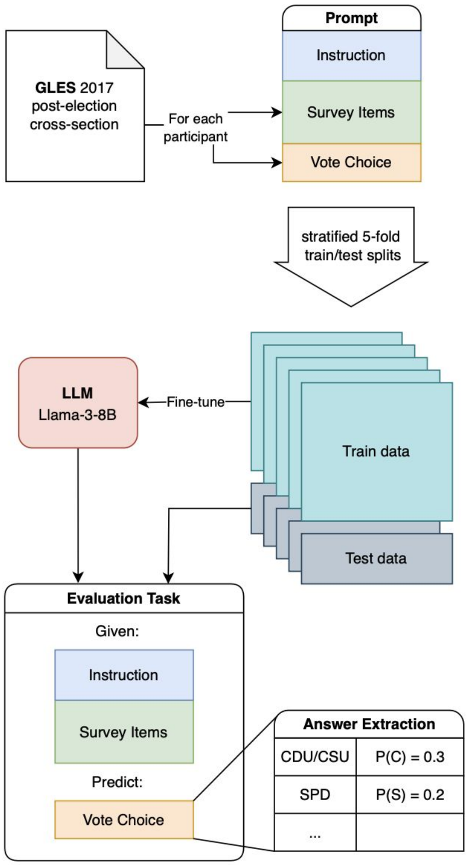
- Try to improve on the GPT-3.5 results from *von der Heyde et al. (2024)*
- Background knowledge of LLMs has potential for missing data problems *(Narayan et al., 2022)*
- Improvements in computational efficiency of fine-tuning with QLoRA *(Dettmers et al., 2023)*
- Open-source models are catching up in performance *(Dubey et al., 2024)*

Implementation

- Train LLMs on prompts generated from 2017 GLES survey data and predict the participants vote choice
- Answer the following research questions:
    - **RQ1**: Do fine-tuned LLMs offer a significant **advantage over zero-shot LLMs** in predicting voting choices in Germany?
    - **RQ2**: Are fine-tuned LLMs **more effective than established methods** for addressing missing data problems in survey research?

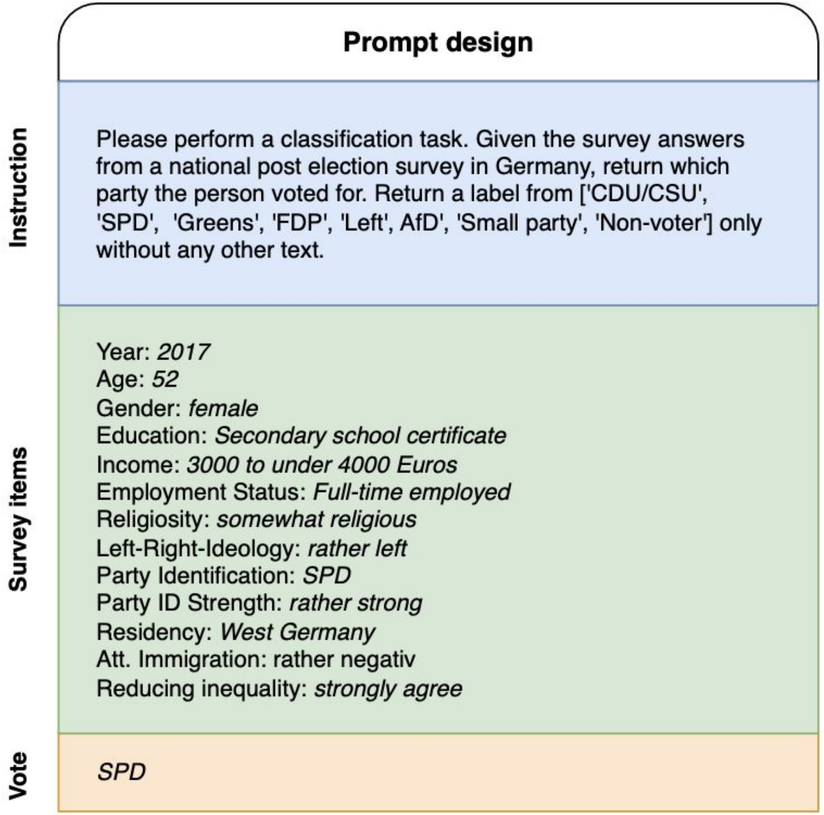
Method: Overview

1. German election survey data: GLES post-election cross-section cumulation 2017
2. We select 12 survey items that were most commonly associated voting behaviour
3. We design an Instruction prompt for each participant
4. We split the data into train-test sets
5. We fine-tune a LLM on the train data
6. We evaluate by letting the fine-tuned model predict the vote choices of the hold-out participants



Method: Prompt Design

- The instruction is added for a strong zero-shot baseline
- Survey questions and answers are reduced to short “item: answer” pairs



Method: Experiments

EQ1 Comparison to zero-shot prediction

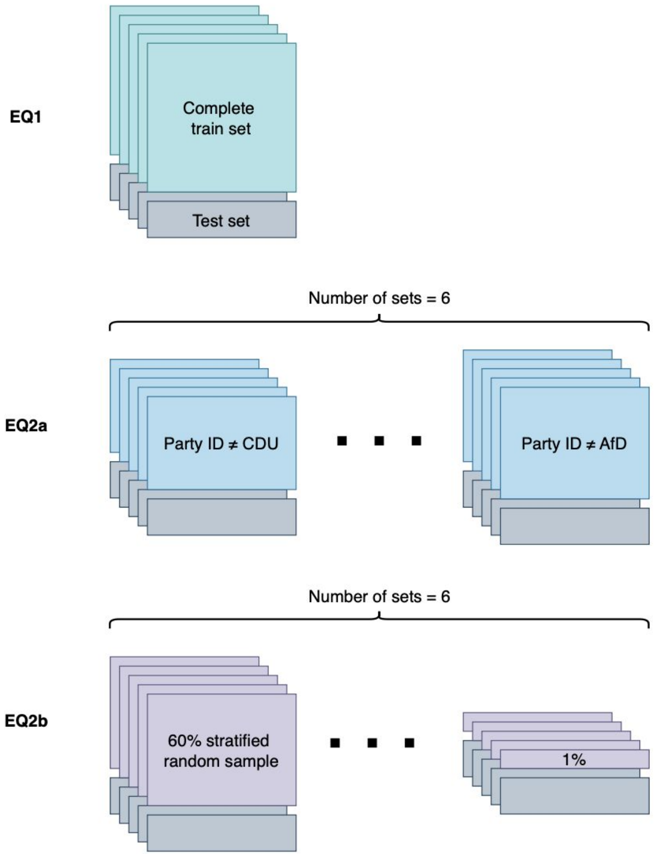
- Data-subsets of stratified 5-fold train/test splits
- Train Llama-3-8B on the train-set
- Evaluate mean performance of fine-tuned Llama against zero-shot Llama and the GPT-3.5 performance reported by *von der Heyde et al. (2024)*

EQ2a Systematic non-responses

- Exclude survey respondents that identify with a certain party
- Evaluate on the same test set as EQ1, against different established tabular data classifiers

EQ2b Sample efficiency

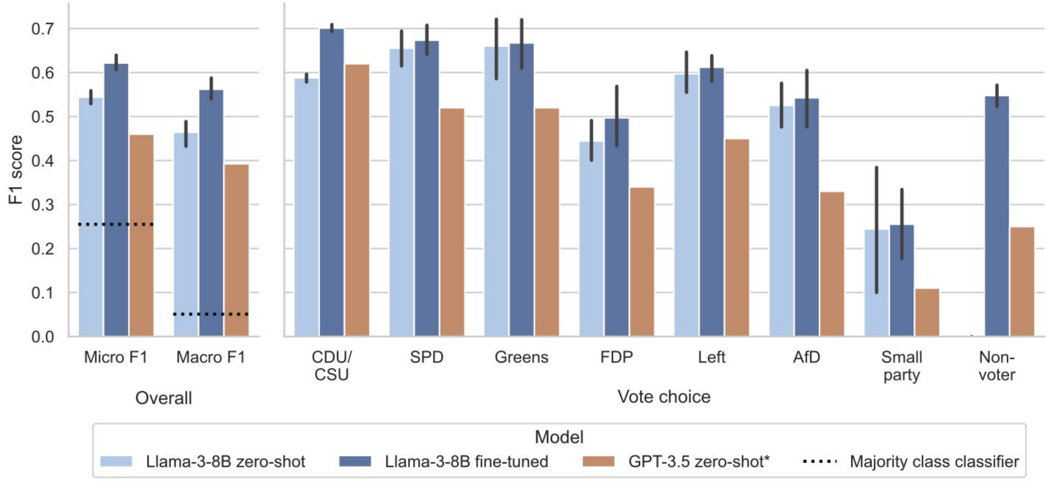
- Exclude a certain ratio of respondents in the training set (stratified)
- Evaluate on the same test set as EQ1, against different established tabular data classifiers



Results: RQ1

- The fine-tuned Llama-3 model outperforms the zero-shot models for all parties
- The fine-tuned Model still struggles with the ideological diverse small parties
- LLMs tend to under-predict right-leaning parties
- The vote distribution of fine-tuned models fits the GELS distribution better (not pictured)

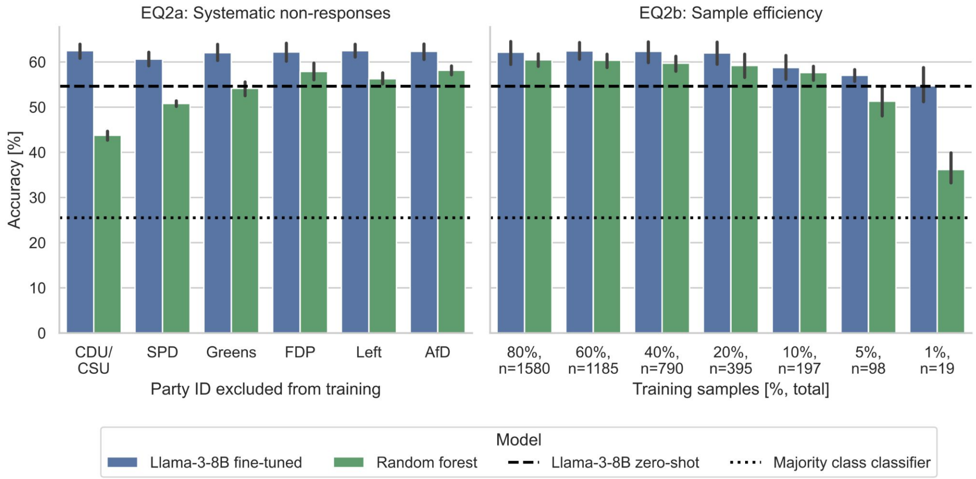
⇨ Fine-tuning increases performance on this task and can reduce bias in responses

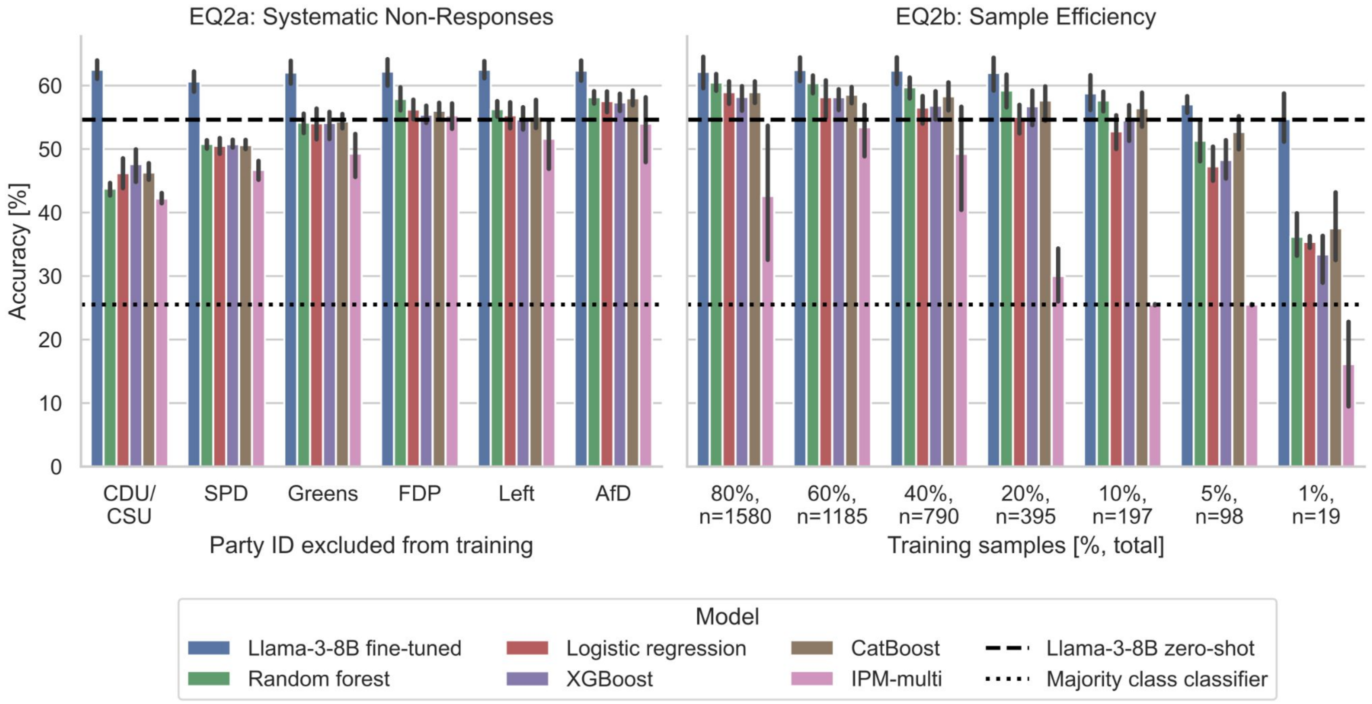


Results: RQ2

- The fine-tuned performance is better than traditional models when the training data is imbalanced.
- The fine-tuned performance is better than traditional methods with heavily reduced sample sizes.

⇨ Fine-tuned models might be able to help with biased or very limited survey data




EQ2a: Systematic Non-Responses
EQ2b: Sample Efficiency
Accuracy [%]
0
10
20
30
40
50
60
CDU/
CSU
SPD
Greens
FDP
Left
AfD
Party ID excluded from training
80%,
n=1580
60%,
n=1185
40%,
n=790
20%,
n=395
10%,
n=197
5%,
n=98
1%,
n=19
Training samples [%, total]
Model
Llama-3-8B fine-tuned
Random forest
Logistic regression
XGBoost
CatBoost
IPM-multi
Llama-3-8B zero-shot
Majority class classifier

Discussion

- **RQ1** Fine-tuned open-source LLMs are more effective in predicting voting behaviour than zero-shot approaches and can reduce their pre-trained political biases
- **RQ2** The fine-tuned model outperformed established methods, showing improved vote prediction when trained with biased data and remaining robust with reduced training data

⇨ Fine-tuned LLMs might enable imputation of previously hard-to-impute survey data and make new planned missing date survey designs possible

Limitations

- Fine-tuning is still considerably more computationally expensive than zero-shot inference and traditional imputation methods
- Requires a certain amount of participants as opposed to zero-shot approaches

Training on Twitter Data to Predict Survey Results

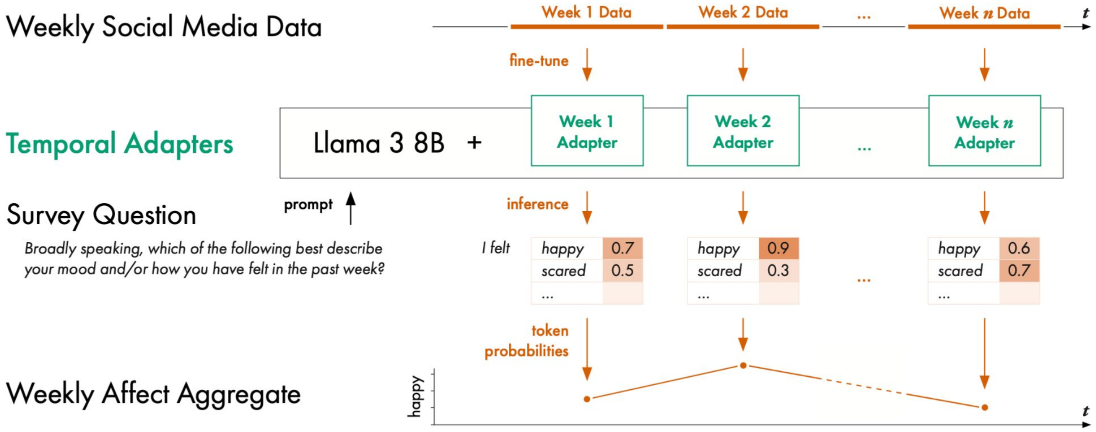


Figure 1: **Illustration of Temporal Adapters.** First, we gather weekly text data from a panel of Twitter users and fine-tune Temporal Adapters for Llama 3 8B with it. Then, we prompt the fine-tuned model with established survey questions, one week at a time, and extract affect aggregates from the answer options' token probabilities. Temporal Adapters enable longitudinal analyses of affect aggregates from social media data by temporally aligning LLMs.

Ahnert et al. (2024)

Training on Twitter Data to Predict Survey Results

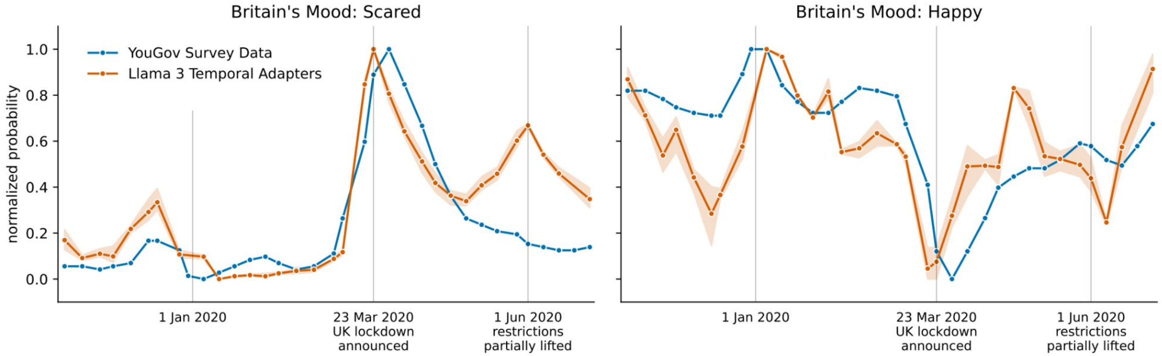


Figure 3: **Affect Aggregates Extracted from Temporal Adapters.** We extract answer probabilities by prompting a weekly fine-tuned Llama 3 8B with the same question wording as in the survey (YouGov 2024a), and compare them to the respective weekly survey data. The time series are min-max normalized and a 3 week rolling average is applied. The shaded orange area indicates minimum and maximum LLM answer probabilities across 3 training seeds. Our results descriptively show in the plot a similar trend of both signals and we find strong positive and significant ($p < 0.01$) cross-correlation between LLM probabilites and the survey data. Additional time series are provided in Figures 7 and 8 in the Appendix.

Ahnert et al. (2024)

Fine-tuning Resources

Huggingface

- Hosts open-source LLMs and Datasets
- Lots of libraries for working with LLMs, e.g., transformers, peft, lighteval

EleutherAI

- Non-profit focusing on training and evaluating completely open source
    - The Pile: open-source 886 GB dataset designed for training large language models
    - Pythia Scaling Suite: https://huggingface.co/collections/EleutherAI/pythia-scaling-suite-64fb5dfa8c21ebb3db7ad2e1
    - LMM evaluations: https://github.com/EleutherAI/lm-evaluation-harness

Links

Workshop

Jupyter Notebook:

https://github.com/tobihol/survai-finetuning

Paper Preprint:

https://doi.org/10.31219/osf.io/udz28

Personal

LinkedIn