Truth and AI: The Impact of Fine-Tuning GPT-3.5 on Lie Detection

Alejandra Olivia Cruces Andrews, Chelsie Adelle Renessa Romain,
Mario Alejandro Tapia Montero, Rovena Llapushi

Abstract

This study evaluates the performance of a fine-tuned GPT 3.5 model in detecting deceptive opinions. Inspired by Loconte et al. (2023), we utilize a dataset of opinions which are labeled as either truthful or deceptive, and apply a fine-tuning process to enhance GPT 3.5's deceptive opinion detection capabilities. Our findings reveal that the fine-tuned GPT 3.5 outperforms the FLAN-T5 model (presented in the aforementioned article) in accuracy, and exhibits the use of statistically significant linguistic features to distinguish between truthful and deceptive statements. This research contributes to the potential applications of large language models (LLM) in fields requiring deception detection.

Table of Contents

Introduction

In this document, we present a study on the performance of a fine-tuned version of GPT 3.5 for the task of lie detection. Our work is inspired by the paper "Verbal Lie Detection using Large Language Models" by *Loconte et al. (2023)*, who used a similar methodology but with a different model (FLAN-T5). We used Scenario 1 from their paper, which consists of a dataset composed by human opinions labeled as either truthful or deceptive.

For the fine-tuning process, a portion of this dataset was used for training different splits of the data and then a test set of opinions from the same dataset was used to assess the performance of the model. Our motivation was to investigate whether the fine-tuned version of GPT 3.5 could improve its ability to detect lies. We also compared its performance with the FLAN-T5 model reported in the cited article. This document describes the methodology and the results of our study, as well as the implications and limitations of our findings.

Materials and Methods

GPT 3.5 Architecture

GPT-n overview

GPT-n is a language model that operates from left to right, utilizing only the decoder portion of the transformer architecture, without the cross-attention layer. The process begins with word embeddings and positional encodings. Attention mechanisms are then applied to these initial inputs. Following this, several feed-forward layers are used, with regularization steps incorporated at the end. This entire process is encapsulated within a 12-layer transformer decoder. The model is trained using the standard method of cross-entropy loss.

Fine-tuning

The fine-tuning loss is composed of two parts: the loss specific to the task at hand, and the loss from language modeling. During the fine-tuning phase, the model’s structure remains unchanged, with the exception of the last linear layer. The input format varies depending on the task. Our focus is on single sentence classification. To classify individual sentences, the data is fed into the model as it was during training, and the label is predicted based on the final representation of the last input token.

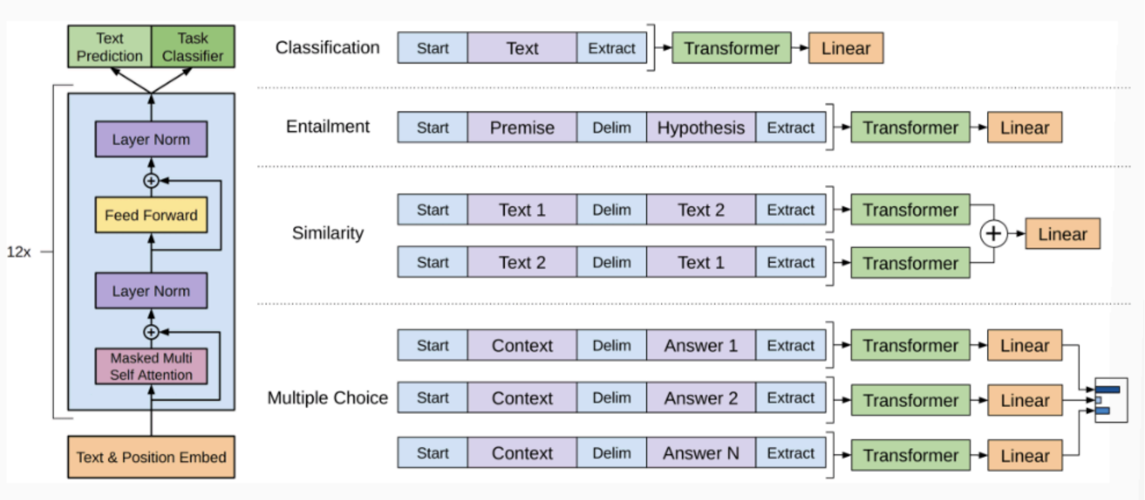


Figure 1. shows the architectural structure of GPT 3.5 model (OpenAI, 2023)

The revolution of GPT-3

The GPT-3 paper by OpenAI (2023), presents a novel concept known as in-context learning, which deviates from traditional artificial intelligence practices. In the conventional supervision paradigm, if we want to classify text as a binary problem, we first create a dataset of positive and negative examples and then train a custom model to make the binary distinction. While this model can be powerful, it may not scale to the complexity of human experience.

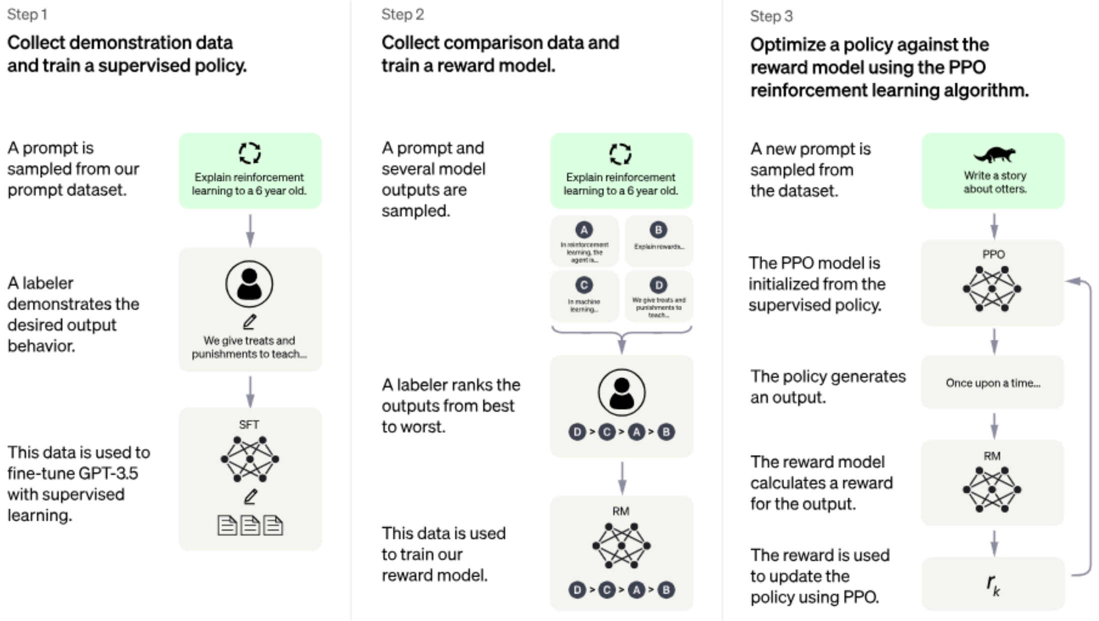
In-context learning promises that a single, large, frozen language model can serve all these purposes. We provide the model with examples of truthful and deceptive opinions, hoping that it will learn in-context about the distinction we’re trying to make.

The types of n-shot learning that can be introduced to the GPT-3 model include:

- ➢ Zero-shot learning: The model can predict the answer given only the task name with no examples.
- ➢ One-shot learning: In addition to the task name and description, we provide the model with one example, and the model will be able to predict the answer.

- ➢ Few-shot learning: A few examples are introduced to the model along with the task description.

Another significant innovation is the concept of “self-supervision”, a powerful mechanism for acquiring rich representations of form and meaning. In self-supervision, the model’s objective is to learn co-occurrence patterns in the sequences it is trained on. The model is simply learning to assign high probability to attested sequences. These models are thought of as generators, but the generation involves sampling from the model. This is powerful because self-supervision requires minimal human effort and has facilitated the rise of another important mechanism, large-scale pretraining.



Step 1

Collect demonstration data and train a supervised policy.

A prompt is sampled from our prompt dataset.

A labeler demonstrates the desired output behavior.

This data is used to fine-tune GPT-3.5 with supervised learning.

Step 2

Collect comparison data and train a reward model.

A prompt and several model outputs are sampled.

A labeler ranks the outputs from best to worst.

This data is used to train our reward model.

Step 3

Optimize a policy against the reward model using the PPO reinforcement learning algorithm.

A new prompt is sampled from the dataset.

The PPO model is initialized from the supervised policy.

The policy generates an output.

The reward model calculates a reward for the output.

The reward is used to update the policy using PPO.

Explain reinforcement learning to a 6 year old.
We give treats and punishments to teach...
SFT
Explain reinforcement learning to a 6 year old.
A
In reinforcement learning, the agent is...
B
Explain rewards...
C
In machine learning...
D
We give treats and punishments to teach...
D > C > A > B
RM
D > C > A > B
Write a story about otters.
PPO
Once upon a time...
RM
r_k

Figure 2. shows Open AI’s schematic representation of the GPT-n LLM (Randford, 2018)

The final component is the role of human feedback. The best models, referred to as Instruct models by OpenAI, are trained with more than just self-supervision. Two important aspects from the ChatGPT blog post diagram include:

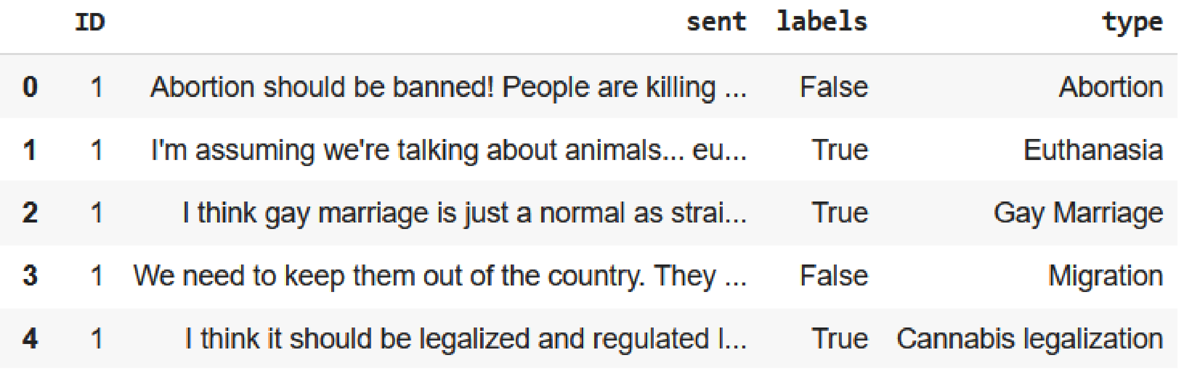
- ➢ The language model is fine-tuned on human-level supervision (Step 2), making binary distinctions about good and bad generations.

- ➢ In a second phase, the model generates outputs and humans rank all of the outputs the model has produced (Step 3). This feedback goes into a lightweight reinforcement learning mechanism.

In both of these phases, there are important human contributions that take us beyond the self-supervision step.

Methodology

We use the approach described in *Loconte et al. (2023)* to analyze the data set "Opinioni" under Scenario 1, and we fine-tune GPT 3.5 on it. We conduct 4-fold cross validation, splitting the data set into 75% for training and 25% for testing each time. We ensure that the same participant ID is only in either the training or the test set to avoid model bias.



	ID	sent	labels	type
0	1	Abortion should be banned! People are killing ...	False	Abortion
1	1	I'm assuming we're talking about animals... eu...	True	Euthanasia
2	1	I think gay marriage is just a normal as strai...	True	Gay Marriage
3	1	We need to keep them out of the country. They ...	False	Migration
4	1	I think it should be legalized and regulated l...	True	Cannabis legalization

Figure 3. shows some opinions taken from the dataset

To configure `GPT-3.5-turbo-0163`, we need to specify the following parameters in the API: System, User, and Assistant. These parameters are defined as follows:

- ➢ **System role** - "Your task is to evaluate if the user is truthful or not about their opinion."
- ➢ **User** - the opinion text
- ➢ **Assistant** - the label, True (T) or False (F)

We convert the data set to a *.json* file where each row, which contains one opinion, has the following format:

```
{"message": [{'role': 'system', 'content': System_role},
{'role': 'user', 'content': Opinion},
{'role': 'assistant', 'content': Labels}, …]}
```

We run 3 epochs for each cross validation. The steps are as follows:

1. We fine-tune the model with the training data set.
2. We test the fine-tuned model on the validation set.
3. We calculate the metrics: accuracy, precision, recall, F-score.

Finally, we performed an explainability analysis, following *Loconte et al. (2023)*, to understand the role of linguistic style in the model's classification process for differentiating truthful and deceptive statements. We follow the DeCLaRative stylometry analysis proposed by *Loconte et al. (2023).* This procedure involves taking 26 linguistic features for each statement (Table 4), and performing a permutation t-test. Finally, the effect size is presented as Common Language Effect Size (CLES).

Results

In the following sections, the results outputted by the fine-tuned model are presented. We interpret the results at model level (Table 1), and at topic level (Table 2). To compare the models by topic, we calculated each metric separately for each topic and then took the average of each metric across all models. This gave us a global metric for each topic.

Fine-tuned model

Table 1. Summary of the metrics for evaluating the models by CV fold

	Accuracy	**Precision**	**Recall**	**F-score**
Model 1	0.8816	0.8692	0.8971	0.8829
Model 2	0.8512	0.8904	0.8100	0.8483
Model 3	0.8688	0.8775	0.8548	0.8660
Model 4	0.8624	0.8558	0.8669	0.8571
Average value	0.8660	0.8732	0.8572	0.8636

Standard deviation	0.0110	0.0126	0.0313	0.0128

Table 2. Summary of the average metrics for evaluating the models by topic

	Accuracy	**Precision**	**Recall**	**F-score**
Abortion	0.8520	0.8788	0.8164	0.8460
Euthanasia	0.8680	0.8778	0.8579	0.8660
Immigration	0.8820	0.8935	0.8676	0.8803
Gay Marriage	0.8900	0.8943	0.8870	0.8893
Cannabis Legalization	0.8380	0.8254	0.8567	0.8400
Standard deviation	0.0213	0.0283	0.0258	0.0212

From Table 1, we can see the average accuracy of the model is 0.87 with a standard deviation of 0.0110. Independently of the split of the data set, the models perform similarly, suggesting that they are not overfitting to the training set and are generalizing results.

As we mentioned before, the opinions are divided into five topics, and we calculate the accuracy across the four models for each topic (Table 2). From the table, we can see that the Fine-tuned models in GPT 3.5 performs best on opinion about Gay Marriage and worst on Cannabis Legalization. The classifier has similar performance on Abortion, Euthanasia, and Immigration, with only slight differences in the metrics.

These results suggests that the classifier is more effective on topics that are more polarized and less ambiguous, such as Gay Marriage, than on topics that are more nuanced and complex, such as Cannabis Legalization. The accuracies vary more across the topics than across models. The standard deviation of accuracies by topic (Table 2) is 0.02 which is twice as large as the deviation across models (Table 1), 0.01.

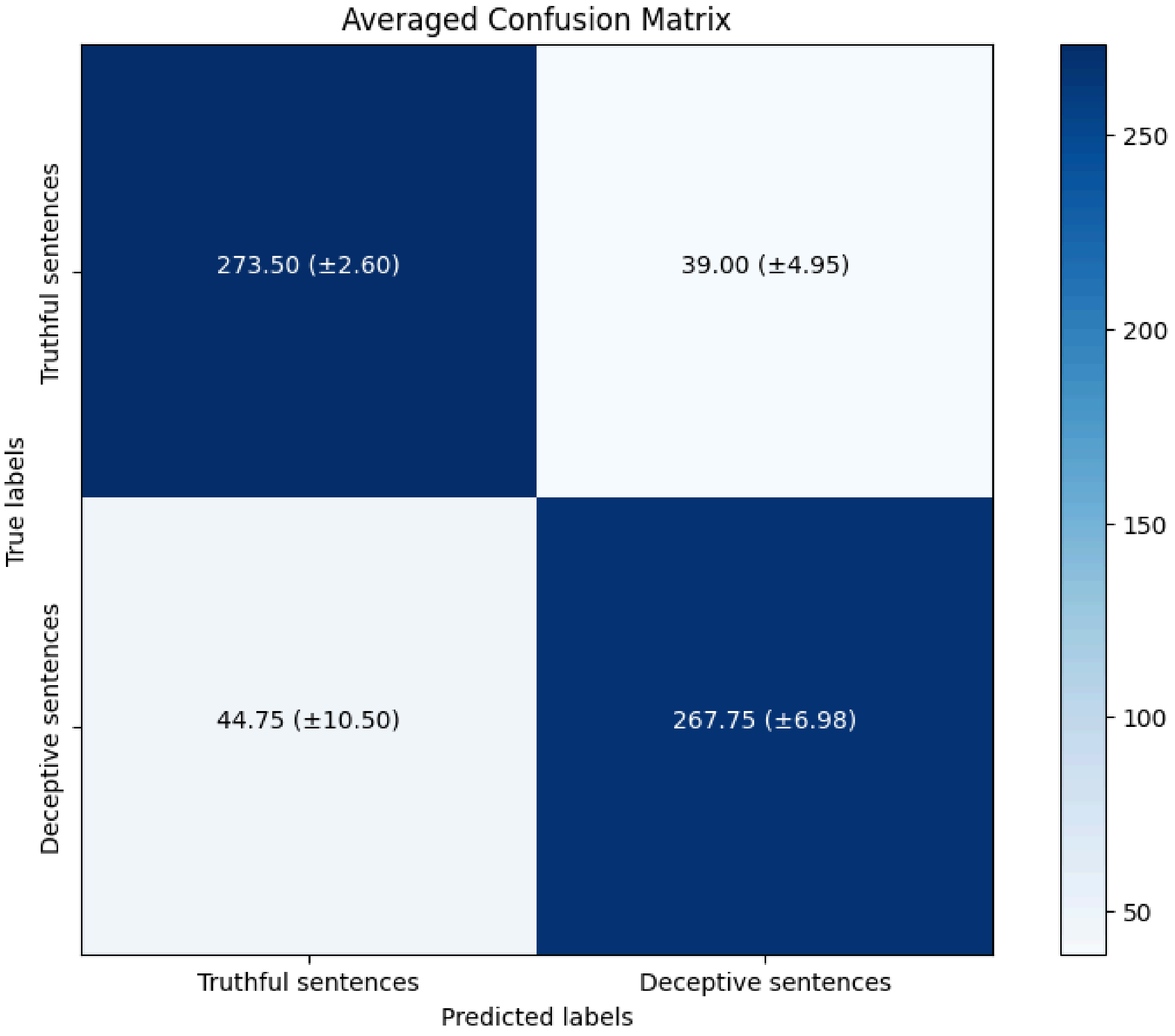


Figure 4. Shows the overall confusion matrix

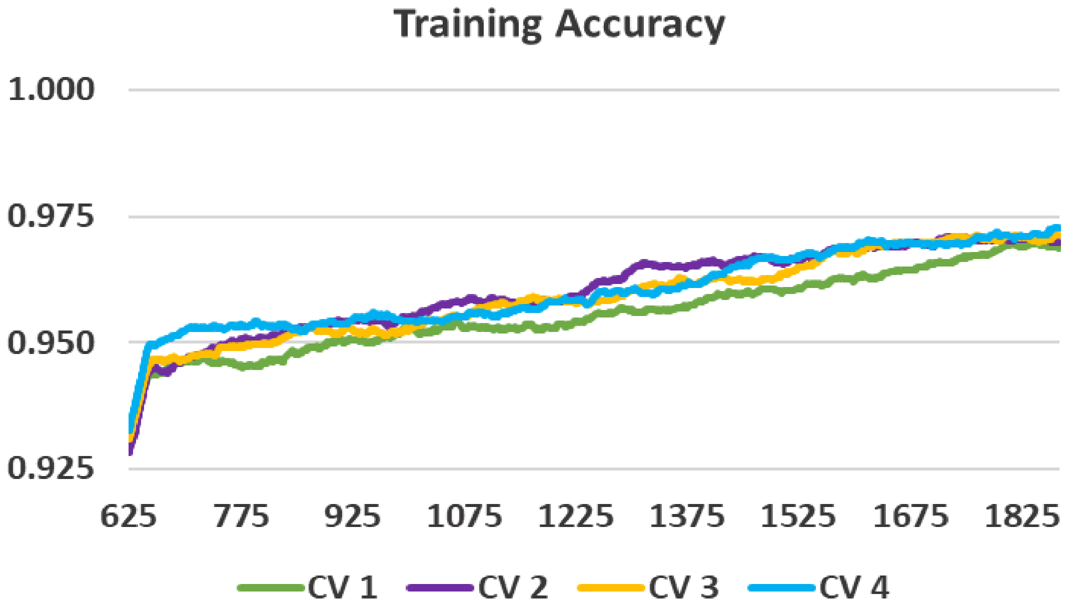


Figure 5(a). Shows the moving average training accuracy for each CV fold (*) Moving Average of 625 steps (1 epoch).

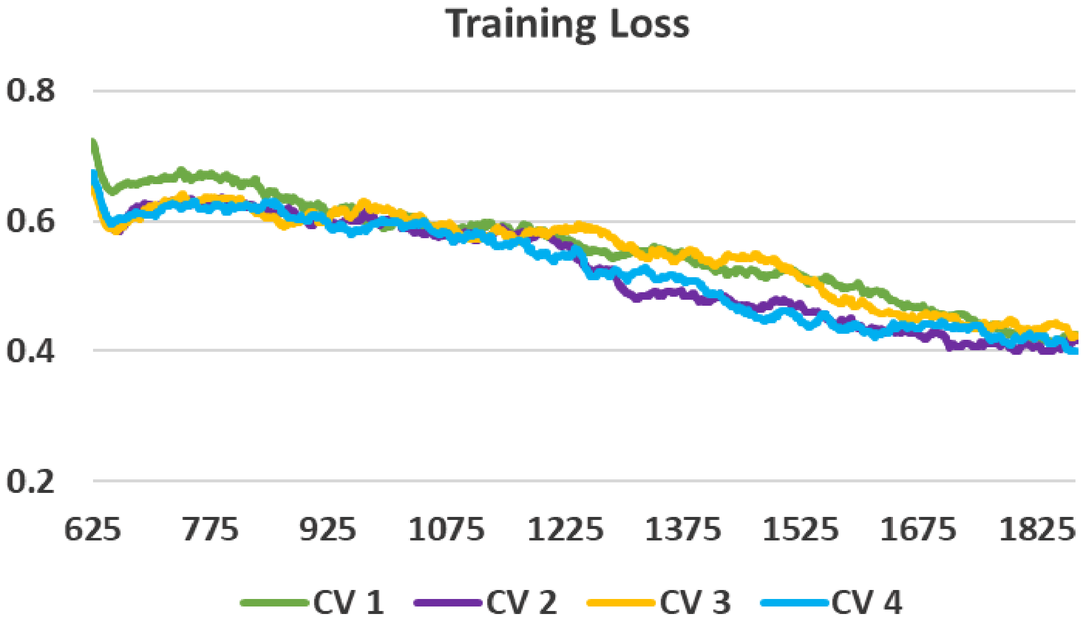


Figure 5(b). Shows the moving average training loss for each CV fold

For each step, the OpenAI API computes a mean token accuracy and a loss. Each step is a batch of three instances. As each step contains few instances, we calculate a moving average of 625 steps, equivalent to one epoch, for the mean token accuracy and a loss. In Figure 5(a) and (b) we can see how accuracy and loss evolve with steps, respectively. As expected we can see how the mean accuracy increases and the loss decreases.

Comparison with FLAN-T5

In the following table, a comparison between the performance of the FLAN-T5 model from *Loconte et al. (2023)* and our results is presented.

Table 3. Test set accuracy comparison between FLAN-T5 and fine-tuned GPT3.5 models on Scenario 1

Model	**Average accuracy**	**Standard deviation**
Flan-T5 small	0.8064	0.02
Flan-T5 base	0.8260	0.03
GPT-3.5-turbo-0163	0.8660	0.0110

These results indicate that, after a fine-tuning process, GPT3.5 Turbo has better generalization capabilities than FLAN-T5 models.

Discussion

Analysis of False Negatives

In comparing false statements (i.e. label = false) and true statements classified as false (i.e label = true and predicted label = false), across all splits the only feature with a significant p-value is the concreteness score in split 2 for which the null hypothesis was rejected.

Analysis of False Positives

In comparing true statements (i.e. label = true) and false statements classified as true (i.e label = false and predicted label = true), across all splits there were no linguistic features with a p-value of sufficient significance to reject the null hypothesis.

Cognitive Biases in GPT 3.5

Truth bias is a psychological tendency where people believe others are telling the truth, regardless of whether or not that person is actually lying or being untruthful. In terms of error classification, GPT 3.5 has a higher number of false positives compared to false negatives (Figure 4). Of all the incorrect predictions, GPT 3.5 was more likely to classify a result as true (T) regardless of whether the opinion was truthful or not. This can be interpreted as a truth bias of the LLM.

Analysis of Correctly Classified Statements

Linguistic features in Truthful and Deceptive statements that were accurately classified by GPT 3.5 are presented in Figure 6. The bar length represents the averaged Common Language Effect Size (CLES) among the 4-folds of linguistic features that survived post-hoc corrections. CLES, the Common Language Effect Size, is a non-parametric effect size, measuring the probability that a specific linguistic feature, in a picked-at-random truthful statement, will have a higher score than in a picked-at-random deceptive one.

For those statements whose true and predicted labels were congruent, there were a number of statistically significant linguistic features across all splits of the data. The top 10 of these are: the word count, number of syllables, concreteness score, grade level, readability, authenticity score, tone, percentage of words related to negative context, perceptual details, and self-reference.

The main three features that have more probability of being in a truthful statement than in a deceptive one are number of words, number of syllables, and the grade level required to understand the text and authenticity. On the other hand, the top three features with a lower probability of appearing in a truthful statement are concreteness, readability of the text, and number of words related to a negative sentiment. Overall, the CLES scores of linguistic features associated with truthfulness were consistently larger than the values scores associated with deception.

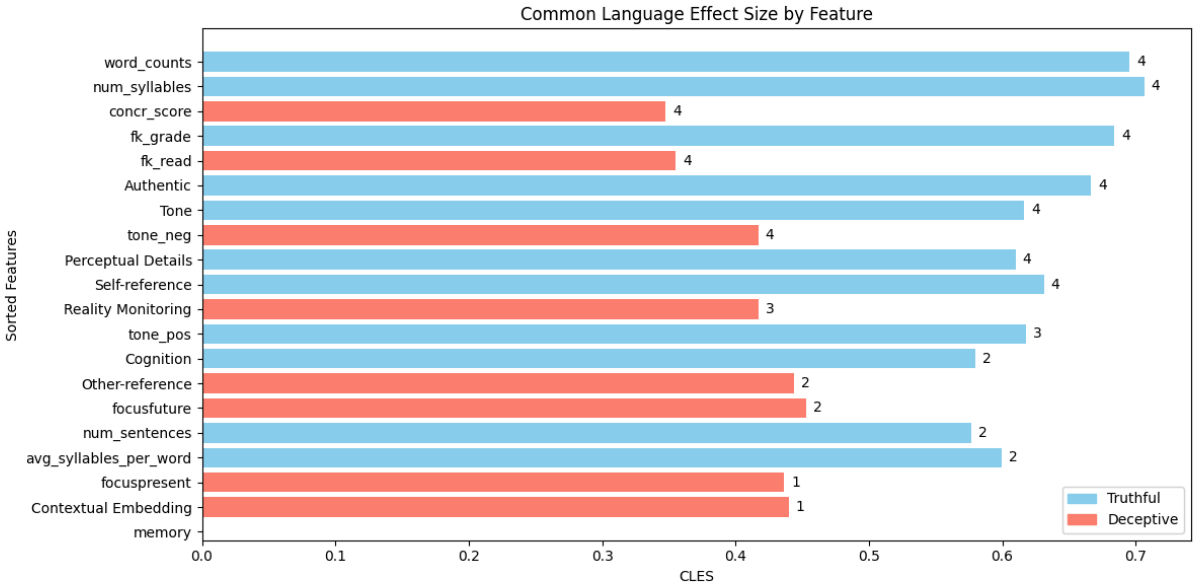


Figure 6. Common Language Effect Size by feature, sorted by frequency of significant appearances across splits.

Possible applications

A possible use case for this fine-tuned LLM is in the domains of Law enforcement and Forensics, where the accuracy of detecting deception is crucial. If the model produces too many false positives, it means that many liars are misclassified as truthful (i.e. guilty people are not caught). Conversely, if the model produces too many false negatives, it means that many truth-tellers are misclassified as liars (i.e. innocent people are wrongly accused). Therefore, it makes sense to use F-score as a metric to balance between precision and recall, aiming to reduce both false positives and false negatives. The model performs well, as indicated by an average F-score of 0.8636 and a low standard deviation of 0.0128. We believe that this outcome demonstrates the robustness and reliability of the model for assisting in these fields. The model could provide valuable insights and guidance for the practitioners and researchers in these domains.

Conclusion

In this document, we studied the performance of a fine-tuned version of GPT 3.5 on the task of lie detection inspired by the paper "Verbal Lie Detection using Large Language Models" by *Loconte et al. (2023)*. We used Scenario 1 from the paper and the opinion dataset, containing opinions labeled as truthful or deceptive. It is on this dataset that we fine tuned GPT 3.5.

We have found that the average accuracy (86.6%) of GPT 3.5 on the task of Lie Detection on opinions is higher than the average accuracy (82.6%) of FLAN-T5 based on the same task. Furthermore, we have found that the most significant linguistic features across all splits of the data are: the number of syllables, grade level, word count, authenticity score, self-reference, perceptual details, tone, percentage of words related to negative context, readability, and concreteness score.

Bibliography

Loconte, Riccardo & Russo, Roberto & Capuozzo, Pasquale & Pietrini, Pietro & Sartori, Giuseppe. (2023). Verbal lie detection using Large Language Models. Scientific Reports. 13. 10.1038/s41598-023-50214-0.

OpenAI. (2023). ChatGPT (Mar 14 version) [Large language model]. https://chat.openai.com/chat

Radford, A., Narasimhan, K., Salimans, T., & Sutskever, I. (2018). Improving language understanding by generative pre-training.

Appendix

Table 4. List and short description of the 26 linguistic features pertaining to the DeCLaRatiVE Stylometry technique.

Label	**Description**
num_sentences	Total number of sentences
word_counts	Total number of words
num_syllables	Total number of syllables
avg_syllabes_per_word	Average number of syllables per word
fk_grade	Index of the grade level required to understand the text
fk_read	Index of the readability of the text
Analytic	LIWC summary statistic analyzing the style of the text in term of analytical thinking (0–100)
Authentic	LIWC summary statistic analyzing the style of the text in term of authenticity (0–100)
Tone	Standardized difference (0-100) of 'tone_pos' - 'tone_neg'
tone_pos	Percentage of words related to a positive sentiment (LIWC dictionary)

Label	Description
tone_neg	Percentage of words related to a negative sentiment (LIWC dictionary)
Cognition	Percentage of words related to semantic domains of cognitive processes (LIWC dictionary)
Memory	Percentage of words related to semantic domains of memory/forgetting (LIWC dictionary)
focuspast	Percentage of verbs and adverbs related to the past (LIWC dictionary)
focuspresent	Percentage of verbs and adverbs related to the present (LIWC dictionary)
focusfuture	Percentage of verbs and adverbs related to the future (LIWC dictionary)
Self-reference	Sum of LIWC categories 'i' + 'we'
Other-reference	Sum of LIWC categories 'shehe' + 'they' +'you' Perceptual details Sum of LIWC categories 'attention' + 'visual' + ' auditory'+ 'feeling'
Contextual Embedding	Sum of LIWC categories 'space' + 'motion' + 'time' Reality Monitoring Sum of Perceptual details + Contextual Embedding + Affect - Cognition
Concreteness score	Mean of concreteness score of words
People	Unique named-entities related to people: e.g., 'Mary', 'Paul', 'Adam'
Temporal details	Unique named-entities related to time: e.g., 'Monday', '2:30 PM', 'Christmas'
Spatial details	Unique named-entities related to space: e.g., 'airport', 'Tokyo', 'Central park'
Quantity details	Unique named-entities related to quantities: e.g., '20%', '5 $', 'first', 'ten', '100 meters'