SCIPEDIA Journal of Dawning Research

Building a Conversational AI Medium to Enhance Psychotherapy Training with Virtual Patients

Michael Petrizzo

Abstract

Psychotherapists in training lack a standardized and formalized method of patient interaction for proper development of empathy, communication, and experience. Currently, training involves residents practicing with each other, where one acts as the patient and one as the psychotherapist, or with simulated patients -actors who replicate patient scenarios. Both methods have shortcomings in availability, reliability, and the accuracy in replicating real scenarios. This project attempted to create virtual patients by utilizing online patient transcripts through the fine-tuning of three modern Artificial Intelligence models, ChatGPT-4o, LlaMa-3.1v-405B, and Gemini 1.5 Pro as well as their miniature versions where applicable. A website interface was created to interact with the fine-tuned models for evaluation. The accuracy of the models was determined using cosine similarities to measure semantic relation between data and model outputs, ranging from 93.3% to 83.11% , with ChatGPT-4o Mini achieving the highest accuracy. These findings highlight the potential for virtual patients to serve as a more accessible, reliable, and effective training method for residents. Further evaluation and continual refinement remain necessary to address current limitations.

OPEN ACCESS

Published: 23/06/2025

INTRODUCTION

Psychotherapists, the umbrella category encompassing social workers, psychologists, psychiatrists, and general therapists, are vital supportive assets for the mental well-being of many. The training of a psychotherapist is standardized under the Accreditation Council for Graduate Medical Education [1], which requires generalized competencies in a variety of skills. To become a licensed psychotherapist, the requirements vary from state to state but generally require at least 60 hours of semester education [2].

Simulated patients/standardized patients have been used to both teach and assess residents [3] , requiring an actor trained for a specific situation but being an effective and meaningful method of education [4]. Simulated patients must themselves go through training to maintain a level of quality and standard between patients [5]. Simulated patients are also expected to give a level of interpersonal connection and have memorized events/encounters.

On the other hand, virtual patients, which are computer-simulated, combine the advantage of patient usage and eliminate the need for an actor. This broad term encompasses from decision-tree-style conversations to interactive virtual avatars. Virtual patients have been utilized for general medical practices, such as performing examinations and patient monitoring [6], while their usage as a psychotherapist training tool has been severely under-explored. Currently, the creation of virtual patients for psychotherapists relies heavily on recording simulated patients and then giving the user a limited array of options [7]. This lacks interactivity typical of patient interaction by forcing the user to select an option that may not be reflective of their response. Hence, generative methods for virtual patients may be highly beneficial as a training tool.

The objective of this study is to explore the potential of modern machine learning models in replicating training methods for psychotherapists. The study aims to evaluate the effectiveness of these models in simulating therapeutic interactions by assessing their accuracy in mimicking patient behaviors, and determine their potential for enhancing psychotherapist training by offering a scalable, interactive alternative to traditional simulated patients.

Machine Learning (ML), a rapidly advancing field, is the concept of teaching machines to perform specific tasks and detect patterns [8]. ML is sub-sectioned into different processes for differing problems, such as Recurrent Neural Networks for textual processing, Convolutional Neural Networks for image processing, and Neural Networks for general pattern processing. The concept of Machine Learning was developed under Alan Turing's 1950 advances [9], and has been ever growing since. The training cycles of a model are entitled epochs, the adjustable aspects are hyperparameters, and often follow a methodology for tuning hyperparameters through optimization with algorithms such as AdamW (Adaptive Movement Estimation with Weight Decay) [10]. Fine-tuning is the process of using a pre-trained model which was developed on a much broader and larger dataset, and then re-training the model on a more specified dataset to develop new characteristics in the model. This retains the model's ability for large language capability while also allowing specialization, such as a medical questionnaire or for mathematical evaluation.

Natural Language Processing (NLP), is the act of preprocessing semantic and textual data into a computer-viable format [11]. This includes lemmatization, embeddings, keyword filtering, truncation, padding, and other methods of cleansing. NLP is sequential, meaning all inputs will receive the same preprocessing techniques. NLP can be effectively split into word embedding and input filtering. Words are often broken into their core components, which processes such as lemmatization

Correspondence: Michael Petrizzo (michaelpetrizzo26@gmail.com). This is an article distributed under the terms of the Creative Commons BY-NC-SA license

and stemming do in differing ways, either by the core word or by removing prefixes, this entire process is called tokenization. Then, these core words are embedded, which is the process of converting characters to integer lists of large dimensionality. The dimensionality maps words in space and ideally can show similarities in meaning by distance between points (Figure 1). The process of embedding is often unique to each model for various purposes, costs, and training data

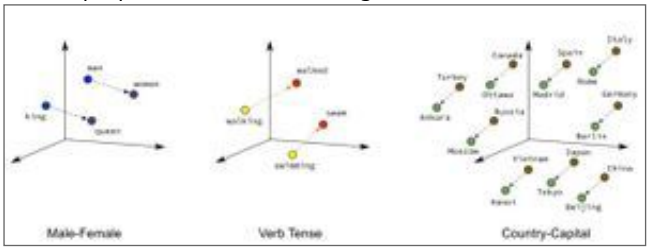

Figure 1: Word Embedding Visualization. (developers.google.com)

Cosine Similarity measures the distance between dimensional vectors to generate a broad accuracy (Figure 2). Depending on the embeddings utilized, this distance measures the semantic similarity between two points, vital for measuring the similarity between two texts which are not written identically. While not providing absolute accuracies which are reliable to a complete model's behavior, it provides an additional metric which can compare relative accuracies.

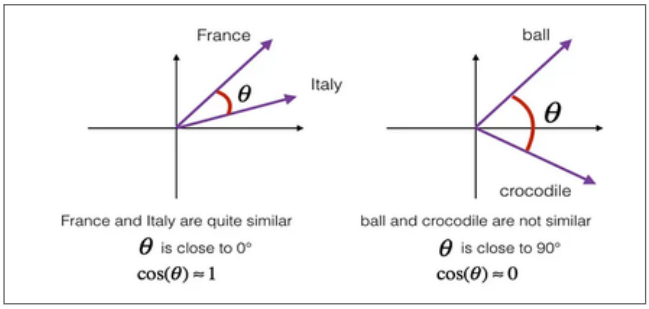

Figure 2: Cosine Similarity Visualization. (https://www.index.dev/blog/best-nlp-algorithms-to-get-document-similarity)

Neural Networks, labeled as the core of modern ML, are a series of nodes connected via inputs and outputs in a web-like structure (Figure 3). These nodes contain activation functions that perform a rounding formula that will either activate or deactivate the node for usage [12]. The most common activation functions include ReLU, Sigmoid, Tanh, and Leaky ReLU which differ depending on the task of the model [13]. The model learns by passing training data through the neural network, and then alters the internal parameters of the node, including the activation function, to create a correct output [14]. Over the course of multiple diverse inputs and accurate outputs, the model can accurately predict the output of a new input.

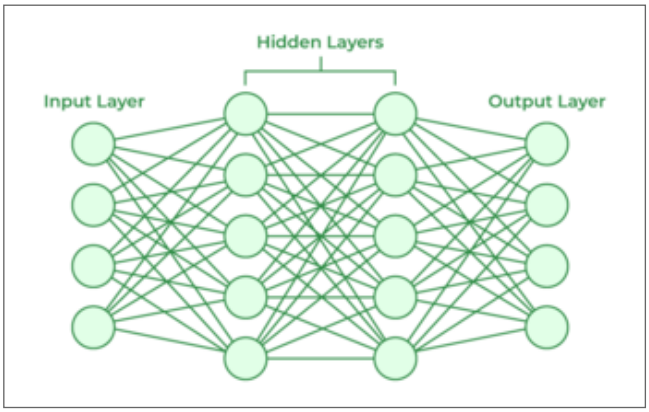

Figure 3: Neural Network Structure. (https://www.geeksforgeeks.org/artificial-neural-networks-and-its-applications/)

Recurrent Neural Networks (RNNs) are utilized predominantly in sequential data, including textual sequences [15]. RNNs utilize the fundamental neural network aspect while exhibiting "attention" or the ability to remember how a series of inputs, rather than one input, can alter an output. This is achieved by processing inputs through the neural network while also passing the unaltered inputs back into the model (Figure 4). This creates a "series" of inputs that the model can process for specific tasks.

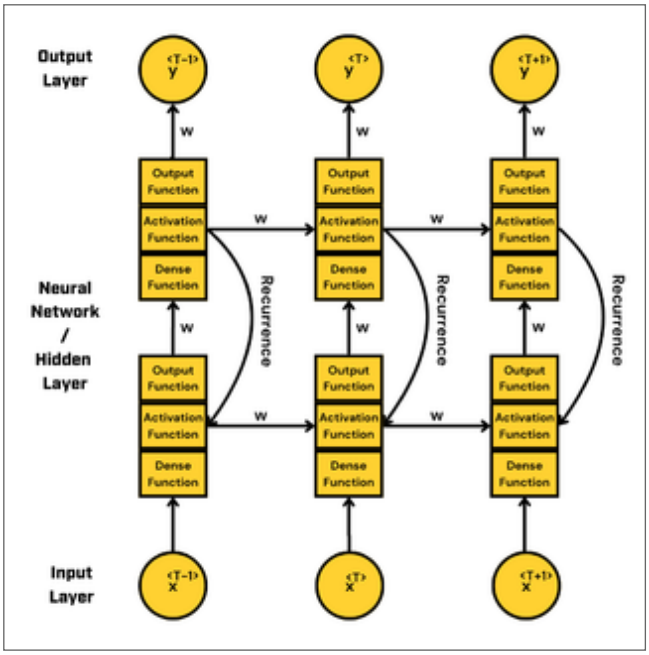

Figure 4: Classical RNN Structure.

Transformers (TRNNs) built off the RNN structure by introducing more sophisticated attention and processing mechanisms (Figure 5). The usage of an encoder-decoder system allows the models independent processing between outputs and inputs, while also introducing customizable tokenization processes that represent the data's features [16]. This process of introducing information back and forth between the encoder and decoder is defined through the feed-forward layers directly into a series of multi-headed attentions. The normalization and add layer serve to prevent exploding or vanishing gradients, and often "contain" the outputs being produced [17], this normalization improves training time by utilizing numbers less than 1, which exponentially decreases rather than increases. The specific

architecture varies among models, but generally includes these series of multi-head attention, feed forwards, normalizations, and lastly ends with an activation function.

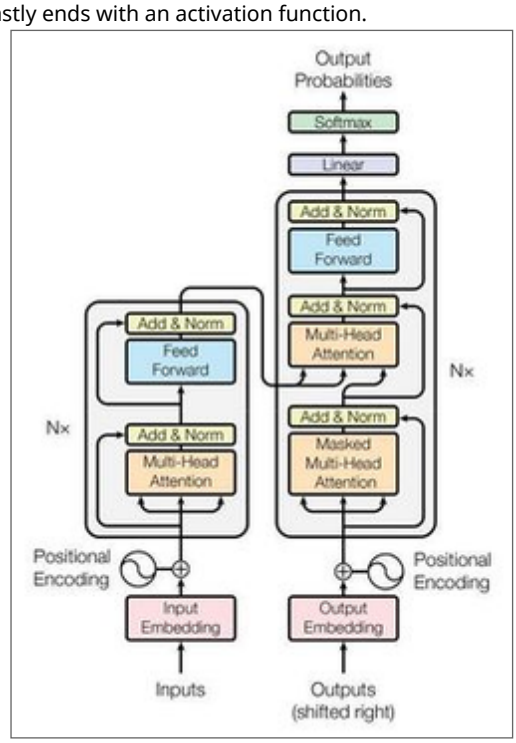

Figure 5: TRNN Structure.

Mixture of Experts (MoE) is a sub-divisional-based architecture comprising multiple networks connected similarly to TRNN to achieve self-attention [18]. This is accomplished through specialized "experts" to perform subdivided tasks of the model. Working similarly to an office, the "leader" flows all input directions, and calls upon "workers" who are the specialized experts. This type of model has lower training times and higher accuracy convergence compared to conventional models [19]. While not a new concept, its difficult conceptualization makes it a novel approach to Machine Learning and is currently deployed in advanced modern models, such as ChatGPT 4.0.

By utilizing the advantages of multiple models which leverage MoE and TRNN architectures, human-like behavior can be replicated with increasing accuracy to potentially provide an additional training tool for resident psychotherapists. Further, this format will be most effective in replicating virtual-therapy sessions and reduce the costs associated with traditional methods. The created Virtual Patients and interface(s) should achieve the following criteria:

- Adaptable and scalable to a variety of interfaces and transcriptions.
- Easy to use with feedback-based quality of life features through an online-interface.
- Open-sourced to allow continual development with multiple parties.

SCIENTIFIC FOUNDATION

Simulated Patients are a fundamental aspect of psychotherapy training which requires a certified actor to be available. Due to this, oftentimes another psychiatric resident loses training time to serve as a simulated patient for another resident. While virtual patients seem promising as an alternative, they lack the interactivity of a simulated patient. As such, this project aimed to develop virtual patients that have the advantage of simulated patients by leveraging AI.

The advancing AI models of ChatGPT-4o, LlaMa-3.1v-405B, and Gemini 1.5 Pro have been shown to have higher levels of problem-solving and human-like characteristics [20]. Furthermore, these models have been used to mimic human behaviors, notably for medical practice with promising potential [21]. The increase in performance can best be attributed to the large parameter counts and data sizes (Table 1). Each of these models has smaller or "mini" alternatives that hold the same architecture but smaller parameters and model sizes. Ideally, the largest model can process inputs more effectively, but due to costs or training time, smaller models were utilized both for testing and to serve as a baseline comparison among models.

Table 1: Model Specifications and Parameter Comparisons. Combination of Multiple Sources [22,23,24])

Model/Parameter	*ChatGPT-4o*	*LlaMa-3.1v-405B*	*Gemini-1.5-Pro*
Architecture	Encoder-Decoder	Decoder Only	*MoE TRNN*
Model Size	~500 GB	~1 TB	N/A*
Parameters	~1.8 Trillion	~405 Billion	N/A*
Speed (Tokens/second)	63.3	27.8	59.8
Cost (Per Million Tokens)	*Input:* $5.00	Free for Research	$1.25
	Output: $15.00	Free for Research	$2.50

- - Not disclosed publicly, but can be estimated to be within the ranges of 140B Parameters and ~250 GB.

These models were chosen for their high accuracy in token processing, problem-solving, replication, and high token density. Each model has specific strengths, which is why three models were chosen rather than a singular (Table 2). Namely, Massive Multi-Language Understanding (MMLU) tests a model's ability to multitask among a variety of subjects (Hendrycks et al., 2020); Instruction Focused Evaluation for LLMs (IFEval) evaluates the ability of a model to follow the given instructions [25]; Zero-Shot Benchmark for Long Text Understanding (ZeroSCROLLS) assesses the ability to hop between a series of paragraphs and answer questions [26]; InfiniteBench is a questionnaire requiring an average token size larger than 100k [27]; and lastly, Needle in Haystack (NIAH), requires the model to find a fact from a large article or dataset [28]. The term N-Shot references the amount of pre-training on the data prior to testing and Chain of Thought (CoT) is a prompting style to encourage problem-solving and sequential processing [29].

Benchmarks demonstrating model efficiency among high tokenization and human-like characteristics are vital for model predictions in fine-tuning and for case-specific uses. For example, Gemini 1.5 Pro's high token context performance allows for conversations to be continuous without attention loss and is tested among InfiniteBench, ZeroSCROLLS, and NIAH. The MMLU benchmarks show a model's ability to be humanistic, to understand and grasp complex semantic relationships in a conversation.

Table 2: Model Comparison Among Various Key Benchmarks. Combination of Multiple Sources [30,31,32,33,34,35,36]

Model/Benchmark	*ChatGPT-4o*	*LlaMa-3.1v-405B*	*Gemini 1.5 Pro*
MMLU (0-shot, CoT)	**88.7**	88.6	86.0
MMLU Pro (5-shot, CoT)	74.0	73.3	**75.8**
IFEval	85.6	**88.6**	87.1
ZeroSCROLLS/8K Alternative	90.5	95.2	**99.7**
InfiniteBench	82.5	**83.4**	-
Needle In Haystack	**100.0**	98.1	99.7

Older generation models, such as GPT 3.5 have been tested in comparative studies between human-like tendencies of models, with mixed results [37]. Although AI models trained on mental health transcripts have yet to be studied extensively, the data required for training these models is publicly available. This may be utilized to create virtual patients that have high similarities to real patients without the possible detrimental effects that can be placed on a patient by a resident in training. To ensure effectiveness, the testing of potential virtual patients among current resident training is incredibly important as well, efforts to reach individuals to test these models are ongoing. For ease of use and accessibility, the creation of a Graphical User Interface (GUI) was equally vital to successful implementation and was subject to the same amount of criticism and feedback as the model. The standardization of psychotherapy training with virtual patients can ensure proper development in the resident's ability for patient communication, empathy, and allows easier access than compared to actors. Further, virtual patients can encompass a larger variety of mental illnesses and specific situations through the use of a singular transcription. The model being adjustable also allows the same model to have unique responses to the same input, given a change in the temperature of the model. Lastly, the integration of resident transcription allows instructors faster evaluations of the competencies and abilities of a resident.

Previous research in utilizing AI to replicate patients for training residents is novel, with very limited conceptualization and application. However, recently a new trial has been published; Patient-Ψ [38] attempts the goal of training mental health professionals via ChatGPT-4.0 specialization and prompt engineering. These prompts reflect specific emotional and situational aspects of the individual to be replicated and then use a "trainer" framework to converse between the patient and a user. The results of the model output were compared with professionals against the default ChatGPT-4.0 model. Further, this paper aims specifically at CBT training and aims for generalization in future research. Vitally, the usage of supervised learning is not explored, instead opted for unsupervised learning. Further, rapid model development warrants the usage of multiple models for analysis.

METHODOLOGY

Overview

1. **Data Collection:** Transcription collection of publicized and anonymized therapy sessions. Must have direct interaction between Patient and Therapist with sufficient detail for training.
2. **Interface Creation:** Website-based HTML, CSS, and Javascript interface with customizability and accurate connections to patients. The backend is supported by Python and respective hosting-software. Must have conversational saving through multiple formats, voice-to-text, and multiple patient options.
3. **Patient Creation:** Python-based patient fine-tuning and storage to be connected in the Interface. The selected models are trained through two methods: fine-tuning and transfer learning. An independent accuracy with Cosine-Similarity is used to compare multiple created virtual patients and determine usage in the interface.
4. **Usage:** Outlines intended work-flow and professional usage in resident training. Further clarifies how an instructor may review conversations by the resident psychotherapist.

Data Collection

Transcripts were gathered through various sources, such as Kaggle, and online publicly available PDFs through direct Google searches for "Therapy Patient Transcripts". In total, there are 2 transcriptions in pdf format which were then converted into Comma Separated Value (CSV) format for easier usage in eventual model training and fine-tuning. The choice of transcripts relied heavily on data quantity, sufficient patient output/communication, and data availability. Each transcript is a direct therapy session between the patient and the therapist in a one-on-one style. The conversion from PDF to CSV was hand-generated, with the assistance of ChatGPT-4.0 for demographic characteristics which included a general age range. The transcripts were formatted into full segments of patient and therapist messages, and the corresponding characteristics as columns. Independent CSV files are then combined into one large file for convenience, which are separated by index (Figure 6).

ID	MentalIllness	AgeRange	ClientText	TherapistText
1	Major Depressive Dis	60-70	X	Hi. Good to see you again.
1	Major Depressive Dis	60-70	Hi.	Can I take a look at your scores?
1	Major Depressive Dis	60-70	Sure.	While I'm looking at these, just tell me, in your own words, how
1	Major Depressive Dis	60-70	Well, I would say I th	Oh, that's wonderful. I'm really glad to hear that. It looks like yo
1	Major Depressive Dis	60-70	Well, I think that wha	Okay.
1	Major Depressive Dis	60-70	Because I had been	Oh, that's great. That's great. Looks like your concentration on t
1	Major Depressive Dis	60-70	Yeah.	Oh, I'm really glad. You said you don't have any idea as to why
1	Major Depressive Dis	60-70	Maybe that's not enti	Oh, that's great. We'll get to those things in just a minute, but le
1	Major Depressive Dis	60-70	Well, I guess to start	What's something that you would like to be able to do, but you'v
1	Major Depressive Dis	60-70	Well, it's kind of like t	Good, okay, so we'll try to get to evenings. Anything else? Any c
1	Major Depressive Dis	60-70	I would say that that'	You want to get your apartment in better order?
1	Major Depressive Dis	60-70	Yeah.	Okay.
1	Major Depressive Dis	60-70	But that seems like s	Well, maybe we could figure out together where to start.

Figure 6: Generated CSV Structure.

The addition of further transcripts is crucial, but due to data availability and required patient anonymity, is difficult to collect without the direct guidance of a professional. Further, transcriptions of specific mental illnesses are rarely provided nor categorized to protect patient information. Due to these difficulties, a customizable patient was created which allows the user to define a background and mental-illness based on transfer-learning strategies. Lastly, additional transcriptions can be readily characterized and converted through the algorithms defined prior.

Every transcription utilized was priorly anonymized and safely released respecting the information of the patient. Information such as the location of therapy, patient name, age-range, and mental-illness are not provided by the transcript and instead classified by model as stated prior. Transcripts are saved on the server to provide information for Python transfer learning. No user-provided data is asked or saved regarding transcriptions.

Interface Creation

The GUI consists of an HTML, CSS, Javascript, and Python co-creation to create a website interface and GUI simultaneously.

HTML and CSS are the predominant combinations for creating professional and aesthetically appealing User Interfaces. The individual pages of the website are split into separate HTML files but utilize the same CSS style to ensure consistency. Javascript was integrated into HTML structure to allow further user input, such as custom form submissions or the creation of templates, headers, and footers which can be imported into pages and reduce redundancy. As HTML and CSS are not back-end languages, Python was utilized to connect the fundamental ML models to the front-end appearance by passing through Javascript. The model was trained in Python, which has a higher availability for ML modules, with usage in Tensorflow, Google-AI-Studio, and HuggingFace. Javascript can pass parameters between these languages through form submissions, features, functions, and queries. The individual pages are also referencing each other; allowing interaction between various components of the GUI, often with unique features that pass the user's intended model and interaction. To create an application with the website-based code, Electron was utilized to create an environment to run the HTML, CSS, Javascript, and Python code in combination. This allows zipping into an executable format for application usage and possible Mobile conversion.

The index page or "initial" state, contains a form submission and description of model choices (Figure 7). This page serves as the connection between the user and their choice of model. The page covers the basic descriptors of each model and redirects to the fundamental "chatBot" page by passing the chosen patient as a feature. For individual model testing, further choices of ML model between GPT-4o and GPT-4o-mini are available. The usage of an account is intentionally avoided to prevent security concerns and database usage but may be added as needed to view previous conversations or create a "classroom" with a supervisor being able to view the "student's" interactions.

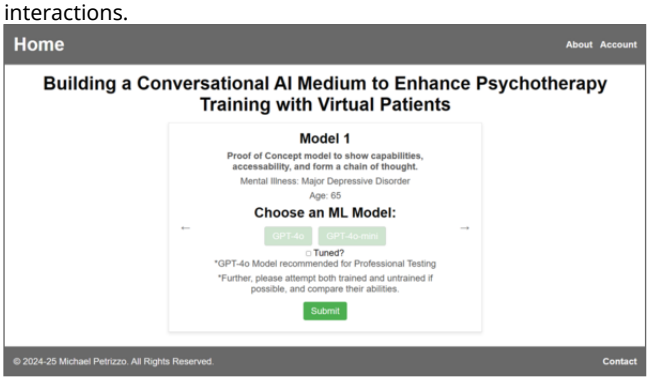

Figure 7: Main Index Page.

The "chatBot" page takes the chosen model and creates a unique interaction between the user and the virtual patient (Figure 8). This contains a chat box with textual or voice input, passes the input into the ML model, and returns a visual and auditory response to the user. The requests between the front-end (user interface) and back-end (python model) are sent with Flask via HTTPS Ports for local development and server-sided ports during complete implementation.

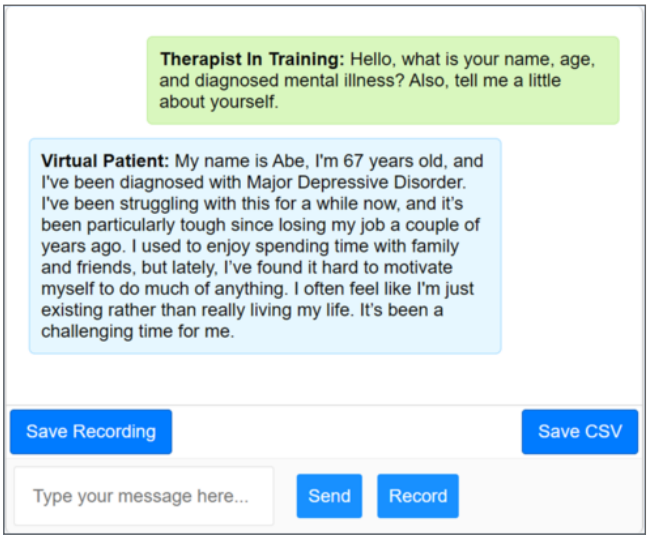

Figure 8: Example Conversation with Transfer-Learning GPT-4o-Mini.

User recording and conversations are not saved beyond local abilities allowing user-download. All information is deleted upon local refresh. Information is never saved permanently. Audio will not record unless allowed by the user, and again, only saved locally to create the video file associated with the Save Recording button or Voice-To-Text.

External pages may be utilized to contain contact, legal, or extraneous information.

Patient Creation

Each patient-model experienced differing training according to API availability, UI aspects, and other various preprocessing required. Llama-3.1-405B was downloaded directly onto the system, while GPT and Gemini required API usage and continual costs per input. The models utilized customized accuracy metrics of cosine similarity comparing the direct semantic relation between the model's outputs and the patient's characteristic behaviors, to determine the most successful model per transcription. Llama-3.1-405B and ChatGPT-4o used smaller versions for testing, specifically, Llama-3.2-1B and ChatGPT-4o mini, and were ultimately incorporated in accuracy metrics as training loss indicated high accuracy. An additional series of models were instead trained through transfer-learning, where transcriptions were instead passed through context rather than explicitly fine-tuned. For continual progression, communication with a professional is an ongoing operation ingrained into the model testing (Figure 9).

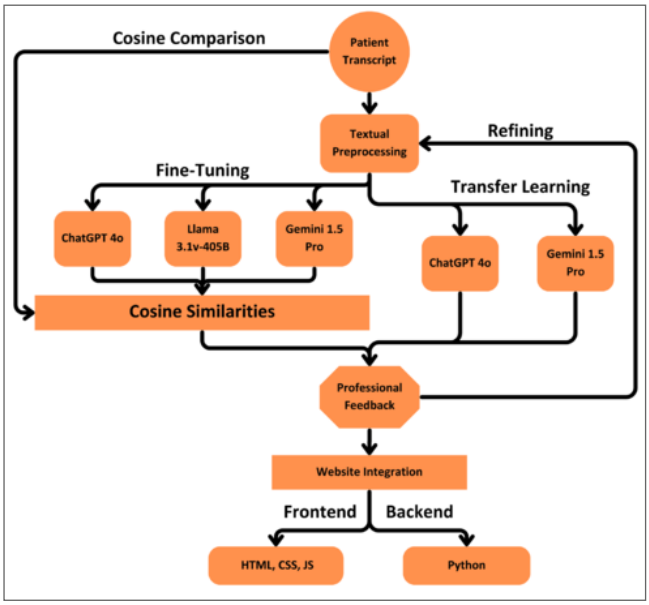

Figure 9: Patient-Training Workflow.

ChatGPT-4o; This model was fine-tuned with the OpenAI API by structuring the dataset into a series of messages between the user and client. The assistant text would follow the patient portion, and the user would follow the therapist. The messages were formatted into a JSON format as {messages : [{role: system/client/user, content: text...}]} with the first dataset being BB3-Session-2-Annotated and containing 230 JSON lines. This file was uploaded into OpenAI and fine-tuned with GPT-4o-mini-2024-07-18 prior to full model use. The model was trained with 3 epochs, contained ~23558 tokens per epoch, and cost roughly ~0.01$; which was calculated through the OpenAI recommended algorithm [39]. In total, the model was fine-tuned through 690 steps, and had a final training loss of 0.1172 with a 1.00 training accuracy. The gpt-4o-full model was trained similarly, with slightly differing parameter; the same 3 epochs, with a 2.0 learning rate, and a final training loss of 0.2085 (Figure 10).

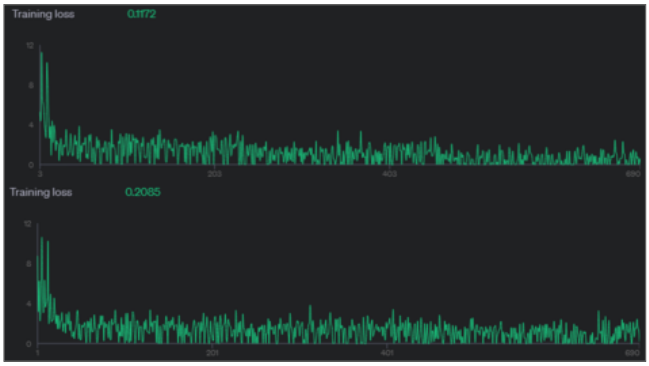

Figure 10: ChatGPT-4o Mini Fine Tuning (Top) & ChatGPT-4o Full Fine Tuning (Bottom). (Adapted from OpenAI API)

LlaMa-3.x; This model is instilled directly into the Hugging Face library and was downloaded into an external Standard Storage Device (SSD) due to the large model size and independence from conflicting environmental settings. The usage for research purposes was approved via request and requires specific API keys to utilize. Visual Studio Code directly hosted the environment for model memory and was loaded via transformers. AutoTokenizer.from_pretrained and transformers. AutoModelForCasualLM.from_pretrained. Inputs were loaded via CSV format and then made into a class compatible with the API format. This was combined with training specifications, with default learning rates, epoch counts, and the use of quantization to decrease model load on GPU and CPU usage.

Gemini 1.5 Pro; This model is fine-tuned through Google AI Studio API with a structured JSON similar to ChatGPT-4o. Key differences include more descriptive classifiers for parameters which are used to fine-tune the model. This file had to be uploaded to Google Cloud, tuned through vertexAI, and used gemini-1.5-pro-002 as the source model. This creates an endpoint through Google AI Studio which allows the model to be used. In all, roughly 7 models were fine-tuned per transcription, each with differing model configurations such as a test-train split or learning rate, with key formatting issues being the primary cause for model error and repeated creation. Almost all tuning specifications were provided via official guides, which held nondescript JSON formatting, tuning, and utilization example code. [40,41,42] The final model cost roughly ~1.50$, and was trained with the default epoch count of 5 and learning rates of 2.0. Each epoch made ~2.5k predictions and reached a final training loss of 1.98 (Figure 11).

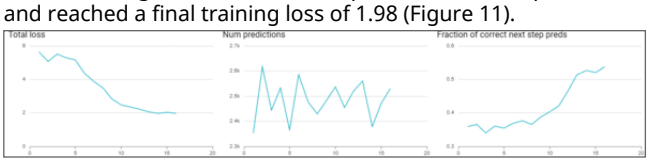

Figure 11: Gemini 1.5 Pro Fine Tuning (Adapted from Google Cloud.)

Transfer-Learning; Each model utilized identical transfer-learning specifications. Firstly, the transcription is loaded according to the selected patient. For each row in the transcription, the patient and therapist message is appended to a string which is then passed into the model as background/system information. The model used is the base-line version which was used for fine-tuning. A series of key background information is also passed, such as instruction to behave like the transcription and to respond as if this is a new therapist. For each sent message, the user-input and background information is passed in its entirety. Gemini 1.5 Pro struggled to differentiate transcription from user-input, in large part due to no way to differ system-content and user-content in chat completion. The Llama series cannot be transfer-learned as it has no prior trained information.

Customizable Model; A descriptable model which allows the user to input background information and a mental illness. This was created to mitigate the need for transcripts and for the user to practice with a patient of a select mental illness or select behaviors. However, these models cannot be compared with the cosine similarity due to a comparative transcript not being available. Instead, professional feedback is the only metric available to inform the quality of these models. Both GPT models are available as customizable models, Gemini and Llama are not for their aforementioned characteristics.

Usage

The GUI is split into two pages, one for initial model selection/customization, and the second to initiate the conversation (Figure 12). The initial page houses multiple patient selections alongside a ML model choice. Further the last selection allows the user to customize the model with a description and mental illness. The "Tuned" checkbox represents the decision between Tuned(Fine-Tuned) and

unTuned(Transfer-Learning) models. Pressing submit redirects the user to the conversational page. This page has options to send messages through text or voice-recording, as well as buttons to save the conversation in both a CSV and video format for instructor evaluation.

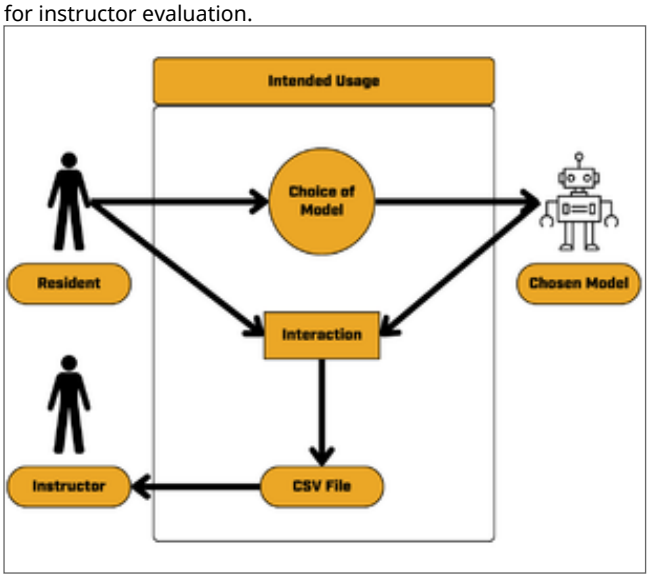

Figure 12: Intended Usage and Work-Flow.

RESULTS AND ANALYSIS

The individual metrics for model accuracy, recall, and instruction following are characteristically different depending on the exact patient transcript utilized (Table 3). This could be due to varying data quality, which cannot be verified for patient anonymization. Likewise, data quality is not standardized nor verified for usage as a virtual patient. Each model received an accuracy through a randomized series of 50 messages from the training set, the messages were randomized with a random number generator. The randomized set utilized identical seeds to ensure model comparison.

Fine-Tuned Models

Table 3: Fine-Tuned Model Metrics.

Model		*ChatGPT-4o Mini*	*ChatGPT-4o Full*	*Gemini 1.5 Pro*	*LlaMa-3.2v-1B*
Transcript 1	Cos. Sim.	93.66	88.88	86.12	83.11
	Model Loss	0.12	0.21	-	1.73
Transcript 2	Cos. Sim.	92.31	86.80	85.18	83.68
	Model Loss	0.06	1.66	-	1.76
Average	Cos. Sim.	92.99	87.84	85.65	83.40
	Model Loss	0.09	0.93	-	1.74

Interestingly, ChatGPT-4o Mini performed higher statistically compared to the larger, superior ChatGPT-4o Full model. Both models were trained identically, utilizing the exact same script and training data besides changing the model specification. The training for both models seems to lack convergence of training loss, which is something that warrants further analysis and fine-tuning more than the recommended settings. This may show a superiority in generalized tasks through a smaller network, rather than an ambitiously large dataset. Further, the cost difference for training and usage makes ChatGPT-4o-mini much more applicable in business-end environments, as the testing cost was near negligible for ChatGPT-4o-mini while reaching upwards of 0.10$ for 50 inputs of ChatGPT-4o-full, showing a stark increase.

Table 4: Transfer-Learning Model Metrics.

Model		*ChatGPT-4o Mini*	*ChatGPT-4o Full*	*Gemini 1.5 Pro*
Transcript 1	Cos. Sim.	83.21	91.25	87.18
Transcript 2	Cos. Sim.	81.44	86.52	80.39
Average	Cos. Sim.	82.33	88.88	83.79

The accuracies in the transfer-learning models are similar to those of the Fine-tuned models. However, the GPT-4o-mini model performed worse in comparison to the larger model, and all of these models have seemingly lower accuracies than the Fine-tuned counterparts. Without professional feedback, it would be incredibly difficult to differentiate the quality of these models. However, some key differences are apparent, such as the transfer-learning models have a much more descriptive and longer output. These models are more deterministic, having closer to identical messages upon the same inputs but much more expensive costs. For comparison, the transfer-learning models were roughly 10x as expensive but had higher quality and more descriptive outputs which may explain semantically dissimilar messages.

The Llama-3.2v-1B was created as a comparative model to the larger and generally stronger GPT and Gemini series. This model achieved the lowest accuracy, which can be attributed to having the smallest model size when compared via parameter count to the other models. However, this model was released to be suitable for business environments with capabilities for fine-tuning accurately due to the model's small size but retained capabilities to the 3.1 series. The training was completely customizable and allowed for much more selections compared to API-based models. The Llama-3.1v models were much more expensive than the 3.2v, and were virtually impossible to train on a singular device.

Gemini 1.5 Pro faced many challenges in tuning, documentation, testing, and model retention. The overall performance of Gemini 1.5 Pro is high, but the model lacked the ability to grasp the contents of small-sized inputs, even with accelerated learning rates. As such, the accuracy of the model is lower than ChatGPT, but a larger transcription may prove highly beneficial. The fine-tuned and transfer-learning model strongly represented the base model more than the patient transcription, having inbuilt descriptors to behave dissimilar to human-like characteristics. Lastly, the safety guidelines for Gemini 1.5 Pro were very aggressive, meaning most inputs ended up being denied inputs. This was with the safety guidelines specifically declared, warnings accepted, and all token counts as high as possible, but most inputs were still declined without a warning output. While not necessarily negative in general usage as a chatbot, the Gemini model failed as a virtual patient.

The ability for a model to respond quickly is also a vital component of a chatbot design, to enhance user satisfaction and make fluid conversation. Having the model entirely installed on the system makes Llama the fastest model, being dependent on the configuration and specification of the system, while

Gemini 1.5 Pro took extensive safety measures; not necessarily negative, but would extend prediction times dramatically. Furthermore, the rate-limiting of API-based models, such as the ChatGPT and Gemini series used in this paper, may incidentally cause inputs to be buffered or lost entirely. The transfer-learning models responded much quicker than the equivalent fine-tuned models. Oftentimes being faster in comparison of milliseconds to multiple seconds.

CONCLUSION

Utilizing modern Artificial Intelligence models to develop virtual patients is a promising applicable concept which warrants further development with higher computational power. The need for such developments is high; especially in ensuring resident training quality and creating availability to simulated patients. Furthermore, the promise for such concepts to be successful increases with the development of more advanced models that can capture semantic relations with even fewer inputs. The creation of a website interface was successfully developed to be utilized; however, there are various hurdles before full deployment into professional environments.

The development of a website interface with customizability was successful, through allowing multiple patient choices and a completely customizable patient. Further, the Cosine-Similarity accuracies of each model have high promise, especially considering the limited data-size and computational power available. This study reflects the ability of virtual patients to be created with additional intractability supported by Transformer and MoE models with prominent applications in current training methods. This new training method may be especially useful in virtual-therapies through messaging, which is an ever more prevalent type of therapy.

The open-source nature of the study allows and invites all individuals to expand upon the training methods utilized in this paper for the development of advanced Psychotherapy training, where the programs can be found in the Addendum.

Limitations

The limitations within this paper reflect those of transcript availability, computational power, and time constraints. This is especially apparent in rate-limiting, training sizes, documentation difficulty, and data availability.

The limitations in creating virtual patients specifically stem from data availability, where finding large amounts of data brings concerns to identification and the minimal data may reflect biases; especially as the transcripts were from the same source. While these transcripts reflect two different individuals, the generalizability may be limited pertaining to differing cultures or ideologies. Further, transcriptions of less common mental illnesses are further difficult to gather and may limit the fine-tuned model capabilities. The addition of the customizable model aims to combat this, but limitations in model bias will also suffer the same limitations with lower magnitude. Finding additional potential transcriptions would best be hosted under a school of therapy or similar environment, where connections are awaiting a response and warrant future research.

Complete training of a large model requires hundreds of hours of fine-tuning, which may then be nullified due to a simple error and then restarted, forcing more time to train. Amplified by novel documentation of model training and usage creates large gaps in public knowledge for utilizing these modern models. Furthermore, the limitations in artificial outputs make standardization of each model difficult; as the probabilistic nature of a Machine Learning model prevents consistent responses. Lastly, the testing from professionals is ongoing for these models, and may dramatically impact reflections about model accuracy and capacity for expansion. Connections to certified psychotherapists have been made, and feedback is awaited.

Cosine Similarity was used as an independent metric for model comparison, but may not completely evaluate the accuracy of models in this context. While strong for semantic similarity, which was used to determine a model's similarity to its training data, the analysis of hallucination and error is largely not included through this metric. However, a common metric beyond Cosine Similarity is largely unavailable in contexts where training is not performed locally, and instead loss methods or accuracies are not provided by the GPT and Gemini models. Comparing these models without training-loss is typically done with a Benchmark test, but this scope is limited in a generalizable benchmark. To mitigate these limitations, subjective professional feedback was acquired through currently practicing Psychotherapists of multiple companies and regions.

Future Research

As stated prior, the need for further evaluation and training adjustments is an ongoing and future focus of research. Notably, the Llama series requires more hours of fine-tuning to be evaluated for accuracy and then deployed. A continuation to evaluate models with psychotherapists and residents is also a focus for future research. More generally; additional time is necessary to develop the limitations and expand the potential to suit the unique challenge of replicating patients with specific characteristics. This could require a dataset of "unique" individuals each with their characteristic traits to warrant development. In addressing transcription needs, Generative Artificial Intelligence is a hesitant alternative that has high potential but demands broad exceptionally large models to be trained. The usage of generation to both create transcriptions and generate images representing the patient's emotions has the potential to further reflect the nuances of in-person conversation, but requires further developments in generative AI.

Potential

There is potential apparent in each model's displayed accuracy and further metrics to evaluate the model's performance are necessary to determine their use in real-life application. For example; ChatGPT-4o mini having a higher accuracy than the base-model equivalent is highly counterintuitive and needs to be evaluated further. The accuracy of the models overall reflects some semantic correspondence successfully gathered by the model, which can also likely be improved through more data. Interaction with the model was successfully implemented and developed intentionally to provide the ability for model addition as technology develops further. The website was also developed with the ability to be converted into an app for mobile devices, further supporting scalability as models are evaluated further. By utilizing Electron as a secondary port, the website has executable files, a website interface, and mobile conversion capacities. This allows easy integration into ongoing training systems for residents. In all, this paper demonstrates the potential of AI to greatly improve the availability of resident training.

ACKNOWLEDGEMENTS

In Loving Memory of Zuri Leung.

Thank you to Ms. Colette, Ms. Beatty, and Dr. Kramer for their amazing guidance as research teachers.

REFERENCES

[1] Accreditation Council for Graduate Medical Education, ACGME Competencies, 2024. [Online]. Available: https://www.acgme.org/programs-and-institutions/programs/common-program-requirements/

[2] New York State Licensed Professionals, NYS Mental Health Counseling, NYSED, 2024. [Online]. Available: https://www.acgme.org/programs-and-institutions/programs/common-program-requirements/

[3] H. S. Barrows, "An overview of the uses of standardized patients for teaching and evaluating clinical skills," Acad. Med., vol. 68, no. 8, pp. 443–451, 1993.

[4] A. Lovink, M. Groenier, A. Niet, H. Miedama, and J. J. Rethans, "How simulated patients contribute to student learning in an authentic way: an interview study," Adv. Simul., vol. 9, no. 4, pp. 1–10, 2024.

[5] A. J. Cleland, K. Abe, and J. J. Rethans, "The use of simulated patients in medical education: AMEE Guide No 42," Med. Teach., vol. 31, no. 6, pp. 477–486, 2009.

[6] Body Interact, "Virtual Patients," 2024. [Online]. Available: https://bodyinteract.com/virtual-patient-simulator/

[7] B. Zalewski, M. Guziak, and M. Walkiewicz, "Developing simulated and virtual patients in psychological assessment – Method, insights and recommendations," PubMed, vol. 12, no. 1, pp. 455–461, 2023.

[8] IBM, "What is Machine Learning?," 2024. [Online]. Available: https://www.ibm.com/topics/machine-learning

[9] A. M. Turing, "Computing machinery and intelligence," Mind, vol. 59, no. 236, pp. 433–460, 1950.

[10] PyTorch, "ADAMW," 2023. [Online]. Available: https://pytorch.org/docs/stable/generated/torch.optim.AdamW.html

[11] C. Stryker and J. Holdsworth, "What is NLP?," IBM, 2024. [Online]. Available: https://www.ibm.com/topics/natural-language-processing

[12] J. Lederer, "Activation functions in artificial neural networks: A systematic overview," arXiv preprint arXiv:2101.09957, 2021.

[13] P. Baheti, "Activation Functions in Neural Networks: 12 Types & Use Cases," v7, 2021. [Online]. Available: https://www.v7labs.com/blog/neural-networks-activation-functions

[14] Amazon Web Services, "What is a Neural Network?," 2024. [Online]. Available: https://aws.amazon.com/what-is/neural-network/

[15] C. Stryker, "What is a recurrent neural network?," IBM, 2024. [Online]. Available: https://www.ibm.com/topics/recurrent-neural-networks

[16] KiKaBeN, "Transformer's Encoder-Decoder," 2021. [Online]. Available: https://kikaben.com/transformers-encoder-decoder/

[17] J. L. Ba, J. R. Kiros, and G. E. Hinton, "Layer normalization," arXiv preprint arXiv:1607.0650, 2016.

[18]–[19] IBM, "What is Mixture of Experts?," 2024. [Online]. Available: https://www.ibm.com/topics/mixture-of-experts

[20]–[21] A. Ahmed, H. Hayat, and D. Hayat, "Comparing GPT-4o, LLaMA 3.1, and Claude 3.5 Sonnet," Walturn, 2024. [Online]. Available: https://www.walturn.com/insights/comparing-gpt-4o-llama-3-1-and-claude-3-5-sonnet

[22] D. Haywood, "Gpt-4o tokens per second comparable to gpt-3.5-turbo. Data and analysis," OpenAI Developer Forum, 2024. [Online]. Available: https://community.openai.com/t/gpt-4o-tokens-per-second-comparable-to-gpt-3-5-turbo-data-and-analysis/768559

[23] L. Kilpatrick and B. S. Mallick, "Updated production-ready Gemini models, reduced 1.5 Pro pricing, increased rate limits, and more," Gemini, 2024. [Online]. Available: https://developers.googleblog.com/en/updated-gemini-models-reduced-15-pro-pricing-increased-rate-limits-and-more/

[24] S. Pichai and D. Hassabis, "Our next-generation model: Gemini 1.5," Google Blog, 2024. [Online]. Available: https://blog.google/technology/ai/google-gemini-next-generation-model-february-2024/#sundar-note

[25] J. Zhou et al., "Instruction-Following Evaluation for Large Language Models," arXiv preprint arXiv:2311.07911, 2023.

[26] U. Shaham et al., "ZeroSCROLLS: A Zero-Shot Benchmark for Long Text Understanding," arXiv preprint arXiv:2305.14196, 2023.

[27] X. Zhang et al., "∞Bench: Extending Long Context Evaluation Beyond 100K Tokens," arXiv preprint arXiv:2402.13718, 2024.

[28] H. Wang et al., "Multimodal Needle in a Haystack: Benchmarking Long-Context Capability of Multimodal Large Language Models," arXiv preprint arXiv:2406.11230, 2024.

[29] V. Gadesha and E. Kavlakoglu, "What is chain of thoughts (CoT)?," IBM, 2024. [Online]. Available: https://www.ibm.com/topics/chain-of-thoughts

[30] A. Ahmed, H. Hayat, and D. Hayat, "Comparing GPT-4o, LLaMA 3.1, and Claude 3.5 Sonnet," Walturn, 2024. [Online]. Available: https://www.walturn.com/insights/comparing-gpt-4o-llama-3-1-and-claude-3-5-sonnet

[31] Artificial Analysis, "Gemini 1.5 Pro (Sep): API Provider Benchmarking % Analysis," 2024. [Online]. Available: https://artificialanalysis.ai/models/gemini-1-5-pro/providers

[32] Artificial Analysis, "Gemini 1.5 Pro (Sep): Quality, Performance % Price Analysis," 2024. [Online]. Available: https://artificialanalysis.ai/models/gemini-1-5-pro

[33] Google Cloud, "Prepare supervised fine-tuning data for Gemini models," 2024. [Online]. Available: https://cloud.google.com/vertex-ai/generative-ai/docs/models/gemini-supervised-tuning-prepare

[34] Google Cloud, "The AI detective: The Needle in a Haystack test and how Gemini 1.5 Pro solves it," 2024. [Online]. Available: https://cloud.google.com/blog/products/ai-machine-learning/the-needle-in-the-haystack-test-and-how-gemini-pro-solves-it

[35] Google DeepMind, "Gemini 1.5: Unlocking multimodal understanding across millions of tokens of context," 2024. [Online]. Available: https://storage.googleapis.com/deepmind-

media/gemini/gemini_v1_5_report.pdf

[36] Scale AI, "Instruction-Following Evaluation," 2024. [Online]. Available: https://scale.com/leaderboard/instruction_following

[37] J. Ye et al., "A Comprehensive Capability Analysis of GPT-3 and GPT-3.5 Series Models," arXiv preprint arXiv:2303.10420, 2023.

[38] R. Wang et al., "PATIENT-Ψ: Using Large Language Models to Simulate Patients for Training Mental Health Professionals," arXiv preprint arXiv:2405.19660, 2024.

[39] M. Wu and S. Fishman, "Data preparation and analysis for chat model fine-tuning," OpenAI, 2024. [Online]. Available: https://cookbook.openai.com/examples/chat_finetuning_data_prep

[40] Google AI, "Fine-tuning tutorial," 2024. [Online]. Available: https://ai.google.dev/gemini-api/docs/model-tuning/tutorial?lang=python

[41] Google Cloud, "Prepare supervised fine-tuning data for Gemini models," 2024. [Online]. Available: https://cloud.google.com/vertex-ai/generative-ai/docs/models/gemini-supervised-tuning-prepare

[42] Google Cloud, "Tune Gemini models by using supervised fine-tuning," 2024. [Online]. Available: https://cloud.google.com/vertex-ai/generative-ai/docs/models/gemini-use-supervised-tuning

ADDENDUM

The Python files used for training and testing can be found in the following Github Link:

https://github.com/MI2yaya/Research

The website attributed to this project is under a Provisional Patent as of 3/14/25.