

Analysis of Limitations of AI Tools for Pediatric Speech Language Pathology Documentation and Mitigation Strategies

by

Tia Tuinstra

A thesis
presented to the University of Waterloo
in fulfillment of the
thesis requirement for the degree of
Master of Applied Science
in
Systems Design Engineering

Waterloo, Ontario, Canada, 2025

© Tia Tuinstra 2025

Author's Declaration

I hereby declare that I am the sole author of this thesis. This is a true copy of the thesis, including any required final revisions, as accepted by my examiners.

I understand that my thesis may be made electronically available to the public.

Abstract

Speech Language Pathology (SLP) is a therapy discipline offered by KidsAbility, a pediatric rehabilitation clinic in Southern Ontario. Documentation is a key part of SLP and other therapy practice guidelines and can take up significant portions of a therapist's time. AI-based clinical documentation aids have been developed to help reduce this burden, and one such tool - MutuoHealth's AutoScribe - has been piloted by KidsAbility. Though this AI tool has been beneficial to some therapy disciplines, the SLP clinicians face unique challenges when using these tools. The model seemed unable to recognize speech therapy strategies or to parse the play-based script of pediatric appointments. This thesis seeks to explore the issues SLPs encounter with AI documentation tools and propose potential approaches to mitigate these issues.

The AI documentation process was divided into the transcription pipeline, where an audio file input produced a corresponding transcript output, and the generation pipeline, where an input transcript produced a draft SOAP note. The SLPs who had participated in the AutoScribe pilot test were interviewed about their experiences with the tool and its integration into their workflows. The issues reported by the therapists were sorted into those more closely related to the transcript and those more closely related to the drafted SOAP note. A set of sample SLP appointments from KidsAbility were gathered from an extended AutoScribe pilot, with 10 selected as examples of appointment data (audio, transcripts, drafted and final SOAP notes) to test the transcription and generation pipelines. An augmented automatic speech recognition (ASR) pipeline based on a Whisper model was used to test improvements to the transcript. However, the generated transcripts were not significantly improved from the pilot test. Instead, ground truth transcriptions were manually created from the audio files to use for testing the generation pipeline. For SOAP note generation, the addition of discipline-specific context tailored to appointment type was tested. This context was curated in collaboration with SLPs from KidsAbility to include SOAP templates, definitions of key concepts, and information about speech data. A Llama 3.3 70B model was used for SOAP note generation with ground truth transcriptions and SLP specific RAG-adjacent information as context. The input context was optimized over several iterations based on clinicians' evaluations of generated SOAP note quality.

KidsAbility's SLPs had flagged sessions targeting speech practice as having particular difficulties with AutoScribe. The model seemed unable to make inferences about the child's speech quality from the transcript alone. Methods of quantitatively assessing speech based on session audio were explored as ways to provide additional context on speech quality to the SOAP generation model. A sample appointment was selected for testing, and child speech samples of the targeted sound were sliced from the audio and assigned quality categories. These samples were then compared against correct productions using the cosine distance between their mel-spectrograms. The samples were also passed through a phoneme-based ASR model to get the layer activations. The cosine distances and layer outputs were then tested as predictive measures of articulation accuracy, with layer outputs yielding the best results. The resulting speech accuracy scores were then passed into the

generation model as additional context, with the output containing correct statements about the nature of the child’s articulations.

Though clinicians’ availability limited the extensiveness of generated SOAP note evaluations, the SOAP notes generated with SLP-specific context showed improvement compared to the basic model generation. The model also tended to repeat information from previous SOAP notes if examples were provided. It was found that quantitative speech analysis does seem possible using phoneme model layer activations and cosine distances between the mel-spectrograms of correct articulations. Based on these findings, further optimizations to the generation pipeline and work on making effective AI tools for KidsAbility’s EY SLPs will continue.

Acknowledgements

I would like to thank Smile Digital Health and KidsAbility for sponsoring this research.

Thank you to KidsAbility's innovation team for their support of the project from start to finish. A special thanks to the clinicians at KidsAbility who gave their time and energy to provide invaluable guidance and feedback, and who patiently answered all of my many questions.

I would like to thank my lab mates and fellow graduate students at the University of Waterloo, who made my entire degree so much brighter.

Thank you especially to my supervisor, Bryan Tripp, for his support and guidance throughout all of this project's ups and downs.

And lastly, I would like to thank my friends and my family for their love and encouragement.

Table of Contents

Author’s Declaration	ii
Abstract	iii
Acknowledgements	v
List of Figures	ix
List of Tables	x
1 Introduction	1
1.1 Motivation	1
1.2 Objectives	2
1.3 Contributions	2
2 Background	3
2.1 SLP Therapy and Documentation	3
2.2 AI in Medical Documentation	5
2.2.1 Natural Language Processing	5
2.2.2 Transformers	5
2.2.3 State-of-the-Art	7
2.2.4 Application of NLP in Healthcare	8
2.2.5 Model Inference	9
2.2.6 Retrieval Augmented Generation	11
2.3 AutoScribe Pilot	13
2.4 Automatic Speech Recognition	14
2.4.1 Speech and Audio Processing	14

2.4.2	Speech Recognition	17
2.4.3	Phoneme Recognition	18
2.4.4	Speaker Diarization	19
2.4.5	State-of-the-Art	19
3	Data	21
3.1	SLP Appointment Data	21
3.1.1	AutoScribe Pilot Data	21
3.1.2	Manual Transcription	22
3.1.3	Redaction	22
3.1.4	Speech Practice	23
3.2	Prompting and SLP-Context Data	23
3.2.1	Overview	23
3.2.2	Task Examples	24
3.2.3	Historical SOAP Notes	24
3.2.4	Universal Resources	24
3.2.5	SLP Definitions	25
3.2.6	Speech Data	25
3.2.7	Templates	26
4	Transcription Pipeline	27
4.1	Overview	27
4.2	Methods	27
4.3	Testing	28
4.4	Conclusion	29
5	SOAP Note Generation	30
5.1	Methods	30
5.2	Results	32
5.2.1	First Round	32
5.2.2	Second Round	35
5.2.3	Third Round	37
5.2.4	Fourth Round	39
5.2.5	LLM Reviewer	41
5.3	Discussion	44

6	Speaker Comparison	45
6.1	Overview	45
6.2	Speech Sample Preparation	45
6.3	Speech Quality Score Prediction	46
6.4	Generation	48
7	Conclusions	50
7.1	Summary	50
7.2	Limitations	51
7.3	Future Work	52
	References	53

List of Figures

5.1	System and user prompt structure with one-shot prompting and context provided to the model in Round 1.	33
5.2	Example of system and user prompts provided to the model for reviewing and revising generated SOAP notes.	43
6.1	Average cosine distance between child and reference speech samples for layer activations by speech quality score.	47
6.2	Average cosine distance between child and reference speech sample mel spectrograms by words and speech quality score.	48
6.3	Predicted vs. true speech quality scores by speaker for regression model trained on layer 9 outputs. Mean score is denoted by 'x'.	48

List of Tables

4.1	Mean Tokenized Length by Transcript Type	29
5.1	SOAP Note Evaluation Rubric	31
5.2	Round 1 Generated SOAP Note Lengths by Transcript and Context Type.	33
5.3	Round 1 Mean Scores for AlayaCare and Generated SOAP Notes.	34
5.4	Round 1 mean scores for generated SOAP notes without context by transcript type. The highest mean score for each rubric category is in bold.	34
5.5	Round 1 mean scores for generated SOAP notes with context by transcript type. The highest mean score for each rubric category is in bold.	35
5.6	Round 1 mean and standard deviation of ROUGE scores for generated SOAP notes compared to AlayaCare and task example notes when context is provided, for all transcript types.	35
5.7	Round 2 mean scores for generated SOAP notes by variation. The highest mean score for each rubric category is in bold.	36
5.8	Round 2 mean and standard deviation of ROUGE scores for generated SOAP notes compared to task example SOAP note with and without task example provided in input.	37
5.9	Round 3 mean and standard deviation of ROUGE scores for generated SOAP notes compared to task example SOAP note with and without task example provided in input.	38
5.10	Round 3 mean scores for generated SOAP notes by variation. The highest mean score for each rubric category is in bold.	38
5.11	Round 3 mean scores for generated SOAP notes by appointment type. The highest mean score for each rubric category is in bold.	39
5.12	Round 4 scores for generated SOAP notes compared to rounds one and three. The highest mean score for each rubric category is in bold.	41
5.13	Round 4 mean scores for generated SOAP notes by appointment type. The highest mean score for each rubric category is in bold.	41
5.14	Fourth round scores for generated SOAP notes by variation and appointment. The highest mean score for each rubric category is in bold.	44

6.1	Scores for generated SOAP notes by speech quality score variation compared to Round 4 generation. The highest mean score for each rubric category is in bold.	49
-----	---	----

Chapter 1

Introduction

1.1 Motivation

KidsAbility is a pediatric rehabilitation centre in southern Ontario that offers therapy services to children and youth with disabilities and developmental delays. Though Speech-Language and Audiology Canada’s maximum benchmark wait time for speech sound disorders is eight months (for low risk school age children), the wait for speech therapy across Ontario can be more than three years [10]. Increasing treatment capacity may be essential for reducing these delays. Clinical documentation is a necessary but time-consuming and cognitively demanding task for clinicians of many disciplines, including those focused on pediatric rehabilitation. SLP stress is contributed to by both documentation demands and workload [37]. A type of documentation used frequently by SLPs is a progress note in SOAP format, written after each appointment and stored to track a client’s treatment process. However, KidsAbility conducted an internal study into clinical documentation and found that therapists can spend 20 minutes on these documents, adding up to significant non-treatment time over the course of a working day. Reducing the time that clinicians spend on documentation may indirectly improve these wait times as clinician time is freed for treatment and help reduce stress levels as documentation becomes less of a burden. In response to this challenge, KidsAbility approached the University of Waterloo research team to explore the use of artificial intelligence (AI) to assist with documentation for their clinicians.

Initially, KidsAbility expressed interest in developing a dedicated in-house model for automatically generating draft clinical notes from session audio recordings. In parallel, the organization had begun pilot testing MutuoHealth’s AutoScribe, a commercially available AI documentation tool [44]. While AutoScribe demonstrated some success with occupational therapy (OT) and physiotherapy (PT) sessions, it performed poorly in speech-language pathology (SLP) contexts. This discrepancy is likely due to inherent differences between SLP sessions and traditional medical appointments, with speech therapy sessions often being more interactive in nature. Speech practice in particular may be difficult to both pick up correctly from an audio recording and to parse in a transcript.

Despite these limitations, the potential of AI to reduce documentation burdens remains promising. Successes in other healthcare disciplines indicate that AI-assisted documentation can significantly improve workflow efficiency and reduce clinician burnout. The challenge lies in adapting these technologies to fit the specific demands of pediatric SLP.

1.2 Objectives

The primary objective of this project was to develop a proof-of-concept AI system capable of supporting SLP documentation workflows at KidsAbility by addressing the limitations identified in existing tools. Therefore the first goal was to explore the underlying reasons why general-purpose AI tools fall short in SLP settings. As the ultimate aim was to improve the quality of draft documentation, based on these findings, the project would demonstrate practical strategies for tailoring AI solutions to SLP discipline-specific clinical contexts.

With this objective, open-source LLMs would be used for generation to showcase how improvements to drafted SOAP notes could be achieved.

1.3 Contributions

This project resulted in a proposed multi-component AI pipeline tailored to pediatric SLP. It sought to address the key issues identified based on consultations with KidsAbility's SLPs who had participated in the AutoScribe pilot study. The following components were integrated:

- **Language Model Pipeline:** A large language model (Llama 3.3 70B) was adapted using a retrieval-augmented generation (RAG)-adjacent approach, enabling it to reason about discipline-specific content within the context of each session. This design improves draft note relevance and structure compared to generic AI summarization tools.
- **Transcription Model Ensemble:** A customized transcription system was developed, combining a state-of-the-art (SOTA) automatic speech recognition (ASR) model with phoneme-level alignment and speaker diarization. This configuration ensures more accurate and complete transcription of multi-speaker pediatric sessions that is informative for sorting information into SOAP note categories.
- **Phoneme-Level Analysis:** A quantitative speech analysis module was introduced to evaluate the accuracy of a child's target phoneme productions within therapy sessions. This output is used to guide SOAP note generation, offering richer insight into speech practice while preserving the clinical interpretability of the final documentation.

Chapter 2

Background

KidsAbility is a pediatric rehabilitation centre in southern Ontario with a head office in Waterloo. They offer various therapy services for children, including speech language pathology. Previously, KidsAbility’s innovation team (referred to as the Rocket Team) had approached the Medical AI lab about a partnership with the goal of using AI tools to reduce time spent by therapists on documentation tasks. This led to a project to develop a custom language model for aiding occupational therapists in writing documentation based on scratch notes taken during appointments. Since then, KidsAbility has also piloted a commercial AI tool called AutoScribe that converts from audio of patient visits to clinical documentation. However, this tool performed poorly for speech language pathologists (SLPs). This project seeks to understand the limitations faced by speech therapists when using AI tools and develop a proof of concept pipeline to address these issues.

Interviews were conducted with SLPs who had participated in the AutoScribe pilot to shed light on the difficulties they faced with this type of tool. They were able to view both the transcription and SOAP note AutoScribe generated from their appointments. The feedback received was sorted into issues relating to the transcription and issues with the generated SOAP note, splitting the pipeline into two main segments, audio-to-transcription and transcription-to-documentation.

2.1 SLP Therapy and Documentation

Pediatric speech-language pathology is a specialized area of healthcare that focuses on assessing, diagnosing, and treating communication and feeding disorders in children, from infancy through adolescence. Speech-language pathologists (SLPs) work with children who have difficulties with speech sounds, language development, social communication, voice, fluency (such as stuttering), and oral-motor or swallowing skills. The goal is to support a child’s ability to communicate effectively and participate fully in daily activities, learning, and social interactions. Pediatric SLPs often collaborate with families, educators, and other professionals to create individualized and developmentally appropriate intervention plans.

KidsAbility’s Early Years (EY) Program focuses on children younger than school age [32]. After an initial assessment, a therapy plan will be put in place tailored to the child’s specific communication needs [18]. In the EY SLP program, parents are considered therapy partners, and appointments typically fall into one of two streams: block or coach therapy. In block therapy, therapy is provided in clustered sessions, focusing on the speech or language areas that the child has difficulty with. The goals are child specific and typically provided directly from the therapist. The child is the recipient of the goals and the target of the therapy. Coaching is a caregiver-focused approach where therapists guide adults in supporting a child’s speech and language development in daily life. The parents are the recipients of a session’s goals, with the therapists teaching parents and coaching them through the strategies for helping the child with their specific speech or language difficulties. The child then receives intervention through the parents.

Within these two streams, the subcategories of the visit are determined by what the child’s goals are. Examples of focus areas are phonology, motor speech, fluency, language, social communication, alternative and augmented communication (AAC), gestalt language processing, AVT, and feeding. The appointment stream and area will determine the overall structure of the visit and what therapy strategies may be used. Specific strategies and programs have been copyrighted by several organizations. However, these typically follow evidence-based approaches, and there has been significant public research conducted in this field. SLPs at KidsAbility reference different programs, such as those offered by Hanen [57], Lidcombe [36] and Palin [43].

Based on the appointment area, the information a therapist would expect to learn from a child’s vocalizations and behaviours will also vary. For example, an appointment targeting social communication may have the therapist pay more attention to how the child interacts with others (e.g. turn-taking) than to their articulation of words, while a motor speech appointment may focus on how the child coordinates speech movements. Children may also have issues that overlap multiple focus areas, such as language and social communication, or speech clarity, which combines articulation (a sub-category of motor speech) and phonology.

For each appointment, an SLP clinician is expected to submit documentation on the patient’s progress to KidsAbility’s electronic health record (EHR) system. These progress notes are often written in a structure called SOAP, where S stands for Subjective, O for Objective, A for Assessment (or Analysis), and P for Plan [49]. This format groups specific types of information under each category. The Subjective section includes updates from or concerns raised by the caregiver or teacher. The Objective section documents observable data from the session, including the goals worked on, therapy activities, clinical interventions, or observations, from the session. In Assessment, clinical impressions (i.e. response to treatment, progress made, clinical analysis of ongoing need) are reported. The Plan contains the plan to progress towards goals, activities the therapist might consider for the next session, or other supports/referrals. This may also include a summary of the visit and next steps (including any appointments scheduled) [33]. To help standardize documentation, most SLPs follow the guideline of documenting by exception, where unchanged information from previous appointments is not reiterated in a new SOAP note unless it impacts the current session.

KidsAbility’s internal review revealed that therapists can spend as much time with clients as on documentation. Finding ways to reduce the burden of documentation can allow for more time to be spent on treatment and help prevent clinician burnout.

2.2 AI in Medical Documentation

2.2.1 Natural Language Processing

Natural language processing (NLP) involves computer processing of human language. Similar to how language competency in humans can be assessed by ability to 1) understand verbal or written language and 2) generate verbal or written language, so too can NLP be divided into a computer’s ability to understand language inputs (Natural language understanding (NLU)) and produce language outputs (Natural language generation (NLG)). NLU uses language inputs to accomplish tasks, with applications such as comprehending user queries in virtual assistants and determining article relevance in search engines [34]. NLG will output language based on input data and is suited to applications such as content creation [14]. There are also many areas where these overlap, such as in chatbots that must both understand and respond to user queries.

Current NLP practice usually involves machine learning models to perform a variety of tasks, from speech recognition to text generation. In order for these models to work with language data, inputs usually undergo tokenization, a process wherein they are broken down into tokens corresponding to a vector ingestible by the model, performed by a tokenizer. The range of tokens known to the tokenizer is the tokenizer’s vocabulary. Most commonly, these tokens represent words, word fragments, or structural information such as punctuation.

2.2.2 Transformers

Transformers are a type of machine learning model that excel at sequence related tasks and have demonstrated competence in modelling long-range dependencies. The transformer architecture solved several issues found in the previous architectures most commonly used for language modelling tasks, namely convolutional neural networks (CNNs) and recurrent neural networks (RNNs). Though CNNs have excelled at the computer vision tasks for which they were invented, in order to properly capture longer-range dependencies they often require increasingly deep stacks of layers, and so their computational burden can increase quickly [25]. Larger receptive fields for CNNs typically require either very large kernels or many stacked layers to directly connect tokens. While RNNs can theoretically store information indefinitely in hidden states, in practice, information from distant tokens is often lost over time [4]. While the gating introduced by LSTMs may help prevent vanishing gradients, long sequences can still cause gradients to explode, limiting the effective context length [22]. RNNs are also impacted by distance between tokens, tending to place greater weight on closer tokens, which can be problematic for tasks that require recalling farther

away context [26]. They are also unable to process tokens in parallel, which slows down both training and inference, especially for longer sequences. Unlike CNNs and RNNs, transformers are able to capture long-range dependencies in text data and can process entire sequences at a time, rather than by individual tokens, directly modeling relationships between tokens through a self-attention mechanism.

This self-attention mechanism is one of the main advances of the transformer architecture. Self-attention allows the transformer to consider the importance of tokens in a sequence relative to one another, creating a representation that takes into account the relationships between tokens [59]. This is done by computing a query, key, and value vector from each input token embedding. To calculate the attention score between two embeddings, the dot product of the query vector of one embedding with the key vector of another is divided by the square root of the dimension of the key vector. After applying a softmax function to this output, each score represents the attention weight for a pair of token embeddings. These attention scores are then used to find the weighted average of the value vectors of the other embeddings in the sequence to get the final output for that token.

Self-attention is an efficient operation, with multiple attention heads able to be processed simultaneously, allowing for faster computation. This multi-headed attention allows for more complex representations to be learned, as each attention head can learn different patterns in the data. The results can then be concatenated and then combined by passing them through a linear layer to bring them to the correct model dimension for further processing.

Self-attention does not include positional information, and so positional encodings are typically incorporated into the embeddings to account for token order. This is done by adding a position embedding to the token embedding. There are a variety of types of position encodings, with sinusoidal position encodings (where sine or cosine functions with different frequencies form the dimensions of the encoding) being frequently used [59]. These embeddings can also be either learned or static.

Transformer models can include both encoders and decoders [59]. The encoder transforms inputs into their context-rich embedding representations. Its processing then captures relationships between input tokens. The decoder takes the encoder outputs and generates a target sequence, where one new token is generated at a time. It uses both self-attention (to consider previous generations) and cross-attention (to refer back to the encoder’s representations). In full sequence-to-sequence tasks (like translation), both are used together. Model architectures for classification or question-answering are often encoder only, while for generative tasks (e.g. NLG) they are decoder-only. In models like BERT, only the encoder is used, while in GPT-style models trained through Causal Language Modelling (CLM), only the decoder is used [12][51].

Transformers scale well and are often used to form foundation models (large models pre-trained on vast quantities of data [5]), such as large language models (LLMs). This term is applied to models that have hundreds of millions or billions of parameters. With this increase in scale, LLMs have also demonstrated improvement in performance, achieving SOTA across a variety of NLP tasks. All SOTA LLMs have a transformer-based backbone.

Continuous improvement of models to better align them to specific downstream tasks has also been implemented in many cases, such as through reinforcement learning from human feedback (RLHF) [48].

2.2.3 State-of-the-Art

OpenAI’s proprietary GPT models and Meta’s open source LLaMA models are SOTA for LLMs. The most recent GPT models released are under GPT-4, larger and more advanced than its predecessors, though its number of parameters is unknown. GPT-4.1 is the latest smaller release, boasting a context window of 1 million tokens and improved understanding for long context [47]. It also significantly outperforms the recent GPT-4o on Scale’s MultiChallenge benchmark, which measures ability to follow instruction. The collection of GPT models fine-tuned for chatting using methods such as RLHF have been called ChatGPT. While GPT-4 remains a SOTA benchmark in LLMs and the ChatGPT release has had significant societal impact, their proprietary nature poses challenges for applications involving sensitive or health-related data, where open systems are preferred. In contrast, Meta’s LLaMA (Large Language Model Meta AI) series offers a family of open-source models with varying sizes and capabilities, providing flexibility for researchers with different computational resources and privacy requirements.

The original LLaMA 3 release in April of 2024 included 8B and 70B parameter models, with the smaller 8B version achieving performance comparable to ChatGPT 3.5 Turbo, a notable milestone for open-weight models. This was a marked improvement from LLaMA 2, with a more efficient encoder and training set that was seven times larger and underwent filtering for quality [40]. Since then there have been several minor releases. LLaMA 3.1 expanded the architecture by introducing longer context windows and support for tool usage, with the addition of a 405B parameter model at the high end with a context window of up to 128K. LLaMA 3.2 followed with specialized models for edge computing (1B and 3B) and vision (11B and 90B), the latter built upon the 8B and 70B LLaMA 3.1 backbones. The latest version, LLaMA 3.3, currently includes only a 70B model, which outperforms many larger models (including the 405B version of LLaMA 3.1) on instruction-following benchmarks like MATH and GPQA [41]. It also holds the highest IFEval score among these open-weight models, which is a benchmark specifically designed to evaluate instruction-following quality.

Each LLaMA model benefits from a comprehensive training pipeline that is split into two stages: pre-training and post-training. This includes pre-training on large-scale datasets and post-training with fine-tuning for domain or task specificity. The post-training method was simplified and based on supervised fine-tuning, rejection sampling, and direct preference optimization [19]. These steps enable strong performance while allowing researchers to customize or extend the models for applications requiring transparency or adaptability.

In addition to architectural and training considerations, model deployment performance can be significantly influenced by quantization techniques, which reduce model size and memory footprint to enable efficient inference. Two leading approaches in 4-bit quantization are Gaussian Post-Training Quantization (GPTQ) and bitsandbytes (bnb) normal-

float 4-bit (NF4). While both aim to make large models like LLaMA 70B more feasible to deploy, their methodologies and resulting performance differ.

GPTQ is a Hessian-aware quantization method that compresses model weights by minimizing reconstruction error on a calibration dataset [17]. This approach often yields outputs that closely resemble those from full-precision (FP16) models, particularly at 4-bit precision, where many other quantization methods experience noticeable performance degradation. GPTQ operates in a post-training, offline stage, using calibration data to optimize quantization parameters such as outlier handling, per-channel granularity, and precision allocation. This enables more stable and accurate inference, especially for large models deployed in memory-constrained environments.

In comparison, bitsandbytes’ NF4 format offers broader compatibility for both training and inference (unlike GPTQ, which is inference-only). NF4 is designed to balance inference quality with training support, making it particularly useful in fine-tuning workflows (e.g. QLoRA) [11]. However, for pure inference tasks, it may introduce slightly more variability in zero-shot output quality, though the difference is often minimal and task-dependent.

For finer control, AutoRound builds upon standard GPTQ by introducing an additional optimization step that refines rounding decisions through gradient-based search. This technique has been shown to improve benchmark scores such as MMLU and GSM8K (complex reasoning tasks), particularly at 4-bit precision [9]. While the gains are modest, they are most noticeable in tasks requiring long context handling or resilience to quantization-induced degradation, potentially reducing hallucinations in these settings.

Overall, for projects prioritizing inference accuracy under compute or memory constraints, such as in clinical or healthcare settings, GPTQ (and particularly AutoRound) can provide a measurable improvement in efficiency while retaining output quality without the storage and compute requirements of full-precision models.

2.2.4 Application of NLP in Healthcare

LLMs have shown promise across a wide range of healthcare applications, including medical writing, clinical decision support, diagnostics, and education. Their ability to process and summarize large volumes of information makes them valuable tools for both clinician and patient education. LLMs have also demonstrated potential in accelerating medical research, such as in drug development. In addition, they are increasingly used to support patient engagement through chatbots or virtual assistants and in patient communications [39].

AI has increasingly been used to help aid clinicians in medical documentation tasks. LLMs are particularly suited to this role, having been trained to deal with language data. A previous University of Waterloo project with KidsAbility focused on developing a custom model for drafting OT SOAP notes based on scratch notes taken by a therapist during an appointment. The model, fine-tuned on historical examples of KidsAbility OT SOAP notes, demonstrated an increase in SOAP note quality [13].

Existing AI scribes and dictation tools have already been used in various health settings, with the majority aimed at primary care physicians for patient visits. These systems

typically take as input an audio recording of a physician consultation and output a draft of clinical documentation. MutuoHealth’s AutoScribe is one such commercial tool [44].

Applications of LLMs in healthcare come with significant privacy and ethical considerations, particularly when handling personal health information (PHI). The use of proprietary or closed-source models often requires submitting sensitive clinical data to third-party servers, raising concerns around data security, regulatory compliance, and patient confidentiality. Open-source models offer a more controlled alternative, enabling local deployment and reducing risk of PHI exposure. Beyond privacy, documentation tasks such as drafting clinical notes introduce domain-specific risks, including hallucination of inaccurate details or omission of critical information, which are issues that can compromise care and accountability [27]. There is also the risk of models returning complete training examples, underscoring the need for thorough de-identification if models are fine-tuned on patient data [55]. Conventional LLM evaluation metrics, such as ROUGE or BLEU, may inadequately reflect the quality of structured documentation like SOAP notes [63]. Previous studies done in collaboration with KidsAbility have developed task-specific rubrics to help address this, designed to assess occupational therapy notes with attention to clarity, completeness, and clinical relevance [13].

2.2.5 Model Inference

Inference is the process of providing an input to a model to get its predicted output. In the context of LLMs, this typically involves giving a prompt to the model to receive its generated text response. This is not deterministic, both at the hardware level from complex computations as well as from the model’s random sampling of words at each step based on their probability distribution rather than choosing only the most likely. The generation can be affected both by what is included in the input prompt as well as the model configuration and settings used for the generation.

These settings or hyperparameters can help shape the model’s output in response to an input. Hyperparameters often require tuning for a given task because they influence both the coherence and flexibility of the model’s responses. When set appropriately, they can improve the output and reduce unwanted behaviours, but when pushed too far, they may distort or suppress useful outputs. For example, applying a repetition penalty can reduce loops where the model restates the same phrase, but if the penalty is too strong, it may also prevent the model from repeating important information (e.g. ”take one tablet daily”) where repetition is contextually appropriate. For generative models, one of the main parameters is temperature, which controls the randomness of a generation. A temperature of 1.0 is considered a benchmark for average randomness and is typically the default setting. Lower temperature values result in responses that are more deterministic, which is suitable for outputs required to be more factual, while higher temperatures allow for more varied and creative outputs. This is because lowering the temperature shifts the distribution of the tokens’ probability to be more narrow and so only tokens with high probabilities are selected. Raising the temperature broadens the distribution so that less likely tokens have an increased chance of being selected [15].

The model will select its output tokens for each time step based on a sampling method. Greedy sampling is a method where the highest probability token is always selected, resulting in often overly rigid results with low diversity. Beam search is a method that will sample at each step and retain the top specified number of candidate sequences rather than single tokens, allowing for a larger pool of potential outputs. It will then update these sequences by generating possible next tokens for the next step, pruning all but the new best sequences. This process will continue until the task is complete. Top-p (or nucleus) is a sampling method that computes the cumulative probability distribution of the most likely tokens in descending order. Once this distribution exceeds the top-value, the rest of the tokens will be cut off. For example, a top-p of 0.9 means that only the tokens comprising the top 10% probability nucleus are considered. Top-p has had excellent practical results. Some models also make use of a repetition penalty that discourages repeated tokens or token sequences in the generated text. There are a few methods of applying this penalty, with the most common applying a penalty based on either the presence or the frequency of a token or sequences that are already part of the output sequence.

Prompting plays a central role in LLMs by guiding the model’s behavior and shaping its output based on how the input is framed. It acts as an interface between the user and the model, making it a powerful tool for adapting LLMs to a wide variety of tasks. Since LLMs are trained to predict the next token given a context, the way a task is phrased can significantly influence performance.

Prompts can include instructions, questions, or examples that define the task, enabling model adaptation without additional training. Zero-shot, one-shot, and few-shot prompting are powerful techniques for leveraging LLMs without needing task-specific fine-tuning. These techniques are especially valuable because they let users adapt language models to new tasks flexibly and efficiently, often with surprisingly strong performance, even in out of distribution domains that the model was not explicitly trained in. In zero-shot prompting, the model is given a task with only instructions and no example. This is useful for generalization when no example data is available of paired inputs and outputs. One-shot prompting provides a single example alongside the instructions, helping the model understand the format or intent more clearly. Few-shot prompting is a technique used with LLMs that involves providing a small number of task examples within the prompt to guide model behavior, inferring patterns or rules. LLMs have demonstrated an ability to replicate the formatting and style of provided few-shot examples, making this a particularly advantageous method when there is insufficient data or computational resources to perform full fine-tuning [8]. By including one or more complete examples of the desired task in the prompt, the model can better understand the expected input to output structure. Performance typically improves with the number of examples provided [58]; however, the method is constrained by the model’s context length, which limits how much information can be included. Additionally, few-shot prompting can lead to overfitting to the specific examples shown, reducing generalizability if the examples are not carefully selected. This makes the quality of the few-shot examples of great importance, as poor examples can introduce biases and lead to decreased performance for summarization-type tasks [46].

In addition to user prompts, many LLMs also support system prompts. Role tokens and ordering are used to differentiate the system prompt from the user prompt, as the model

is trained with these tokens to distinguish between prompt sections and prioritize system prompts. System prompts are usually hidden from end users and are provided to the model before user prompts. They set persistent context that will condition all responses such as establishing high-level behaviour or constraints like tone, role, or style. As system prompts provide global guidance to the model across an entire conversation or session, this makes them especially useful in multi-turn dialogue or applications with specific goals (e.g. being a helpful assistant or staying within domain boundaries). Together, prompting and system prompts form the primary interface for controlling LLM output.

Meta’s prompting guide provides tips for writing prompts for the Llama models, as most models have specific formatting that they have been trained on and that should be followed for optimal results [42]. Even with these guidelines, adjusting a prompt to best fit a task can still take testing to find optimal wording and configurations.

Prompting is important to facilitate effective communication with an LLM. Prompt patterns have been compared to software patterns, allowing for reusable ways of conversing with models for different types of tasks. [61] demonstrates how these patterns can be applied to a variety of problems with the potential to cross domains. It is important for each prompt to identify the task for a model. Example prompt patterns are to ascribe a role to the LLM that it will respond as, or to provide a template that can then be filled in by the LLM.

Text generation inferences (TGIs) can help streamline the inference process for trained LLMs. They often offer support for libraries and packages designed to improve inference speed and efficiency, such as FlashAttention and quantization. A user can provide an input prompt in text format that is then tokenized and sent to the model to predict the output token sequence. The generation will then be converted to text and returned to the user. One example of a TGI is available for HuggingFace through the Python Transformers library [23]. This TGI also allows the user to adjust several parameters controlling the generation (such as temperature, sampling methods, and the maximum number of generated tokens) through an API rather than needing to reload models.

2.2.6 Retrieval Augmented Generation

Retrieval augmented generation (RAG) is a strategy involving two components, a retriever model and a generative model, with the term first introduced in [35]. During inference time, the retriever can search a database of domain-relevant information for data related to the current input. It then provides these additional resources to the generative model to help inform its response. As an external database is being used, this database can be kept updated with no need of model fine-tuning and can supplement a lack of domain-specific knowledge in a foundational model.

The retriever can significantly affect RAG performance, as it governs how information relevance is determined. The two main types of retriever models are sparse and dense retrievers. With sparse retrievers, text is represented as high dimensional vectors where the dimensions correspond to words in the model’s vocabulary. Only dimensions corresponding to words present in the input have nonzero values. These nonzero values are usually

based on word frequency statistics, such as term frequency–inverse document frequency (TF–IDF). Sparse retrievers are typically faster and better at capturing direct similarity measures between queries and related terms, as they use token-level representations of input [53]. However, they are less effective at understanding semantic relevance and longer or more conversational queries. Query expansion can be used to help assuage these limitations by incorporating synonyms in queries. These terms are often added using LLMs or domain-specific knowledge sources. Dense retrievers use machine learning (usually neural networks) to convert queries and resources into vector embeddings. The similarities of these resource embeddings to the query embeddings are then computed, such as with cosine similarity, to find the most relevant resources. Due to this embedding matching, dense retrievers are often better at capturing similarities in semantics between queries and documents, even for more conversational queries [30]. Common dense retrievers are based on BERT [52] models due to their high quality embeddings, such as SBERT and ColBERT [31]. Hybrid retrievers have also been developed that seek to combine sparse and dense retrievers to maximize the quality of the retrieved documents. For more specialized tasks, such as those that are domain-specific like healthcare specialties, training a retriever may be appropriate.

One paper aimed to address the current gap in best practices for selecting optimal RAG configurations across different medical use cases [62]. Despite the growing popularity of RAG systems in clinical and biomedical applications, there remains little consensus on how best to implement and tailor these models to specific medical tasks. The study offers a comprehensive benchmarking framework that evaluates various retrievers and configurations, ultimately providing practical insights for medical AI developers and researchers. Among the evaluated systems, MedCPT, a contrastively pre-trained transformer model for biomedical information retrieval, emerged as the most reliable retriever, delivering consistently strong performance across tasks [24]. Traditional methods such as BM25 also demonstrated robust results and, when fused with neural retrievers, can further enhance retrieval quality [62]. These findings suggest that hybrid retrieval strategies may be well-suited for clinical domains where both accuracy and interpretability are critical.

While some forms of RAG involve fine-tuning to directly integrate the retriever with the generative model, prompt-based RAG avoids this. Though fine-tuning may improve alignment between retriever and generator, it can be costly to fine-tune or retrain to accommodate new data or a different retriever. Prompt-based RAG is another method for enhancing LLMs with contextual information through prompt engineering, requiring minimal integration as there is no fine-tuning involved. This approach retrieves relevant text segments and inserts them directly into the prompt, often using structured formats such as: Given the following context: [context]. Write a progress note: [progress note]. This method combines the strengths of a retrieval system with the generative capabilities of an LLM. Because it does not require fine-tuning, it avoids risks related to memorizing sensitive data, making it a safer option in regulated domains like healthcare. It has also outperformed existing dense retriever models in quality of retrieved content [28]. However, its performance is constrained by the context window length and the quality of the prompt design, which can affect how well the model utilizes the retrieved context.

2.3 AutoScribe Pilot

KidsAbility entered into a pilot test of MutuoHealth’s AutoScribe tool, an AI scribe for clinical documentation, with representation from each of their therapy disciplines. When using the tool, the audio from a client session is recorded and a transcript is generated. Clinicians are able to select the type of documentation to draft, and AutoScribe will return a draft of the documentation based on the visit transcript. During the first KidsAbility pilot, the tool was used for 186 appointments (with 18 SLP appointments) and had reasonable results overall for some of KidsAbility’s therapy disciplines, such as OT and PT, but performed particularly poorly for SLP appointments. Participating SLPs found the generated note in many cases to be unusable. In some instances, AutoScribe failed to generate a note altogether, returning a blank output.

Due to these results, MutuoHealth agreed to continue piloting AutoScribe for KidsAbility’s SLPs, as well as to collect the audio recordings and corresponding transcripts for these appointments. This meant that audio recordings, AutoScribe transcripts, AutoScribe note generations, and final edited AlayaCare notes could all be collected and matched to an appointment for further study. The goal was to determine why SLPs face difficulties with using these AI tools and also to suggest approaches to help mitigate these issues.

Interviews were conducted with several SLPs who participated in the pilot to better understand their clinical workflows and experiences with the AutoScribe tool. SLPs were asked to describe their experiences using AutoScribe, how it fit into their existing workflow, as well as how they envisioned such a tool could provide value for them and what improvements could make this a reality. Common issues reported by SLPs were that it put information in incorrect sections of the SOAP note, reported incorrect information, included information not relevant to their discipline, or failed to include relevant information. For reporting information in incorrect places, observations were a point of difficulty; even when summaries were dictated to the model at the end of a session, they would often get put in the wrong sections (Subjective or Analysis).

The success of the model seemed to vary greatly with the type of appointment. One SLP noted that an instance where they had had success with the model, only needing to make minor edits, was when using it during a service planning appointment. This was with no client present and so no therapist-observed information to be reported in the note. For appointments where the client is present and playing, the therapists theorized that the model may have difficulty in parsing the play script, as KidsAbility focuses on play-based therapy. In these appointments, the model often fails to pull out any of the main points from the session and confuses details. Therapists commented that it seemed unable to tell that a strategy was being employed within the conversation. For example, in one instance the model reported an update related to feeding due to an ice cream game present in the transcript that was actually targeting speech sounds.

Speech therapy seems to present particular difficulty both for transcription and note generation. It was hypothesized that the automatic speech recognition (ASR) system may not be able to properly transcribe speech sounds that are relevant to speech therapy. For example, it may be unable to capture mispronunciations (intentional or unintentional) of

words during exercises. Frequently therapists will use corrective feedback when working on a target sound (e.g. /s/), where they will still say the right word but may stretch it out or make an otherwise distorted version of it. Similarly, a child with /s/ sound difficulty may also have distorted production, such as interdental (sounding like a /th/), or in some other way that lacks clarity. As most ASR systems match audio waveforms to the most likely word or word sequence, these distortions may not be captured in the transcript as the model ‘corrects’ these slight errors.

These tools also often fail to capture child speech samples. This was confirmed when, upon inspection of the transcripts themselves, it was clear that not all speech information was correctly transcribed by the model, with utterances made by the child omitted more frequently than those by the therapist or parent. This is consistent with the child speech segments often being quieter than the adult speech contained in the audio files.

Some of these issues, particularly with the actual generation of the transcription, could be due to poor microphone quality in addition to environmental factors. For example, therapists and clients may move around the room during appointments and may be out of optimal microphone range. In the future, collecting samples with a better microphone may improve results.

It was important to narrow down the field of investigation, as speech therapy encompasses a broad range of encounters, from assessments to therapy appointments, and from those focusing on articulation to non-articulation. These can be school-based or in-centre. There are also joint appointments with other disciplines or individual appointments with a single SLP. AutoScribe’s generated notes for joint coaching appointments did not seem to suffer from the same issues, perhaps because those appointments are more medical based, with updates from other professionals on the team. The clients tend to be more complex, and with more people present the transcript may tend to be clearer and stay more on track.

For the second round of the pilot, AutoScribe added in additional prompting cues that therapists were encouraged to use in their appointments with the intent of improving some of the issues encountered by SLPs. This included specific use of ‘Observations are’ and ‘Analysis is’ to cue the model as to what SOAP sections information belonged under. The efficacy of this was unclear, with therapists reporting inconsistent results. A comprehensive SLP-specific SOAP template was also drafted and reviewed by SLPs at KidsAbility, that was more tailored to a pediatric population (e.g. specifying that reports should not include alcohol/caffeine consumption). This template was used in the AutoScribe prompt to guide the model in reporting only relevant information in the correct sections. However, this did not seem to resolve the issues.

2.4 Automatic Speech Recognition

2.4.1 Speech and Audio Processing

Mel-spectrogram generation is a common preprocessing step in audio and speech processing pipelines and is widely used in applications such as ASR [50], speaker identification [7], and

language assessment. A mel-spectrogram is a visual representation of an audio waveform that displays how the energy of various frequency components changes over time, with frequencies mapped onto the mel scale, which is a perceptual scale designed to reflect how humans experience pitch. Unlike a linear frequency scale, the mel scale provides greater resolution at lower frequencies, aligning more closely with human auditory sensitivity. By capturing key acoustic features such as pitch, intensity, and formant structure, mel-spectrograms offer a compact yet information-dense representation of speech. They are particularly useful for comparing articulation across speakers, enabling objective analysis of pronunciation, clarity, and phonetic variation in both clinical and research settings.

Though not specific to audio signals, Dynamic Time Warping (DTW) is a technique used to compare temporal sequences by aligning them non-linearly to minimize the distance between corresponding elements [45]. This can allow for articulation comparison of target phrases or sounds between speech samples. This is done by warping the time dimension of one sequence to match the other. In speech analysis, DTW is particularly helpful for handling variations in speaking rate or timing, allowing for flexible comparison of features like mel-spectrograms across different utterance samples. DTW can be applied to align entire sequences, ensuring global consistency, or to sub-sequences of interest, allowing for fine-grained comparison of a specific phoneme or phrase while ignoring unrelated portions of the signal. For investigation of phonemes, it is recommended to apply DTW to only the relevant phoneme segment, as otherwise DTW can incorporate variability from unrelated phonemes, coarticulation effects, pitch fluctuations, and word duration. As a result, the DTW cost reflects the variation of the entire sequence rather than articulatory differences specific to a target sound, such as /s/. For applications focused on phoneme-level articulation, including the full word may obscure meaningful differences and lead to imprecise comparison. Therefore, snipping the audio to isolate only relevant phoneme segments (e.g., the /s/ in sun or bus) should be done prior to applying DTW. Distance measures can then be calculated between the adjusted sequences, such as cosine distance.

Noise can affect audio processing or analysis (such as the performance of DTW) in different ways, depending on the type and severity of the interference. Mild background noise, such as a low hum or air conditioning, typically has an impact only on overall magnitude of the signal without distorting its frequency shape. However, transient noises like coughs or (in the case of several SLP appointments) Microsoft Teams notifications can produce sharp spikes in the mel frequency bands, significantly distorting the local spectral shape and potentially leading to errors in downstream tasks. Broadband noise, such as wind or amplified microphone noise, tends to raise the energy floor (particularly in the higher frequencies), making it harder to distinguish between fricatives (e.g. /th/, /v/, /f/). Reverberation poses a more substantial challenge, as it blends the time-frequency representation, blurring the clarity of fricatives and plosives (e.g. /t/, /k/, /d/) and reducing temporal precision. Microphone inconsistencies, such as variations in gain or the distance of a speaker to the microphone, can also distort the frame shape if the signal-to-noise ratio (SNR) is low, though in most cases the effect is small.

There are several mitigation strategies to minimize the impact of noise, and two will be discussed here. The first is additive noise matching, where identical background noise is introduced to both audio samples before comparison. Rather than removing background

noise, this method aims to equalize noise profiles so that differences in recording environments, such as variations in microphone static, room acoustics, or other environmental factors, do not bias the analysis. This is particularly useful when comparing subtle speech distortions, such as an interdental /s/ versus a correct /s/, where even minor differences in background noise could mask or exaggerate variations in articulation. By adding the same noise to both recordings, the comparison is placed on more equal acoustic footing, improving consistency in cosine-based DTW measures and isolating true articulation effects. Since cosine similarity is sensitive to spectral shape, which can be influenced by differences in noise floor, this approach helps ensure that the DTW cost reflects speech production rather than environmental artifacts. When the speech distortion is subtle but important, a higher signal-to-noise ratio (e.g. 25–30 dB) should be used to prevent excessive masking.

The SNR of a recording, typically expressed in decibels, can be estimated by comparing the average power of the desired signal to that of the background noise:

$$\text{SNR (dB)} = 10 \log_{10} \left(\frac{P_{\text{signal}}}{P_{\text{noise}}} \right)$$

This usually involves isolating a segment of the recording containing only noise (such as a pause between speech) to estimate P_{noise} , then using a speech-containing segment to estimate P_{signal} . Additive noise matching extends this by taking a representative noise sample from one recording and scaling its amplitude so that, when added to another recording, the resulting SNR matches that of a reference file. By adjusting noise levels in this way, differences in noise conditions between recordings are standardized, enabling fairer comparisons of signal features.

The second approach for noise mitigation is Voice Activity Detection (VAD). VAD is often a component of ASR pipelines that involves identifying whether or not audio input contains speech. This is an important step that allows audio segments without speech to be filtered out from a model’s inputs, as otherwise a model may introduce hallucinations by attempting to predict speech from silence. As deep learning based ASR models often have limits on input audio length, VAD also allows for better chunking of input data. Segments of larger audio inputs can be flagged as containing speech, which can prevent splitting audio files over syllables and potentially corrupting the transcription output. It is also useful for identifying speakers in a conversation, which will be discussed further in the speaker diarization section.

VAD’s identification of speech-containing audio segments works by analyzing short, overlapping frames (typically 10 to 30 ms) and classifying them as speech or non-speech. Traditional VAD methods use acoustic features such as short-term energy, zero-crossing rate, and spectral distribution, applying threshold-based rules to distinguish speech from silence or noise. Modern deep learning-based VAD systems instead extract more complex features (e.g. mel spectrograms, MFCCs) and train neural networks and demonstrate an improved ability to detect speech more accurately in noisy environments. Post-processing techniques, such as maintaining speech labels briefly after speech ends (hangover schemes) or using separate thresholds for speech onset and offset (hysteresis), further reduce segmentation errors. The resulting time-stamped speech intervals enable more effective audio

chunking, noise removal, and downstream tasks like ASR and speaker diarization. Pyannote has a SOTA open-source model for VAD. Silero is another strong contender, with Pyannote considered more accurate while Silero has faster processing [7][54].

2.4.2 Speech Recognition

ASR is the processing of human speech into text, making it a vital component of applications such as voice assistants, transcription services, and accessibility tools. Previously, hidden Markov models (HMMs) combined with Gaussian Mixture Models (GMMs) for acoustic modeling, with separately trained language models to capture linguistic structure, were the basis of modern speech recognition systems [38]. These modular pipelines involved distinct components (acoustic, pronunciation, and language models) that worked together to decode speech. However, end-to-end pipelines have been an increasing focus of development, with impressive results. With advances in machine learning technologies, deep learning approaches to ASR have become SOTA. These strategies are free of the assumptions HMMs make about feature properties. Deep learning’s increasing ability to capture long-term dependencies are key to its more recent success in the field, with Long Short-Term Memory models (LSTMs) [20] and more recently with transformers and attention-based models [29]. With attention-based models, a single model is able to learn all pipeline components, mapping directly from audio waveforms to text. By combining the phonetic, acoustic, and language components into one, these parts can be learned together and with greater cohesion, a shift that has led to more streamlined and often more accurate ASR systems [29]. Though this reduces intermediate steps and manual feature engineering, these end-to-end approaches remove the modularity of those that separate out the acoustic and language modelling, reducing customizability and potentially requiring re-training for specific use cases.

For both traditional and modern ASR pipelines, forced alignment plays an important supporting role. Forced alignment is the process of precisely matching a known transcript to its corresponding audio at a fine-grained level, such as phonemes or words. The process begins by using an acoustic model to estimate frame-wise label probabilities for each speech unit, forming a trellis matrix that represents the likelihood of aligning each label to each time frame. The most probable alignment path through this matrix is then found using the Viterbi algorithm, which computes the sequence of labels that best fits both the transcript and the acoustic evidence [21]. Forced alignment provides labeled data with accurate timing information important for tasks such as training data preparation, phonetic research, and pronunciation assessment, without the need for manual annotation [38]. The Montreal Forced Aligner (MFA) is one such tool that became a gold standard in alignment due to its accuracy and robustness. Though unlike modern deep learning-based approaches, MFA relies on manually engineered acoustic features and statistical models rather than neural networks. While effective in many settings, these methods are labour intensive and have limited ability to capture complex speech variability, such as coarticulation or noisy environments. Despite these drawbacks, MFA remains widely used as a benchmark and a data preparation tool for training contemporary deep learning models in speech research [38].

Transcribing SLP appointment audio faces limitations with typical ASR due to the nature of speech therapy practices. The speech-to-text models are often unable to recognize sections of speech practice and either mistake sounds for other words or omit these sections altogether from the transcript. Mispronunciations (intentional or otherwise) may be 'corrected' to the most probable word by the language model, producing a transcript that does not capture client articulation.

2.4.3 Phoneme Recognition

Phonemes are the smallest subunits of speech. For example, the words sun and bun differ by a single phoneme at the beginning (/s/ vs. /b/), which changes their meaning entirely. Understanding and identifying phonemes is fundamental in speech processing because they provide a structured, discrete representation of spoken language that bridges raw audio and higher-level linguistic units like words or sentences. There are several standardized alphabets used to record phonemes, including ARPAbet and the International Phonetic Alphabet (IPA), which is the most commonly used by KidsAbility's speech therapists. Some words have multiple pronunciations and therefore multiple IPA notations that are recognized as correct. Phoneme transcription may get around the issues typical language modelling pipelines have with certain speech therapies, as language modelling attempts to match sounds to specific words and phrases, potentially obscuring articulation details.

Historically, many speech processing systems output phonetic transcriptions as an intermediate representation. These systems map speech audio into sequences of phonemes (typically using a standardized alphabet such as IPA), simplifying the variability in speech and allowing models to focus on the phonological structure. This step occurs before language modeling or other downstream tasks. As discussed in the previous section for regular ASR, SOTA was typically achieved using systems built on HMMs combined with GMMs to statistically model the temporal and acoustic features of speech. Tools like the MFA use these methods to perform forced alignment, where phonemes are mapped to precise time intervals in audio.

As in regular ASR, the introduction of deep learning revolutionized phoneme recognition by enabling models to learn acoustic representations directly from data. Neural networks (particularly convolutional and recurrent architectures), improved the accuracy and robustness of phoneme classification. More recently, end-to-end models using techniques like Connectionist Temporal Classification (CTC) and attention-based sequence-to-sequence architectures have further advanced the field by removing the need for forced alignment, instead learning alignments implicitly, and allowing flexible mapping from audio to phoneme sequences [2]. These models better capture context and are more resilient to noise and speaker variability.

Despite these advancements, phoneme recognition still comes with significant challenges that have contributed to its typical association with high error rates. These include the handling of coarticulation effects (where phonemes influence each other) and variability from accents and dialects and the recognition of rare phonemes.

2.4.4 Speaker Diarization

Speaker diarization is the process of identifying and distinguishing individual voices within an audio recording, regardless of whether the number of speakers is known beforehand. Traditional diarization systems relied on classical signal processing techniques, manual feature engineering, and simple statistical models (e.g. HMMs) to cluster speakers [16]. While somewhat effective, these methods required significant manual intervention and often struggled to generalize to out-of-domain data.

SOTA today is a dual-model approach that is able to capture distinctive spectral features (e.g. pitch), with one model for segmentation and one for speaker embedding generation [6]. In recent years, Pyannote has emerged as a leading and open source toolkit for speaker diarization research and applications, and its pipeline will be used to illustrate the speaker diarization process. The pipeline begins with speech activity detection (i.e. VAD) to identify speech-containing regions. Then, during segmentation, the segmentation model analyzes short overlapping windows of audio, typically lasting milliseconds and spaced slightly apart, to detect when one voice transitions to another. Each window may contain up to a set limit of local speakers (e.g. Pyannote defaults to three), independent of speaker identities identified in other windows. To link these local speakers across windows and determine whether they represent the same individual (global speaker), an embedding model generates vector representations capturing speaker-specific characteristics such as pitch and tone. These embeddings are compared across windows, with similar embeddings clustered together to represent the same speaker while dissimilar ones indicate distinct speakers. To improve clarity, segments with overlapping speech are often excluded before embedding extraction. After clustering, embeddings associated with each global speaker are merged to form a final representation, followed by a binarization step that consolidates overlapping windows and assigns global speaker labels to continuous audio segments [7].

There are also distinctions between mechanical split diarization, where audio is divided based on simple rules or channel separation, and AI-based diarization, which uses learned voice footprinting to identify speakers. For example, many speech models (such as Whisper) process audio as mono-channel by downmixing multi-channel inputs, which makes for more efficient processing while typically maintaining recognition accuracy [50]. However, this can limit the use of spatial cues that might otherwise aid diarization in multi-speaker or multi-microphone scenarios.

Despite these improvements, remaining challenges in diarization include managing overlapping speech segments and the memory demands associated with processing long audio.

2.4.5 State-of-the-Art

Whisper, developed by OpenAI, is widely regarded as a SOTA open-source model for ASR, approaching human accuracy and robustness. Its architecture is based on an encoder-decoder transformer design, where the encoder processes 80-channel log-mel spectrograms extracted from 30-second audio segments, and the decoder uses a sequence-to-sequence framework to generate text. Whisper was trained on a diverse dataset of 680,000 hours

of multilingual and multitask paired audio-text data, achieving strong performance across many languages and robustness to noise, accents, and varied recording conditions. The model family includes multiple sizes, from the lightweight tiny (39M parameters) and base models for real-time or resource-constrained scenarios, to medium and large (1550M parameters) models for maximum accuracy. These models have “.en” versions optimized for English-only tasks. Whisper supports a variety of inference settings that affect output style and accuracy, such as temperature sampling for adjusting diversity, beam search for setting level of deterministic decoding, and task modes for transcription versus translation. It also includes options for VAD and timestamp generation, making it flexible for integration into both research and production ASR systems [50].

WhisperX is a Whisper-based pipeline that includes or refines several features that Whisper either does not include directly out of the box, such as speaker diarization or phoneme transcription, or that have room for improvement [3]. For example, Whisper generates timestamps for utterances rather than per word, and these can be several seconds out of sync. The WhisperX pipeline applies forced alignment for better matching of transcript to audio timing. For fine-grained alignment, WhisperX employs phoneme-based transcription with phoneme fine-tuned wav2vec2, matching the phoneme recognition predictions to the Whisper transcription for final results [21]. Though Whisper does perform VAD, WhisperX introduces an additional VAD preprocessing step that reduces hallucinations. It also uses Faster-Whisper in the backend, which is a faster batched version of Whisper that removes silent segments longer than two seconds with Silero VAD, further improving speed [56]. Multispeaker ASR is an additional supported feature, with speaker diarization from pyannote-audio integrated to assign speaker ID labels. WhisperX has performed well, being awarded first place at Ego4d transcription challenge.

Chapter 3

Data

3.1 SLP Appointment Data

3.1.1 AutoScribe Pilot Data

From the continuing AutoScribe pilot, KidsAbility was able to provide the audio files, transcriptions, and generated SOAP notes from AutoScribe for 30 EY SLP appointments, along with their matching finalized SOAP notes from KidsAbility’s EHR, AlayaCare. Of these appointments, 10 were selected to be used for testing the transcription and generation pipelines. This selection was made to cover a range of SLP EY appointment types, making sure to include sessions with speech practice as these scenarios were flagged as having particularly poor performance by AutoScribe. 3 appointments were for block therapy targeting articulation and phonology, while the remaining 7 were coaching sessions targeting language and speech clarity, social communication, fluency, and AAC. Several visits had multiple target areas. The audio files provided for each appointment were mono-channel and in .wav format. These were normalized to help improve quality, as the speakers could be difficult to hear.

When tokenized, the AutoScribe transcriptions had a mean length of 4769 tokens, with a standard deviation of 2591. Though AutoScribe did not assign speakers to the transcription, child speech in particular seemed poorly captured by AutoScribe in these appointments from comparison with the audio recording. However, additional factors such as microphone and therapy room set up could have affected the quality of the audio. Neither speaker or timestamp information was captured in the AutoScribe transcript.

The tokenized AutoScribe generated notes had a mean length of 306 tokens, with a standard deviation of 51. This was shorter than the tokenized AlayaCare notes, which had a mean of 467 tokens with a standard deviation of 171. Upon inspection of the generated notes, AutoScribe had written only “N/A” for the Subjective section in 8 out of the 10 notes. In contrast, only one of the AlayaCare notes lacked any details in the Subjective section.

3.1.2 Manual Transcription

To create ground truth transcriptions to compare against the proposed transcription pipeline and AutoScribe’s generated transcriptions, I manually transcribed each appointment from the provided audio recordings. However, due to audio quality and human error during transcription, there may still be missing or incorrectly transcribed information. The transcript was divided into speaker segments, with every new line containing a speech segment with a label corresponding to the current speaker (e. g. [Therapist], [Parent], [Child]). No timestamps were included in the transcript.

To determine the importance of capturing child speech, the ground truth transcription was adapted to create two additional versions. While the ground truth contained child speech as recorded with the standard English alphabet, another version replaced these sections with the phonemes transcribed by a phonetic model during the periods of child speech. The third version removed segments of child speech from the transcription entirely.

The phonetic transcript was created using a wav2vec2 model fine-tuned to output phonemes for audio input [60]. Segments of the audio recordings that contained child speech were extracted and run through this model to produce the corresponding phonemes. These phoneme outputs were manually filtered for child speech from other speakers in that section, and the cleaned phoneme output was used to replace the manually recorded child speech segments in the full transcript.

By creating these alternate sets of transcripts, the impact of incorporating child speech into the transcript and the difference in capturing phonetically correct child speech segments rather than simply the approximated word from a typical ASR mode could be better assessed.

Two transcripts (ID10 and ID17) were trimmed so that the full context input would be within the model’s maximum context length that would fit on the GPU instance (ml.g5.48xlarge). The manual transcriptions were significantly longer than the AutoScribe transcriptions for each appointment, with a mean of 9417 tokens (SD=1704) for the regular version. The final lengths for the phoneme transcripts had a mean of 9935 tokens (SD=1658), while the version without child speech had a mean of 8728 tokens (SD=2041).

3.1.3 Redaction

All data was stored and processed within KidsAbility’s Amazon Web Services (AWS) environment. KidsAbility removed all externally linked PHI from the SOAP notes before the research team was able to access them. Any PHI contained within the SOAP notes was redacted with a corresponding category label (e.g. [NAME] for names, [ADDRESS] for physical addresses).

However, as the research team performed manual transcription directly from appointment audio files, there was unavoidable access to incidental PHI mentioned during the appointment, such as first names. Any PHI encountered while transcribing was redacted in the same manner as the SOAP notes, keeping the labeling system consistent.

3.1.4 Speech Practice

To support a more detailed analysis of speech production, audio samples of the phonemes targeted during sessions were isolated. One of the block articulation sessions was chosen as the appointment to draw these samples from, as it targeted speech production of the /s/ sound. Segments where the target sound was practiced within the larger audio file were identified and corresponding instances of the child’s speech extracted. These were paired with reference pronunciations from the therapist as well as a recording from a researcher, which was modified using additive noise matching to better reflect appointment session acoustics (discussed more in Speaker Comparison).

Each of the child’s phoneme attempts was manually categorized into one of five levels (1: very poor, 2: poor, 3: fair, 4: good, 5: perfect). This was done by the same KidsAbility SLP who had been the therapist for the appointment. The SLP listened to the audio segments where the child said a word containing the target sound and based the speech quality rating on perceived accuracy and intelligibility. This categorization enabled subsequent quantitative comparisons across speaker types and provided labels for evaluating pronunciation quality using learned audio representations.

3.2 Prompting and SLP-Context Data

3.2.1 Overview

Providing additional context to the model may help improve its outputs, particularly for a specialized task it may not have received explicit training on. To enhance the relevance and clinical accuracy of language model outputs, a collection of SLP- and SOAP note-specific materials were curated to serve as contextual input during inference. These resources were chosen to reflect the language, structure, and priorities of pediatric SLP documentation. The contextual set included historical EY SOAP notes written by clinicians, task examples of paired appointment transcripts with SOAP notes, and parent-facing handouts that reflect how progress and goals are communicated outside the clinical setting.

In addition, SLP definitions of key terms and therapy strategies were incorporated, offering insight into the language and concepts used within the discipline. Templates outlining standard SOAP note structures were also included to reinforce the expected organization and tone of clinical documentation, providing guidance for SLP-specific information belonging in each section.

These materials were not used to fine-tune the model, but rather to guide its generation behavior during inference. With discipline-specific context, the goal was to produce outputs that are more aligned with existing clinical expectations.

3.2.2 Task Examples

To be able to provide full examples of the task for the model, the 10 appointment transcripts and SOAP notes were re-visited. SLPs matched each of these appointments with their most closely related appointment. Due to the length of the transcripts and context length constraints, only one full example would be able to fit for each appointment. Most appointments had clear matches within the other appointments, though not all matches were one-to-one. However, the appointment for ID15 did not have a strong match within the other 9 examples, though ID5 was determined to be the closest and so was used as its task example.

3.2.3 Historical SOAP Notes

Historical EY SLP SOAP Notes were selected and reviewed by KidsAbility’s SLP clinical manager to be of a style and quality that KidsAbility approves and encourages. These notes were chosen to cover key areas of SLP within EY. 15 SOAP notes were provided and de-identified, spanning several appointment categories. 9 of these notes were for Block therapy appointments, targeting language (receptive and expressive), phonology, articulation, motor speech, and fluency. 1 note was for feeding. 5 notes were for parent coaching appointments, targeting language (social communication, receptive language, and expressive language), speech production, social communication, and permanent hearing loss. As the area of each SOAP note was recorded, this allowed the research team to match these example notes with similar sample appointments to provide additional guidance for SOAP note structure and content.

3.2.4 Universal Resources

KidsAbility maintains an internal repository of universal resources, sorted by therapy area and composed of various information packages targeted at parents. Relevant handouts may be provided to parents after sessions, as therapists will often give parents one or two strategies to work on. Handouts usually contain general strategies, with multiple handouts often existing for a single strategy, or multiple strategies in a single handout, though a clinician will usually have particular handouts that they prefer to disperse. The relevance of a strategy is determined by the main concerns of the therapist for how the child is presenting and what has been worked on in a given session. These documents are typically pdfs, with some videos, assessments, and worksheets as well. Though available to all KidsAbility SLPs, therapists noted that these handouts were not always distributed as they were considered text-heavy.

It is important to note that, as these handouts are specifically designed for parents, they may use different vocabulary to describe conditions and activities than an SLP would report in their SOAP notes. However, information they contain may still be relevant for documentation. To process this information and determine what may be useful as context for the model, KidsAbility SLPs were asked to go through the SLP-specific resources.

Specific areas targeted by these handouts are fluency/stuttering, language disorders, speech sound disorders, and voice. They also included the resources for the SPOT program, which combines SLP and OT. For each handout, the SLPs recorded whether all, most, some, or none of the document contained relevant SOAP note information. If only part of a document was relevant, they were also asked to note down which sections. Additionally, the SLPs recorded which appointment categories the information was relevant for, along with the IDs of any related historical SOAP notes.

Based on this assessment of relevance, universal resource context was developed for each of the 10 sample appointments. This was done by taking the most relevant handouts and handout sections for each appointment type and combining them to create the finalized parent resource context, matched to each sample appointment. All information was thoroughly checked to make sure it was properly formatted when stored as input context.

3.2.5 SLP Definitions

To provide additional clinically relevant context for SOAP note generation, a set of definitions and descriptions written by practicing SLPs at KidsAbility was created. This aimed to address the gaps in universal resources, particularly related to clinical terminology, as the parent handouts used more common phrasing.

SLPs were asked to define key terms in their discipline, such as common disorders, strategies, and therapy tools. A draft document was provided by the research team as a guide, containing SLP therapy categories and terms that had appeared in the historical SLP SOAP notes. However, the clinicians were encouraged to provide any terms and sections that might be missing yet were relevant for SOAP notes.

The resulting glossary included brief, clear definitions for clinical terms, as well as intervention-specific strategies. It also described typical appointment types along with strategies associated with each, providing a structured reference for how certain therapeutic approaches are applied in practice. The collected material was reviewed to ensure consistency and clarity. Although SLPs were given flexibility in phrasing, definitions were edited only lightly for grammar and formatting. Where possible, redundant entries were merged, and similar concepts were compared to reduce ambiguity.

These definitions were used as contextual input to enhance the relevance and interpretability of model-generated content. By grounding the model in discipline-specific language from actual clinicians, the approach aimed to improve terminology use and align SOAP note generation with the expectations of practicing clinicians.

3.2.6 Speech Data

In addition to discipline-specific terms, another context piece added for SLP appointments was information about what is looked for in child speech samples and how this varies by type of session. SLPs were asked to provide a brief explanation of what a therapist would

learn from a child’s vocalizations during an appointment for each of the appointment sub-categories represented in the 10 sample appointments. This information could better equip the model to understand what parts of the transcript are important for certain sections of the SOAP note, as the attention given to different aspects of a child’s vocalizations vary widely. For example, block therapy for articulation and phonology may look at the phoneme level of speech productions, while a language appointment may involve commenting on the length of phrases and types of words used. Though what is reported in the note is highly goal dependent, and not all appointment categories focus on speech, if certain errors are observed that are not age appropriate they would still be mentioned in the note.

3.2.7 Templates

Templates for SOAP notes were included in the system prompt to guide the model in producing outputs aligned with current SLP documentation practices. Two main variants were used: one adapted from the AutoScribe pilot (to help improve model outputs for SLPs), and another drafted by a practicing SLP at KidsAbility. The pilot-based template offered detailed guidance for each section. It gave comprehensive explanations and examples of what could be included in each category for different SLP appointment types. However, due to the prescriptive wording and breadth of information covered in the template, it was theorized that it may contribute to the model’s repeating of template instructions verbatim or including information not relevant to the current session (e.g. reporting on articulation in an AAC session). This led to the development of the second template, which provided brief section-level descriptions with the aim of being practical and informative but more general. This way, the content was relevant to all appointment types. These templates served as archetypes, informing the model about the expected structure and content of SOAP notes without prescribing exact wording. A hybrid template was also developed that made use of the SLP template definitions but drew from the AutoScribe template’s examples of what could be included in each section, with the aim of capturing both versions’ strengths.

Chapter 4

Transcription Pipeline

4.1 Overview

As the appointment transcript was to be the primary source of appointment specific information to the model when drafting a SOAP note, improving the quality of the transcription was of significant value. With this in mind, ASR pipeline was created to test whether a better transcript than the one produced by the AutoScribe system could be produced using SOTA methods, and if so, how it might impact SOAP note generation.

The AutoScribe transcripts did not include speaker diarization, but this information may be valuable for accurate SOAP note generation. Several therapists remarked on the model’s misattribution of what they said to parents and vice versa in the drafted SOAP notes. This is a barrier to note quality, as information relayed by parents and clinicians do not have the same clinical authority, with information from caregivers treated as a different category and reported in certain sections of the note. Having this information as part of the transcript could improve transcript clarity and help the model create better organized and more accurate SOAP notes.

As most ASR systems rely on language models, they may struggle with the speech practice found in SLP appointments. To help assuage this, the research team set up an audio to phoneme model. These phoneme outputs would replace areas of text where word confidence was low.

4.2 Methods

A custom transcription pipeline was developed to address the challenges of transcribing pediatric speech therapy sessions, particularly those involving child speech and overlapping speakers. The pipeline was modified from WhisperX [3], which aligns Whisper ASR outputs with wav2vec2 embeddings [60] to generate accurate word-level timestamps necessary for speaker diarization. This alignment method offers improved temporal precision and long-form transcription, making it well-suited to extended therapy sessions.

The core ASR model used in the pipeline was Whisper, selected for its SOTA performance across diverse acoustic conditions. Whisper’s outputs were supplemented with timestamp alignment using a standard wav2vec2 model, following the strategy introduced by WhisperX. This allowed for consistent word-level alignment without requiring separate decoding stages.

To improve recognition of child speech and low-confidence regions in the transcript, a phoneme-level wav2vec2 model was incorporated. Whisper’s confidence scores for predicted token sequences were returned from the model, and in cases where Whisper assigned low confidence scores to predicted words (typically in child utterances), these words were replaced with phonetic representations using IPA symbols generated by the phoneme wav2vec2 model. This substitution enabled the representation of speech that may not be fully intelligible or correctly transcribed by conventional ASR models. The ability of wav2vec2 to effectively capture the acoustic characteristics of child speech was a central consideration in this component of the pipeline.

Speaker diarization was integrated using WhisperX’s implementation of pyannote/speaker-diarization-3.1, which assigns speaker labels to the aligned transcript. Pyannote operates offline by default, which is important for maintaining data privacy. This step was particularly important in distinguishing between the speech of the therapist and the child, as well as identifying overlaps and interruptions during dialogue. Including speaker information in the transcript improved overall clarity and was expected to support the language model in producing more organized and accurate SOAP notes, especially when attributing specific observations or actions to the correct speaker.

Multiple versions of the transcript were to be generated and compared to assess the impact of different system components. These comparisons included: 1) transcripts generated by AutoScribe versus those from the custom Whisper-based pipeline, 2) versions with and without speaker diarization, 3) transcripts that included or excluded child speech, and 4) versions where child speech was replaced with IPA phonemes in low-confidence regions.

These comparisons would inform decisions about pipeline configuration and be used to evaluate the effectiveness of each strategy in enhancing transcript accuracy and interpretability for SOAP note generation.

4.3 Testing

Each of the 10 audio files were passed through the transcription pipeline to output final transcriptions. The four variations previously discussed were created by adjusting settings for the pipeline to allow for assessment of different components. In one version, the phoneme-replacement was unused, with sections of low-confidence tokens transcribed as the main Whisper model’s chosen outputs rather than replaced with the corresponding phonetic transcription. In others, the threshold for phoneme replacement was varied to understand which parts of the audio file the model struggled with.

However, for each variation of the pipeline, the resulting transcriptions were far from the quality of the manually generated transcriptions. They were not able to capture more

Table 4.1: Mean Tokenized Length by Transcript Type

Transcript Type	Mean Tokenized Length
Manual	9416.5
AutoScribe	4768.8
Whisper	2347.0

detail than the AutoScribe transcript, lacking significant content from the session. As seen in Table 4.1, the mean tokenized length is much lower for the transcripts generated by the Whisper-based pipeline.

4.4 Conclusion

Based on the results of the tests conducted on the example audio files, the transcription pipeline does not seem to produce significantly better results than the AutoScribe ASR system. It is suspected that the transcription issues experienced here are significantly contributed to by poor audio quality, as even manual transcription was difficult. This could be contributed to by audio issues such as low microphone quality or environmental factors such as microphone placement and room set-up, in addition to rate of speech and accents. Many therapists use the built-in microphone in their laptops, which is not carried around as they move about the room during appointments. The current audio quality prevents determining what are barriers to transcription due to the nature of speech therapy and what are the result of equipment and environmental factors. Therefore, in depth analysis of the quality of phonetic child speech and speaker diarization will not be further explored in this thesis. If AI scribes continue to be piloted for SLPs, investigating optimal set-up to reduce errors in the transcript would be highly valuable, as the transcript is the main source of visit-specific information. At this time, KidsAbility does intend to test the system with improved microphones to see how this impacts transcription issues.

For investigating issues with SOAP note generation in the following sections, the research team will move forward with an idealized version of the pipeline to prevent magnification of errors in the pipeline. The inputs used will be the manually curated transcriptions and context, as if the transcription pipeline could perfectly recognize human speech, convert speech to phonemes, recognize speakers and their roles, and if the system could perfectly recognize appointment type and assign relevant RAG info.

Chapter 5

SOAP Note Generation

5.1 Methods

I compared SOAP note generations based on AutoScribe transcripts and ground truth transcripts. AutoScribe’s generation pipeline was not accessible, but the same LLM was used to generate SOAPS for all transcript types to allow for direct comparison of the outputs.

Llama 3.3 70B was chosen as the generative model for these experiments. It was accessed through the HuggingFace transformers library and hosted on an AWS SageMaker endpoint, a ml.g5.48xlarge GPU instance. With this set-up, an input context length of 30k tokens could be processed at one time, which was enough space for an input transcript, a task example, and the additional SLP context deemed relevant for an appointment.

SOAP notes were evaluated using a rubric based on [1], with five categories covering important aspects of SOAP note quality: Clear, Complete, Concise, Relevant, and Organized. An additional category for Accuracy was added after request by the SLP evaluators, to help capture whether information was true to the events of the appointment. However, given the small sample size and to capture more nuanced information, the 3 point Likert scale was adapted to a 7 point scale. The criteria for each of these categories and point scores were developed in conjunction with the SLPs. Details of the rating system and categories are shown in Table 5.1.

Two SLPs from KidsAbility participated in ranking the SOAP notes, dividing the task by rubric categories into those dealing with note content (Complete, Relevant, Accurate) and note structure (Clear, Concise, Organized). One SLP evaluated note content, while the other assessed note structure. This separation was made because the SLP that evaluated content had been the clinician for the majority of the appointments and so had a more thorough understanding of the underlying appointment, while the second SLP would have a less biased perspective between appointments for how the note was structured.

To simulate an idealized RAG pipeline, the context gathered previously was organized and matched with the appointments and SOAP notes they were most relevant for. For each appointment, context could include a task example (matching transcript and SOAP note),

Table 5.1: SOAP Note Evaluation Rubric

Category	1	2	3	4	5	6	7
Clear	Very Poor: Confusing, ambiguous, difficult to interpret	Poor: Frequently unclear, contains vague or imprecise language	Fair: Occasionally unclear, requires effort to interpret	Adequate: Mostly clear, minor ambiguities present	Good: Clear and understandable throughout	Very Good: Highly clear, precise and professional language	Excellent: Exceptionally clear, well-articulated and easy to follow
Complete	Very Poor: Critically incomplete, major components missing	Poor: Key elements absent, significant information gaps	Fair: Some important elements present, others lacking	Adequate: Covers essential content but may lack depth	Good: Includes all expected content with minor omissions	Very Good: Thorough and well-detailed, most necessary information included	Excellent: Fully comprehensive, no omissions, exceeds expectations
Concise	Very Poor: Excessively wordy, redundant, lacks focus	Poor: Verbose, contains unnecessary repetition or tangents	Fair: Some unnecessary detail or repetitive phrasing	Adequate: Generally focused, minor verbosity	Good: Well-edited, avoids unnecessary repetition	Very Good: Efficient and streamlined presentation	Excellent: Maximally concise without loss of clarity or meaning
Relevant	Very Poor: Off-topic, includes unrelated or distracting content	Poor: Marginal relevance, includes non-clinical or irrelevant data	Fair: Generally relevant, but includes occasional unrelated details	Adequate: Largely relevant, with minimal extraneous content	Good: Appropriate and purposeful information included	Very Good: All content directly supports clinical focus	Excellent: Every detail is targeted and supports clinical reasoning
Organized	Very Poor: Disorganized, lacks logical flow, hard to follow	Poor: Inconsistent structure, difficult navigation between sections	Fair: Some structure present but lacks consistency	Adequate: Mostly follows expected structure, some transitions unclear	Good: Logical structure, sections well-delineated	Very Good: Seamless flow, easy to navigate and understand	Excellent: Exemplary structure, intuitive, professional, and coherent layout
Accuracy	Very Poor: Major components of the visit are fabricated or misrepresented; does not match visit type or purpose	Poor: Multiple significant discrepancies between note content and actual visit; visit type, findings, or actions are inaccurately portrayed	Fair: Some alignment with the visit, but important factual errors reduce reliability	Adequate: Generally accurate, but minor factual inconsistencies or assumptions are present	Good: Accurately reflects the visit with small, non-critical or minor inferred details	Very Good: Strong alignment with actual visit; details are factually correct, appropriate to the visit type, and well-documented	Excellent: Completely accurate; all included information is factual, specific to the visit, and verifiably reflects what occurred in real time

SLP definitions, and parent handouts specific to the appointment category. Templates to guide SOAP note structure were also included in the system prompt.

5.2 Results

5.2.1 First Round

In the first round of SOAP note generation, four different transcript variations were used:

- **AutoScribe:** original AutoScribe-generated transcript.
- **Manual:** manually created ground truth transcript.
- **Phonetic child speech:** manual transcript with child speech segments replaced by their IPA representation.
- **No child speech:** manual transcript with child speech segments removed.

For each transcript variation, the input was tested with and without providing context (matched task example, SLP definitions, and parent resources), resulting in 8 possible variations for each of the 10 appointments. A low-randomness temperature setting of 0.5 was used to curb potential hallucinations and encourage adherence to formatting guidelines.

The initial instructions to the model regarding its role were based on the system prompt used in [13]. The system and user prompts with full context provided are shown in Figure 5.1.

System Prompt

You are a helpful, respectful, and honest assistant for a speech language pathologist (SLP) in a pediatric rehabilitation clinic. You follow these rules:

1. Follow directions meticulously.
2. Write in point-form. Do not write in paragraphs.
3. Do not produce any extra text. Only write what is asked for.
4. Use the following headings: Subjective, Objective, Analysis, and Plan. After these sections, write "End of Note."
5. Answer as helpfully as possible, while being safe.

Your task is to produce SOAP notes from transcriptions of appointments. If context is provided, use it to help inform the SOAP note.

I am going to provide a template for your output in SOAP format.

Try to fit the output into the relevant sections of the SOAP template.

Please preserve the formatting and overall template that I provide.

This is the template:

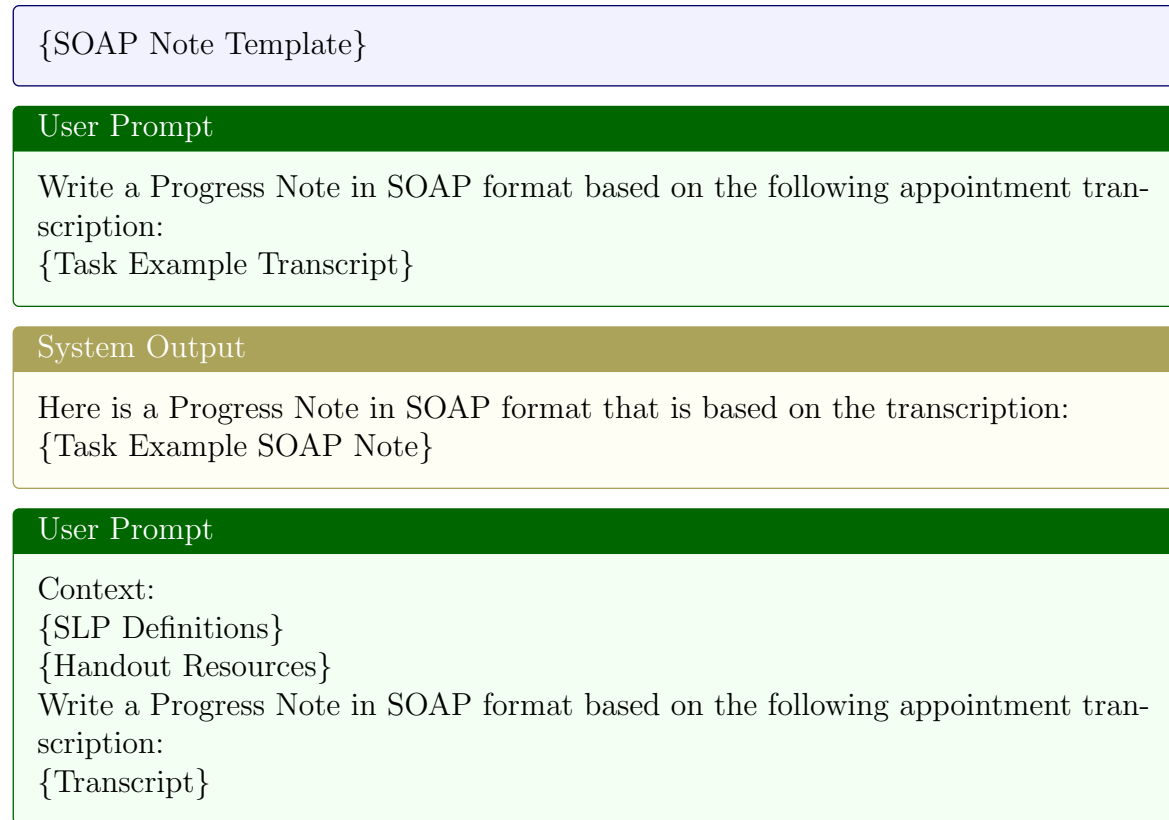


Figure 5.1: System and user prompt structure with one-shot prompting and context provided to the model in Round 1.

80 SOAP notes were generated in the first round, with the 10 AlayaCare SOAP notes for each appointment also needing to be ranked by therapists for comparison. Two SLPs from KidsAbility evaluated all 90 SOAP notes, dividing the task by the six rubric categories into those dealing with note content (Complete, Relevant, Accurate) and note structure (Clear, Concise, Organized). The SLPs discussed between themselves to ensure the rankings were consistent and accurate.

The generated note lengths for each transcript type can be found in Table 5.2. SOAP notes generated from manual transcripts were typically the longest both with and without provided context, with notes generated from AutoScribe and phonetic transcripts the shortest.

Transcript Type	No Context	Context
AutoScribe	1226.9	1380.6
Manual	1421.8	2086.1
Phonetic	1226.9	1382.9
No Child Speech	1275.3	1454.5

Table 5.2: Round 1 Generated SOAP Note Lengths by Transcript and Context Type.

Input	Clear	Complete	Concise	Relevant	Organized	Accurate
AlayaCare	6.700	6.500	6.600	7.000	6.400	6.500
All AI-Generated Notes	2.838	2.538	2.988	2.938	3.325	2.762
Notes Generated without Context	2.100	2.050	2.200	2.475	2.35	2.250
Notes Generated with Context	3.575	3.025	3.775	3.400	4.300	3.275

Table 5.3: Round 1 Mean Scores for AlayaCare and Generated SOAP Notes.

Transcript Type	Clear	Complete	Concise	Relevant	Organized	Accurate
AutoScribe	2.1	2.1	2.6	2.3	2.7	2.1
Manual	1.8	1.8	1.7	2.3	2.1	2.2
Phonetic	2.2	2.2	2.3	2.6	2.5	2.1
No Child Speech	2.3	2.1	2.2	2.7	2.1	2.6

Table 5.4: Round 1 mean scores for generated SOAP notes without context by transcript type. The highest mean score for each rubric category is in bold.

From Table 5.3, the AlayaCare notes (SOAP notes submitted by SLPs to KidsAbility’s EHR) were rated significantly higher than the generated notes (with and without context) across all categories, and so there is clear room for improvement. Table 5.4 shows results for the notes generated without additional context, where AutoScribe had better organization and conciseness than the other transcript versions, while the phonetic and no child speech transcripts had better relevance and the manual transcript performed the worst overall. The SOAP notes generated using context consistently ranked higher than those without context across all six categories, with better accuracy for non-AutoScribe transcripts, as seen in Table 5.5. Manual and phoneme transcripts were more complete, while manual performed the worst for the note structure categories.

However, the SOAP notes generated with context were found to have information repeated from the task example SOAP notes. This was confirmed from inspecting the notes and is supported by Table 5.6, which shows the mean and standard deviation for ROUGE scores comparing the notes generated with context to the AlayaCare and task example SOAP notes. The task example comparison yields significantly higher ROUGE scores. This may have confused the evaluations, as the note structure may have been of high quality and the content relevant to the sections, but only because the model had stolen from the example and not because it had generated a SOAP note that matched well with the input transcript.

A trend noticed by the SLPs was that model-generated notes tended to repeat the same information for Subjective, Objective, and Analysis sections, or to put information in the incorrect section. The model also sometimes failed to pull out the main purpose of the notes. The SLPs noted that it seemed to struggle the most with pulling the main goal for notes in the fluency category, instead converting them to speech and language. This could be because fluency was discussed less during the visit than other topics.

Additionally, strategies during these appointments typically involve asking parents to

Transcript Type	Clear	Complete	Concise	Relevant	Organized	Accurate
AutoScribe	3.5	2.8	4.3	3.3	4.6	2.8
Manual	3.1	3.2	2.7	3.3	3.5	3.4
Phonetic	3.8	3.4	3.9	3.6	4.5	3.6
No Child Speech	3.9	2.7	4.2	3.4	4.6	3.3

Table 5.5: Round 1 mean scores for generated SOAP notes with context by transcript type. The highest mean score for each rubric category is in bold.

Score Type	Alaya Care		Task Example	
	Mean	Standard Deviation	Mean	Standard Deviation
ROUGE-1	0.289	0.070	0.400	0.166
ROUGE-2	0.065	0.048	0.199	0.206
ROUGE-L	0.153	0.043	0.273	0.188
ROUGE-LSUM	0.276	0.064	0.387	0.169

Table 5.6: Round 1 mean and standard deviation of ROUGE scores for generated SOAP notes compared to AlayaCare and task example notes when context is provided, for all transcript types.

model strategies rather than directly asking the child. However, the model would often report that it was the child who had been the target of the strategy. The generated notes also tended to pull wording directly from the template that led to confusing outputs, repeating the template instructions rather than filling in information from the session.

5.2.2 Second Round

A second round of SOAP note generation and evaluation sought to address the repetition of the example SOAPS. It was hypothesized that the model was overfitting to the single example SOAP note from the task example, and so the example historical SOAP notes were used at this stage to provide further guidance on SOAP note structure. Additionally, the long context length could also have been overwhelming to the model, and so condensing and decreasing the amount of additional information included was also tested by removing parent handouts from the input. A new template was introduced in this round that was more general and less prescriptive and included section explanations written by a KidsAbility SLP rather than the AutoScribe template. This was to prevent the model from recycling template wording and details that were not relevant for every session. Another adjustment made was to increase the temperature to 0.6, as low temperatures encourage the model to favour high-probability outputs, which can lead to repeating few-shot examples or ignoring slight differences in prompts. The goal of the temperature increase was to have less rigid and more natural responses in addition to less repeated information between sections while still having a temperature low enough to discourage hallucinations.

For this iteration, only the transcripts with the child speech represented as phonemes were used, as they had scored relatively highly in the first round and choosing a single

transcript would allow for more fine-grained analysis of the other input components. For each of the 10 appointments, 4 variations were tested with the phonetic transcripts and raised temperature. For each variation, the temperature was set to 0.6. The Round Two variations are summarized as:

- **Variation 1:** Same full context input as Round One.
- **Variation 2:** Parent handouts removed and replaced with four example historical SOAP notes chosen for their detail and adherence to standard SOAP structure (the number of historical notes for this case was limited due to context length constraints).
- **Variation 3:** Task example removed and replaced with 24 example SOAP notes (the 15 historical SOAP notes and the 9 other example SOAPS from the appointments).
- **Variation 4:** Task example and parent handouts removed and replaced with same 24 example SOAP notes as Variation 3.

The 40 notes were generated and ranked, with all categories scored this time by the SLP who had been the clinician for most of the appointments. Though the second SLP was unable to continue, the SLPs were both involved in setting up the criteria for each category and had discussed how the first round evaluations were ranked to help with consistency in subsequent rounds.

Variation	Clear	Complete	Concise	Relevant	Organized	Accurate
Variation 1	2.6	2.9	2.9	2.7	3.3	2.4
Variation 2	2.8	2.1	3.1	1.8	4.2	1.7
Variation 3	2.5	1.5	3.4	1.3	3.8	1.1
Variation 4	2.1	1.6	4.2	1.4	4.2	1.2

Table 5.7: Round 2 mean scores for generated SOAP notes by variation. The highest mean score for each rubric category is in bold.

It was found that even with the increase in temperature and number of examples, the model frequently repeated information from the task example SOAP note (and to a lesser extent the historical SOAP notes), seemingly without regard to the current appointment. This is reflected in the ROUGE scores of the generated SOAP notes compared to the task example in Table 5.8, with the scores significantly higher when the task example was included in the input. When the task example was not provided, it still repeated information from the other example SOAP notes even when no related content was mentioned in the transcript. After review by the researcher, all 10 of the appointments were found to have blatant copying from example SOAP notes. This copying likely had different impacts on the note evaluation depending on how similar the example visit was. This was the main issue highlighted by the therapist, making it difficult to judge the model’s ability to find relevant information within the visit transcript and also confusing the impact of other input components. There was no clear evidence that the model was referencing the template or the SLP definitions, and limited evidence of content from parent handouts. The

only instance where there was notable contribution from a parent handout, the clinician had flagged it as being irrelevant for that SOAP note. In one case, the model also used phonemes in a nonsensical way, repeating verbatim phoneme content from the transcript out of context and with no interpretation. However, the therapist did note that there was less repetition between SOAP sections in this round (though this may have been due to the copying of previous SOAPs).

Score Type	No Task Example		Task Example	
	Mean	Standard Deviation	Mean	Standard Deviation
ROUGE-1	0.240	0.089	0.576	0.182
ROUGE-2	0.051	0.050	0.411	0.254
ROUGE-L	0.138	0.055	0.460	0.240
ROUGE-LSUM	0.236	0.090	0.564	0.189

Table 5.8: Round 2 mean and standard deviation of ROUGE scores for generated SOAP notes compared to task example SOAP note with and without task example provided in input.

An important note is that, though the model’s repeating of task example information often negatively affected evaluations of note content for the generated SOAP, in several occasions where the example came from a closely related visit for the same client, the inclusion of this stolen information improved the note content scores. These related appointments may be relevant enough to the specific session such that their SOAP notes overlap and can provide valuable context to the model that may not be directly included in the transcript, such as a client’s current goals and progress. Unfortunately, most progress notes are not sufficiently related, and such direct inclusion of their information is out of place.

5.2.3 Third Round

In the first phase of the third round, the goal was to explore methods of preventing the model from stealing example SOAP note content and improving note quality without the use of example SOAPs. The clinicians would perform another evaluation on the best variations from this exploration. It was hypothesized that the matched example transcripts and SOAPs were too similar to each other, and so rather than learning guidelines for forming a SOAP note from a transcription, the model would reuse the example output. To help address this, task examples were given from visits that were dissimilar from the target appointment and phrased explicitly as examples rather than a previous user interaction. Two coaching appointments were chosen to be the task examples for the others as their SOAP notes had both scored relatively highly. The two visits were paired with each other, and each of the remaining appointments was paired with the example most dissimilar to it (i.e. where the therapy practices were most distinct). For example, this meant pairing appointments that had significant speech practice with the appointment that mostly involved caregiver updates. Slight variations to the instructions before the template were tested, switching the ordering and wording. Within the prompt, more explicit directions

not to copy content from the task example were also provided, as well as to only report information in the sections it was relevant to. Clearer labeling of sections of the input was also added, with tags in the system prompt such as [Role] before role instructions were given and [Task] before providing task specific instructions. Within the task instructions, an explanation of the transcript structure was included (i.e. speaker tags before sections of dialogue) to aid in transcript interpretability. Parent handouts were also removed from the input for this round, as they had had no clear benefit. The temperature was raised to 0.7 to encourage more diverse model output and further curb repetition between sections.

It was found that even with a dissimilar task example and explicit instructions against copying, the model frequently repeated information from the task example SOAP note that did not match the current appointment transcript. This is reinforced by the ROUGE scores for the generated SOAP notes compared to the task example SOAP note in Table 5.9, with the scores again being significantly higher when the task example is provided. This also happened if example SOAP notes were provided rather than a full task example - if an example was included, it would copy. However, the outputs that used only the template and the SLP definitions as additional context seemed improved from the previous rounds, pulling information from the current transcript. The two most promising versions used the briefer template, differing from each other only slightly in the wording and order of instructions provided in the system prompt.

Score Type	No Task Example		Task Example	
	Mean	Standard Deviation	Mean	Standard Deviation
ROUGE-1	0.220	0.024	0.509	0.151
ROUGE-2	0.017	0.009	0.359	0.199
ROUGE-L	0.103	0.010	0.414	0.179
ROUGE-LSUM	0.214	0.026	0.503	0.155

Table 5.9: Round 3 mean and standard deviation of ROUGE scores for generated SOAP notes compared to task example SOAP note with and without task example provided in input.

In phase two of the third round, the 20 SOAP notes generated for these variations were sent to the clinicians for evaluation. Table 5.10 reports the mean scores for the generations from the two main input variations, showing modest improvements over the previous round for note content quality. The second template version seemed to be slightly higher than the first for most note content categories. The first template variation also achieved comparable scores to the second round in note structure despite having been given neither a task example or example historical SOAP notes.

Variation	Clear	Complete	Concise	Relevant	Organized	Accurate
Template 1	3.2	3.1	3.4	2.8	3.9	2.8
Template 2	2.7	3.1	3.2	3.1	3.7	3.0

Table 5.10: Round 3 mean scores for generated SOAP notes by variation. The highest mean score for each rubric category is in bold.

Regardless of input variation, performance also seemed to vary by appointment type, as shown in Table 5.11. The model seemed to perform best at the appointments that were most similar to a typical doctor patient visit. In this subset, those were the AAC sessions. The therapist described these appointments as more straightforward visits where the model was able to hit on all of the main points, while other appointments seemed to have nuances that the model was unable to pick up on. This was further reinforced after the therapist briefly reviewed the joint OT-SLP or PT-OT-SLP appointments from the earlier AutoScribe trial and found the generated SOAP notes to be typically more accurate than the AutoScribe-generated SLP SOAP notes (though it still struggled with nonverbal Observations). The model seemed to struggle the most with the block appointments, which all targeted articulation and phonology in this subset. This may be because these appointments typically involve direct speech practice, play-based therapy, and observations not directly recorded in the transcript.

Variation	Clear	Complete	Concise	Relevant	Organized	Accurate
Coaching - AAC	3.5	4.5	3.8	4.5	4.0	4.3
Coaching - Mixed	3.1	2.7	3.1	2.5	4.0	2.6
Block	2.3	2.8	3.3	2.7	3.3	2.5

Table 5.11: Round 3 mean scores for generated SOAP notes by appointment type. The highest mean score for each rubric category is in bold.

For many appointments, the model reported on object permanence or that the child was able to identify animals and objects. After review, this was not found to be present in the SLP-specific context, yet in many appointment transcripts the therapist does play games with the child that involve object identification. Though in some cases this information is technically supported by the transcript, it is typically not clinically relevant for an SLP SOAP note.

5.2.4 Fourth Round

The fourth round would explore new variations to improve note quality, this time with the researcher’s qualitative review rather than clinician ranking. Many of the remaining issues with the SOAP notes did not require clinical expertise to recognize (e.g. repeated information, incorrect activities). The researcher also relied on direct comparison with the matching AlayaCare SOAP note.

In the first phase, the manual transcript was tested along with the phoneme transcript to see the impact of an alternative child speech representation. A new context component tested this round was information specific to child speech that an SLP would be looking for based on the appointment category. The template was once again revised, with a new version combining both the SLP definitions and the AutoScribe version (rephrasing the AutoScribe template’s details as examples of information to include rather than prescriptively). The system and user prompt instructions were refined to emphasize being concise, to only report information found in the transcript, and to avoid general statements. Parent

handouts were also added back in for one variation to test whether they were able to add value to the generations. One final variation also added the task example again.

Once again, the variation with the task example copied inaccurate details not from the current visit. The model did not seem to be making use of the phonemes in the current format, and so for subsequent iterations only the manual transcript was used. For the variation with parent handouts, some generations seemed to have improved quality, though there were several instances of strategies mentioned from the handouts that were irrelevant for the current session. Based on these findings, the parent handouts were further revised to be more targeted for each session, keeping only those most closely related. This also helped to reduce context length, which would potentially allow the model to better focus on the rest of the input.

In the second phase, the manual transcript was tested with two versions of the template, one still based off of the SLP provided definitions and the other with the combined details of the AutoScribe template, with no example SOAPS. The revised SLP definitions, parent handouts, and child speech information were also included. The variations tested both together and separately were the inclusion of speech information and parent handouts.

After reviewing the previous outputs for some of the AAC appointments, further refinements were made to the parent handouts associated with them (i.e. removing language handouts for a nonverbal client). For the fluency sessions, the provided SLP definitions were given more information provided by KidsAbility’s SLPs. The variations tested were the two template versions and the inclusion or exclusion of parent handouts. Multiple SOAP notes were generated for each variation for better comparison.

The added fluency information was beneficial, with a generated SOAP note even including the fluency outcomes reporting tool mentioned in the transcript and corresponding AlayaCare SOAP note. After reviewing the generated outputs from these variations, the added speech information seemed most impactful for the motor speech appointments, which use speech information most directly. Speech practice was a significant component of the session, and the generated outputs were now more focused on the targeted sounds, with fewer irrelevant details.

In general, it seemed that more relevant details were captured in the fourth round’s SOAP notes, with refinements from each of the phases leading to small improvements. The model seemed better able to identify session activities in this round’s generations, correctly noting specific games or books. The more detailed template and inclusion of parent handouts led to the best results for the coaching appointments, while the briefer template without parent handouts seemed best for the block appointments. This may be because the selected set of block appointments were all related to motor speech, and the sessions centred around child speech practice rather than parent strategies.

The best generation for each of the 10 sample appointments was chosen to be included in a final evaluation by the SLP that had ranked the previous rounds’ outputs. Three slightly different generation methods were represented, with half of the appointments having the best results from the briefer template with no parent handouts (4 articulation and 1 fluency), and the other half benefiting from the more detailed template either with (2 AAC) or without (1 AAC and 2 articulation) parent handouts. Table 5.12 shows the mean

results for Round 4 compared to the highest scores in each category from Round 3 as well as the no context generations for the manual transcription inputs from Round 1. Round 2 results were not included, as the task example and example SOAP notes may have confused results.

Variation	Clear	Complete	Concise	Relevant	Organized	Accurate
Round 1 No Context	1.8	1.8	1.7	2.3	2.1	2.2
Round 3 Template 1	3.2	3.1	3.4	2.8	3.9	2.8
Round 3 Template 2	2.7	3.1	3.2	3.1	3.7	3.0
Round 4	3.5	2.6	3.9	2.9	3.7	2.8

Table 5.12: Round 4 scores for generated SOAP notes compared to rounds one and three. The highest mean score for each rubric category is in bold.

These results show a clear increase in note quality when discipline- and task-specific context is included, with scores increasing across all rubric categories, comparing row 1 (no context) to the other rows. Though Round 4 (row 4) only surpasses both Round 3 variations (rows 2 and 3) in clarity and conciseness, it does match or surpass the lower Round 3 variation score for each category excepting completeness.

Variation	Clear	Complete	Concise	Relevant	Organized	Accurate
Coaching - AAC	4.5	3.5	5.0	4.5	5.0	4.5
Coaching - Mixed	3.4	2.2	3.6	2.4	3.6	2.4
Block	3.0	2.7	3.7	2.7	3.0	2.3

Table 5.13: Round 4 mean scores for generated SOAP notes by appointment type. The highest mean score for each rubric category is in bold.

In Table 5.13, the mean scores for broad appointment categories are reported. Coaching appointments that only target AAC have the highest scores across all rubric categories. As previously hypothesized, this may be due to the more straightforward nature of these visits, which often involve more discussion between caregiver and therapist rather than play-based therapy. Coaching appointments that targeted other and mixed focus areas, such as language and fluency, scored similarly to the block appointments (targeting articulation and phonology). Notably, the block appointments scored higher for Complete and Relevant categories and only slightly lower for Accuracy, which is an improvement from previous rounds. This may be due to the inclusion of speech information that may have helped guide the model in what to look for and report on in that type of session.

5.2.5 LLM Reviewer

The fifth and final round involved having the LLM review and revise previously generated outputs. In the review phase, the model should look for missing or fabricated content in reference to the input transcript, as well as issues with formatting based on the provided SOAP note template. Based on these findings, the model should then revise the generated

SOAP note. Chat-GPT helped in refining the new system and user prompts to be optimized for this two-step task, with the final versions displayed in Figure 5.2. The LLM was given the role of an expert clinical documentation reviewer specializing in SOAP note evaluation for pediatric SLP. Step one was to analyze the SOAP note for accuracy and completeness, looking out for hallucinations, omissions, or issues with section content. Once it had conducted its review, step two was to revise the SOAP note based on the review findings. The model was also provided a template of SOAP structure to guide its review (both templates used in round 4 were tested).

System Prompt

You are a licensed pediatric speech-language pathology assistant and expert in clinical documentation review.
You specialize in evaluating SOAP notes for accuracy, completeness, and clarity based on session transcripts.
Use professional, clinical language appropriate for inclusion in a patient record. Be concise, factual, and avoid speculation.
Always distinguish between what was observed and what was inferred.
If you detect hallucinated or fabricated content, clearly label it. If information is missing, include it only if it is directly supported by the original transcript.
Your revised SOAP notes should be clean, structured, and clinically appropriate. Eliminate any unnecessary repetition across sections. Each part of the SOAP note should contain distinct, non-redundant information. For example, avoid restating the same goals or activities unless it serves a new purpose in the section (e.g., analyzing performance in the Assessment after listing tasks in the Objective).
Write in point-form. Do not write in paragraphs.
Use the following section-specific guidance to ensure each part of the SOAP note is appropriately structured and clinically relevant:
{SOAP Note Template}

User Prompt

Your task is to review and revise a SOAP note generated from an SLP rehabilitation appointment transcription.

Transcript of Pediatric SLP Appointment:
{Transcript}

Initial SOAP Note (Generated):
{Generated SOAP note}

Step 1: Critical Review

Analyze the SOAP note for accuracy and completeness. Identify:

1. Hallucinations — Any content in the SOAP note that is not supported by the transcript, including:

- Fabricated symptoms, behaviors, parent statements, or therapy goals
- Inferred diagnoses or plans not explicitly mentioned

2. Omissions — Any clinically relevant information from the transcript that was left out, such as:

- Key parental concerns or child behaviors
- Observations about articulation, language, or communication attempts
- Details about cues, prompts, supports, or outcomes

3. Section placement errors — Any content that belongs in a different SOAP section (e.g., subjective info in the Objective).

Present your review as bullet points under these three headings.

Step 2: Revised SOAP Note Based on your findings, re-write the SOAP note so that it:

- Reflects only what is supported by the transcript
- Includes relevant missing information
- Clearly separates Subjective, Objective, Assessment, and Plan
- Is appropriate for a pediatric speech-language therapy context
- Uses concise, clinical language that would be suitable for a real pediatric SLP patient record

Return your output in two parts:

1. Review Findings
2. Revised SOAP Note

Figure 5.2: Example of system and user prompts provided to the model for reviewing and revising generated SOAP notes.

The best round 4 outputs for each appointment were selected based on how closely their details matched with the AlayaCare SOAP notes, though most had issues recognizable to the researcher (repeated lines, hallucinated content, missing information, etc.).

From inspecting the outputs, the model’s review was often able to identify missing information and some hallucinations, however, it did not often make use of what it found in the review when rewriting the note. In no instance did it add information to the revised note not present in the original SOAP note. Even if the review mentioned that the note had missed information for a note category, it did not generally appear in the revised note. Content identified as hallucinations was not always removed, either. The model in general did have some success in making the SOAP notes more concise, removing unsupported statements and duplicate lines between sections. It also frequently removed lines that were copied from the template. However, it also occasionally removed relevant points as well while failing to remove vague statements that added little value to the SOAP note. There was no clear trend for the information the model would remove, leave in, or add, though positively it most often seemed to remove redundant information.

For each appointment, the final LLM-reviewed output of the best generated SOAP note was evaluated by an SLP. The mean results are compared to the results of the original round 4 generations in Table 5.14. The LLM-reviewed notes seem to result in an overall decrease

in rubric scores, suggesting that the current capabilities of the LLM for revising the SOAP note are not strong. This could be a combination of poor model instructions in addition to the model not understanding how to carry out the task. This is reflected in the output itself, as the model did not seem to make use of its review in revising the SOAP note. The review was occasionally able to correctly identify both hallucinations and missed topic areas.

Variation	Clear	Complete	Concise	Relevant	Organized	Accurate
Round 4	3.5	2.6	3.9	2.9	3.7	2.8
LLM Reviewer	2.9	2.2	3.7	2.8	3.7	2.7

Table 5.14: Fourth round scores for generated SOAP notes by variation and appointment. The highest mean score for each rubric category is in bold.

As the LLM frequently output the generated SOAP note in paragraph format rather than point form, it was theorized that perhaps this would impact the note structure ratings. The SLP evaluating the SOAP notes commented that if they were to change the rating based on paragraph vs. point form, it may have been in conciseness. However, while conducting the evaluations they noted that there were paragraph-form notes that they would have not changed the rating for, and so this cannot be the sole contributor to the decrease in note structure scores, as the biggest decrease was in clarity.

5.3 Discussion

SLP-specific context had a noticeable improvement on note quality across all six rubric categories. Defining SLP terminology and therapy strategies, explaining aspects of speech relevant to appointment types, and providing a template for SOAP note structure all seemed to have positive impacts. Adding information about what a therapist would look for within a child’s speech based on the appointment type was, as expected, most helpful for the appointments that most directly drew information from child speech (i.e. articulation). It seemed to have little effect on AAC appointments, where the focus was on communication technology usage.

SLP is a broad discipline, and it was vital that any details provided to the model were directly applicable to the session. For example, having a template that explained sections of a SOAP note with generic example information was beneficial for note structure, but when a prescriptive and detailed template was used that covered appointment type-specific content, the model would repeat content from the template that was irrelevant to the session. This is also demonstrated by the model repeating information irrelevant to the current session from example SOAP notes when provided as context. Providing only the relevant resources in the context was the only consistent method of preventing repetition of unrelated content during testing.

Chapter 6

Speaker Comparison

6.1 Overview

The goal of this comparison was to investigate ways of quantitatively analyzing child speech quality for speech therapy sessions. The desired output was a way to score child speech samples such that it could be passed in as an additional input to the model, providing deeper insight into a child’s performance during speech therapy sessions.

SLP appointments that focused on speech therapy, particularly articulation and phonology, were flagged by therapists as having particular difficulties with AI tools. For appointments that target specific sounds or speech areas, a SOAP note will often include details about how well the child was able to produce a target sound in different situations. If the model was able to understand more of the child’s speech production during practice sessions, the SOAP note may be able to draft more informed Observation and Analysis sections and include more detail relevant to these types of appointments. Determining measures of speech quality from session audio would be a step towards providing the model with this more nuanced information that may be relevant for SOAP notes.

Child speech samples were compared against correct reference speech samples, assessing how the cosine distance between their mel-spectrogram representations varied with speech quality score. These samples were also used as inputs to a phoneme ASR model, and the activations of the model layers for the speech samples were recorded. The cosine distances and layer activations were tested as predictive features of speech sample quality using a regression model, with layer activations achieving better results. These predicted speech quality scores were then used as additional context for SOAP note generation.

6.2 Speech Sample Preparation

One of the block appointments that focused on articulation and phonology was chosen as a sample to develop and test this speech comparison, as it included clear examples of speech practice involving the child, therapist, and caregiver. 30 sections of child and therapist

speech where the main target sound was practiced were isolated from the appointment audio file. Each child speech sample was assigned a scalar label by an SLP that reflected articulation accuracy. This scale was from 1 to 5, where [1: very poor, 2: poor, 3: fair, 4: good, 5: perfect].

For each of the 30 child speech samples, matching therapist samples were collected for the same target words. To provide a consistent reference for comparison, an additional recording was created for each word, produced by a researcher with typical adult articulation. In this way, a quantitative comparison could be made between typical and distorted articulation. To ensure acoustic consistency with the therapy session recordings, additive noise matching was applied to the researcher recordings. Background noise segments were extracted from moments of silence during the original therapy appointments and added to the researcher samples to match the SNR and better simulate the recording environment. The estimated SNR for the sample appointment was 13.06 dB, and so this was the target SNR for adjusting the reference speaker samples.

Following this, all audio samples (child, therapist, and researcher productions) were manually sliced to isolate the target phoneme being practiced in the session. One or two surrounding phonemes were retained when necessary to preserve natural transitions and consistency between speaker samples, while the remainder of each word was excluded to minimize interference from unrelated sounds. Removing these extra phonemes allowed for a more direct analysis of the child’s articulatory accuracy for the target speech sound.

6.3 Speech Quality Score Prediction

To measure how different a child’s articulation was from typical speech, the final audio segments of the target speech sounds were converted into mel-spectrograms, and DTW was used to align them temporally. Cosine distance was then calculated between the child’s articulation and both the therapist and researcher samples. These audio segments were given a score of 1 to 5 based on articulation quality: very poor, poor, fair, good, and perfect, based on previously assigned ratings. This enabled analysis of acoustic similarity as a function of articulation accuracy.

The audio segments were fed into the wav2vec2 phoneme transcription model, and the activations were recorded for each layer. A comparison of the average cosine distance between the network activations of child speech samples and reference speech samples for each layer is shown in Figure 6.1, grouped by child speech quality score.

To investigate whether embedding layer representations could be used to estimate phoneme production quality, a Ridge regression model was developed using cosine distance values derived from language model layer outputs. Cosine distances were computed between child phoneme articulations and reference articulations (therapist and researcher) across model layers 1 through 25. Visualizations of these distances showed that cosine distance decreased monotonically with increasing quality score only in the final two layers, though consistent separation between higher and lower productions were seen across earlier layers as well. Ridge regression was then applied to map embedding distances to these

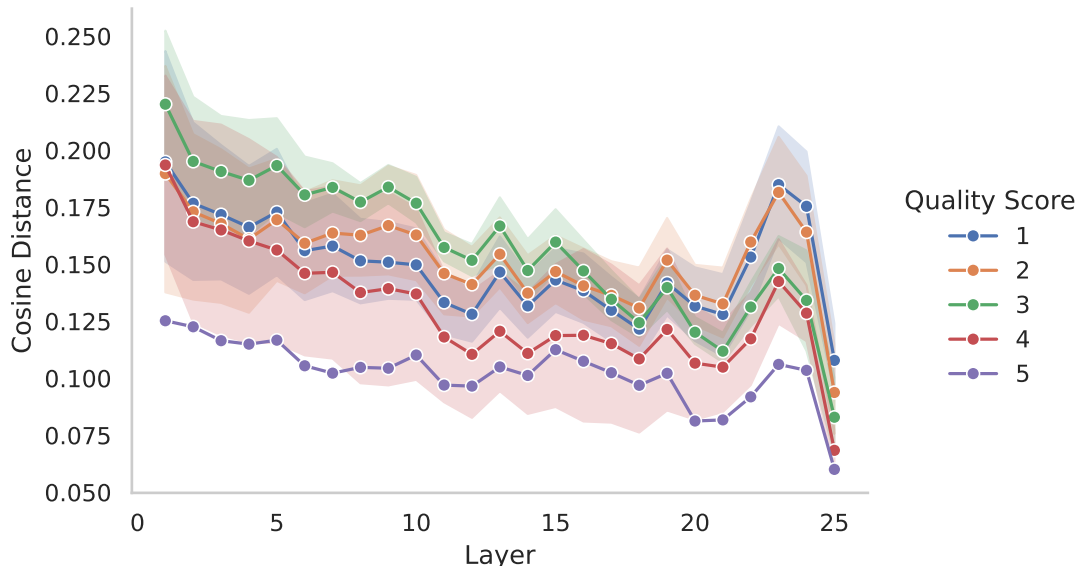


Figure 6.1: Average cosine distance between child and reference speech samples for layer activations by speech quality score.

scalar scores. Several configurations were evaluated using leave-one-out cross-validation (LOOCV). A baseline using the average cosine distance across all layers produced modest results ($\text{RMSE}=1.112$, $R^2=0.033$). Using all eight layers as input improved performance ($\text{RMSE}=0.345$, $R^2=-0.030$), with a slight further gain when averaging their outputs ($\text{RMSE}=0.332$, $R^2=0.044$). When assessing layers individually, layer 9 yielded the best predictive performance ($\text{RMSE}=0.756$, $R^2=0.881$). These results suggest that embedding representations from middle model layers, particularly those surrounding layer 9, contain information useful for quantifying articulation quality. Further work may explore whether using multiple layers or alternate modeling approaches can improve regression accuracy.

In LOOCV, the model was trained 30 times, each time leaving out one of the examples, and then predicting a score for that left-out example. Therefore, every data point gets a predicted score, but it was never used in the model that predicted it, giving a better estimate of generalization performance. This also left a prediction for each of the 30 samples that could then be tested as an additional input to the SOAP generation model.

Figure 6.2 plots the average cosine distance between mel-spectrograms of child and reference speakers for each sample word, organized by speech quality score. There is high variability across scores, particularly in the lowest score category, with no discernible correlation between cosine distance and speech quality.

The predictions of the model (trained using the layer 9 embeddings) for each word by each speaker had varied results. The reference speakers all performed fairly well, with the regression model predicting appropriately high scores, as seen in 6.3. However, the predictions made for the child speech samples (in the left panel) tended to be lower overall even if the actual speech sample had been of higher quality. Though apart from the lowest score category, the predictions do follow a trend of lower to higher predictions for lower to

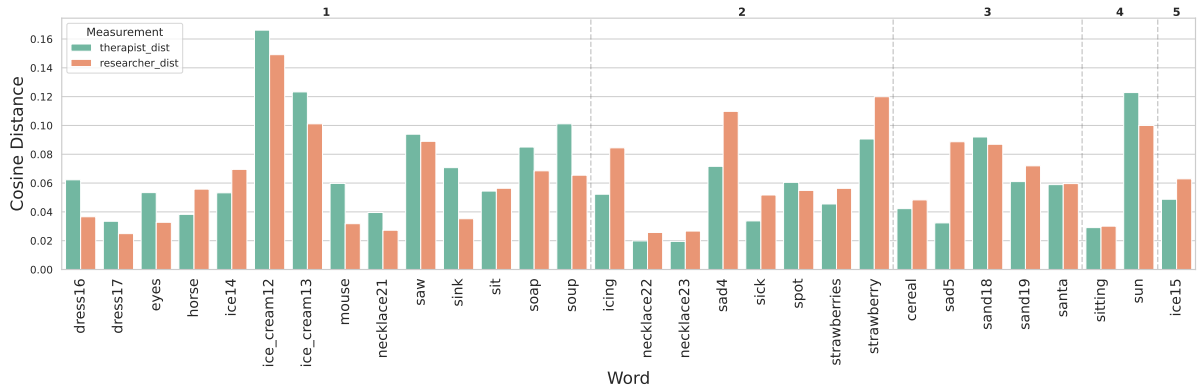


Figure 6.2: Average cosine distance between child and reference speech sample mel spectrograms by words and speech quality score.

higher quality samples. This could perhaps be improved with a larger sample size.

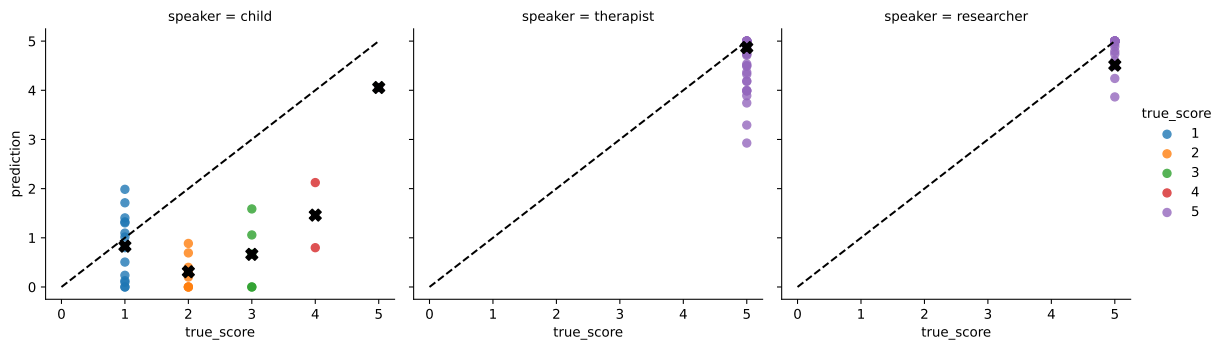


Figure 6.3: Predicted vs. true speech quality scores by speaker for regression model trained on layer 9 outputs. Mean score is denoted by 'x'.

6.4 Generation

The scores from the regression model were used as additional inputs to the LLM for drafting SOAP notes. Two versions for incorporating these scores into the user prompt were tested. In the first, the scores were averaged across the speech samples for the session, providing an overall session score. For the second version, the list of words and their corresponding scores were included, to provide more nuanced information for where the child did well and where further support may be needed.

These progress notes were evaluated by an SLP, and their results are reported in Table 6.1. The incorporation of speech quality information did not result in a clear improvement in note quality. From inspecting the notes, this added information did not seem to have a strong impact on the generated SOAP note contents. Though the notes did seem

to contain more correct (and fewer incorrect) inferences about child speech quality, the language was vague, failing to specify the child’s performance in the manner an SLP would typically report. Further testing with more examples and more comprehensive speech analysis could perhaps improve these results.

Variation	Clear	Complete	Concise	Relevant	Organized	Accurate
Round 4	5	3	5	3	3	3
Average Score	4	2	4	3	3	3
Set of Scores	4	2	3	2	3	2

Table 6.1: Scores for generated SOAP notes by speech quality score variation compared to Round 4 generation. The highest mean score for each rubric category is in bold.

Chapter 7

Conclusions

7.1 Summary

In this thesis, the implementation of AI documentation tools for pediatric SLP workflows was analyzed. First, issues with existing solutions were explored through interviews with the SLPs who had participated in KidsAbility’s AutoScribe pilot to better understand how the tool fit with their workflows and behaved during their appointments. The issues were categorized into those primarily affecting the transcription phase of the system and those affecting the documentation generation phase. Potential approaches to help mitigate these shortcomings were then proposed for each phase. For transcription, this included the addition of speaker labels for utterances, the phonetic transcription of child speech, and a more accurate and complete transcript (as the AutoScribe transcript missed significant detail) were suggested. For note generation, the model input would be tailored to include SLP-specific context, such as example SLP SOAP notes and terminology, to better inform the SOAP note.

10 sample SLP appointments that had struggled during the AutoScribe pilot were identified, and key pieces of data were gathered for each: AutoScribe audio file, transcript, generated note, and final AlayaCare note. Ground truth transcriptions were created for the appointments. Datasets of SLP-specific context were curated with the assistance of KidsAbility’s SLPs. A transcription pipeline based on WhisperX was tested, however, the results did not yield any notable improvements over the AutoScribe transcript, and so the note generation was tested with the AutoScribe and ground truth transcripts rather than those from the custom transcription pipeline. Several rounds of SLP SOAP notes were generated for the 10 appointments, with SLPs evaluating their quality. Adjustments were made to the model settings and input context based on the feedback from the clinicians.

As issues with the model failing to capture meaningful information for speech therapies had been raised by the clinicians, additional work was done to quantitatively analyze child speech quality for sounds targeted during therapy sessions such that this information could be conveyed to the model during SOAP note generation. This was done by isolating speech practice segments in one of the sample appointments and assigning labels of quality (based

on SLP review), generating the mel-spectrograms of the target sound for each segment, and extracting the activations of a phoneme recognition model for each input. These were mapped to a quality score with a regression model, and these scores were fed as input to the model when generating a SOAP note for the appointment. The notes generated with additional information seemed to make appropriate though general speech statements.

In general, the SOAP notes seemed to improve in quality when given appropriate context. The ability of the model to capture the relevant details of an appointment seemed to vary significantly for different types of appointments, with those that had more direct conversation and less play-based therapy or direct speech practice producing higher quality notes. This thesis has demonstrated that phoneme model activations and mel-spectrogram speaker distances show promise as ways of quantitatively assessing speech production. They also have potential to serve as an additional model input for improving reports on speech therapy sessions.

7.2 Limitations

One of the major limiting factors of this work was the availability of clinicians. Their expertise was invaluable for helping curate the SLP-specific context for model inputs as well as evaluating generated SOAP notes. However, their time did not allow for ranking of all versions of SOAP note outputs. This meant that the research team reviewed some of the output rounds to determine noticeable impacts of variation on note quality that did not rely on clinical expertise, such as repetition of lines or information stolen from example notes, or on direct comparison with the matching AlayaCare SOAP note.

The evaluation of SOAP notes can also be a subjective task, with the value placed on different rubric categories and what meets the criteria for different scores varying between clinicians. Due to this, ratings may not be representative of all SLPs for all categories.

A limitation of applying ASR-based documentation aids to this task is that they are completely reliant on verbally communicated information. This means that, in lieu of additional information sources, the model will always fail to capture unstated observations, which was a common complaint from SLPs.

Another limitation was the small sample size of appointments. This was largely constrained by the appointments carried out by SLPs participating in the pilot, though another contributing factor was the labour intensive work of adding new samples. The process to create ground truth transcriptions was tedious and time consuming, and significantly increasing the sample pool would have added considerable time to the overall project. However, having additional sample appointments could have allowed for more robust conclusions to be drawn from the results, particularly if there was better representation from each appointment type.

The quality of the audio files from these sample appointments was also a limitation that complicated both manual and ASR-based transcription. Lower quality and stationary microphones were used (typically built-in laptop microphones), with high amounts of background noise present and soft speaker voices.

7.3 Future Work

This partnership with KidsAbility to improve clinicians’ experiences with AI scribes will continue. Next steps will be to continue optimizing the SLP specific system to produce progress notes, building on the current prototype. There may be further iterations on instructions and hyperparameters for SOAP generation, along with further experiments in multi-step reasoning approaches. Post-processing or prompting to address systematic errors may also be explored, such as duplicate lines or removing lines repeated from the template. Ongoing work will also focus on teaching the LLM to accurately interpret child phonemes during SLP encounters, beginning with the development of more examples and instructions.

Additional work may be done to improve transcriptions and more concretely test the impact of transcription quality (e.g. missing details, importance of speaker diarization, pre-processing of audio file). Improving automated transcription quality could also lead to creation of a larger testing set and/or training set.

For SOAP notes generated with a task example, the model would steal information from the task example SOAP note. Though this often resulted in a decrease in note content ratings for the generated SOAP notes, in the few cases where the visits were very closely related (i.e. sequential visits for the same client), the impact of this was actually positive for evaluations of note content quality. These linked sessions may be able to provide highly relevant context to the model, such as previous goals a client has completed or is working towards that may not be restated in the transcript. KidsAbility has shown interest in linking an LLM to a client’s record, such that the model would have access to previous assessments and progress reports for a client. This would be valuable for understanding progress towards goals and providing a baseline of a client’s status. This work may also be expanded both to other therapy disciplines within KidsAbility as well as to generation of other forms of clinical documentation and reporting (e.g. emails, referrals).

References

- [1] Solomon Amenyo, Maura R. Grossman, Daniel G. Brown, and Brendan Wylie-Toal. Assessment of ai-generated pediatric rehabilitation soap-note quality, 2025.
- [2] Alexei Baevski, Henry Zhou, Abdelrahman Mohamed, and Michael Auli. wav2vec 2.0: A framework for self-supervised learning of speech representations, 2020.
- [3] Max Bain, Jaesung Huh, Tengda Han, and Andrew Zisserman. Whisperx: Time-accurate speech transcription of long-form audio, 2023.
- [4] Y. Bengio, P. Simard, and P. Frasconi. Learning long-term dependencies with gradient descent is difficult. *IEEE Transactions on Neural Networks*, 5(2):157–166, 1994.
- [5] Rishi Bommasani, Drew A. Hudson, Ehsan Adeli, Russ Altman, Simran Arora, Sydney von Arx, Michael S. Bernstein, Jeannette Bohg, Antoine Bosselut, Emma Brunskill, Erik Brynjolfsson, Shyamal Buch, Dallas Card, Rodrigo Castellon, Niladri Chatterji, Annie Chen, Kathleen Creel, Jared Quincy Davis, Dora Demszky, Chris Donahue, Moussa Doumbouya, Esin Durmus, Stefano Ermon, John Etchemendy, Kawin Ethayarajh, Li Fei-Fei, Chelsea Finn, Trevor Gale, Lauren Gillespie, Karan Goel, Noah Goodman, Shelby Grossman, Neel Guha, Tatsunori Hashimoto, Peter Henderson, John Hewitt, Daniel E. Ho, Jenny Hong, Kyle Hsu, Jing Huang, Thomas Icard, Saahil Jain, Dan Jurafsky, Pratyusha Kalluri, Siddharth Karamcheti, Geoff Keeling, Fereshte Khani, Omar Khattab, Pang Wei Koh, Mark Krass, Ranjay Krishna, Rohith Kudipudi, Ananya Kumar, Faisal Ladhak, Mina Lee, Tony Lee, Jure Leskovec, Isabelle Levent, Xiang Lisa Li, Xuechen Li, Tengyu Ma, Ali Malik, Christopher D. Manning, Suvir Mirchandani, Eric Mitchell, Zanele Munyikwa, Suraj Nair, Avanika Narayan, Deepak Narayanan, Ben Newman, Allen Nie, Juan Carlos Niebles, Hamed Nilforoshan, Julian Nyarko, Giray Ogut, Laurel Orr, Isabel Papadimitriou, Joon Sung Park, Chris Piech, Eva Portelance, Christopher Potts, Aditi Raghunathan, Rob Reich, Hongyu Ren, Frieda Rong, Yusuf Roohani, Camilo Ruiz, Jack Ryan, Christopher Ré, Dorsa Sadigh, Shiori Sagawa, Keshav Santhanam, Andy Shih, Krishnan Srinivasan, Alex Tamkin, Rohan Taori, Armin W. Thomas, Florian Tramèr, Rose E. Wang, William Wang, Bohan Wu, Jiajun Wu, Yuhuai Wu, Sang Michael Xie, Michihiro Yasunaga, Jiaxuan You, Matei Zaharia, Michael Zhang, Tianyi Zhang, Xikun Zhang, Yuhui Zhang, Lucia Zheng, Kaitlyn Zhou, and Percy Liang. On the opportunities and risks of foundation models, 2022.

- [6] Hervé Bredin. pyannote.audio 2.1 speaker diarization pipeline: principle, benchmark, and recipe. In *Interspeech 2023*, pages 1983–1987, 2023.
- [7] Hervé Bredin, Ruiqing Yin, Juan Manuel Coria, Gregory Gelly, Pavel Korshunov, Marvin Lavechin, Diego Fustes, Hadrien Titeux, Wassim Bouaziz, and Marie-Philippe Gill. pyannote.audio: neural building blocks for speaker diarization, 2019.
- [8] Tom B. Brown, Benjamin Mann, Nick Ryder, Melanie Subbiah, Jared Kaplan, Prafulla Dhariwal, Arvind Neelakantan, Pranav Shyam, Girish Sastry, Amanda Askell, Sandhini Agarwal, Ariel Herbert-Voss, Gretchen Krueger, Tom Henighan, Rewon Child, Aditya Ramesh, Daniel M. Ziegler, Jeffrey Wu, Clemens Winter, Christopher Hesse, Mark Chen, Eric Sigler, Mateusz Litwin, Scott Gray, Benjamin Chess, Jack Clark, Christopher Berner, Sam McCandlish, Alec Radford, Ilya Sutskever, and Dario Amodei. Language models are few-shot learners, 2020.
- [9] Wenhua Cheng, Weiwei Zhang, Haihao Shen, Yiyang Cai, Xin He, Kaokao Lv, and Yi Liu. Optimize weight rounding via signed gradient descent for the quantization of llms, 2024.
- [10] Children’s Health Coalition. Make kids count action plan. Technical report, Children’s Health Coalition, Canada, October 2022. Accessed: 2025-08-11.
- [11] Tim Dettmers, Artidoro Pagnoni, Ari Holtzman, and Luke Zettlemoyer. Qlora: Efficient finetuning of quantized llms, 2023.
- [12] Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. Bert: Pre-training of deep bidirectional transformers for language understanding, 2019.
- [13] Rachel DiMaio. Creation of a custom language model for pediatric occupational therapy documentation. Master’s thesis, University of Waterloo, Waterloo, Ontario, Canada, 2024. Presented in fulfillment of the thesis requirement for the degree of Master of Applied Science.
- [14] Chenhe Dong, Yinghui Li, Haifan Gong, Miaoxin Chen, Junxin Li, Ying Shen, and Min Yang. A survey of natural language generation. *ACM Computing Surveys*, 55(8):1–38, December 2022.
- [15] Weihua Du, Yiming Yang, and Sean Welleck. Optimizing temperature for language models with multi-sample inference, 2025.
- [16] Emily B. Fox, Erik B. Sudderth, Michael I. Jordan, and Alan S. Willsky. A sticky hdp-hmm with application to speaker diarization. *The Annals of Applied Statistics*, 5(2A), June 2011.
- [17] Elias Frantar, Saleh Ashkboos, Torsten Hoefer, and Dan Alistarh. Gptq: Accurate post-training quantization for generative pre-trained transformers, 2023.

- [18] Government of Ontario. Preschool speech and language and children’s rehabilitation services guidelines: Program elements. Accessed: 2025-08-11, April 2024. Guidelines document published online.
- [19] Aaron Grattaffiori, Abhimanyu Dubey, Abhinav Jauhri, Abhinav Pandey, Abhishek Kadian, Ahmad Al-Dahle, Aiesha Letman, Akhil Mathur, Alan Schelten, Alex Vaughan, Amy Yang, Angela Fan, Anirudh Goyal, Anthony Hartshorn, Aobo Yang, Archi Mitra, Archie Sravankumar, Artem Korenev, Arthur Hinsvark, Arun Rao, Aston Zhang, Aurelien Rodriguez, Austen Gregerson, Ava Spataru, Baptiste Roziere, Bethany Biron, Binh Tang, Bobbie Chern, Charlotte Caucheteux, Chaya Nayak, Chloe Bi, Chris Marra, Chris McConnell, Christian Keller, Christophe Touret, Chunyang Wu, Corinne Wong, Cristian Canton Ferrer, Cyrus Nikolaidis, Damien Allonsius, Daniel Song, Danielle Pintz, Danny Livshits, Danny Wyatt, David Esiobu, Dhruv Choudhary, Dhruv Mahajan, Diego Garcia-Olano, Diego Perino, Dieuwke Hupkes, Egor Lakomkin, Ehab AlBadawy, Elina Lobanova, Emily Dinan, Eric Michael Smith, Filip Radenovic, Francisco Guzmán, Frank Zhang, Gabriel Synnaeve, Gabrielle Lee, Georgia Lewis Anderson, Govind Thattai, Graeme Nail, Gregoire Mialon, Guan Pang, Guillem Cucurell, Hailey Nguyen, Hannah Korevaar, Hu Xu, Hugo Touvron, Iliyan Zarov, Imanol Arrieta Ibarra, Isabel Kloumann, Ishan Misra, Ivan Evtimov, Jack Zhang, Jade Copet, Jaewon Lee, Jan Geffert, Jana Vranes, Jason Park, Jay Mahadeokar, Jeet Shah, Jelmer van der Linde, Jennifer Billock, Jenny Hong, Jenya Lee, Jeremy Fu, Jianfeng Chi, Jianyu Huang, Jiawen Liu, Jie Wang, Jiecao Yu, Joanna Bitton, Joe Spisak, Jongsoo Park, Joseph Rocca, Joshua Johnstun, Joshua Saxe, Junteng Jia, Kalyan Vasuden Alwala, Karthik Prasad, Kartikeya Upasani, Kate Plawiak, Ke Li, Kenneth Heafield, Kevin Stone, Khalid El-Arini, Krithika Iyer, Kshitiz Malik, Kuenley Chiu, Kunal Bhalla, Kushal Lakhotia, Lauren Rantala-Yeary, Laurens van der Maaten, Lawrence Chen, Liang Tan, Liz Jenkins, Louis Martin, Lovish Madaan, Lubo Malo, Lukas Blecher, Lukas Landzaat, Luke de Oliveira, Madeline Muzzi, Mahesh Pasupuleti, Mannat Singh, Manohar Paluri, Marcin Kardas, Maria Tsimpoukelli, Mathew Oldham, Mathieu Rita, Maya Pavlova, Melanie Kambadur, Mike Lewis, Min Si, Mitesh Kumar Singh, Mona Hassan, Naman Goyal, Narjes Torabi, Nikolay Bashlykov, Nikolay Bogoychev, Niladri Chatterji, Ning Zhang, Olivier Duchenne, Onur Çelebi, Patrick Alrassy, Pengchuan Zhang, Pengwei Li, Petar Vasic, Peter Weng, Prajjwal Bhargava, Pratik Dubal, Praveen Krishnan, Punit Singh Koura, Puxin Xu, Qing He, Qingxiao Dong, Ragavan Srinivasan, Raj Ganapathy, Ramon Calderer, Ricardo Silveira Cabral, Robert Stojnic, Roberta Raileanu, Rohan Maheswari, Rohit Girdhar, Rohit Patel, Romain Sauvestre, Ronnie Polidoro, Roshan Sumbaly, Ross Taylor, Ruan Silva, Rui Hou, Rui Wang, Saghar Hosseini, Sahana Chennabasappa, Sanjay Singh, Sean Bell, Seohyun Sonia Kim, Sergey Edunov, Shao-liang Nie, Sharan Narang, Sharath Raparthy, Sheng Shen, Shengye Wan, Shruti Bhosale, Shun Zhang, Simon Vandenhende, Soumya Batra, Spencer Whitman, Sten Sootla, Stephane Collot, Suchin Gururangan, Sydney Borodinsky, Tamar Herman, Tara Fowler, Tarek Sheasha, Thomas Georgiou, Thomas Scialom, Tobias Speckbacher, Todor Mihaylov, Tong Xiao, Ujjwal Karn, Vedanuj Goswami, Vibhor Gupta, Vignesh Ramanathan, Viktor Kerkez, Vincent Gonguet, Virginie Do, Vish Vogeti, Vitor Al-

biero, Vladan Petrovic, Weiwei Chu, Wenhan Xiong, Wenyin Fu, Whitney Meers, Xavier Martinet, Xiaodong Wang, Xiaofang Wang, Xiaoqing Ellen Tan, Xide Xia, Xinfeng Xie, Xuchao Jia, Xuwei Wang, Yaelle Goldschlag, Yashesh Gaur, Yasmine Babaei, Yi Wen, Yiwen Song, Yuchen Zhang, Yue Li, Yuning Mao, Zacharie Delpierre Coudert, Zheng Yan, Zhengxing Chen, Zoe Papakipos, Aaditya Singh, Aayushi Srivastava, Abha Jain, Adam Kelsey, Adam Shajnfeld, Adithya Gangidi, Adolfo Victoria, Ahuva Goldstand, Ajay Menon, Ajay Sharma, Alex Boesenberg, Alexei Baevski, Allie Feinstein, Amanda Kallet, Amit Sangani, Amos Teo, Anam Yunus, Andrei Lupu, Andres Alvarado, Andrew Caples, Andrew Gu, Andrew Ho, Andrew Poulton, Andrew Ryan, Ankit Ramchandani, Annie Dong, Annie Franco, Anuj Goyal, Aparajita Saraf, Arkabandhu Chowdhury, Ashley Gabriel, Ashwin Bharambe, Assaf Eisenman, Azadeh Yazdan, Beau James, Ben Maurer, Benjamin Leonhardi, Bernie Huang, Beth Loyd, Beto De Paola, Bhargavi Paranjape, Bing Liu, Bo Wu, Boyu Ni, Braden Hancock, Bram Wasti, Brandon Spence, Brani Stojkovic, Brian Gamido, Britt Montalvo, Carl Parker, Carly Burton, Catalina Mejia, Ce Liu, Changan Wang, Changkyu Kim, Chao Zhou, Chester Hu, Ching-Hsiang Chu, Chris Cai, Chris Tindal, Christoph Feichtenhofer, Cynthia Gao, Damon Civin, Dana Beaty, Daniel Kreymer, Daniel Li, David Adkins, David Xu, Davide Testuggine, Delia David, Devi Parikh, Diana Liskovich, Didem Foss, Dingkan Wang, Duc Le, Dustin Holland, Edward Dowling, Eissa Jamil, Elaine Montgomery, Eleonora Presani, Emily Hahn, Emily Wood, Eric-Tuan Le, Erik Brinkman, Esteban Arcaute, Evan Dunbar, Evan Smothers, Fei Sun, Felix Kreuk, Feng Tian, Filippos Kokkinos, Firat Ozgenel, Francesco Caggioni, Frank Kanayet, Frank Seide, Gabriela Medina Florez, Gabriella Schwarz, Gada Badeer, Georgia Swee, Gil Halpern, Grant Herman, Grigory Sizov, Guangyi, Zhang, Guna Lakshminarayanan, Hakan Inan, Hamid Shojanazeri, Han Zou, Hannah Wang, Hanwen Zha, Haroun Habeeb, Harrison Rudolph, Helen Suk, Henry Aspegren, Hunter Goldman, Hongyuan Zhan, Ibrahim Damla, Igor Molybog, Igor Tufanov, Ilias Leontiadis, Irina-Elena Veliche, Itai Gat, Jake Weissman, James Geboski, James Kohli, Janice Lam, Japhet Asher, Jean-Baptiste Gaya, Jeff Marcus, Jeff Tang, Jennifer Chan, Jenny Zhen, Jeremy Reizenstein, Jeremy Teboul, Jessica Zhong, Jian Jin, Jingyi Yang, Joe Cummings, Jon Carvill, Jon Shepard, Jonathan McPhie, Jonathan Torres, Josh Ginsburg, Junjie Wang, Kai Wu, Kam Hou U, Karan Saxena, Kartikay Khandelwal, Katayoun Zand, Kathy Matosich, Kaushik Veeraraghavan, Kelly Michelen, Keqian Li, Kiran Jagadeesh, Kun Huang, Kunal Chawla, Kyle Huang, Lailin Chen, Lakshya Garg, Lavender A, Leandro Silva, Lee Bell, Lei Zhang, Liangpeng Guo, Licheng Yu, Liron Moshkovich, Luca Wehrstedt, Madian Khabsa, Manav Avalani, Manish Bhatt, Martynas Mankus, Matan Hasson, Matthew Lennie, Matthias Reso, Maxim Groshchev, Maxim Naumov, Maya Lathi, Meghan Keneally, Miao Liu, Michael L. Seltzer, Michal Valko, Michelle Restrepo, Mihir Patel, Mik Vyatskov, Mikayel Samvelyan, Mike Clark, Mike Macey, Mike Wang, Miquel Jubert Hermoso, Mo Metanat, Mohammad Rastegari, Munish Bansal, Nandhini Santhanam, Natascha Parks, Natasha White, Navyata Bawa, Nayan Singhal, Nick Egebo, Nicolas Usunier, Nikhil Mehta, Nikolay Pavlovich Laptev, Ning Dong, Norman Cheng, Oleg Chernoguz, Olivia Hart, Omkar Salpekar, Ozlem Kalinli, Parkin Kent, Parth Parekh, Paul Saab, Pavan Balaji, Pedro Rittner, Philip Bontrager, Pierre Roux, Piotr Dollar, Polina Zvyagina, Prashant

Ratanchandani, Pritish Yuvraj, Qian Liang, Rachad Alao, Rachel Rodriguez, Rafi Ayub, Raghotham Murthy, Raghu Nayani, Rahul Mitra, Rangaprabhu Parthasarathy, Raymond Li, Rebekkah Hogan, Robin Battey, Rocky Wang, Russ Howes, Ruty Rinott, Sachin Mehta, Sachin Siby, Sai Jayesh Bondu, Samyak Datta, Sara Chugh, Sara Hunt, Sargun Dhillon, Sasha Sidorov, Satadru Pan, Saurabh Mahajan, Saurabh Verma, Seiji Yamamoto, Sharadh Ramaswamy, Shaun Lindsay, Sheng Feng, Shenghao Lin, Shengxin Cindy Zha, Shishir Patil, Shiva Shankar, Shuqiang Zhang, Shuqiang Zhang, Sinong Wang, Sneha Agarwal, Soji Sajuyigbe, Soumith Chintala, Stephanie Max, Stephen Chen, Steve Kehoe, Steve Satterfield, Sudarshan Govindaprasad, Sumit Gupta, Summer Deng, Sungmin Cho, Sunny Virk, Suraj Subramanian, Sy Choudhury, Sydney Goldman, Tal Remez, Tamar Glaser, Tamara Best, Thilo Koehler, Thomas Robinson, Tianhe Li, Tianjun Zhang, Tim Matthews, Timothy Chou, Tzook Shaked, Varun Vontimitta, Victoria Ajayi, Victoria Montanez, Vijai Mohan, Vinay Satish Kumar, Vishal Mangla, Vlad Ionescu, Vlad Poenaru, Vlad Tiberiu Mihailescu, Vladimir Ivanov, Wei Li, Wenchen Wang, Wenwen Jiang, Wes Bouaziz, Will Constable, Xiaocheng Tang, Xiaojian Wu, Xiaolan Wang, Xilun Wu, Xinbo Gao, Yaniv Kleinman, Yanjun Chen, Ye Hu, Ye Jia, Ye Qi, Yenda Li, Yilin Zhang, Ying Zhang, Yossi Adi, Youngjin Nam, Yu, Wang, Yu Zhao, Yuchen Hao, Yundi Qian, Yunlu Li, Yuzi He, Zach Rait, Zachary DeVito, Zef Rosnbrick, Zhaoduo Wen, Zhenyu Yang, Zhiwei Zhao, and Zhiyu Ma. The llama 3 herd of models, 2024.

- [20] A. Graves and J. Schmidhuber. Framewise phoneme classification with bidirectional lstm networks. In *Proceedings. 2005 IEEE International Joint Conference on Neural Networks, 2005.*, volume 4, pages 2047–2052 vol. 4, 2005.
- [21] Moto Hira. Forced alignment with wav2vec2. TorchAudio Documentation, 2025. TorchAudio tutorial documentation; includes CTC forced alignment API guidance.
- [22] Sepp Hochreiter and Jürgen Schmidhuber. Long short-term memory. *Neural Computation*, 9(8):1735–1780, 1997.
- [23] Hugging Face. Text generation inference. Hugging Face Documentation, 2025. Accessed: 2025-08-11.
- [24] Qiao Jin, Won Kim, Qingyu Chen, Donald C Comeau, Lana Yeganova, W John Wilbur, and Zhiyong Lu. Medcpt: Contrastive pre-trained transformers with large-scale pubmed search logs for zero-shot biomedical information retrieval. *Bioinformatics*, 39(11), November 2023.
- [25] Rie Johnson and Tong Zhang. Deep pyramid convolutional neural networks for text categorization. In Regina Barzilay and Min-Yen Kan, editors, *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 562–570, Vancouver, Canada, July 2017. Association for Computational Linguistics.
- [26] Rafal Jozefowicz, Oriol Vinyals, Mike Schuster, Noam Shazeer, and Yonghui Wu. Exploring the limits of language modeling, 2016.

- [27] Kyung Hee Jung. Large language models in medicine: Clinical applications, technical challenges, and ethical considerations. *Healthcare Informatics Research*, 31(2):114–124, April 2025. Epub 2025 Apr 30.
- [28] Bongsu Kang, Jundong Kim, Tae-Rim Yun, and Chang-Eop Kim. Prompt-rag: Pioneering vector embedding-free retrieval-augmented generation in niche domains, exemplified by korean medicine, 2024.
- [29] Shigeki Karita, Nanxin Chen, Tomoki Hayashi, Takaaki Hori, Hirofumi Inaguma, Ziyang Jiang, Masao Someki, Nelson Enrique Yalta Soplin, Ryuichi Yamamoto, Xiaofei Wang, Shinji Watanabe, Takenori Yoshimura, and Wangyou Zhang. A comparative study on transformer vs rnn in speech applications. In *2019 IEEE Automatic Speech Recognition and Understanding Workshop (ASRU)*, page 449–456. IEEE, December 2019.
- [30] Vladimir Karpukhin, Barlas Ögüz, Sewon Min, Patrick Lewis, Ledell Wu, Sergey Edunov, Danqi Chen, and Wen tau Yih. Dense passage retrieval for open-domain question answering, 2020.
- [31] Omar Khattab and Matei Zaharia. Colbert: Efficient and effective passage search via contextualized late interaction over bert, 2020.
- [32] KidsAbility Centre for Child Development. Speech language therapy – early years. Accessed: 2025-08-11, 2024. Program description page.
- [33] KidsAbility Centre for Child Development. Slp definitions. Provided by KidsAbility Centre for Child Development, 2025. Internal document, not publicly available.
- [34] Alessandro Lenci. Understanding natural language understanding systems. a critical analysis, 2023.
- [35] Patrick Lewis, Ethan Perez, Aleksandra Piktus, Fabio Petroni, Vladimir Karpukhin, Naman Goyal, Heinrich Küttler, Mike Lewis, Wen tau Yih, Tim Rocktäschel, Sebastian Riedel, and Douwe Kiela. Retrieval-augmented generation for knowledge-intensive nlp tasks, 2021.
- [36] Lidcombe Program Trainers Consortium. About the lidcombe program. Accessed: 2025-08-11, 2025. Official website describing the Lidcombe Program for early stuttering intervention.
- [37] Mahrukh, Shumaila Malik, Hafsa Khan, Rizwana Khalil, and Aasma Akram. Level of stress among speech and language pathologists working in clinical setups. *Journal of Health and Rehabilitation Research*, 3(2):30–36, Nov. 2023.
- [38] Michael McAuliffe, Michaela Socolof, Sarah Mihuc, Michael Wagner, and Morgan Sonderegger. Montreal forced aligner: Trainable text-speech alignment using kaldi. In *Interspeech 2017*, pages 498–502, 2017.

- [39] Xiaoxuan Meng, Xiaoyu Yan, Kun Zhang, Dan Liu, Xiaohui Cui, Ying Yang, Ming Zhang, Cheng Cao, Jun Wang, Xiaoxiao Wang, Jing Gao, Yiguang Wang, Jianming Ji, Zhiyong Qiu, Min Li, Chun Qian, Ting Guo, Shuang Ma, Zhong Wang, Zhen Guo, Ying Lei, Chen Shao, Wei Wang, Hu Fan, and Yudong Tang. The application of large language models in medicine: A scoping review. *iScience*, 27(5):109713, April 2024.
- [40] Meta AI. Introducing meta llama 3. Meta AI Blog, April 2024. Accessed: 2025-08-11.
- [41] Meta AI. Llama 3.3 model card. GitHub repository, 2024. Accessed: 2025-08-11.
- [42] Meta AI. Meta llama 3 model card and prompt formats. Meta AI Documentation, 2025. Accessed: 2025-08-11.
- [43] Michael Palin Centre for Stammering. Assessment and therapy at the mpc. Accessed: 2025-08-11, 2025. Official website describing the Palin PCI approach for childhood stammering.
- [44] Mutuo Health Solutions. Putting ai to work for clinicians, 2025. Web page; information current as of August 11, 2025.
- [45] Meinard Müller. Dynamic time warping. *Information Retrieval for Music and Motion*, 2:69–84, 01 2007.
- [46] Abdurrahman Odabaşı and Göksel Biricik. Unraveling the capabilities of language models in news summarization, 2025.
- [47] OpenAI. Introducing gpt-4.1 in the api. OpenAI Blog, April 2025. Accessed: 2025-08-11.
- [48] Long Ouyang, Jeff Wu, Xu Jiang, Diogo Almeida, Carroll L. Wainwright, Pamela Mishkin, Chong Zhang, Sandhini Agarwal, Katarina Slama, Alex Ray, John Schulman, Jacob Hilton, Fraser Kelton, Luke Miller, Maddie Simens, Amanda Askell, Peter Welinder, Paul Christiano, Jan Leike, and Ryan Lowe. Training language models to follow instructions with human feedback, 2022.
- [49] Vivek Podder, Valerie Lew, and Sassan Ghassemzadeh. Soap notes. Accessed: 2025-08-11, August 2025. StatPearls Publishing; updated August 28, 2023.
- [50] Alec Radford, Jong Wook Kim, Tao Xu, Greg Brockman, Christine McLeavey, and Ilya Sutskever. Robust speech recognition via large-scale weak supervision, 2022.
- [51] Alec Radford and Karthik Narasimhan. Improving language understanding by generative pre-training. 2018.
- [52] Nils Reimers and Iryna Gurevych. Sentence-bert: Sentence embeddings using siamese bert-networks, 2019.
- [53] Stephen Robertson and Hugo Zaragoza. *The Probabilistic Relevance Framework*. Now Foundations and Trends, 2009.

- [54] Silero Team. Silero vad: pre-trained enterprise-grade voice activity detector (vad). GitHub repository, 2024. Accessed: 2025-08-11.
- [55] Victoria Smith, Ali Shahin Shamsabadi, Carolyn Ashurst, and Adrian Weller. Identifying and mitigating privacy risks stemming from language models: A survey, 2024.
- [56] SYSTRAN. Faster whisper transcription with ctranslate2. GitHub, 2025. GitHub repository; up to $4\times$ faster transcription with lower resource use :contentReference[oaicite:0]index=0. Accessed: 2025-08-11.
- [57] The Hanen Centre. The hanen centre. Accessed: 2025-08-11, 2025. Official website describing Hanen programs for speech and language development.
- [58] Daphne van Zandvoort, Laura Wiersema, Tom Huibers, Sandra van Dulmen, and Sjaak Brinkkemper. Enhancing summarization performance through transformer-based prompt engineering in automated medical reporting, 2024.
- [59] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N. Gomez, Lukasz Kaiser, and Illia Polosukhin. Attention is all you need, 2023.
- [60] Vitouphy. vitouphy/wav2vec2-xls-r-300m-timit-phoneme. <https://huggingface.co/vitouphy/wav2vec2-xls-r-300m-timit-phoneme>, 2022. Accessed: 2025-08-20.
- [61] Jules White, Quchen Fu, Sam Hays, Michael Sandborn, Carlos Olea, Henry Gilbert, Ashraf Elnashar, Jesse Spencer-Smith, and Douglas C. Schmidt. A prompt pattern catalog to enhance prompt engineering with chatgpt, 2023.
- [62] Guangzhi Xiong, Qiao Jin, Zhiyong Lu, and Aidong Zhang. Benchmarking retrieval-augmented generation for medicine. In Lun-Wei Ku, Andre Martins, and Vivek Sriku-mar, editors, *Findings of the Association for Computational Linguistics: ACL 2024*, pages 6233–6251, Bangkok, Thailand, August 2024. Association for Computational Linguistics.
- [63] E. Yu, X. Chu, W. Zhang, X. Meng, Y. Yang, X. Ji, and C. Wu. Large language models in medicine: Applications, challenges, and future directions. *International Journal of Medical Sciences*, 22(11):2792–2801, May 2025.