Locating and interpreting factual association in Speech Language Models

Extending Mechanistic Interpretability to the Speech Modality: A Causal Study of Factual Recall in the Spirit LM model

Master's thesis in Computer science and engineering

LUCA MODICA

FILIP LANDIN

Department of Computer Science and Engineering
CHALMERS UNIVERSITY OF TECHNOLOGY
UNIVERSITY OF GOTHENBURG
Gothenburg, Sweden 2025

Master's thesis 2025

Locating and interpreting factual association in Speech Language Models

Extending Mechanistic Interpretability to the Speech Modality: A Causal Study of Factual Recall in the Spirit LM model

LUCA MODICA
FILIP LANDIN

UNIVERSITY OF
GOTHENBURG

CHALMERS
UNIVERSITY OF TECHNOLOGY

Department of Computer Science and Engineering
Chalmers University of Technology
University of Gothenburg
Gothenburg, Sweden 2025

Locating and interpreting factual association in Speech Language Models
Extending Mechanistic Interpretability to the Speech Modality: A Causal Study of Factual Recall in the Spirit LM model

Supervisor: Mehrdad Farahani, Department of Computer Science and Engineering
Examiner: Richard Johansson, Department of Computer Science and Engineering

Master's Thesis 2025
Department of Computer Science and Engineering
Chalmers University of Technology and University of Gothenburg
SE-412 96 Gothenburg
Telephone +46 31 772 1000

Cover: A sample waveform with rectangles and triangles connecting each other, symbolizing the joint speech features in encoding and recalling knowledge (the image has been generated with AI).

Typeset in LaTeX
Gothenburg, Sweden 2025

Locating and interpreting factual association in Speech Language Models
Extending Mechanistic Interpretability to the Speech Modality: A Causal Study of Factual Recall in the Spirit LM model
LUCA MODICA
FILIP LANDIN
Department of Computer Science and Engineering
Chalmers University of Technology and University of Gothenburg

Abstract

Recent advances have enabled Speech Language Models (SLMs) to both understand and generate text and speech by representing audio as discrete tokens learned from raw waveforms without supervision. As these multimodal systems become increasingly common in real-world applications, it is crucial to understand how they encode and retrieve factual knowledge, insights that are key to improve their factual accuracy and reliability. While previous research has explored these mechanisms in traditional Large Language Models (LLMs) by observing their responses to targeted prompts (such as "The capital of Italy is _____"), much less is known about how these processes work in multimodal models such as SLMs, particularly regarding interactions between different modalities in cross-modal scenarios (e.g., speech-to-text).

This thesis aims to explore how Speech Language Models store and recall factual associations by applying Causal Mediation Analysis (CMA), a method inspired by causal inference used to quantify the contribution of model components to factual predictions. We introduce *MultimodalCausalTracer*, an adaptation of CMA to also handle discrete speech tokens. We use a CTC-based forced alignment algorithm to locate targeted words in a spoken utterance, map discrete speech tokens to text equivalents, and visualize CMA results across speech and text modalities.

We applied MultimodalCausalTracer to the Spirit LM model using a new speech-based version of the Known dataset which we constructed, covering spoken factual prompts about countries, famous people, and places. The results, measured in terms of Average Indirect Effect (AIE) of the model's components, show evident discrepancies between text-to-text and speech-to-text tasks, suggesting that the emergent mechanisms for factual recall are only partially carried over from the text to the speech modality. Our findings highlight key areas for future work, including extending experiments to other cross-modal scenarios and investigating factual recall in different SLMs and factual datasets.

Keywords: Machine Learning, Deep Learning, Causal Inference, Speech Language Models, Discrete Speech Tokens, Mechanistic Interpretability, Multimodal Learning, Model Analysis.

Acknowledgements

First and foremost, we want to express our sincere gratitude to our supervisor, Mehrdad Farahani. We are thankful not only for his meticulous guidance throughout our thesis work but also for introducing us to the fascinating research field of Mechanistic Interpretability and tools such as Causal Mediation Analysis. His support proved invaluable, from helping us define our research direction, to providing constant assistance during experiments in this novel and challenging area, and finally helping us submit our work as a paper for publication. We are truly grateful for his dedication and professionalism throughout the thesis work, and we hope to continue learning from him in our future professional and academic careers.

Secondly, we are also grateful to our examiner, Richard Johansson, for proposing the main idea behind our thesis work and for providing us with insightful feedback at key milestones: the thesis proposal, planning report, and half-time report. We would also like to sincerely thank our opponent, Zhixing Li, for thoughtful and constructive feedback, offered from a fellow students perspective. His careful reading and insightful suggestions helped us refine the content of our thesis and improve its overall clarity and coherence.

Finally, we want to thank the C3SE division at Chalmers University for supporting our work through access to computational resources, including the Alvis and Vera GPU clusters. We acknowledge the National Academic Infrastructure for Supercomputing in Sweden (NAISS) for enabling us to conduct Mechanistic Interpretability research in speech language models.

Luca Modica, Filip Landin, Gothenburg, 2025-06-15

Contents

List of Figures **xi**

List of Tables **xiii**

1 Introduction **1**
1.1 Research topics and motivation 2
1.2 Goals and challenges 2
1.3 Limitations and risks 3
1.4 Thesis outline 4

2 Background **5**
2.1 Factuality in Large Language Models 5
2.2 Mechanistic Interpretability 6
2.3 Modern Speech Language Models 9

3 Theory **13**
3.1 Preliminaries 13
3.1.1 What is a Language Model? 13
3.1.2 What is a Tokenizer? 14
3.1.3 Text Encoder 14
3.1.4 What is a Contextualized Representation? 14
3.1.5 Transformer architecture 15
3.2 Speech Language Models with Discrete Speech Language Modeling . 16
3.2.1 Speech encoder and feature extraction 17
3.2.2 Spirit LM: Model structure 17
3.2.3 Training HuBERT and Spirit LM 18
3.2.4 Downstream tasks 19
3.3 Causal Mediation Analysis 19

4 Methods **23**
4.1 Dataset selection and preparation 23
4.2 Forced Alignment for Cross-modal Token Mapping 24
4.3 MultimodalCausalTracer 26
4.4 Experimental setup 29

5 Results **31**

5.1 Initial factuality benchmark 31
5.2 Within-modality experiments: text-to-text 32
5.3 Cross-modality experiments: speech-to-text 33
5.4 Discussion 36

6 Conclusion **39**
6.1 Future Directions 40

Bibliography **43**

A 1: extended results plot for speech-to-text Causal mediation Analysis **I**

List of Figures

2.1 Representation of a car detecting neuron in the vision model *InceptionV1* [37]. Using features from the previous layers, it looks for wheels at the bottom of its convolutional window and windows at the top. 6
2.2 The evolution of Speech Language Models: from Pre-LLM stage with separated acoustic and language model to the current Speech LLM stage where audio and text are integrated in a unified Large Language Model. 9
2.3 A high-level overview of the two approaches to integrate speech into an LLM: *discrete speech modeling* (left) and *continuous speech modeling* (right) (from [29]). 10

3.1 A generic language model [55] 13
3.2 The original Transformer architecture [60] 15
3.3 (i) Speech tokenizer; (ii) SpeechLM; (iii) Token-to-speech decoder [62] 16
3.4 Structure of a suitable general speech-language model architecture for our study, alongside the possible tasks they are usually evaluated on [17]. 17
3.5 Overview of Speech LLM Architectures 18
3.6 In the example, one wants to estimate the effect of a drug on a patient, as well as the side effects of the aspirin assumption, which also impact the monitored health outcome. To disentangle the effect of the drug treatment from the aspirin intake, we employ causal mediation analysis (right panel) (image from the work done in [9]). 20

4.1 Results of the forced alignment for a speech utterance (transcript: "The capital of Roman Republic is"). The plot on top shows the trellis matrix, with the highlighted optimal path and score for each labeled letter; on the bottom, instead, we show the mel spectrogram of the spoken utterance, with the corresponding boundaries between each (spoken) text token. 27
4.2 Example of the causal graph representation in the case of text-based models (from [8]). 28
4.3 Causal traces plot, showing the **Average Indirect Effect** of individual model components over a sample of 1000 factual statements (AIE measured using probabilities; model used: GPT-2-XL, from [8]). . . . 28

5.1 **Baseline experiment with the unimodal backbone text LLM**, to study factual recall in multimodal models. It shows the log-scaled (log-probability) Average Indirect Effect over the sample of 754 correct factual statements in Known-t2t on the `Spirit LM` backbone language model, `Llama2-7b` [63], decomposed by single hidden state, MLP, and attention. As expected, the results align with results in [8] in text-only LLMs. 32
5.2 **Text-to-Text log AIE over the samples in Known-t2t** of individual components of `Spirit LM`. The results align with unimodal backbone model behavior, suggesting the preservation of text factual recall capabilities even after fine-tuning on another modality (speech). 33
5.3 **Example of a single CMA experiment in a Speech-to-text scenario**, with related tokens aggregated by their spoken text tokens (prompt transcription: *Windows Media Player* is developed by ___). The attribution map reveals a strong trace at the late site driven by the modalitydeclaration token used to interleave modalities during `Spirit LM` training alongside a weaker signal at the early site on the subject token. This earlysite signal mirrors the behavior observed when the model processes textonly inputs. 34
5.4 **Speech-to-Text Average Indirect Effect over the samples in Known-t2t**, per individual components of `Spirit LM`. As expected from the initial factual benchmark, we can notice a considerably smaller value magnitude, although a consistent causality signal around the last subject token as in the input text modality. 35
5.5 **Speech-to-Text Average Indirect Effect** (log-scale) of individual components of `Spirit LM`, over the 70 samples in the Known-s2t subset. Although a slightly reduced diffusion effect of the causal traces across multiple layers of the model, we can still observe the same AIE magnitude as in Figure 5.4, as well as the same causal effect pattern around the last subject token. 35

A.1 Impact of restoring state after the corrupted input. II
A.2 Impact of restoring MLP after the corrupted input. III
A.3 Impact of restoring Attn after the corrupted input. IV

List of Tables

4.1 Examples of data points from the Known dataset. 24

5.1 Factual benchmark performance of `Spirit LM` accuracy and average probability of the correct object for different input-output settings. A prediction is considered correct if the annotated object appears within the first five generated tokens. 31

5.2 **Average Total Effect (ATE)** in different experimental conditions. The ATE reflects the overall causal influence of targeted components on factual prediction performance, averaged across tokens and layers. 36

1

Introduction

Where do Speech Language Models store facts? In this thesis, we present valuable insights from our investigation into how factual knowledge is encoded and represented in the speech modality, with a Mechanistic Interpretability (MI) approach.

Large Language Models (LLMs) have demonstrated exceptional capabilities across a wide range of natural language tasks, such as fact-checking, text generation, summarization, and question answering [1]. These modelsmost notably large, autoregressive transformer-based architecturesare often able to predict facts about the world with impressive performance, including answering factual (prefix) prompts such as *"The capital of Italy is"* by relying on information stored in their parameters [2]. However, despite these successes, LLMs still suffer from hallucination, factual errors, and various types of biases: those represent limitations on their trustworthiness and practical use, especially in critical domains, real-world scenarios and knowledge which is less discussed on the web [3], [4]. Therefore, understanding the mechanisms underlying how these models encode, store, and recall factual knowledge has become a central topic in NLP, motivated by the need to improve their factual accuracy and reliability [2].

An emerging area of research called **Mechanistic Interpretability (MI)** aims to uncover how neural networks encode and process information by reverse-engineering their internal computations, converting them into human-understandable algorithms [5], [6]. Notably, recent works in this field often leveraging intervention-based methods inspired by causal inference have shown that factual knowledge can be localized within specific model structures (such as MLPs), rather than only identifying components responsible for phenomena like gender bias [7]–[10]. Beyond deepening our understanding of how models store knowledge, these insights have led to the development of model editing techniques, which make it possible to directly modify factual associations by precisely adjusting model parameters. This ability to target and intervene within the model marks significant progress toward creating language models that are more accurate, controllable, and reliable [8], [11], [12].

While the above advances have driven our understanding of text-based LLMs, another particularly remarkable and emerging capability is the adaptability of LLMs to integrate new modalities beyond text, by projecting the new modality into the same latent space [13]. **Speech (Large) Language Models (S(L)LMs)** exemplify this trend: by integrating audio feature extraction and speech-text multimodality, they have proven to reach SOTA performance in complex Spoken Language Understand-

ing (SLU) tasks [14]. A recent interesting approach for training modern spoken language models (SLMs) involves representing audio as discrete speech tokens using self-supervised speech encoders [15], [16]. These models learn both semantic and acoustic features by modeling sequences of discrete tokens derived directly from raw waveforms, without relying on text or labels [17]. Research into this paradigm of discrete representations has led to models that effectively bridge written and spoken modalities [18]–[20], enabling complex tasks such as turn-taking in conversations [21].

1.1 Research topics and motivation

While purely text-based LLMs have seen progress in the interpretability of their mechanisms, this investigation is still in early stages in Multimodal Large Language Models (MLLMs) [22], [23], particularly for modalities other than vision. Interpretability for speech LLMs remains underexplored compared to the text counterparts: in particular, investigating to what extent multimodal text/speech systems share underlying mechanisms remains unexplored. Recent works [24], [25] have explored speech model interpretability at the speech features level, considering different granularities (e.g., word boundaries, pronunciation), finding, for example, that frame-level representations within each word segment are not all equally informative. These studies lead to open questions more related to how sentence-level properties (e.g., subject) are encoded and the mechanisms behind a specific behavior, a gap our work contributes to fill.

The increased complexity and popularity of these types of speech systems make understanding how they encode, store, and retrieve information even more crucial, for two main reasons. In the first place, it would aid our understanding of knowledge representation in audio and multimodal models, and help to bridge the gap between speech and text modality [14]. Second, this knowledge can be transferred to real critical domains: from the medical domain to voice-based information retrieval (speaker identification), other than the design of reliable and real-time conversational AI [26]–[28].

1.2 Goals and challenges

We aim to employ Causal Mediation Analysis (CMA), a model-based intervention technique used in MI, to examine how factual associations are encoded in Speech-Language Models. Similarly to Meng et al. [8], our study focuses on pinpointing factual knowledge within these architectures and determining if insights from text-based models apply to the speech domain. To concretize the goal of the thesis, the main research question to be answered is formulated as follows:

> *How knowledge and factual association are encoded and memorized in Speech Language Models, and can they be localized?*

This, alongside the following sub-questions, which detail the main one:

- *Can Causal Mediation Analysis be used to identify and manipulate causal pathways of factual associations in Speech Language Models, similarly for text-based LMs?*
- *For multimodal speech (L)LMs, can the text modality influence how knowledge is localized in the speech one?*
- *Within-modality factual recall (e.g.: Text → Text, Speech → Speech): How is factual association recall supported when the input and output modalities are the same?*
- *Cross-modality factual recall (e.g.: Speech → Text, Text → Speech): How does causal attribution behave when input and output modalities differ?*

Considering the `Spirit LM` model as a case study, the main contribution of this work is the adaptation of CMA to autoregressive speech language models with discrete speech tokens. This enables the conduct of factual knowledge studies in different modalities (speech) or cross-modal scenarios by systematically intervening on hidden state activations while processing factual prompts, both for within- and cross-modal scenarios, providing valuable insights into factual recall mechanisms in SLMs.

Compared to text-based LLMs, such an investigation applied to Speech Language Models also poses remarkable challenges. On a language features level, speech LLMs face obstacles in their speech feature extraction process: using discrete speech tokens, there is an inherent difficulty in retaining nuanced semantic information [29], which can cause the internal mechanisms for factual recall to operate differently or less effectively compared to traditional LLM scenarios. Additionally, speech token sequences are generally much longer than their text counterparts [30], leading to significantly higher computational expenses and potential context window limitations. Moreover, in the speech domain and the related discrete speech tokens, it is no longer obvious which portions of a spoken utterance correspond to specific words or linguistic information (e.g., identifying the subject of a sentence). This creates additional complexity for intervention-based methodologies, as they depend on precisely locating and manipulating specific information within the model's representation space to establish causal relationships across the architecture.

1.3 Limitations and risks

Due to time constraints and computational expenses, our experiments focus primarily on within-modality and cross-modality scenarios where text serves as the output modality. Furthermore, our investigation is confined to a single Speech Large Language Model (`Spirit LM`), which means our findings may not generalize across the broader spectrum of speech systems of the same type. These limitations are important to acknowledge when interpreting our results, as different speech language model architectures, training methodologies, and multimodal fusion techniques could potentially yield different factual association patterns and behaviors.

Since this is a purely research-oriented project, direct risks are very limited. The study belongs to the area of Model Analysis of existing, open-source speech models:

we didn't release any new models, nor new datasets. Therefore, we don't see any trivial or obvious ways of abusing the presented results, nor exploiting them for harmful intentions.

1.4 Thesis outline

The thesis is organized into 6 chapters.

Chapter 1, gives an overview of our study, the context of the research, and our main results and discussion.

Chapter 2 is dedicated to an in-depth review of the background of the thesis, from the importance of factuality to related works to our investigation.

Chapter 3 contains all the preliminary theory behind the thesis is explained: this is mainly in terms of how modern speech Language Models work and how the technique used for the model analysis (Causal Mediation Analysis (CMA)) in the broader field of Mechanistic Interpretability (MI).

Chapter 4 outlines the methodologies of our research: model and dataset selection and details on how Causal Mediation Analysis is applied to the a multimodal (speech) model.

Chapter 5 presents the results of causal inference leveraged to explore factual associations in speech language models, and discusses how these relate to our research questions.

Chapter 6 concludes by summarizing the main takeaways, highlighting potential limitations, and promising future directions.

2

Background

2.1 Factuality in Large Language Models

Factuality has emerged as a core issue in LLMs due to their tendency to produce hallucinated or ungrounded outputs. User trust and adoption of such systems rely heavily on their ability to generate factually accurate content, especially in sensitive applications like healthcare and legal systems. Also for this reason, assessing and improving factuality in LLMs is currently considered as *"an emerging and crucial research area"* [3].

Current established approaches focus on mitigating hallucinations and improving factual accuracy by evaluating and enhancing factually stored knowledge within LLMs.

- Evaluating factuality in LLMs' text generation presents significant challenges, as it requires determining how accurately a generated statement aligns with established truths. Depending on which factual aspects we prioritize (such as knowledge grounding or uncertainty awareness), researchers have developed various benchmark datasets for assessment, including for long-form text generation. Two notable examples are *StrategyQA*, which tests multi-hop reasoning in Question Answering tasks, and *PopQA*, which evaluates a model's ability to handle questions about long-tail knowledge.
- Various approaches have been developed to improve model factuality, typically by updating internal knowledge or mitigating the effects of inaccurate facts. A possible strategy includes fine-tuning models on a small portion of the `CommonCrawl` dataset that shows similarity to high-quality reference corpora [1]. Implementing Retrieval-Augmented Generation (RAG) represents a reasonable alternative, which enhances factuality by accessing external knowledge sources, effectively bridging the gap between parametric memory (internal to the model) and non-parametric memory (retriever-based) [31], [32].

In addition to the approaches described above, the field of Mechanistic Interpretability provides additional pathways for investigating factuality in LLMs by revealing the underlying mechanisms of behaviors such as factual recall and Indirect Object Identification (IOI) tasks [8], [10], [33]. These insights enable two valuable applications: developing model editing techniques that directly intervene in the models' weights to update knowledge, as demonstrated by Rank-One Model Editing (ROME), MEND,

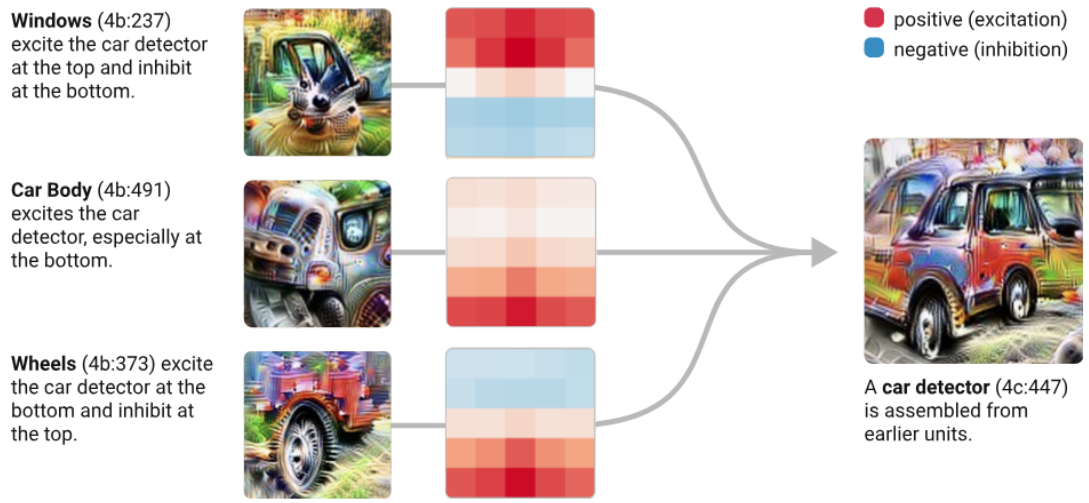


Figure 2.1: Representation of a car detecting neuron in the vision model *InceptionV1* [37]. Using features from the previous layers, it looks for wheels at the bottom of its convolutional window and windows at the top.

and MEMIT [8], [11], [12]; and enhancing AI safety by creating systems to monitor factuality-related behaviors such as truthfulness [34].

2.2 Mechanistic Interpretability

Mechanistic Interpretability (MI) a fairly new area of interpretability research. It aims to reverse-engineer the computations performed by large transformer-based models (definition in Section 3.1.5) to provide human-interpretable explanations of their behavior, in the same way as programmers reverse-engineer software binaries into human-readable code. Typically, an MI approach breaks down complex models into smaller, understandable components to analyze how different parts contribute to overall behavior, alongside discovering simple algorithmic pattern in toy models that can be applied to larger ones [5], [6], [35]. The term *"Mechanistic Interpretability"* was coined by Chris Olah in a series of blog posts in the Distil.pub Circuit Thread: in particular, in one of the first articles [36], MI was first introduced as "understand the *mechanistic* implementations of neurons in terms of their weights, as in the car detecting neuron example in Figure 2.1.

The rapid growth in terms of popularity and performance of transformer-based Language Models (LMs) has raised also concerns about their safety, generalizability and, as described in Section 2.1, factuality and reliability in their content generation; that's the main reason why MI gains considerable attention as research field that try to interpret not only what LMs learn, but also their behavior, enabling users to leverage the explanations to address LM shortcomings and improve their behavior [38].

In their influential article *Zoom In: An Introduction to Circuits*, Olah et al. from OpenAI [36] raise a foundational question in interpretability research: whether individual neurons and their connections can be meaningfully treated as a serious object of study. Building on this premise, they propose three exploratory claims that frame emerging research directions in mechanistic interpretability.

- **Features**. A feature in this framework represents a human-interpretable input property encoded within LM activations (e.g., "dog", "has four legs"). The primary objective in MI is to interpret LM representations by identifying and decoding the features they contain.
- **Circuits**. A circuit represents computational pathways (or graph) that connect features or transformer components, providing insights into specific LM behaviors ranging from factual recall to arithmetic reasoning [8], [39].
- **Universality**. This aspect examines the extent to which studies of features and circuits can be generalized across different models or tasks, and whether similar computational structures emerge across these varied scenarios.

To study the three main objectives of Mechanistic Interpretability, numerous techniques have been developed to study the fundamental claims of Mechanistic Interpretability described above and understand transformer-based models: we describe the most commonly used below.

- **Vocabulary Projection Methods**. The main aim of this class of techniques is to perform a layer-by-layer analysis to discover the intermediate latent representations of a model by projecting these internal states into a more human-interpretable space (also called vocabulary space). The idea behind this approach lies in an iterative inference perspective [40], where the model progressively develops a latent representation for a prediction: these methods aim to examine how this developmental process unfolds across the model's intermediate stages. *Logit Lens* [41] represents the representative technique of this class, where it projects intermediate hidden states through the final layer's unembedding matrix to produce logits over the vocabulary, revealing how token predictions stabilize throughout the network's depth rather than simply undergoing gradual refinement.
- **intervention-based techniques**. Those methods systematically intervene on model activations (*patching*), to identify which model components (*mediators*) are responsible for particular outputs [42]. These kinds of techniques are most commonly used for circuit study, both at the level of crucial model components of the study (nodes of the computation graph) or the connection between them (edges of the computation graph).
- **Sparse Autoencoders (SAEs)**. For feature discovery, Sparse Autoencoders are external neural networks employed to address the superposition problem, where individual neurons in language models simultaneously encode multiple concepts [43]. The core idea involves training these autoencoders to decompose the model's dense activations into a sparse, higher-dimensional space where each learned feature corresponds to a more interpretable, monoseman-

tic concept. The primary objective of SAEs is to disentangle the polysemantic representations within language models, transforming them into explicit, individually meaningful features that can be analyzed in isolation. To achieve this sparse and interpretable representation, SAEs are trained using a dual loss function: a reconstruction loss that ensures faithful recovery of the original activations, and a sparsity penalty that constrains the autoencoder to activate only a small subset of features for any given input.

In this thesis, we utilize Causal Mediation Analysis (CMA), also referred to as Causal Tracing, a fundamental intervention-based method that facilitates the causal exploration of how intermediate variables and model components influence specific outputs. A key reason for selecting this approach is its ability to enable both structural analysis, investigating the internal architecture of the model, and behavioral analysis, evaluating the models reactions to specific inputs. While at the same time, emphasizing the relationships between components across different layers of the model, unlike Logit Lens, and without the need to train an external estimator, as required by methods like Sparse Autoencoders [9]. By systematically intervening on model activations (a process known as patching), CMA highlights which internal components (mediators) are responsible for certain outputs. The model is treated as a computational graph, and the influence of individual components is quantified using the concepts of Total Effect (TE) and Indirect Effect (IE) [44]. Further methodological details are provided in Section 3.3.

Causal Tracing has been successfully used across diverse domains to uncover the inner mechanisms of neural models. For example, researchers applied CMA to understand gender bias in neural NLP models, identifying specific neurons and attention mechanisms that mediate biased behavior; this, alongside applications in vision-language models to assess and mitigate bias, demonstrating its utility in multimodal settings by tracing causal pathways between image and textual representations [9]. Moreover, the technique enables the exploration of arithmetic reasoning in language models, identifying how numerical knowledge propagates through transformer architectures [39]. Several studies have focused on Multimodal Large Language Models, including efforts to measure and trace the pathways of bias generation and propagation within Vision Language Models (VLMs), as well as to develop mitigation strategies based on these findings [45]. Other research has investigated the core components of VLMs responsible for factual recall, particularly in models like LLaVa [23], [46].

The examples above demonstrate the versatility of CMA and its applicability to both unimodal and multimodal systems. However, most existing studies have concentrated on vision-based multimodal models, leaving a gap in the literature regarding the use of these techniques to analyze speech Language Models and uncover the underlying circuits responsible for their behavior. Extending CMA to speech-based systems can reveal how these models encode multimodal knowledge and whether patterns observed in text language models (LMs) carry over to speech architecturesa key aim of this thesis.

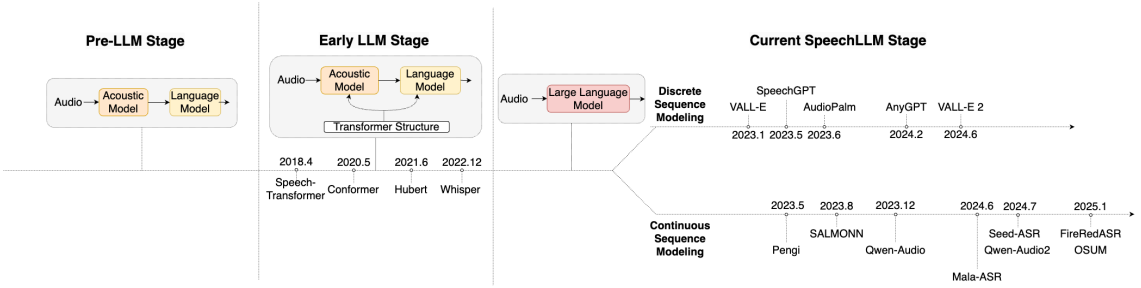


Figure 2.2: The evolution of Speech Language Models: from Pre-LLM stage with separated acoustic and language model to the current Speech LLM stage where audio and text are integrated in a unified Large Language Model.

2.3 Modern Speech Language Models

Given the success of text-based language models (LLMs) a popular approach to speech language models has been to run a three-stage Automatic Speech Recognition (ASR), text Language Model, Text-To-Speech (TTS) pipeline. The ASR model transcribes the speech input to text, passes it to the language model, which generates a textual response that is converted to speech with the TTS model. This pipeline was for example used by OpenAI with the release of *"ChatGPT can now see, hear, and speak"* [47] for GPT4 in 2023 which relied on Whisper [48] for ASR.

However, this pipeline introduces latency, compounds errors, and loses aspects of speech that have proven useful to many speech tasks: prosody, emotional content, and the degree of background noise [14]. Consequently, there has been a number of approaches focusing on *end-to-end* speech models.

A popular example of this is the transition from ChatGPT Voice Mode to GPT4o (o for omni handling text, vision, and audio natively) with *"Hello GPT-4o"* in 2024 [49]. An additional reported benefit with an end-to-end approach is the reduction in average inference time for audio inputs: from 5.4 seconds for GPT4 to 320 milliseconds (similar to human response time) [49].

In more general terms, recent innovations in LLMs have also shown impressive potential in speech-based language tasks by integrating speech features extracted by the audio feature extractor, through models such as `Spirit LM` ([18]) or *SpeechGPT* ([19]): those kinds of systems, by integrating audio feature extraction and speech-text multimodality, have proven to reach SOTA performance in complex Spoken Language Understanding (SLU) tasks (long-form speech understanding, hot-word recognition) while retaining capabilities in the original text modality ([14]).

As illustrated in Figure 2.2, current research integrating speech in LLMs converged into two different approaches: *discrete sequence modeling* and *Continuous Sequence Modeling*. In both cases, the major motivation for this paradigm is its ability to bypass traditional transcription pipelines, expanding language coverage to low-resource settings and enabling expressive speech modeling that captures aspects beyond linguistic content [50].

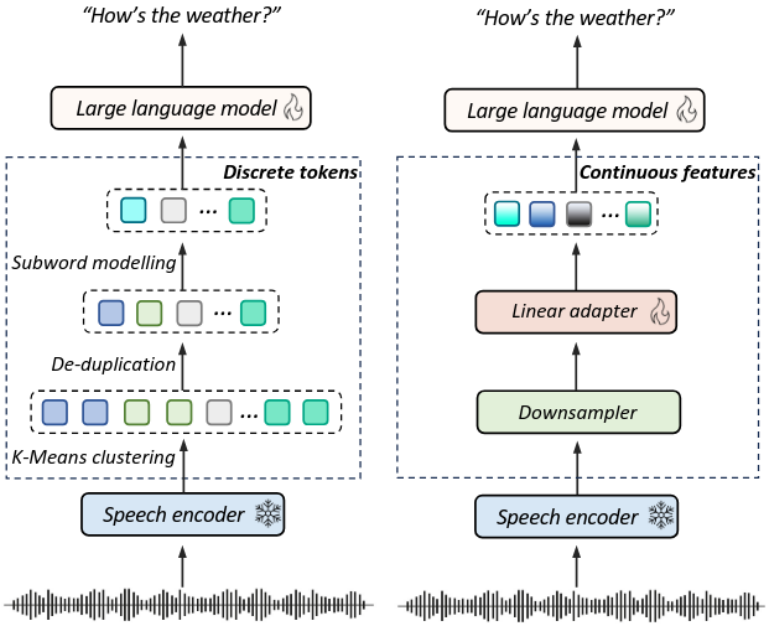


Figure 2.3: A high-level overview of the two approaches to integrate speech into an LLM: *discrete speech modeling* (left) and *continuous speech modeling* (right) (from [29]).

Discrete sequence modeling. This paradigm was first introduced with the *Generative Spoken Language Modeling (GSLM)* framework and further advanced through research from the *Textless NLP project*[1][17]. The main idea of the approach is to learn acoustic and linguistic features of a language by transforming the raw continuous speech signals into discrete unit representations, using self-supervised speech encoders such as *HuBERT* or *Wav2Vec2* [15], [16]. These units enable LLMs also to operate in a familiar token-based paradigm, bridging the gap between speech and text processing. Moreover, these discrete tokens are not only suitable for language modeling but also allow reconstruction of the original audio [51]. We present an overview of the high-level pipeline in Figure 2.3 (left): following feature extraction using a speech encoder, these representations are quantized through unsupervised clustering algorithms such as K-means and subsequently refined via post-processing techniques (de-duplication, subword modeling) to enhance training efficiency and improve generalization [29]. Research into discrete representations has driven models like the ones already mentioned above, which bridge written and spoken modalities [18]–[20]. Other notable examples include systems such as *Generative Spoken Dialogue Language Modeling (dGSLM)*, which extend this paradigm by generating speech, paralinguistic signals (e.g., laughter), and facilitating fluid turn-taking in conversations [21].

Continuous sequence modeling. In the second paradigm, called continuous sequence modeling, we similarly begin by encoding audio through a pre-trained and

[1]Link of the project: `https://speechbot.github.io/`

fine-tuned speech encoder such as Whisper [48] or a Conformer model [52]; the resulting features are then transformed via projection layers to achieve alignment with the LLM's embedding space [14]. As shown in Figure 2.3 (right), before the features projection step, a downsampler is applied to the speech features extracted by the speech encoder, to lower computational complexity while retaining semantic and acoustic information. If the speech encoder and downsampler perform enough compression of the audio sequence, this transformation can often be done with a simple linear layer. Whereas, if the encoder produces a continuous representation without discretizations or temporal compression, a more complex *speech modality adapter* is typically required. This approach has also seen speech LMs that can reach State-Of-The-Art (SOTA) in tasks such as Automatic Speech Recognition (ASR) or Speech Translation (ST), since continuous embeddings can preserve a richer and more fine-grained representation of the audio: representative examples are *Pengi* and more recently, the Qwen-Audio model [14], [53], [54].

In this study, we focus on speech language models that integrate the new modality through discrete sequence modeling, since the representation is more suitable and interpretable for applying current Mechanistic Interpretability techniques. The development of discrete speech token-based SLMs opens new pathways for techniques like CMA to speech-based systems, with potential deeper insights into how factual knowledge is encoded, stored, and retrieved. By combining these methodologies, we aim to bridge the existing gap between speech and text modalities, discovering new circuits behind the SLMs' behavior, and advancing the broader understanding of factuality in speech and multimodal language models.

3

Theory

3.1 Preliminaries

Modern speech language models mainly rely on transformers. This section gives a brief overview of this architecture and how a language model works.

3.1.1 What is a Language Model?

A Language model is a machine learning (ML) model trained on human language. Specifically, the model is given a vocabulary set V of tokens (i.e., words and subwords) and trained on token sequences to output a probability distribution over the vocabulary. In other words, for each token v_j in the vocabulary, what is the conditional probability of that token being the next token t_{i+1}, given the input tokens $t_1, ..., t_i$. During inference, this probability $P(t_{i+1}|t_1, ..., t_i)$ is used to generate text one token at a time using either a greedy strategy (always selecting *argmax*), *beam search* (argmax given the joint probability of multiple tokens), or by sampling from the distribution. Jurafsky and Martin give a nice visualization [55], see figure 3.1.

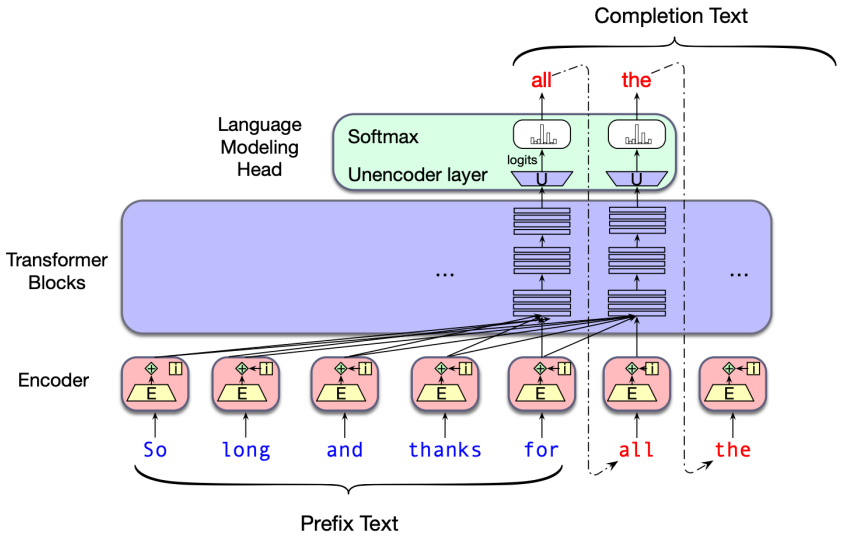


Figure 3.1: A generic language model [55]

3.1.2 What is a Tokenizer?

When a human interacts with a language model, they typically input a string of text (e.g., *"The Capital of Italy is*). The expansiveness of language means that the input is likely to not have been seen by the model before. Still the model is expected to understand (or at least process) the input. This understanding is facilitated by breaking down the input string into a sequence of individual tokens which the model has seen many times before. This is done by a *tokenization algorithm* like *Byte-Pair Encoding (BPE)* [56] to produce e.g., `['<s>', 'The', 'capital', 'of', 'Italy', 'is']`. The tokenizer also performs several steps, including *pre-tokenization* (splitting the raw text into chunks, e.g., by whitespace) and *Normalization* (removing accents, punctuation, and converting to lowercase). Finally, each token is mapped to a numerical token ID (an integer) within the vocabulary set. e.g. [450, 7483, 310, 12730, 338].

3.1.3 Text Encoder

Because a neural network is trained with (stochastic) gradient descent, it is necessary to be able to take the gradient (derivative) of the loss function. For the function to be differentiable, we need a continuous representation of text. This is achieved with an *Encoder* which maps each discrete token ID of the vocabulary from the input sequence to a continuous representation vector of size d referred to as the *embedding dimension* or *hidden size*. This vector encodes semantic meaning of the token across a d-dimensional space.. As that initial vector - the first embedding layer - is passed through the network, it gets a *contextualized representation* of the word which depends on the attention given to the surrounding (context of) tokens.

3.1.4 What is a Contextualized Representation?

Early continuous embeddings of tokens like *Word2vec* and *GloVe* were *static* meaning that the vector representing the token was fixed. But a word (token) can have multiple semantic meanings depending on context. A classic example is the word bank in the context of money and in the context of rivers [55]. This led to the development of *contextualized embeddings* with RNN-based encoder-decoder sequence-to-sequence models, using either GRUs [57] or LSTMs [58]. These models were similar to modern language in that they processed the input sequence autoregressively (from left to right). But unlike modern transformer-based LMs, they did so sequentially leading to problems with *vanishing gradients* and long-range dependencies.

These two approaches to neural machine translation (NMT) encoded the entire input sentence into a fixed-length context vector (c_i) from which a translation was decoded. Bahdanau et al [59] conjectured that the use of a fixed-length context vector was problematic for translating sentences of varying length and introduced the (cross-)*attention mechanism* to generate a distinct context vector c_i for each target word y_i. This allowed the model to shift its focus (in parallel) to different parts of (the encoder-output of) the input, depending on what token it was generating. Contextual representation are important in transformer architectures because it allows

the model to add increasingly rich token representation, in the embedding space, as tokens flows (i.e., are processed) through the *'residual stream'* of the network [5]. Leading interpretability researchers at Anthropic studying transformer models, conceptualize transformer components (token embedding, attention heads, MLP layers, and unembedding) as "communicating with each other by reading and writing to different subspaces of the residual stream" [5].

3.1.5 Transformer architecture

Vaswani et al [60], in the now ubiquitous work *Attention is all you need* [60], extended this approach by proposing *the Transformer* architecture. By allowing both the encoder and the decoder to use their own type of (self-)attention, they completely removed the need for sequential RNNs, greatly facilitating parallel processing and scalable contextual representations. The original architecture can be seen below in figure 3.2. the model consists of multiple *transformer blocks* where each block contains multiple attention layers, multilayer perceptron (MLP) layers, and layer normalization. An attention layer in turn consists of multiple attention heads that can *attend to* different parts of the context independently and the outputs are then concatenated.

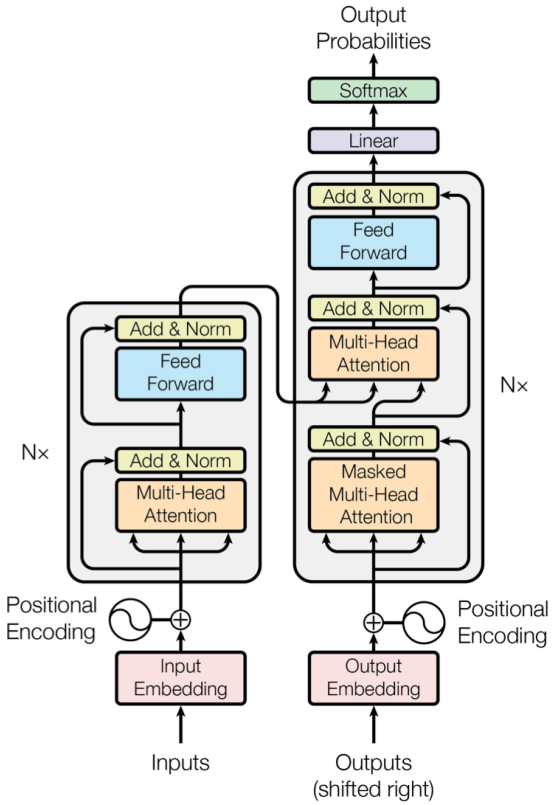


Figure 3.2: The original Transformer architecture [60]

$Q_i^\top K_j$ On a more technical level, the attention mechanism is implemented with four parameterized linear projection matrices for each attention head: W_Q, W_K, W_V, W_O. For each layer (attention head) in the network, and for each token, the token embed-

ding x_i, is projected into a query vector $Q_i = W_Q\, x_i$. All other token embeddings in the context are projected into key vectors $K_j = W_K\, x_j$, and the querykey dot product $Q_i^\top K_j$ is computed to produce a similarity score. This score is scaled by the inverse square root of the dimensionality of the queries and keys, $\sqrt{d_k}$, and normalized using the softmax function. The resulting attention score a_{ij} quantifies how much token i attends to token j, and is used to compute a weighted sum of value vectors $V_j = W_V x_j$. This determines how much of each attended token embedding should be passed into the *residual stream* (so-called because of residual connections; cf. [61]). This is referred to as *scaled dot-product attention* and is shown below in equation 3.1. Finally, after concatenating the outputs of all attention heads, the result is projected back into the model dimension d_{model} using W_O (see equation 3.2).

$$\text{Attention}(Q, K, V) = \text{softmax}\left(\frac{QK^T}{\sqrt{d_k}}\right)V \tag{3.1}$$

$$\text{MultiHead}(Q, K, V) = \text{Concat}(\text{head}_1, \ldots, \text{head}_h)W^O \tag{3.2}$$

3.2 Speech Language Models with Discrete Speech Language Modeling

Unlike text, speech is a continuous waveform without separation between linguistic information and other acoustic aspects and with no self-evident tokenization algorithm to break the input down into subwords/phonemes.

As mentioned in Section 2.3, one of the key technical breakthrough in modern Speech Language Models is discrete speech language modeling, where raw audio can be leveraged by LLMs directly without text or label supervision to build a *Generative Spoken Language Modeling System* [17]. Lakhotia et al. introduced a new SLM pipeline that has proven successful and been reiterated and expanded on by both TWIST [62] and `Spirit LM` [18].

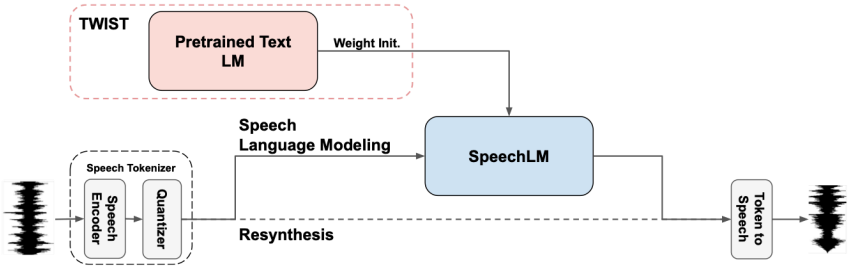


Figure 3.3: (i) Speech tokenizer; (ii) SpeechLM; (iii) Token-to-speech decoder [62]

Figure 3.3 shows the overview of the GSLM framework. The first component is a discrete speech encoder generating pseudo-text tokens to be used by an LLM. They evaluated three such encoders (CPC, wav2vec 2.0, HuBERT) and found that HuBERT [15] performed best. Another noteworthy contribution by the same authors

is the introduction of textually pretrained speech models [62], which continues training on a pretrained text LLM like Llama2 [63] and fine-tunes it on a speech dataset. Spirit LM uses this approach by expanding the vocabulary of Llama2 with discrete speech tokens generated by the Hubert speech encoder.

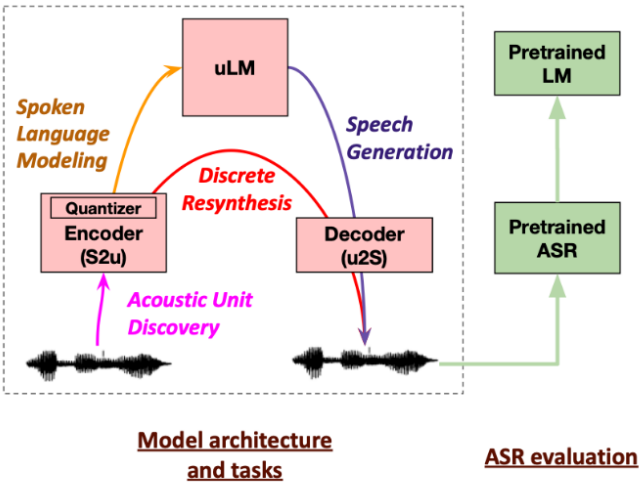


Figure 3.4: Structure of a suitable general speech-language model architecture for our study, alongside the possible tasks they are usually evaluated on [17].

3.2.1 Speech encoder and feature extraction

A *waveform* is sampled from an audio signal with a predetermined *sample rate* (e.g., Spirit LM is trained with 22kHz) generating a sequence (array) of values corresponding to the audio signal's amplitude. To be processed by a language model there must be some form of feature extraction (also referred to as: acoustic unit discovery) and temporal compression of the audio sequence [64]).

A speech encoder (e.g., *HuBERT* [15], *wav2vec2* [16]) encodes the waveform and produces a continuous latent representation. In *discrete sequence modeling*, an audio tokenizer, also known as *Quantizer*, is then used to discretize this continuous representation into a sequence of discrete audio tokens via a learned *codebook* (e.g., using k-means, *VQ-VAE* [65], etc.), similar to how text tokens are modeled with a vocabulary set (sometimes the word audio tokenizer is used to refer to the whole encoder+quantizer block).

In the case of Spirit LM, this is achieved with HuBERT speech encoder and utilizing k-means clustering and self-supervised speech representation learning [14] to tokenize the audio. Discrete tokens have the benefit that they are more easily integrable with the language model without requiring additional adapters.

3.2.2 Spirit LM: Model structure

Spirit LM uses the previously outlined model structure (see figure 3.3) introduced in GSLM and TWIST: (i) Speech tokenizer; (ii) SpeechLM; (iii) Token-to-speech

decoder.

Speech Models can also be categorized depending on the input and output modalities, see the overview below 3.5 taken from [14]. Spirit LM [18] is a ST2ST model trained on interleaved inputs of speech and text meaning it can support both inputs and outputs natively end-to-end.

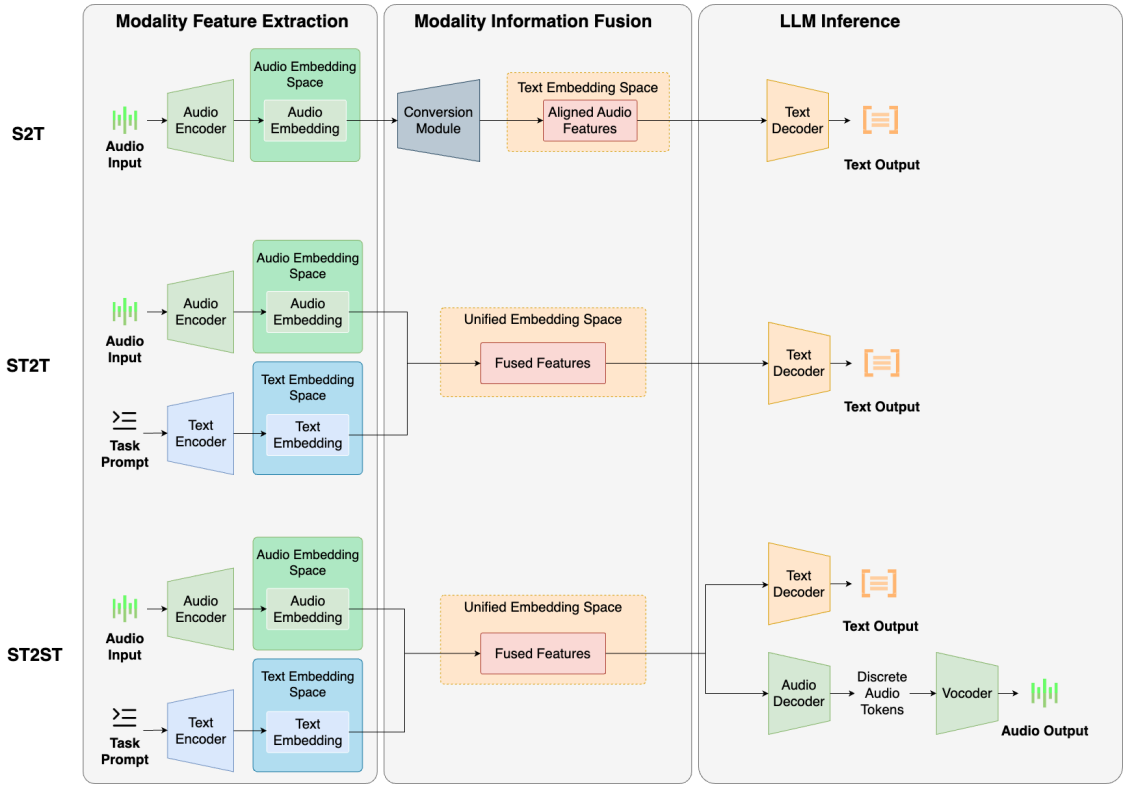


Figure 3.5: Overview of Speech LLM Architectures

When processing speech inputs, Spirit LM relies on the previously mentioned HuBERT based speech encoder to produce hidden units, which are then fed into the underlying language model. The modality fusion is done implicitly by the model within the network and it keeps track of the input and output modality through the use of special modality tokens [SPEECH] and [TEXT] preceding the other tokens.

3.2.3 Training HuBERT and Spirit LM

HuBERT is a form of self-supervised learning (SSL), where labels are constructed through unsupervised learning on an unlabeled dataset and then used by in a supervised learning manner [66]. This is the underlying paradigm for how modern LLMs like GPT and BERT [67] have been trained on massive unlabeled text corpora. More specifically, HuBERT uses K-means clustering to find clusters in sequences of audio data from (60,000h of) Librispeech [68]. The number of discrete hidden units (cluster centroids) used is referred to as the size of the codebook, a common choice that works well in practice is 100 clusters [15], [17]. The K-means model is then used as a teacher model for BERT-style SSL with a masked prediction loss to predict these

hidden units and thereby learn what type of *Hubert-tokens* follows each other. These Hubert tokens can be thought of as encoding phonemes or sub-phonemes which are the smallest components of human speech.

Spirit LM [18] is a multimodal LM trained with a *warm start*, previously found to be an effective way to train speech models [62], starting from the underlying text-based Llama2 model [63]. The speech modality is incorporated into the model by a *token vocabulary expansion* adding the previously mentioned Hubert-tokens to Llama2's existing Byte-Pair Encoding (BPE) token vocabulary. The model is then trained from the Llama2 checkpoint by training on interleaved inputs of speech and text preceded by modality declaration tokens.

3.2.4 Downstream tasks

Some common examples of where speech language models show strong performance is ASR, SLU (Spoken Language Understanding) which includes tasks like emotion and intent classification, slot filling, and semantic parsing directly from speech inputs [14]. End-to-end speech translation skipping the intermediate transcription step thereby reducing error propagation and improving latency [69]. Lastly the development of native speech models is interesting because it heralds the advent of natively multimodal models which can combine text, speech, and vision which can already be seen in the strong multimodal reasoning abilities of GPT-4o [49].

3.3 Causal Mediation Analysis

As mentioned in Section 2.2, Causal Mediation Analysis (CMA) (or *Causal Tracing*), a technique derived from *causal inference* [44], seeks to study the change in a response variable caused by an intervention on another variable X referred to as a *"do"* operation *do(X=x)*. Before giving an example, it's worth pointing out that this causal investigation allows the separation of *direct effects* and *indirect effects* induced by intermediaries (also referred to as *mediators*) [44].

Consider a clinical trial analyzing the health outcome following a drug treatment that causes headache as a side effect. In this case, taking the drug (*do(X=x)* may cause the patient to take an aspirin (the mediator variable), which, in turn, impacts the health outcome of the patient. For a visualization, see the example in Figure 3.6; similarly to its right panel, we employ the capitalized letters of the graph as notation: define X as a control variable that affects an outcome Y based on its change of value, together with a mediator Z between X and Y.

Similarly to the clinical trial described above, a neuron or any other transformer component (e.g., attention heads or feed-forward layers) is a mediator that is influenced by the model input and, in turn, affects its output prediction. For any input sentence, we can measure the effect of an intervention (e.g., a text edit) by comparing the model output with and without the input change and treating model components to discover any indirect effect.

More in particular, in the field of Mechanistic Interpretability, CMA is considered

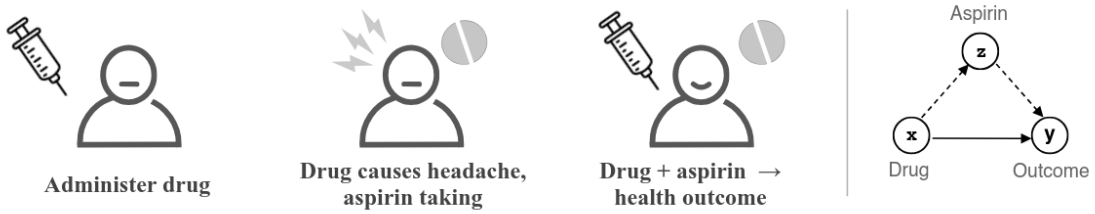


Figure 3.6: In the example, one wants to estimate the effect of a drug on a patient, as well as the side effects of the aspirin assumption, which also impact the monitored health outcome. To disentangle the effect of the drug treatment from the aspirin intake, we employ causal mediation analysis (right panel) (image from the work done in [9]).

an Intervention-based method: It represents one of the most effective techniques to study the fundamental objects described in Section 2.2, with a focus on meaningful computational pathways described by circuits. The idea is to alter the value of an intermediate computation during a forward pass (inference) and observe how this change will impact the model output. In this context, an LM of L layers is visualized as a computational *causal graph* G [44], a Directed Acyclic Graph (DAG) that describes the dependencies between the intermediate computations of the model components. Moreover:

- X represents the input token sequence $[x_0, \ldots, x_T]$ to the language model;
- we consider the outcome Y as the probability of the model to output a token o;
- a mediator Z is a model component's internal state. All the transformer components $\{C_1, \ldots, C_N\}$ represent the nodes of the causal graph G. Since we intervene on the nodes to explore the causality between any type of model component and the output, a node C_i ($i \in [1, N]$) can be defined at various levels of granularity: from a single neuron to a component such as a feed-forward network or an attention head, or also an entire transformer layer.

A typical procedure for applying Causal Mediation Analysis (CMA) with model interventions to investigate language model behavior consists of three runs [8], intending to measure the impact of a component on the final predicted output.

Clean run. The language model is provided a clean prompt $X = x$, producing an output y. The outcome is measured as the probability $\mathbb{P}_x[o]$, where o denotes the expected decoded text token. The selection of o is usually based on annotated ground-truth answers corresponding to the input prompt, as exemplified by the target tokens in the *Known* dataset that will be described in Section 4.1 and utilized in this work. The corresponding hidden states from this inference are cached.

Corrupted run. The model receives a corrupted input prompt $X = x^*$, resulting in a prediction y^*, with the outcome denoted as $\mathbb{P}_{x^*}[o]$. The corruption is a targeted

intervention on a specific part of the text that we want to study (for example, adding noise to the subject embedding of a prompt).

Corrupted-with-restoration run. The same corrupted prompt $X = x^*$ is passed to the model, but with selected values of one or more values of components C_i restored (patched) from the *clean run*. The resulting outcome is denoted as $\mathbb{P}_{x^*,\text{clean } C_i}[o]$, where "clean C_i" refers to the value of the hidden state at layer l and position i from the clean inference.

The results of the three runs allow us to quantify both the overall and mediated effects of interventions. The impact of changing the prompt from clean ($X = x$) to corrupted ($X = x^*$) on the target probability Y is known as the *Total Effect* (TE) [44], defined as:

$$\text{TE} = \mathbb{P}_x[o] - \mathbb{P}_{x^*}[o].$$

Similarly, the contribution of a component mediator can be measured by the *Indirect Effect* (IE), defined as:

$$\text{IE} = \mathbb{P}_{x^*,\text{clean } C_i}[o] - \mathbb{P}_{x^*}[o].$$

Averaging over a set of statements, we obtain the *Average Total Effect (ATE)* and the *Average Indirect Effect (AIE)* for each targeted component C_i, respectively. It is worth mentioning that there are multiple ways to define both Direct and Indirect Effect, including formulations that can partially handle unmeasured treatment outcome confounding [70]; in our case, the natural framework proposed in [44] represents a more suitable choice to investigate LMs in a Mechanistic interpretability point of view.

To measure the patching effect after the three runs and evaluate the importance of specific model components, there are established metrics to analyze the difference of the output between y and y^*, other than the probability, also through the restoration of C_i:

- Softmax **probability, log probability or logit** change of y before and after the patching;
- **probability, log probability or logit difference** change between the value of Y in the clean run and the corrupted one, both before and after the patching;
- **Kulleback-Leiber (KL) divergence** between the probability (or logits) distribution before and after the corrupted-with-restoration run (note that in this case we are not focusing on a single token probability or logits, but rather the full output distribution).

Logits difference is usually a preferred choice: not only because it allows us to analyze components that mediate the value of Y both in clean and corrupted; from empirical results from [42], probability as a measurement might not detect negative components that play an important role in suppressing the correct output. On the other hand, the KL divergence represents a reasonable and complementary metric to the logits difference, since it can provide an evaluation of the overall effect of the patching.

4

Methods

This study is mainly inspired by previous works that apply *Causal Mediation Analysis* to investigate how language models process knowledge stored in their parameters [8]. We revised and adapted this method to create *MultimodalCausalTracer*, which extends the traditional LLM causal tracing to the speech Language Models and allows investigating research questions about how they encode factual information across modalities. The following sections describes the frameworks and methodologies that underpin the mechanistic interpretability study conducted on `Spirit LM`:

- the data selection and preparation
- a mechanism to locate targeted words in audio, as well as mapping discrete speech tokens to corresponding text tokens;
- an overview of the *MultimodalCausalTracer* framework, with focus on how it enables handling speech as input and the related CMA results;
- the experimental setup of the CMA experiments.

4.1 Dataset selection and preparation

The experiments are conducted on the *Known* dataset [8]. *Known* consists of 1209 factual prompts prefix to be completed [1], covering topics such as countries, famous people, and companies, as well as different types of binary relations in the form of "X born-in Y" or "X works-for Y". Each sample is annotated with the subject s of the prompt and the object o: the expected prediction answer. This represents a knowledge tuple (s, r, o) with a prompt (s, r) from which the object o is meant to be predicted in the factual recall experiment (r is the relation). Additionally, each data point contains the relation ID, derived from the *ParaRel* dataset [71]. We present some examples of Known data points in Table 4.1.

To evaluate Spirit LM in the speech-to-text setting, we created an audio version of the dataset by synthesizing the prompts, subjects, and objects of *Known* into utterances with MeloTTS [72]. MeloTTS is a high-quality text-to-speech (TTS) model based on architectures that leverage adversarial learning to improve expressive power, and high-quality natural-speech synthesis.[73], [74].

[1] Link to set of factual prompts: `https://rome.baulab.info/data/dsets/known_1000.json`

Prompt	Subject	Object	Relation ID
Windows Media Player is developed by ___	Windows Media Player	Microsoft	P178 (developed by)
NTFS is developed by ___	NTFS	Microsoft	P178 (developed by)
Lexus's owner, ___	Lexus	Toyota	P127 (is owned by)
Catalonia belongs to the continent of ___	Catalonia	Europe	P30 (is located in)
Eavan Boland was born in ___	Evan Boland	Dublin	P19 (was born in)

Table 4.1: Examples of data points from the Known dataset.

Furthermore, to ensure that the collected dataset is accurate for the investigation across modalities, we select and divide the factual statements of Known into two subsets. The first subset, named *Known-t2t*, includes data points where the model readily generates either an exact correct answer or a close variant in a Text → Text scenario. For instance, if the model generates answers such as "Rome", "Rome, Italy" or "the city of Rome" to the prompt "The capital of Italy is _____", the latter will be included in Known-t2t. The second subset, called *Known-s2t*, follows the same selection criteria, but in a Speech → Text setting.

4.2 Forced Alignment for Cross-modal Token Mapping

Establishing causal effects through CMA requires a *do-operation* (see theory section 3.3), in our case: corrupting the subject s of the input prompt. To conduct the CMA experiment on the speech modality, it was therefore necessary to locate specific parts (i.e., the subject s) of the prompt within the spoken utterance. Forced alignment provides an effective solution by creating a time-aligned representation between speech and words within its transcription. Recent forced aligners employ Connectionist Temporal Classification (CTC) [75] to model the speech-text alignment, since CTC allows for labeling variable-length sequences without requiring explicit temporal boundaries. This capability is particularly valuable for extracting utterance segments from speech corpora containing unsegmented speech, and scales effectively to multilingual settings [76], [77].

We adopted the CTC-based forced alignment approach described in [76] for our work with the `Spirit LM` model. This approach enables a precise mapping from speech tokens to their corresponding text tokens, an essential step for identifying the mediator of interest (here, the subject of a sentence) and carrying out the CMA experiment in the speech modality. The creation of speech-text alignment and the related cross-modal token mapping consists of the following steps.

Text preprocessing for CTC. For the transcription to be compatible with the forced alignment, a text preprocessing is necessary to ensure all characters are included in the CTC model vocabulary. For example, digits and special characters such as "%" are converted to their written format (e.g, "0" becomes "zero", or "%" becomes "percent"). On the text token-level, preprocessing can lead to a longer or shorter sequence of tokens, compared to the original text being tokenized. We later refer to the preprocessed text tokens as *spoken text tokens*. The preprocessing step concludes by joining the spoken text tokens into a single string, using the word boundary character defined by the CTC model as a separator.

Frame-wise label probability estimation from audio waveform. We generate emission probabilities per audio frame, using the pre-trained `HuBERT-LARGE`[2] model as a speech tokenizer. This model is fine-tuned for automatic speech recognition (ASR) with CTC loss, representing a suitable candidate for this use case [15]. In other words, the CTC network computes the probabilities $p(c|t, X)$, where c represent any character label in the vocabulary model, $\mathbf{X} = (x_1, \ldots, x_T)$ the frame-wise audio input and $t \in [1; T]$ an arbitrary time step of the input.

Trellis matrix generation with log-probability of label alignments at each time step. Consider the same previous speech input $\mathbf{X} = (x_1, \ldots, x_T)$ and transcript labels $(c_1, \ldots, c_N)$ at the character level. We now compute through dynamic programming and map all possible joint probabilities in the trellis diagram matrix $K \in \mathbb{R}^{T \times N}$, where $K_{(t,j)}$ represents the maximum log-probability of aligning the first labels $j \in [1; N]$ up to time $t \in [1; T]$. To compute the probability at time step $t + 1$ for label c_{j+1}, we consider two possible transitions: either we stayed on the same label c_{j+1} or transitioned from c_j to c_{j+1}. Based on these criteria, the trellis is updated as follows:

$$K_{(t+1,j+1)} = \max \begin{cases} K_{(t,j)}\, p(t+1, c_{j+1}) \\ K_{(t,j+1)}\, p(t+1, \text{repeat}) \end{cases} \tag{4.1}$$

where $p(t + 1, c_{j+1})$ is the probability of emitting label c_{j+1} at time $t + 1$, and $p(t + 1, \text{repeat})$ is the probability of emitting no label change.

Merge repetitions and segments into words (spoken text tokens). The final step involves postprocessing the output from the optimal path. Because the path may contain consecutive repetitions of the same label, we merge path points corresponding to repeated characters into a single segment to make it close to the original transcript.[3] Similarly, we group segments that correspond to the same spoken text token, using the word boundary character as a guide. The result is a sequence of segments, each representing a spoken text token from the transcript

[2]Link to the model checkpoint used: `https://huggingface.co/facebook/hubert-large-ls960-ft`

[3]When merging path points into a single segment, we use the average probability of all frames in that segment.

and annotated with the corresponding range of audio frames and average emission probability.

The final result of CTC-based forced alignment is illustrated in Figure 4.1. Beginning with the preprocessed transcript "THE|CAPITAL|OF|ROMAN|REPUBLIC|IS," we align each text token with its corresponding segment in the audio. For each text token, we label the aligned speech segment with the average probability over the merged segment, clearly indicating its position within the utterance as a highlighted segment on the spectrogram, with boundaries marking its start and end. This segmentation process allows us to determine the precise time range for each text token.

- Given the frame range (f_{start}, f_{end}) of a spoken text token segment, we can first compute its time range in seconds (s_{start}, s_{end}) in the utterance with the following formula:

$$s_{start} = \frac{\lfloor ratio \cdot f_{start} \rfloor}{sr}, s_{end} = \frac{\lfloor ratio \cdot f_{end} \rfloor}{sr}, \tag{4.2}$$

  Where sr represents the sample rate of the original sampled waveform $Z = (z_1, \ldots, z_M)$, while $ratio = \frac{M}{T}$ represents the number of samples contained in a frame.

- Then, considering the token rate of the speech tokenizer tr and the previously computed time range (s_{start}, s_{end}), the corresponding speech token range (stk_{start}, stk_{end}) is given by:

$$stk_{start} = \lfloor s_{start} \cdot tr \rfloor, stk_{end} = \lceil s_{end} \cdot tr \rceil. \tag{4.3}$$

This direct mapping provides a clear correspondence between each text token and its associated speech tokens, linking elements of the transcript to their acoustic realizations in the audio.

4.3 MultimodalCausalTracer

To localize facts in `Spirit LM` using Causal Mediation Analysis, we developed *MultimodalCausalTracer*, which extends the original LLM-focused technique to handle both the speech modality and multimodal setups. Our approach leverages Spirit LM's pseudo-text tokenization of audio, enabling us to apply analysis using a workflow similar to that of the text modality. We also added functionality for cross-modal experiments, where a special token in the prompt specifies the target modality for generation. When speech is selected as the input modality, the CMA experiment results are post-processed by aggregating the speech tokens that correspond to each text token. This mapping is based on the alignment obtained from the CTC-based forced aligner, ensuring results are directly comparable across modalities.

Figure 4.2 shows an example of the causal graphs and the factual prediction runs, and how the 3 different runs change the information flow to obtain an output. Similarly to [8] and as described in Section 3.3, experiments are conducted based on **clean run**

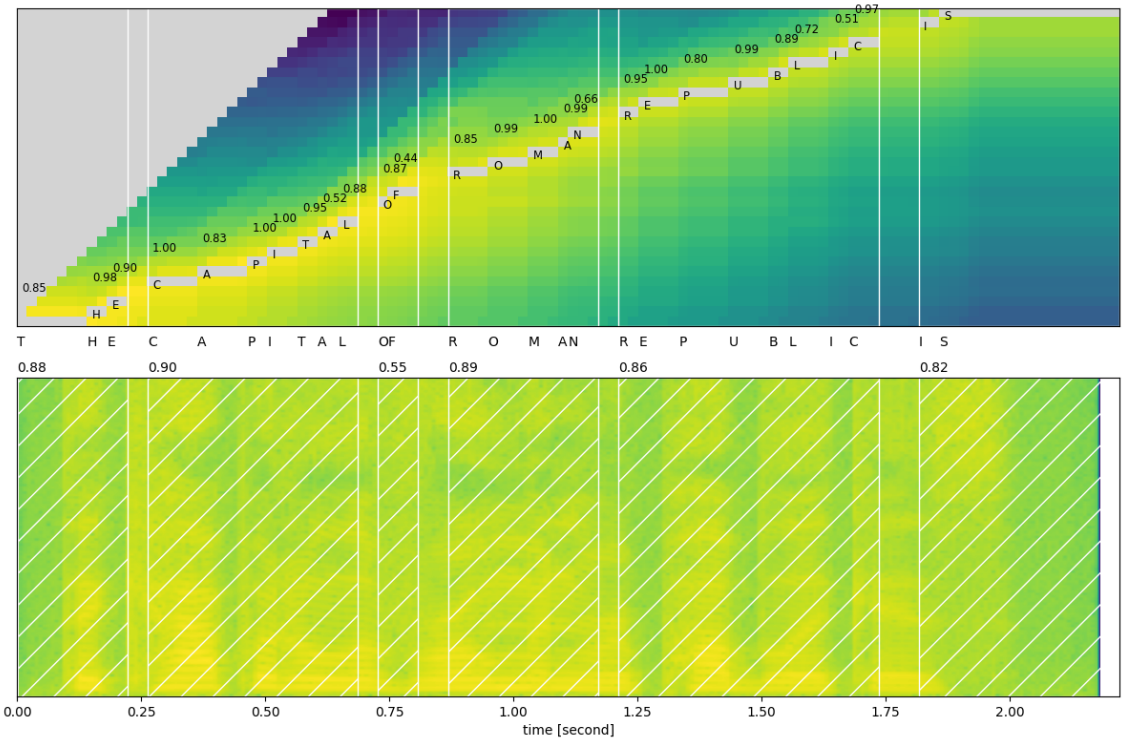


Figure 4.1: Results of the forced alignment for a speech utterance (transcript: "The capital of Roman Republic is"). The plot on top shows the trellis matrix, with the highlighted optimal path and score for each labeled letter; on the bottom, instead, we show the mel spectrogram of the spoken utterance, with the corresponding boundaries between each (spoken) text token.

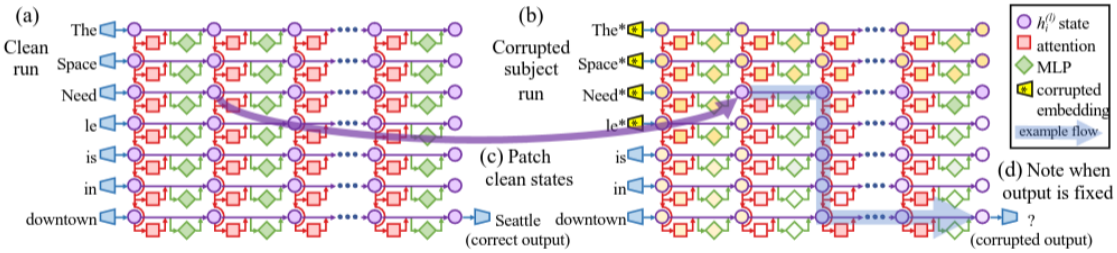


Figure 4.2: Example of the causal graph representation in the case of text-based models (from [8]).

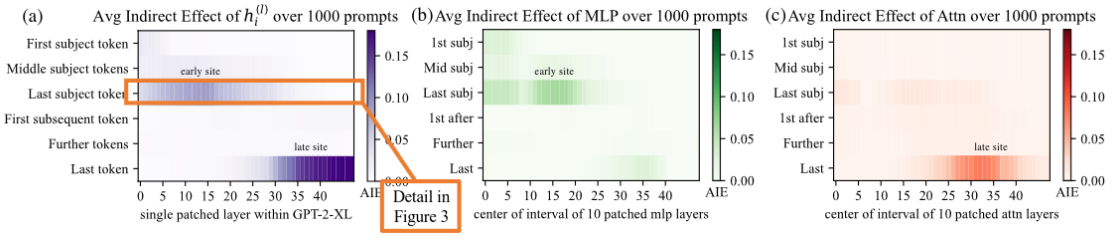


Figure 4.3: Causal traces plot, showing the **Average Indirect Effect** of individual model components over a sample of 1000 factual statements (AIE measured using probabilities; model used: GPT-2-XL, from [8]).

(Figure 4.2(a)), **corrupted run** (Figure 4.2(b)), and **corrupted-with-restoration run** (Figure 4.2(c)). The investigation is focused on the following component: single transformer layer (in particular, the hidden state value), and the decomposition into MLP and self-attention sub-layers.

Within this framework, the factual association study is carried out using *Multimodal-CausalTracer* through two primary experiments, designed to address the research questions presented in Section 1.2.

Experiment 1: Within-modality factual recall (Text → Text) In the first experiment, the text prompts are fed into the model, and the log probability of predicting the corresponding attribute is computed for each of the three runs. The IE is aggregated by the position of the token in the sentence: *first subject token, middle subject tokens, last subject token, first subsequent token, further tokens, and last token*, averaged over all prompts (AIE), and presented as log AIE for readability and comparison. The results are also visualized through attribution maps (similar to Figure 4.3), which are heatmaps that show the restoration effectiveness across tokens (or groups of tokens, as mentioned previously) and layers.

Experiment 2: Cross-modality factual recall (Speech → Text) The second experiment is similar, but it uses the synthesized version of the same set of prompts, which are converted to audio. Each utterance is encoded and discretized by `HuBERT`), and the resulting tokens are fed into the language model, where the CMA pipeline is applied as in the previous experiment. The causal traces of the speech tokens are

aggregated, as for text, by the corresponding text tokens using the cross-modal token mapping, which facilitates direct comparison and interpretation of causal influence across modalities.

Notably, these experiments were repeated with varying hyperparameters, to assess the effects on the mediator (subject speech features) outcomes. We explore the effect of three central hyperparameters.

- **noise**: the level of noise to apply to the embedding, to be corrupted.
- **samples**: how many different noise values need to be sampled for the corruption, with a magnitude defined from the previously described hyperparameter. For more stable reliable results, different "corrupted" and "corrupted-with-restoration" runs will be first performed with different noises sampled from a Gaussian distribution and then averaged.
- **window**: hyperparameter used when we decompose the effects into MLP and Attention lookups. It defines the range of layers over which restoration is performed and analyzed. Specifically, for each run, activations are averaged across a window of layers centered around the layer of interest ℓ_*, covering the range $[\ell_* - (\text{window}/2) - 1, \ldots, \ell_* + \text{window}/2]$. This allows for identifying whether the decisive contributions to factual recall are accumulated across multiple layers or localized to specific ones, avoiding potential negligible effects.

4.4 Experimental setup

We implement the code for *MultimodalCausalTracer* based on the Causal Tracing methodology developed by Meng et al. [8], adopting an Object-Oriented Programming (OOP) paradigm to enhance modularity and facilitate the necessary modifications for handling the discrete speech modality. For all Causal Mediation Analysis experiments presented in this chapter, the "samples" and "window" hyperparameters are set to 5, as they represent a trade-off between computational efficiency and obtaining meaningful causal effect results based on our empirical observations.

The prompt is corrupted by adding Gaussian noise to the subject s embeddings. To avoid Out-Of-Distribution (OOD) issues, which often occur when Gaussian noise is used as a corruption technique in intervention-based methods [42], the noise level is set to three times the standard deviation 3σ of the subject embeddings.

The average total causal effect (ATE) on the prediction of the object o given the corruption of the input (see the theory section 3.3), is presented as a percentage value.

The causal effects of specific model components (attention and MLP) per layer and token, the Average Indirect Effect (AIE), is expressed in log-probabilities. Although this metric is less directly interpretable, it facilitates the comparison between different probability scales. For example, this occur when the effects of different components or different modalities have significant differences. Higher AIE values correspond to layers and computational positions that have greater influence on factual

recall processes.

All tests and experimental procedures were conducted on the Alvis server cluster provided by NAISS. In particular, NVIDIA A100 Tensor Cores GPU (40GB), compute nodes were used.

5

Results

This chapter presents the outcomes of the Causal Tracing experiments, alongside an analysis and discussion of insights gained.

5.1 Initial factuality benchmark

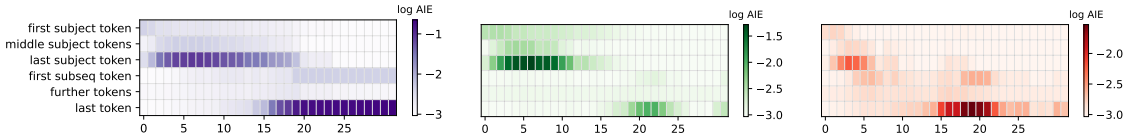
To assess generation performance in factual statement completion for both text and speech inputs, a factuality benchmark was run with the Known dataset. The main goal of the benchmark is to obtain a first glimpse of `Spirit LM`'s factual recall capability. The factual correctness criteria follow the same selection criteria used for the Known-t2t and Known-s2t datasets: a model generation is considered correct if the expected answer (the object o) is included in the model generation, specifically within the first five generated tokens. The results are shown in Table 5.1, presenting both accuracy scores and average token probability of o.

Benchmark Experiments	**Accuracy**	**Avg. Prob.**
Within-modality (Text → Text)	73.86%	37.77%
Cross-modality (Speech → Text)	11.00%	1.76%

Table 5.1: Factual benchmark performance of `Spirit LM` accuracy and average probability of the correct object for different input-output settings. A prediction is considered correct if the annotated object appears within the first five generated tokens.

Switching input modalities reveals a substantial performance gap in both accuracy and average probability of correct answers[1]: specifically, we observe accuracy and average probability percentage point drop of **62.86** and **36.01**, respectively. This indicates that the model exhibits lower accuracy and confidence when processing audio-based content compared to text inputs. Additionally, the performance disparity can be partially attributed to the limited capability of discrete speech tokens in capturing semantic nuances, as discussed in previous research [17], [19], [29]. Additionally, the inherent noise and variability found in the acoustic features of speech inputs [14] introduce further complications. These factors collectively inhibit the

[1]Some answers are represented by multiple tokens: in that case, the probability computed is the joint probability among them.



(a) single patched hidden state of a layer

(b) center of interval of 5 patched MLP layers

(c) center of interval of 5 patched attention layers

Figure 5.1: **Baseline experiment with the unimodal backbone text LLM**, to study factual recall in multimodal models. It shows the log-scaled (log-probability) Average Indirect Effect over the sample of 754 correct factual statements in Known-t2t on the `Spirit LM` backbone language model, `Llama2-7b` [63], decomposed by single hidden state, MLP, and attention. As expected, the results align with results in [8] in text-only LLMs.

model's capability in factual question answering, alongside broader challenges for Spoken Language Understanding (SLU) tasks.

From a mechanistic interpretability point of view, these findings suggest that speech-based factual retrieval may involve more distributed or less coherent computational pathways, with potentially weaker causal relationships between specific model components and factual knowledge access compared to text-based processing.

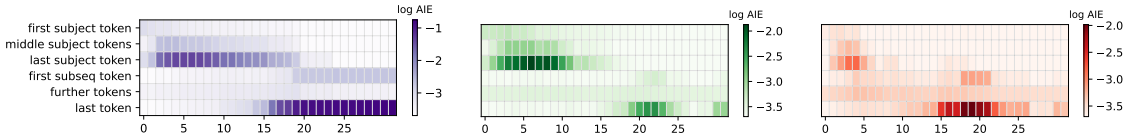
5.2 Within-modality experiments: text-to-text

We begin with withinmodality experiments (i.e., text-to-text) to verify factual recall circuits. We also want to see whether integrating speech modality to the LLM alters the factualrecall pathways. As a baseline for these and subsequent crossmodality experiments (Section 5.3), we perform CMA with the query prompts in Known-t2t and the original backbone model of Spirit LM, `Llama2 7b` [2] [63]. The observed ATE of this experiment is 51%. As expected from previous work by Meng et al. [8], we observe a causal signal (AIE) of comparable magnitude around the last subject token (see Meng's results in Figure 4.3), concentrated in mid-layer MLPs. Additionally, we find strong causal influence in the final token, particularly evident in the attention layers, as shown in Figure 5.1.

We then extend the Causal Tracing experiment to `Spirit LM` in the $T \rightarrow T$ scenario, computing the log Average Indirect Effect (AIE) over the same sample of 754 correct factual statements in Known-t2t. The causal traces are always shown by hidden states of each layer, as well as the effect decomposition into MLP layers and attention layers (Figure 5.2).

The overall causal effect fell slightly, with ATE = 49%. Although incorporating an additional modality (speech) into the generative model, our results remain consistent with the findings reported in [8] using a decoder-only unimodal model (Figure 4.3). Most notably, `Spirit LM` exhibits the same behavioral patterns as its original

[2]Link to the model checkpoint used: `https://huggingface.co/meta-llama/Llama-2-7b-hf`



(a) single patched hidden state of a layer

(b) center of interval of 5 patched MLP layers

(c) center of interval of 5 patched attention layers

Figure 5.2: **Text-to-Text log AIE over the samples in Known-t2t** of individual components of `Spirit LM`. The results align with unimodal backbone model behavior, suggesting the preservation of text factual recall capabilities even after fine-tuning on another modality (speech).

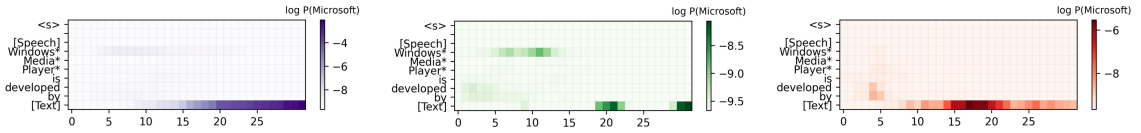
backbone (Figure 5.1) regarding causal signals around subject tokens across hidden states, MLP, and attention layers, with an almost identical AIE value scale.

The findings from this within-modality experiment verified that the text-processing pathways for factual recall, which can be identified in unimodal LLMs, remain almost intact in speech language models like `Spirit LM` even after fine-tuning for the speech modality, while also retaining the original capabilities of the backbone model. This can be attributed to the interleaving strategy used in `Spirit LM` to train the model across both modalities, enabling it to maintain text-only tasks capabilities while incorporating the new speech modality and ensuring reasonable model alignment simultaneously [18], [63].

5.3 Cross-modality experiments: speech-to-text

Moving beyond our analysis of text-only scenarios ($T \rightarrow T$), we now investigate the model's behavior when processing spoken language as input while maintaining text as the output format. Our goals are to measure how well the model performs with speech input and to analyze how the crossmodality setup affects causality. Furthermore, these findings will help us understand if the text modality impacts the speech modality, as well as how the model's behavior changes when transitioning between modalities from input to output. In *MultimodalCausalTracer*, the input is processed in two different stages.

- **Before the CMA experiment**. The raw audio is first converted into speech units using the speech encoder, followed by the translation into a sequence of pseudo-text tokens included in the generative model vocabulary. Finally, at the beginning and the end of the sequence, we added the modality declaration tokens for the input and output modality, respectively.
- **After the CMA experiment**. The results of the CMA experiment are post-processed to obtain the aggregated results of speech tokens corresponding to the text "spoken" ones. The aggregation is done by using the speech-to-text mapping obtained with the forced alignment between the spoken utterance of the prompt and its corresponding transcription. Note that the tokens shown for aggregation are the postaligned spoken text tokens. Because nonCTC



(a) single patched hidden state of a layer

(b) center of interval of 5 patched MLP layers

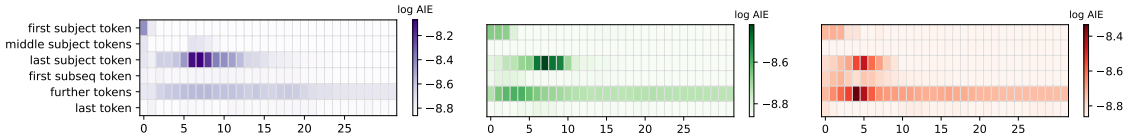
(c) center of interval of 5 patched attention layers

Figure 5.3: **Example of a single CMA experiment in a Speech-to-text scenario**, with related tokens aggregated by their spoken text tokens (prompt transcription: *Windows Media Player* is developed by ___). The attribution map reveals a strong trace at the late site driven by the modalitydeclaration token used to interleave modalities during `Spirit LM` training alongside a weaker signal at the early site on the subject token. This earlysite signal mirrors the behavior observed when the model processes textonly inputs.

characters (e.g., digits, special punctuation) are replaced during preprocessing, the total token count can differ from a straightforward textonly tokenization. Another reason behind this choice is to represent a set of text features with text that can be "spoken", rather than words or characters that have no meaning in audio (e.g., digits or specific punctuation characters).

We first present in Figure 5.3 an example of Causal Tracing results in this cross-modal setting with log probabilities, for all three target model component decompositions (hidden states from transformer layer, MLP, and attention subcomponents). The results are shown by aggregating the speech tokens with the criteria described; for more detailed plots of the same example, we report the causal trace for each single speech token in Appendix A.1.

One of the main insights from this example, as well as many other cases, is the presence of a slight early site trace on the subject level, similar to what has been found in the text modality; this alongside the main activations as a set of speech tokens in the extended raw results (Figure A.1-A.2-A.3), highlighting the granularity of such modality. Although the promising patterns in several of the factual statements of the dataset, we also report a *high level of uncertainty, given by the significantly smaller magnitude and the presence of slightly more traces in the plot compared to unimodal text results*. The behavior is caused by the lack of semantic-related information retention from the discrete speech tokens might not retain enough semantic-related information, as hinted by the benchmark result (Section 5.1) and the work in [29], where key factors behind the under-performance of discrete tokens have been studied. A last observation from this $S \rightarrow T$ attribution map example is the presence of a strong trace on the last token in all 3 effect decompositions, which corresponds to the modality declaration token used by `Spirit LM` to change modality (`[Text]`). During the training of `Spirit LM`, these special tokens were used to prefix both text and speech tokens within a mixed-modality sequence; therefore, we expected to have a significant causal impact on that site, as they serve as a clear indicator distinguishing and transitioning between modalities. However, since they do not carry meaningful information for studying factual recall circuits, we have chosen to exclude their effects from the attribution maps in the upcoming results. This deci-



(a) single patched hidden state of a layer (b) center of interval of 5 patched MLP layers (c) center of interval of 5 patched attention layers

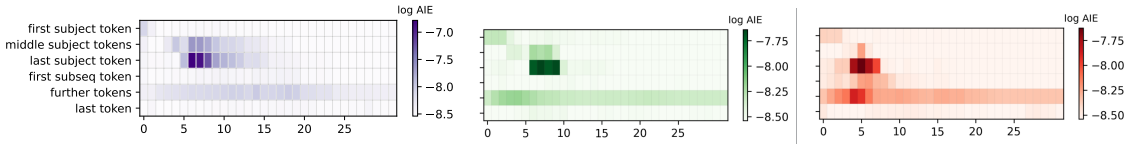
Figure 5.4: **Speech-to-Text Average Indirect Effect over the samples in Known-t2t**, per individual components of `Spirit LM`. As expected from the initial factual benchmark, we can notice a considerably smaller value magnitude, although a consistent causality signal around the last subject token as in the input text modality.

sion allows more relevant causal trace patterns to stand out in the plots generated from the CMA experiment.

As in the $T \rightarrow T$ scenario, we proceed by conducting the Causal Tracing experiment with `Spirit LM` in the cross-modal scenario, starting to calculate the AIE over the same Known-t2t dataset. The resulting attribution maps are shown in Figure 5.4.

As expected from the initial factual benchmark and a first inspection of a CMA experiment with speech as input, the overall Average Total Effect drops drastically, with a value in probability of around 1.9%. We can observe a similar outcome for the AIE calculation, with a more diffuse and lower-magnitude signal. Nevertheless, a noticeable effect around the subject tokens in the MLP and attention layers persists, suggesting a behavior comparable not only to within-modality scenarios but also, more broadly, to patterns seen when using text as input (Figure 5.2).

To investigate this resemblance to the $T \rightarrow T$ scenario, we repeat the CMA experiment under the same cross-modal conditions but using the Known-s2t subset. This second phase evaluates whether a more accurate speech-modality dataset preserves the previously observed pattern or instead reveals an audio-specific signature. Figure 5.5 displays the log-scaled AIE attribution maps, averaged over the 70 prompts in Known-s2t.



(a) single patched hidden state of a layer (b) center of interval of 5 patched MLP layers (c) center of interval of 5 patched attention layers

Figure 5.5: **Speech-to-Text Average Indirect Effect** (log-scale) of individual components of `Spirit LM`, over the 70 samples in the Known-s2t subset. Although a slightly reduced diffusion effect of the causal traces across multiple layers of the model, we can still observe the same AIE magnitude as in Figure 5.4, as well as the same causal effect pattern around the last subject token.

The Average Total Effect had a significant increase due to the more accurate dataset for the speech modality, reaching 16.9%; for the same reason, the causal traces appear less diffuse across multiple layers, for the three model components analyzed. However, the log Average Indirect Effect maintains its low-value scale, alongside the characteristic causality signal that persists around the final subject token in the MLP layers.

Finally, ATE for both modalities is summarized in Table 5.2

Experiment	**Input → Output**	**ATE (%)**
Baseline Backbone (LLaMA2-7b)	Text → Text	51.0
`Spirit LM` (Within-modality)	Text → Text	49.0
`Spirit LM` (Cross-modality, Known-t2t)	Speech → Text	1.9
`Spirit LM` (Cross-modality, Known-s2t)	Speech → Text	16.9

Table 5.2: **Average Total Effect (ATE)** in different experimental conditions. The ATE reflects the overall causal influence of targeted components on factual prediction performance, averaged across tokens and layers.

5.4 Discussion

Using the (`MultimodalCausalTracer`) pipeline, we demonstrate that causal pathways underlying factual recall can be traced in a Speech Language Model, both withinmodality (Text → Text) and crossmodally (Speech → Text).

Compared to previous work [8] and the causal traces observed in the backbone model, the first experiment ($T \rightarrow T$) demonstrates that the factual recall circuits of the text modality are preserved in a speech-text multimodal context such as the `Spirit LM` model.

The outcomes from the Causal Mediation Analyses applied to the cross-modal ($S \rightarrow T$) setting exhibit a very similar causal pattern to $T \rightarrow T$, albeit with much smaller AIE. We do not see evidence for modality-dependent factual association circuits. In other words, these results show that while Speech LMs can retrieve some knowledge from speech, their factual associations remain predominantly textdriven. Furthermore, these factual recall circuits are considerably more readily activated to text input, compared to speech.

This work also suggests novel research directions. Higher AIE values in the attribution maps correspond to layers and computational positions that have greater influence on factual recall processes. These components are therefore key targets for further mechanistic interpretability analysis of factual association retrieval circuits.

In conclusion, we found that for the `Spirit LM` model, there are not novel factual recall circuits for the speech-modality; rather, the text modality serves as a more structured and reliable trigger for recalling facts. This further suggests that the

speech-based fine-tuning of `Spirit LM` does not enable the model to fully utilize the fact-recalling mechanisms learned by the text-based, backbone Llama2 model.

However, it remains unclear whether this limited transfer of factual capabilities arises from(1) noise introduced by adding Gaussian perturbations to token embeddings, (2) imperfections in the speechtotext alignment, or (3) fundamental acoustic/semantic limitations of discrete speech tokens. We therefore encourage more research into factual recall of speech language models [14].

6

Conclusion

In this thesis, we used `Spirit LM` as a case study of SLMs to investigate how factual knowledge is encoded in Speech Language Models (SLMs) and to what extent results obtained in the text domain by Meng et al. ([8]) can be transferred to the speech domain. We proposed a speech-based version of the *Known* dataset and the *MultimodalCausalTracer*, an extension of LLM-based Causal Mediation Analysis [8], [44] to a multimodal setting to also handle discrete speech tokens, thereby providing insights into the behavior of SLMs and the interaction between text and speech. *MultimodalCausalTracer* employs a cross-modal token mapping created by a CTC-based forced aligner, which enables the localization of target words in a spoken utterance, the mapping of discrete speech tokens to text equivalents, and the visualization of CMA results across speech and text modalities.

Preliminary benchmarking on the *Known* dataset revealed a significant performance gap between the text-to-text (t2t) and speech-to-text (s2t) factual sentence completion tasks, with a 62.86% accuracy difference and a 36.01% probability gap in predicting the correct answer. This disparity can be attributed to the complexity of audio features and the inefficiency of discrete speech tokens in capturing semantic information [14], [17], [29]. Increased uncertainty in the s2t prediction resulted in weaker causal traces in the s2t CMA experiments.

We conducted two different CMA experiments using *Known-t2t*, a more accurate subset of the *Known* [8] dataset for Causal Tracing. The within-modal text-to-text CMA experiment demonstrated an identical Average Indirect effect (AIE) value compared with the unimodal backbone LM *Llama2-7b* [63], affirming that SLMs like `Spirit LM` preserve their original text capabilities after speech fine-tuning.

The cross-modal speech-to-text scenario experiments, on the other hand, revealed significantly weaker and more diffuse causal effects, as measured by AIE. However, small causal effects were still observed for model components related to the last subject token, similar to text-only models [8]. These results not only suggest that textually pretrained SLMs' factual recall mechanisms remain primarily text-driven, but they also tend to be modality invariant: speech integration has minimal influence on how factual knowledge is processed and retrieved.

In addition to the complexity of the speech modality, one of the major challenges when working with Speech Language Models is the computational load, since the higher speech token frequency compared to text entails significantly longer sequences.

For instance, the average input sequence length for speech was 63 tokens, compared to 10 for text. Given that the CMA experiment patched the hidden state activation for each layer and token, this significantly increased computational demand. Even with access to *NAISS* GPU clusters, inference time was a major bottleneck to the completion of the project.

6.1 Future Directions

We propose the following future next steps, also based on the limitations described in Section 1.3, aiming to further investigate factual encoding carries over speech, the behavior of such modality, and making our experiment pipeline more efficient.

- **Investigating different scenarios and input corruption techniques.** Due to time constraints, our study is limited to experiments with textual output; extending our experiment for $T \rightarrow S$ and $S \rightarrow S$ factual completion tasks could offer a deeper understanding of modality interactions, revealing layers with significant causal effect for SLU and speech-only inference tasks. Another promising direction involves modifying the corruption approach: either by applying speech-specific masking techniques like Speech-shaped noise (SSN) directly to the raw audio waveforms before tokenization, or by replacing the target tokens entirely with semantically related alternatives rather than applying Gaussian noise [9], [42].
- **Dataset selectivity**. We conducted our experiment on a single speech dataset. Conclusions about how modality interactions affects factual association may not generalize beyond the categories covered in *Known* (e.g., countries, people). The literature offers several other datasets that could provide valuable insights in this context: Spoken SQuAD [78] is a question answering dataset commonly used to assess SLU capabilities, while PopQA [79] contains 14,000 long-tail factual questions aimed at evaluating factual memorization in model parameters. For our purposes, a synthesized speech version of PopQA could be created. Exploring a variety of datasets may reveal new and interesting patterns in how SLMs encode factual information, rather than limiting findings to behaviors specific to certain domains (such as places or famous people).
- **Generalization across different models.** The concept of universality is a key goal in MI [80], and an improved understanding of SLMs necessitates the analysis of other models beyond `Spirit LM`. Notable open source alternatives with both speech and text capabilities and discrete feature embeddings include SpeechGPT and Moshi [19], [81]. Extending this analysis to these models could provide valuable insights into the shared circuits and features that enable cross-modal generalization and elucidate how these models utilize discrete speech tokens for seamless integration of speech and text functionalities.
- **Pipeline optimizations and alternative CMA implementations**. As mentioned above, long speech token sequences present a computational bottleneck in Causal Mediation Analysis experiments. Adapting CMA implementa-

tion pipeline to support batched inference and improved GPU parallelization enables more capacity for future experiments. Additionally, exploring more scalable alternatives to CMA, such as *Attribution Patching* (AtP*)* [82] can allow to uncover causal relationships with reduced resource demands. A final potential enhancement is combining multiple MI techniques, exemplified by *PatchScope* [83], which integrates causal interventions with *Logit Lens* to enhance expressivity and interpretability of model activations.

Bibliography

[1] T. B. Brown, B. Mann, N. Ryder, *et al.*, *Language models are few-shot learners*, 2020. arXiv: 2005.14165 [cs.CL]. [Online]. Available: https://arxiv.org/abs/2005.14165.

[2] F. Petroni, T. Rocktäschel, P. Lewis, *et al.*, *Language models as knowledge bases?* 2019. arXiv: 1909.01066 [cs.CL]. [Online]. Available: https://arxiv.org/abs/1909.01066.

[3] Y. Wang, M. Wang, M. A. Manzoor, *et al.*, *Factuality of large language models: A survey*, 2024. arXiv: 2402.02420 [cs.CL]. [Online]. Available: https://arxiv.org/abs/2402.02420.

[4] N. Kandpal, H. Deng, A. Roberts, E. Wallace, and C. Raffel, *Large language models struggle to learn long-tail knowledge*, 2023. arXiv: 2211.08411 [cs.CL]. [Online]. Available: https://arxiv.org/abs/2211.08411.

[5] N. Elhage, N. Nanda, C. Olsson, *et al.*, "A mathematical framework for transformer circuits," *Transformer Circuits Thread*, 2021, https://transformer-circuits.pub/2021/fra

[6] D. Rai, Y. Zhou, S. Feng, A. Saparov, and Z. Yao, *A practical review of mechanistic interpretability for transformer-based language models*, 2025. arXiv: 2407.02646 [cs.AI]. [Online]. Available: https://arxiv.org/abs/2407.02646.

[7] M. Geva, R. Schuster, J. Berant, and O. Levy, *Transformer feed-forward layers are key-value memories*, 2021. arXiv: 2012.14913 [cs.CL]. [Online]. Available: https://arxiv.org/abs/2012.14913.

[8] K. Meng, D. Bau, A. Andonian, and Y. Belinkov, *Locating and editing factual associations in gpt*, 2023. arXiv: 2202.05262 [cs.CL]. [Online]. Available: https://arxiv.org/abs/2202.05262.

[9] J. Vig, S. Gehrmann, Y. Belinkov, *et al.*, *Causal mediation analysis for interpreting neural nlp: The case of gender bias*, 2020. arXiv: 2004.12265 [cs.CL]. [Online]. Available: https://arxiv.org/abs/2004.12265.

[10] M. Geva, J. Bastings, K. Filippova, and A. Globerson, *Dissecting recall of factual associations in auto-regressive language models*, 2023. arXiv: 2304.14767 [cs.CL]. [Online]. Available: https://arxiv.org/abs/2304.14767.

[11] E. Mitchell, C. Lin, A. Bosselut, C. Finn, and C. D. Manning, *Fast model editing at scale*, 2022. arXiv: 2110.11309 [cs.LG]. [Online]. Available: https://arxiv.org/abs/2110.11309.

[12] K. Meng, A. S. Sharma, A. Andonian, Y. Belinkov, and D. Bau, *Mass-editing memory in a transformer*, 2023. arXiv: 2210.07229 [cs.CL]. [Online]. Available: https://arxiv.org/abs/2210.07229.

[13] S. Yin, C. Fu, S. Zhao, *et al.*, "A survey on multimodal large language models," *National Science Review*, vol. 11, no. 12, Nov. 2024, ISSN: 2053-714X. DOI: 10.1093/nsr/nwae403. [Online]. Available: http://dx.doi.org/10.1093/nsr/nwae403.

[14] J. Peng, Y. Wang, Y. Xi, X. Li, X. Zhang, and K. Yu, *A survey on speech large language models*, 2025. arXiv: 2410.18908 [eess.AS]. [Online]. Available: https://arxiv.org/abs/2410.18908.

[15] W.-N. Hsu, B. Bolte, Y.-H. H. Tsai, K. Lakhotia, R. Salakhutdinov, and A. Mohamed, *Hubert: Self-supervised speech representation learning by masked prediction of hidden units*, 2021. arXiv: 2106.07447 [cs.CL]. [Online]. Available: https://arxiv.org/abs/2106.07447.

[16] A. Baevski, H. Zhou, A. Mohamed, and M. Auli, *Wav2vec 2.0: A framework for self-supervised learning of speech representations*, 2020. arXiv: 2006.11477 [cs.CL]. [Online]. Available: https://arxiv.org/abs/2006.11477.

[17] K. Lakhotia, E. Kharitonov, W.-N. Hsu, *et al.*, *Generative spoken language modeling from raw audio*, 2021. arXiv: 2102.01192 [cs.CL]. [Online]. Available: https://arxiv.org/abs/2102.01192.

[18] T. A. Nguyen, B. Muller, B. Yu, *et al.*, *Spirit lm: Interleaved spoken and written language model*, 2024. arXiv: 2402.05755 [cs.CL]. [Online]. Available: https://arxiv.org/abs/2402.05755.

[19] D. Zhang, S. Li, X. Zhang, *et al.*, *Speechgpt: Empowering large language models with intrinsic cross-modal conversational abilities*, 2023. arXiv: 2305.11000 [cs.CL]. [Online]. Available: https://arxiv.org/abs/2305.11000.

[20] P. K. Rubenstein, C. Asawaroengchai, D. D. Nguyen, *et al.*, *Audiopalm: A large language model that can speak and listen*, 2023. arXiv: 2306.12925 [cs.CL]. [Online]. Available: https://arxiv.org/abs/2306.12925.

[21] T. A. Nguyen, E. Kharitonov, J. Copet, *et al.*, *Generative spoken dialogue language modeling*, 2022. arXiv: 2203.16502 [cs.CL]. [Online]. Available: https://arxiv.org/abs/2203.16502.

[22] Z. Lin, S. Basu, M. Beigi, *et al.*, *A survey on mechanistic interpretability for multi-modal foundation models*, 2025. arXiv: 2502.17516 [cs.LG]. [Online]. Available: https://arxiv.org/abs/2502.17516.

[23] S. Basu, M. Grayson, C. Morrison, B. Nushi, S. Feizi, and D. Massiceti, *Understanding information storage and transfer in multi-modal large language models*, 2024. arXiv: 2406.04236 [cs.CV]. [Online]. Available: https://arxiv.org/abs/2406.04236.

[24] A. Pasad, C.-M. Chien, S. Settle, and K. Livescu, *What do self-supervised speech models know about words?* 2024. arXiv: 2307.00162 [cs.CL]. [Online]. Available: https://arxiv.org/abs/2307.00162.

[25] G. Shen, M. Watkins, A. Alishahi, A. Bisazza, and G. Chrupaa, "Encoding of lexical tone in self-supervised models of spoken language," in *Proceedings of the 2024 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies (Volume 1: Long Papers)*, K. Duh, H. Gomez, and S. Bethard, Eds., Mexico City, Mexico: Association for Computational Linguistics, Jun. 2024, pp. 4250–4261. DOI: 10.18653/v1/

2024.naacl-long.239. [Online]. Available: https://aclanthology.org/2024.naacl-long.239/.

[26] K. Ding, M. Chetty, A. Hoshyar, T. Bhattacharya, and B. Klein, "Speech based detection of alzheimers disease: A survey of ai techniques, datasets and challenges," *Artificial Intelligence Review*, vol. 57, Oct. 2024. DOI: 10.1007/s10462-024-10961-6.

[27] E. Loweimi, M. Qian, K. Knill, and M. Gales, *Speaker retrieval in the wild: Challenges, effectiveness and robustness*, 2025. arXiv: 2504.18950 [cs.SD]. [Online]. Available: https://arxiv.org/abs/2504.18950.

[28] L. Mai and J. Carson-Berndsen, *Real-time textless dialogue generation*, 2025. arXiv: 2501.04877 [cs.CL]. [Online]. Available: https://arxiv.org/abs/2501.04877.

[29] D. Wang, M. Cui, D. Yang, X. Chen, and H. Meng, *A comparative study of discrete speech tokens for semantic-related tasks with large language models*, 2024. arXiv: 2411.08742 [cs.CL]. [Online]. Available: https://arxiv.org/abs/2411.08742.

[30] S. Arora, K.-W. Chang, C.-M. Chien, *et al.*, *On the landscape of spoken language models: A comprehensive survey*, 2025. arXiv: 2504.08528 [cs.CL]. [Online]. Available: https://arxiv.org/abs/2504.08528.

[31] Y. Gao, Y. Xiong, X. Gao, *et al.*, *Retrieval-augmented generation for large language models: A survey*, 2024. arXiv: 2312.10997 [cs.CL]. [Online]. Available: https://arxiv.org/abs/2312.10997.

[32] M. Farahani and R. Johansson, *Deciphering the interplay of parametric and non-parametric memory in retrieval-augmented language models*, 2024. arXiv: 2410.05162 [cs.CL]. [Online]. Available: https://arxiv.org/abs/2410.05162.

[33] K. Wang, A. Variengien, A. Conmy, B. Shlegeris, and J. Steinhardt, *Interpretability in the wild: A circuit for indirect object identification in gpt-2 small*, 2022. arXiv: 2211.00593 [cs.LG]. [Online]. Available: https://arxiv.org/abs/2211.00593.

[34] L. Bürger, F. A. Hamprecht, and B. Nadler, *Truth is universal: Robust detection of lies in llms*, 2024. arXiv: 2407.12831 [cs.CL]. [Online]. Available: https://arxiv.org/abs/2407.12831.

[35] N. Saphra and S. Wiegreffe, *Mechanistic?* 2024. arXiv: 2410.09087 [cs.AI]. [Online]. Available: https://arxiv.org/abs/2410.09087.

[36] C. Olah, N. Cammarata, L. Schubert, G. Goh, M. Petrov, and S. Carter, "Zoom in: An introduction to circuits," *Distill*, 2020. DOI: 10.23915/distill.00024.001. [Online]. Available: https://distill.pub/2020/circuits/zoom-in/.

[37] C. Szegedy, W. Liu, Y. Jia, *et al.*, *Going deeper with convolutions*, 2014. arXiv: 1409.4842 [cs.CV]. [Online]. Available: https://arxiv.org/abs/1409.4842.

[38] A. Templeton, T. Conerly, J. Marcus, *et al.*, "Scaling monosemanticity: Extracting interpretable features from claude 3 sonnet," *Transformer Circuits Thread*, 2024. [Online]. Available: https://transformer-circuits.pub/2024/scaling-monosemanticity/index.html.

[39] A. Stolfo, Y. Belinkov, and M. Sachan, *A mechanistic interpretation of arithmetic reasoning in language models using causal mediation analysis*, 2023. arXiv: 2305.15054 [cs.CL]. [Online]. Available: https://arxiv.org/abs/2305.15054.

[40] S. Jastrzbski, D. Arpit, N. Ballas, V. Verma, T. Che, and Y. Bengio, *Residual connections encourage iterative inference*, 2018. arXiv: 1710.04773 [cs.CV]. [Online]. Available: https://arxiv.org/abs/1710.04773.

[41] nostalgebraist, *Interpreting GPT: The logit lens*, AI Alignment Forum, 2020. [Online]. Available: https://www.lesswrong.com/posts/AcKRB8wDpdaN6v6ru/interpreting-gpt-the-logit-lens.

[42] F. Zhang and N. Nanda, *Towards best practices of activation patching in language models: Metrics and methods*, 2024. arXiv: 2309.16042 [cs.LG]. [Online]. Available: https://arxiv.org/abs/2309.16042.

[43] Z. Yun, Y. Chen, B. Olshausen, and Y. LeCun, "Transformer visualization via dictionary learning: Contextualized embedding as a linear superposition of transformer factors," in *Proceedings of Deep Learning Inside Out (DeeLIO): The 2nd Workshop on Knowledge Extraction and Integration for Deep Learning Architectures*, E. Agirre, M. Apidianaki, and I. Vuli, Eds., Online: Association for Computational Linguistics, Jun. 2021, pp. 1–10. DOI: 10.18653/v1/2021.deelio-1.1. [Online]. Available: https://aclanthology.org/2021.deelio-1.1/.

[44] J. Pearl, *Direct and indirect effects*, 2013. arXiv: 1301.2300 [cs.AI]. [Online]. Available: https://arxiv.org/abs/1301.2300.

[45] Z. Weng, Z. Gao, J. Andrews, and J. Zhao, *Images speak louder than words: Understanding and mitigating bias in vision-language model from a causal mediation perspective*, 2024. arXiv: 2407.02814 [cs.AI]. [Online]. Available: https://arxiv.org/abs/2407.02814.

[46] H. Liu, C. Li, Q. Wu, and Y. J. Lee, *Visual instruction tuning*, 2023. arXiv: 2304.08485 [cs.CV]. [Online]. Available: https://arxiv.org/abs/2304.08485.

[47] OpenAI. "Chatgpt can now see, hear, and speak." Accessed: 2025-05-13. (Sep. 2023), [Online]. Available: https://openai.com/index/chatgpt-can-now-see-hear-and-speak/.

[48] A. Radford, J. W. Kim, T. Xu, G. Brockman, C. McLeavey, and I. Sutskever, "Robust speech recognition via large-scale weak supervision," in *Proceedings of the 40th International Conference on Machine Learning*, ser. ICML'23, Honolulu, Hawaii, USA: JMLR.org, 2023.

[49] OpenAI. "Hello gpt-4o." Accessed: 2025-05-13. (May 2024), [Online]. Available: https://openai.com/index/hello-gpt-4o/.

[50] E. Kharitonov, J. Copet, K. Lakhotia, *et al.*, *Textless-lib: A library for textless spoken language processing*, 2022. arXiv: 2202.07359 [cs.CL]. [Online]. Available: https://arxiv.org/abs/2202.07359.

[51] A. Polyak, Y. Adi, J. Copet, *et al.*, *Speech resynthesis from discrete disentangled self-supervised representations*, 2021. arXiv: 2104.00355 [cs.SD]. [Online]. Available: https://arxiv.org/abs/2104.00355.

[52] A. Gulati, J. Qin, C.-C. Chiu, *et al.*, "Conformer: Convolution-augmented transformer for speech recognition," in *Interspeech 2020*, 2020, pp. 5036–5040. DOI: 10.21437/Interspeech.2020-3015.

[53] S. Deshmukh, B. Elizalde, R. Singh, and H. Wang, *Pengi: An audio language model for audio tasks*, 2024. arXiv: 2305.11834 [eess.AS]. [Online]. Available: https://arxiv.org/abs/2305.11834.

[54] Y. Chu, J. Xu, X. Zhou, *et al.*, *Qwen-audio: Advancing universal audio understanding via unified large-scale audio-language models*, 2023. arXiv: 2311.07919 [eess.AS]. [Online]. Available: https://arxiv.org/abs/2311.07919.

[55] D. Jurafsky and J. H. Martin, *Speech and Language Processing: An Introduction to Natural Language Processing, Computational Linguistics, and Speech Recognition with Language Models*, 3rd. Stanford, 2025, Online manuscript released January 12, 2025. [Online]. Available: https://web.stanford.edu/~jurafsky/slp3/.

[56] P. Gage, "A new algorithm for data compression," *C Users J.*, vol. 12, no. 2, pp. 23–38, Feb. 1994, ISSN: 0898-9788.

[57] K. Cho, B. van Merrienboer, C. Gulcehre, *et al.*, *Learning phrase representations using rnn encoder-decoder for statistical machine translation*, 2014. arXiv: 1406.1078 [cs.CL]. [Online]. Available: https://arxiv.org/abs/1406.1078.

[58] I. Sutskever, O. Vinyals, and Q. V. Le, *Sequence to sequence learning with neural networks*, 2014. arXiv: 1409.3215 [cs.CL]. [Online]. Available: https://arxiv.org/abs/1409.3215.

[59] D. Bahdanau, K. Cho, and Y. Bengio, *Neural machine translation by jointly learning to align and translate*, 2016. arXiv: 1409.0473 [cs.CL]. [Online]. Available: https://arxiv.org/abs/1409.0473.

[60] A. Vaswani, N. Shazeer, N. Parmar, *et al.*, *Attention is all you need*, 2017. arXiv: 1706.03762 [cs.CL]. [Online]. Available: https://arxiv.org/abs/1706.03762.

[61] K. He, X. Zhang, S. Ren, and J. Sun, *Deep residual learning for image recognition*, 2015. arXiv: 1512.03385 [cs.CV]. [Online]. Available: https://arxiv.org/abs/1512.03385.

[62] M. Hassid, T. Remez, T. A. Nguyen, *et al.*, *Textually pretrained speech language models*, 2024. arXiv: 2305.13009 [cs.CL]. [Online]. Available: https://arxiv.org/abs/2305.13009.

[63] H. Touvron, L. Martin, K. Stone, *et al.*, *Llama 2: Open foundation and fine-tuned chat models*, 2023. arXiv: 2307.09288 [cs.CL]. [Online]. Available: https://arxiv.org/abs/2307.09288.

[64] Y. Guo, Z. Li, H. Wang, *et al.*, *Recent advances in discrete speech tokens: A review*, 2025. arXiv: 2502.06490 [eess.AS]. [Online]. Available: https://arxiv.org/abs/2502.06490.

[65] A. van den Oord, O. Vinyals, and K. Kavukcuoglu, *Neural discrete representation learning*, 2018. arXiv: 1711.00937 [cs.LG]. [Online]. Available: https://arxiv.org/abs/1711.00937.

[66] C. M. Bishop, *Pattern Recognition and Machine Learning*. New York: Springer, 2006, ISBN: 978-0-387-31073-2.

[67] J. Devlin, M.-W. Chang, K. Lee, and K. Toutanova, "BERT: Pre-training of deep bidirectional transformers for language understanding," in *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)*, J. Burstein, C. Doran, and T. Solorio, Eds., Minneapolis, Minnesota: Association for Computational Linguistics, Jun. 2019, pp. 4171–4186. DOI: 10.18653/v1/N19-1423. [Online]. Available: https://aclanthology.org/N19-1423/.

[68] V. Panayotov, G. Chen, D. Povey, and S. Khudanpur, "Librispeech: An asr corpus based on public domain audio books," in *2015 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 2015, pp. 5206–5210. DOI: 10.1109/ICASSP.2015.7178964.

[69] Y. Jia *et al.*, "Direct speech-to-speech translation with a sequence-to-sequence model," in *Interspeech*, 2019.

[70] J. M. Peña, *Alternative measures of direct and indirect effects*, 2023. arXiv: 2306.01292 [stat.ME]. [Online]. Available: https://arxiv.org/abs/2306.01292.

[71] Y. Elazar, N. Kassner, S. Ravfogel, *et al.*, *Measuring and improving consistency in pretrained language models*, 2021. arXiv: 2102.01017 [cs.CL]. [Online]. Available: https://arxiv.org/abs/2102.01017.

[72] W. Zhao, X. Yu, and Z. Qin, *Melotts: High-quality multi-lingual multi-accent text-to-speech*, 2023. [Online]. Available: https://github.com/myshell-ai/MeloTTS.

[73] J. Kim, J. Kong, and J. Son, *Conditional variational autoencoder with adversarial learning for end-to-end text-to-speech*, 2021. arXiv: 2106.06103 [cs.SD]. [Online]. Available: https://arxiv.org/abs/2106.06103.

[74] J. Kong, J. Park, B. Kim, J. Kim, D. Kong, and S. Kim, *Vits2: Improving quality and efficiency of single-stage text-to-speech with adversarial learning and architecture design*, 2023. arXiv: 2307.16430 [cs.SD]. [Online]. Available: https://arxiv.org/abs/2307.16430.

[75] A. Graves, S. Fernández, F. Gomez, and J. Schmidhuber, "Connectionist temporal classification: Labelling unsegmented sequence data with recurrent neural networks," in *Proceedings of the 23rd International Conference on Machine Learning*, ser. ICML '06, Pittsburgh, Pennsylvania, USA: Association for Computing Machinery, 2006, pp. 369–376, ISBN: 1595933832. DOI: 10.1145/1143844.1143891. [Online]. Available: https://doi.org/10.1145/1143844.1143891.

[76] L. Kürzinger, D. Winkelbauer, L. Li, T. Watzel, and G. Rigoll, "Ctc-segmentation of large corpora for german end-to-end speech recognition," in *Speech and Computer*. Springer International Publishing, 2020, pp. 267–278, ISBN: 9783030602765. DOI: 10.1007/978-3-030-60276-5_27. [Online]. Available: http://dx.doi.org/10.1007/978-3-030-60276-5_27.

[77] V. Pratap, A. Tjandra, B. Shi, *et al.*, *Scaling speech technology to 1,000+ languages*, 2023. arXiv: 2305.13516 [cs.CL]. [Online]. Available: https://arxiv.org/abs/2305.13516.

[78] C.-H. Li, S.-L. Wu, C.-L. Liu, and H.-y. Lee, *Spoken squad: A study of mitigating the impact of speech recognition errors on listening comprehension*, 2018. arXiv: 1804.00320 [cs.CL]. [Online]. Available: https://arxiv.org/abs/1804.00320.

[79] A. Mallen, A. Asai, V. Zhong, R. Das, D. Khashabi, and H. Hajishirzi, *When not to trust language models: Investigating effectiveness of parametric and non-parametric memories*, 2023. arXiv: 2212.10511 [cs.CL]. [Online]. Available: https://arxiv.org/abs/2212.10511.

[80] C. Olah, N. Cammarata, L. Schubert, G. Goh, M. Petrov, and S. Carter, "Zoom in: An introduction to circuits," *Distill*, 2020, https://distill.pub/2020/circuits/zoom-in. DOI: 10.23915/distill.00024.001.

[81] A. Défossez, L. Mazaré, M. Orsini, *et al.*, *Moshi: A speech-text foundation model for real-time dialogue*, 2024. arXiv: 2410.00037 [eess.AS]. [Online]. Available: https://arxiv.org/abs/2410.00037.

[82] J. Kramár, T. Lieberum, R. Shah, and N. Nanda, *Atp*: An efficient and scalable method for localizing llm behaviour to components*, 2024. arXiv: 2403.00745 [cs.LG]. [Online]. Available: https://arxiv.org/abs/2403.00745.

[83] A. Ghandeharioun, A. Caciularu, A. Pearce, L. Dixon, and M. Geva, *Patchscopes: A unifying framework for inspecting hidden representations of language models*, 2024. arXiv: 2401.06102 [cs.CL]. [Online]. Available: https://arxiv.org/abs/2401.06102.

A

1: extended results plot for speech-to-text Causal mediation Analysis

Here we present the extended results described in the preliminary results for a Speech-to-Text context, by showing the probability for each single speech token. The purpose is to show that, even with low probability, there are multiple cases where multiple contiguous subject speech tokens are highlighted when this part of the sentence is relatively important to restore the original output of the model. The prompt transcription is the same as in Figure 5.3.

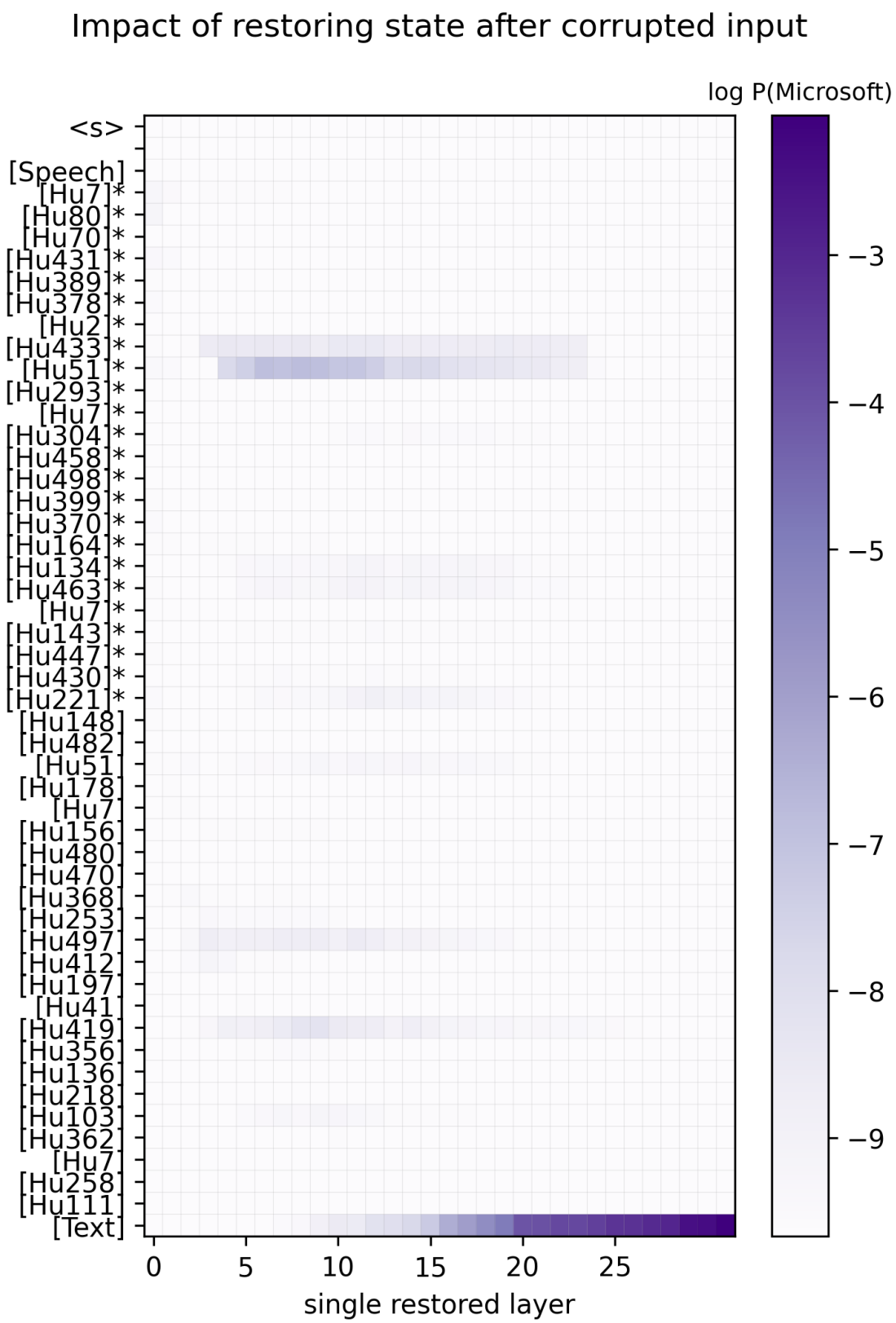


Figure A.1: Impact of restoring state after the corrupted input.

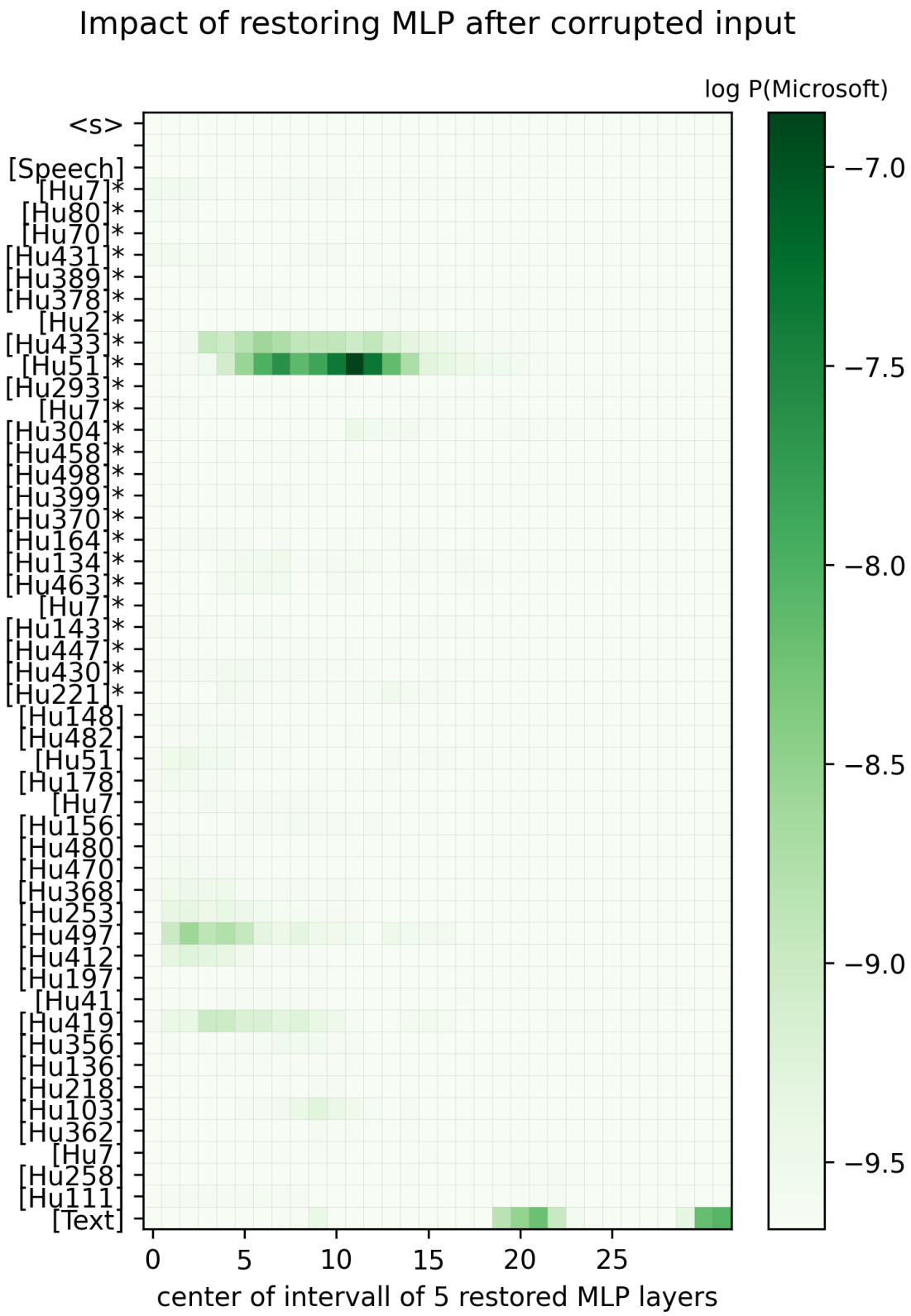


Figure A.2: Impact of restoring MLP after the corrupted input.

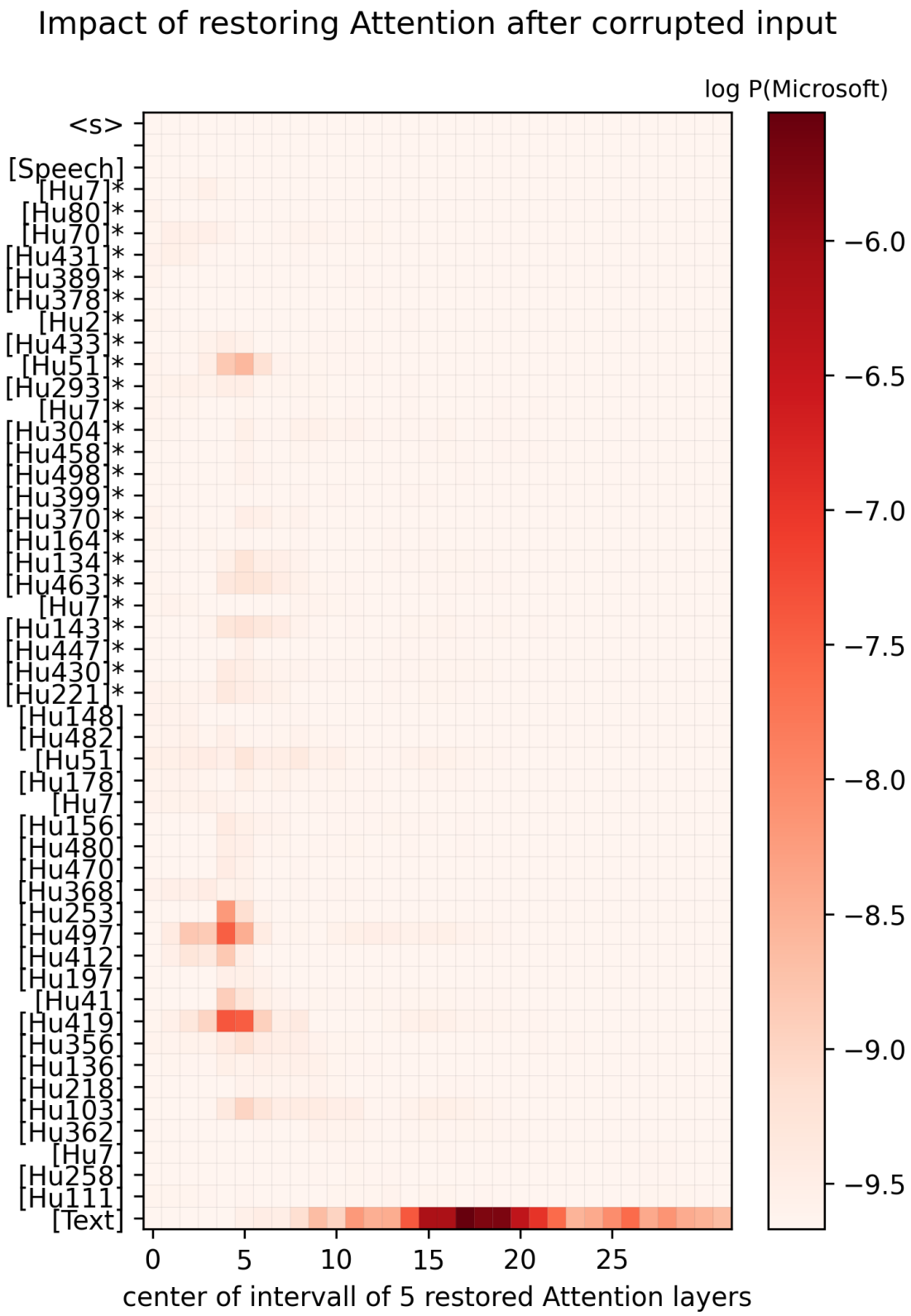


Figure A.3: Impact of restoring Attn after the corrupted input.