Thesis for The Degree of Licentiate of Engineering

Context Is Complex

From Dialogue Histories to Knowledge Integration in NLP

Mehrdad Farahani

Department of Computer Science and Engineering
Chalmers University of Technology and University of Gothenburg
Gothenburg, Sweden, 2024

Context Is Complex
From Dialogue Histories to Knowledge Integration in NLP

Mehrdad Farahani

ISSN 1652-876X

Department of Computer Science and Engineering
Division of Data Science and AI
Some Research Group
Chalmers University of Technology and University of Gothenburg
SE-412 96 Göteborg,
Sweden
Phone: +46(0)31 772 1000

Printed by Chalmers Digitaltryck,
Gothenburg, Sweden 2024.

To my parents, Shahrzad and Mohammad
تقدیم به مادرم شهرزاد و پدرم محمد

Context Is Complex

From Dialogue Histories to Knowledge Integration in NLP

Mehrdad Farahani

Department of Computer Science and Engineering
Chalmers University of Technology and University of Gothenburg

Abstract

Today, Language Models (LMs) have shown impressive results in many Natural Language Processing (NLP) tasks. Recent advancements in scaling up the language models (large language models) suggest they can be relatively reliable tools to assist humans. However, do these models truly "understand" context in the sense of knowing it? To answer this question, we need to understand the definition of context. Although there is no universal definition, making it a complex concept, in NLP, it can take many forms, including exchanged conversations, external knowledge, linguistic structure, and more.

In this thesis, we address the complexity of context from an LM point of view in two central research questions: (1) How can LMs better incorporate dialogue histories and personas in conversational AI tasks? (2) How do LMs balance internal and external knowledge, and when do they prioritize one over the other? We present two studies to address these questions from different perspectives. First, we introduce a new training strategy to encourage the model to consider context in its responses. Then, we apply dissection methods, such as causal mediation analysis, to explore the internal mechanisms of LMs and understand how they interact with context.

Our findings from the first study demonstrate that introducing a relevant training strategy can slightly improve the model's overall performance. However, it does not indicate that the model consistently considers context in its responses. In contrast, the second study provides a clearer understanding of how the model interacts with context. It shows that the model first evaluates the context to ensure its relevance and, if deemed appropriate, incorporates it into its responses. In such cases, the model tends to rely heavily on the context, often ignoring its internal knowledge.

Keywords

model analysis, causal mediation analysis, retrieval-augmented models, context

List of Publications

Appended publications

This thesis is based on the following publications:

[**Paper I**] **Mehrdad Farahani**, Richard Johansson, *An Empirical Study of Multitask Learning to Improve Open Domain Dialogue Systems*
Proceedings of the 24th Nordic Conference on Computational Linguistics (NoDaLiDa) (May 2023), 347-357.

[**Paper II**] **Mehrdad Farahani**, Richard Johansson, *Deciphering the Interplay of Parametric and Non-parametric Memory in Retrieval-augmented Language Models*
Proceedings of the 2024 Conference on Empirical Methods in Natural Language Processing (November 2024), 16966–16977.

Other publications

The following publications were published during my PhD studies, or are currently in submission/under revision. However, they are not appended to this thesis, due to contents overlapping that of appended publications or contents not related to the thesis.

[**a**] Peter Samoaa, **Mehrdad Farahani**, Antonio Longa, Philipp Leitner, Morteza Haghir Chehreghani, *Analysing the Behaviour of Tree-Based Neural Networks in Regression Tasks*
Submitted to IEEE Transactions on Neural Networks and Learning Systems.

Acknowledgment

I would like to thank my main supervisor, Richard Johansson, for his limitless and kind support and insightful guidance. I also sincerely thank my co-supervisor, Gabriel Skantze, and my examiner, Aarne Ranta, for their valuable feedback and encouragement–special thanks to Ashkan for our spontaneous talks.

Thanks to my colleagues in the DSAI division–Farzaneh, Yossra, Juan, Peter, Linus, Jack, Denitsa, Lovisa, Lena, Newton, Adam, Anton, David, Hampus, Markus, Kelsey, Mohammad, Shirin, Nicolas, Alec, Daniel, Filip, Fazeleh, Dag, Kolbjörn, Gerardo, Simon, Rocio, Birgit, Morteza, Peter, Graham, Fredrik, Selpi, Devdatt, Ola and rest of the division, faculty, postdocs and administrative staff.

Special thanks to my friends in the WASP program–Hoomaan Maskan, Saeed Razavi, Amandine Caut, Shivam Mehta, and Livia Qian, for our productive discussions and the enjoyable moments we shared

A shoutout to Milad, Mohammad Ali, Siavash, Firooz, Arman, Seyed Mohammad Mehdi, Arsham, Samira, and Ehsan for making my PhD journey more than just academic and personal growth.

My heartfelt gratitude goes to my parents; no words fully express my appreciation for your sacrifices and unconditional love. To my little sister, Shayna, thank you for all our wonderful memories and always being my partner in laughter and adventure. To her fiancé, Pejman, you are more than family; you are a true friend. And to my soon-to-arrive niece, I cannot wait for our adventures together!

This work was partially supported by the Wallenberg AI, Autonomous Systems, and Software Program (WASP), funded by the Knut and Alice Wallenberg Foundation. Most of the computations for this work were made possible by resources provided by the Swedish National Infrastructure for Computing (SNIC), partially funded by the Swedish Research Council through grant agreement no. 2022-221003.

Mehrdad Farahani
Göteborg, Sweden
November, 2024

Contents

Part I

Summary

Chapter 1

Introduction

Context is a term widely used in Natural Language Processing (NLP). Although there is no universal definition for context, in natural language, it is defined as a fundamental element in human communication with various factors that shape the meaning of an utterance or interaction (Airenti et al., 2017; Y. Huang, 2015). But what does it mean in practice in NLP? *context* can refer to multiple facets, such as the dialogue history in conversation, external knowledge required for answering questions, or even the linguistic structure surrounding an utterance. Indeed, this multifaceted behavior of context makes it complex to study.

Recent developments in building complex Language Models (LMs) or even different scales such as Large Language Models (LLMs) have shown promising improvement in NLP tasks. This has led the NLP communities to claim that these models comprehend the natural language and, by extension, context. However, as highlighted in the position paper by Bender and Koller (2020), understanding meaning cannot be achieved only through analyzing forms[1] in LMs. But what do these models *know* and *how* they process information? Still remain important questions.

Disclaimer: In this thesis, we are not concerned with considering whether LMs "understand" context in the way humans do. Instead, the focus is on investigating what LMs "know" and how they process and use contextual information. Whenever the term "understand" or "understanding context" is used in this thesis, it refers to the computational and representational processes within the model, not to human-like comprehension or cognition.

This thesis explores the context as a multifaceted entity in different scopes. In dialogue systems (Chapter 2, Section 2.5), context allows LM to maintain coherence, align with user intent, and incorporate relevant prior exchanges. For instance, producing paradoxical responses with the personas or previous conversations in an LM highlights the gaps in its contextual comprehension. In

[1] Form refers to any visible or measurable expression of language, like written marks, digital text (pixels or bytes), or speech movements.

the question-answering system (Chapter 2, Section 2.4), context often involves integrating external knowledge like Wikipedia to provide accurate and up-to-date responses. To address these contextualized challenges, we investigate two main research questions in this thesis:

1. **How can LMs better incorporate dialogue histories and personas to generate coherent and consistent responses in Conversational AI tasks?**

2. **How well do LMs balance between what they already know and what they fetch, and when and why do they favor one over the other?**

To address these questions, we conducted two studies. The first study (Chapter 3, Section 3.1) examines how auxiliary tasks, such as Utterance Masking (UM) and Utterance Permutation (UP), can improve the ability of LMs (i.e., richer general contextual representations) to handle dialogue context to generate more consistent responses. In the second study (Chapter 3, Section 3.2), we shift our focus to LM augmented with a retriever to explore how LM balances internal and external knowledge by applying causal mediation analysis.

This thesis is organized as follows: Chapter 2 provides the background, including foundational concepts and required related work for this thesis. Chapter 3 summarized the two studies in more detail, including the methods, findings, and implications. Finally, Chapter 4 outlines the overall discussion around this thesis and future directions for further exploration.

Chapter 2

Background

In this chapter, we will discuss the essential knowledge required to understand the foundation of this thesis. We begin by exploring the evolution of LMs in recent decades, focusing mainly on Transformer-based models in Section 2.1. Recent advancements in LMs have enhanced their capacity to encode more knowledge within their parameters. However, their inability to remain up-to-date and access expert knowledge has led to approaches that aim to overcome these limitations by combining internal information with external knowledge in Section 2.2.

So far, we have discussed the shift in perspective regarding LMs over the past few years. However, even when we achieve a highly accurate LM, can we claim that it is reliable? This question leads us to the next part. We will explore methods that help us interpret both the external behavior and internal workings of these models in Section 2.3, which play a significant role in this thesis.

Sections 2.4 and 2.5 introduce Question-Answering systems and Conversational AI, as this thesis examines the concept of context within two systems. Finally, Section 2.6 presents the Multi-Task Learning paradigm, which serves as an initial step toward improving contextual understanding.

2.1 Language Models (Rise of Transformers)

LMs are designed to predict and generate text by modeling the probability of a sequence of tokens[1]. LMs historically were developed mainly for applications such as speech recognition (Rosenfeld, 2000), machine translation (Koehn et al., 2003), and text prediction (Chen & Goodman, 1999). However, with the increasing complexity of LMs, they can now solve more complex problems (Kaddour et al., 2023).

Initially, LMs like n-grams represented by the conditional probability of the next token given all preceding tokens $\boldsymbol{w}^T = (w_1, w_2, \ldots, w_T)$ (Bengio et al.,

[1] A word is a semantic unit in natural language, while a token is a unit of text that a language model processes. For instance, the word "running" can be represented as a whole-word token "running" or as subword tokens "run" + "ning" in the token space.

2000) is estimated as follows:

$$P(\boldsymbol{w}^T) = \prod_{t=1}^{T} P(w_t \mid w_{t-n+1}^{t-1}) \tag{2.1}$$

Where w_t represents the t-th token, and w_{t-n+1}^{t-1} represents the preceding sequence of $n-1$ tokens.

One limitation of these models is the curse of dimensionality. N-gram models estimate the probability of a token based on the preceding $n-1$ words. As n increases, possible word combinations grow exponentially, requiring vast data to produce reliable probability estimates (Bengio et al., 2000).

Bengio et al. (2000) introduced the concept of neural LMs to address this issue and overcome contextual limitations. They suggested linking each token in the vocabulary to a distributed feature vector, which became the basis for Transformer-based LMs (Vaswani et al., 2017). Similarly, the probability of a sequence in Transformer-based models is represented as:

$$P(\boldsymbol{w}^T) = \prod_{t=1}^{T} P(w_t \mid \boldsymbol{W}_{t-1}; \theta) \tag{2.2}$$

Where w_t represents the t-th token in the sequence, $\boldsymbol{W}_{t-1}$ represents the subsequence of all preceding tokens, and θ indicates the learnable parameters of the model.

As the core of these models, the attention mechanism allows the model to focus on the most relevant parts of the input. This capability helps the model understand long-range relationships between words and adapt to changes in context effectively. The attention mechanism was introduced by Bahdanau et al. (2014) for the first time in the text domain, which allows the models to focus on relevant input parts selectively. However, early implementations of attention are combined with more complex models like Recurrent Neural Networks (RNNs) (Cheng et al., 2016) or Convolutional Neural Networks (CNNs) (Parikh et al., 2016), which makes them less scalable and computationally efficient. Transformers, however, changed the game by introducing self-attention and removing the need for complex models.

The Transformer model, as shown in Figure 2.1, introduces as stacks of encoder and decoder blocks (Vaswani et al., 2017). The input sequence first passes through a Positional Encoding to retain word order information and feeds to encoder and decoder blocks. Each encoder block processes the transformed input sequence with:

1. **Multi-Head Self-Attention (MHA)** to capture contextual relationships between all words in the sequence

2. **Feed-Forward networks (FFN)** to apply nonlinear transformations independently to each token's representation

3. **Residual Connections** that bypass the MHA and FFN layers, stabilizing gradient flow.

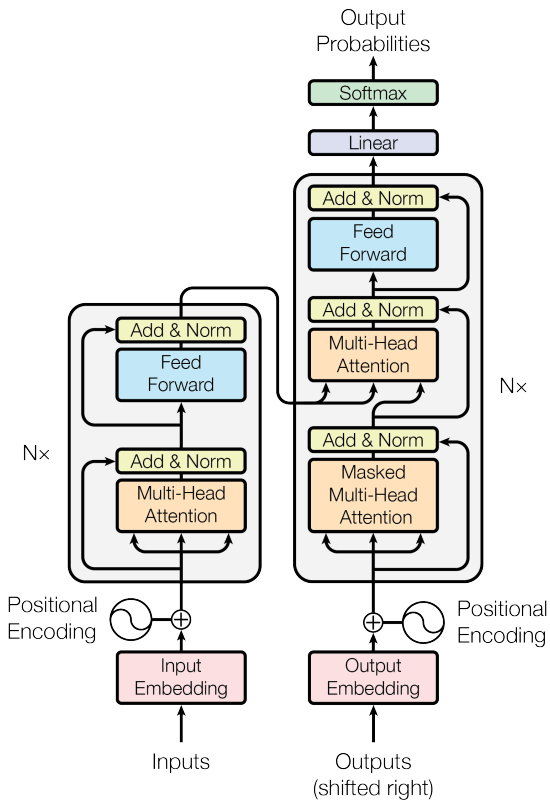

Figure 2.1: This figure illustrates the Transformer architecture, comprising both stacks of encoder and decoder.

Then, the decoder receives the encoder's output and incorporates it with its self-attended outputs to generate a sequence token by token. The decoder incorporates two forms of attention:

- **Masked Self-attention** ensures that each token can only attend to all the visited tokens to the current token.
- **Cross-attention** allows the decoder to integrate information from the entire input sequence already processed by the encoder.

Transformers can capture complete patterns by stacking multiple layers of these components n times.

As we have demonstrated, the Transformer was initially introduced as an encoder-decoder model. However, it can also be divided into distinct components, serving as the basis for three types of models:

- **Encoder-only models** like BERT (Devlin et al., 2019) are designed for tasks that require bidirectional attention.

- **Decoder-only models** Like GPT-1/2/3 (Mann et al., 2020; Radford, 2018; Radford et al., 2019) are focused on autoregressive generation with casual attention (masked self-attention).
- **Encoder-decoder models** like T5 (Raffel et al., 2020) integrate both components for understanding and generation.

The term "decoder" generally refers to an autoregressive model with causal attention, as seen in GPT models, and does not include cross-attention layers.

Before moving on to the next section, it is essential to discuss one of the key aspects of these Transformer-based LMs: their ability to store knowledge within their parameters. Petroni et al. (2019) reveal that LMs like BERT can perform as knowledge bases and recall factual information without fine-tuning. Furthermore, in another study by Kandpal et al. (2023), they show that LMs have memorization facts that are exposed more frequently. These findings show that LMs can store and recall much knowledge effectively.

2.2 Retrieval-Augmented Generation Models

As discussed in Section 2.1, LMs achieved remarkable performance across a wide range of natural language tasks based on the knowledge stored in their parameters. Although these parametric memory models capture a large amount of information in their parameters–often referred to as parametric memory–they struggle to handle cases that need up-to-date or specialized knowledge.

Retrieval-augmented generation (RAG) is an approach designed to address these challenges by augmenting LM to have access to an external source of information, such as Wikipedia–often referred to as non-parametric memory–alongside the internal information of the LM itself. This access to both parametric and non-parametric memory enables the model to produce contextually relevant responses.

In general, the RAG paradigm consists of two components (Guu et al., 2020; Karpukhin et al., 2020; Lewis et al., 2020):

- **Retriever**: Searches an external document index to fetch relevant documents based on the input query.
- **Generator**: A parametric generative model that conditions the input query and each retrieved document to generate the output.

RAG has many variants, but the general workflow of this approach includes the following steps (Siriwardhana et al., 2023):

- **Query Encoding**: The input query is encoded into a representation that the retriever uses to locate relevant external information.
- **Document Retrieval and Ranking**: The retriever searches an external corpus and selects a subset of relevant documents or knowledge elements. These retrieved documents are ranked based on their relevance to the encoded query using a ranking method.

- **Response Generation**: The generator processes the query and the retrieved context to produce a response.

Atlas, introduced by Izacard et al. (2023), is a retrieval-augmented generation model designed explicitly for knowledge-intensive tasks and few-shot learning scenarios. Atlas improves the RAG paradigm by introducing two different components:

- **Contriever** (as the retriever) (Izacard et al., 2021): It uses contrastive learning objectives to learn better contextual alignments between queries and documents. This allows Atlas to retrieve more relevant documents to the input queries.

- **Fusion-in-Decoder (FiD) Mechanism** (as the generator) (Izacard & Grave, 2020): Each retrieved document is encoded separately and fused within the decoder. This mechanism enables Atlas to effectively synthesize information from multiple sources and optimize for few-shot learning.

Atlas has been selected for this thesis as a representative example of a capable RAG approach due to its fine-tuned version for Question-Answering (QA) systems and its aim to balance the model's parametric and non-parametric memory.

2.3 Model Analysis

In Section 2.1, we discussed how LMs have recently indicated major improvements in various NLP tasks thanks to the emergence of Transformer-based architectures. As this progress continues to advance, an important question arises: How reliable are these models? Do they truly understand natural language? Answering these questions cannot be achieved only using evaluating metrics (Belinkov & Glass, 2019).

To illustrate, consider a hypothetical example in sentiment analysis: Suppose we train an LM using labeled data for positive and negative opinions. Based on evaluation metrics, the trained model may demonstrate high performance in identifying positive and negative sentiments. However, how well does the model truly understand sentiment? Will it still perform effectively on paraphrased opinions or those with ambiguity? Can the model exhibit biases related to gender, ethnicity, or other sensitive attributes?

Metrics alone are insufficient to address these questions. We need additional tools to analyze models comprehensively. Specifically, these tools can be categorized into two major approaches: Behavioral analysis (Section 2.3.1) and Structural analysis (Rogers et al., 2020), also referred to as Mechanistic Interpretability analysis (Geiger et al., 2024) (Section 2.3.2).

2.3.1 Behavioral analysis

Examines models' outputs under controlled conditions to understand how models perform across different settings. This analysis typically involves varied datasets and tasks. An early work by Ribeiro et al. (2016) proposed the Local Interpretable Model-Agnostic Explanation (LIME) framework, which helps explain model predictions by approximating the model locally rather than globally. This is achieved by generating samples through perturbations of the original data. Building on this, Ribeiro et al. (2020) presented CHECKLIST, another line of work that constructs a suit of tests based on linguistic capabilities such as reacting to negations, a temporal chain of events, co-references, and many more that facilitate the comprehensiveness of models. This approach shows that even promising Transformer-based models like BERT and RoBERTa (Liu et al., 2019) fail at many linguistic tasks.

2.3.2 Structural analysis

Provides tools to understand how LMs process and generate outputs beyond examining only the outputs. These methods reveal the inner mechanisms contributing to why models behave as they do. Tenney, Xia et al. (2019) introduced edge probing that probes word-level contextual representations to identify which layers in Transformer-based LMs encode linguistic features such as syntax or semantics. They found that these models have rich representations for syntactic over semantics. In a similar study, Tenney, Das and Pavlick (2019) introduced two measurements applied to the traditional pipeline order, demonstrating that LMs can exhibit complex interactions between different levels of hierarchical information. In another line of work, Marjanovic et al. (2024) used a probing method to examine how LMs with access to external context using a retriever handle conflicting knowledge, particularly temporal and disputable facts. This study reveals that the most frequent facts are rarely updated using the context. In a similar objective, Y. Zhao et al. (2024) used a probing method to focus on the flow of information in LMs to detect and understand knowledge conflicts between the model's internal information and external context. In general, a probe is usually a linear or simple neural model trained to predict a target property (e.g., the presence of knowledge conflicts, as mentioned in the examples above) from a specific layer in the model. This method does not necessarily indicate whether the model relies on the information in the representations for the predictions. That leads us to another structural analysis method, causal mediation analysis, which we will discuss about this method in detail in Section 2.3.2.1. This method has recently garnered significant attention as it offers a more mechanistic understanding of model behavior. Yu et al. (2024) employed this approach to investigate the internal causes of non-factual hallucinations in LMs by examining components such as attention mechanisms and feedforward layers. In another study by Meng et al. (2022b), the authors introduced the Rank-One Model Editing (ROME) method. This approach employs causal mediation analysis to identify which parts of the model are responsible for recalling facts (which serves as an inspirational foundation for this thesis and we will discuss in Chapter 3, Section 3.2)

and then uses ROME to update the associated facts. Their findings reveal that certain feedforward layers in the mid-layers are crucial for processing subject tokens during factual recall.

2.3.2.1 Causal Mediation Analysis

Causal Mediation Analysis (CMA) is a general statistical technique initially developed for inferring the effects of any treatment in medicine and social sciences (Pearl, 2001; Peña, 2023) CMA helps disentangle the direct and indirect effects of a treatment or intervention on an outcome.

CMA in General Framework As Pearl (2001) presented, consider that a new drug is developed to lower blood pressure. During clinical trials, it is observed that the drug causes a side effect, such as headaches, in some patients. As a result, these patients begin taking aspirin. On the other hand, due to its anti-inflammatory properties and specific effects on blood circulation, aspirin has an independent impact on alleviating blood pressure symptoms. In this scenario, the drug influences blood pressure through a direct effect via its primary mechanism (controlling blood pressure) and an indirect effect by prompting aspirin use, which contributes to disease improvement. Therefore, it is essential to separate and examine these effects. In this clinical example, the treatment (control variable) is the under-development drug, the outcome is the reduction in blood pressure, and the mediator is the use of aspirin by patients, which built the foundation of CMA. The goal of this setup is to separate the drug's direct effect ($X \rightarrow Y$) and its indirect effect via aspirin use ($X \rightarrow M \rightarrow Y$). Similarly, this method can be applied to the internal analysis of LMs between inputs, internal representations, and outputs (J. Huang et al., 2024; Meng et al., 2022a; Stolfo et al., 2023; Vig et al., 2020; Yao et al., 2021; Yu et al., 2024).

CMA in LM Following the "do" notation of Pearl (2001) ($X \leftarrow 1$ means set X to 1) similar to Meng et al. (2022b), we might be interested in how the intervention on a control variable X (e.g., the actual object vs. a counterfactual object or a noised subject vs. an unnoised subject representations) in the input (e.g., a query like "What is the capital of Sweden? Context: Stockholm is the capital of Sweden.") affects an outcome Y (e.g., the probability of a specific word answer like "Stockholm"). We define the Total Effect (TE) to measure the effect of X on Y regardless of any specific encoding in the latent layers.

$$TE = Y(X \leftarrow 1) - Y(X \leftarrow 0) \tag{2.3}$$

Here:

- $(X \leftarrow 1)$: Represents the intervention (e.g., replacing the "Stockholm" representation at the token embedding representation with a counterfactual representation like "Milan" or adding noise to the "Sweden" token representation).

- $(X \leftarrow 0)$: Represents the absence of intervention.

The intervention does not always affect the outcome directly; instead, its influence may pass through an intermediary variable M (often called a mediator) modifies which part of the model contributes to the effect of X on Y. Based on Pearl (2001) notation, we define Indirect Effect (IE) with the following expression:

$$IE = Y(X \leftarrow 0, M(X \leftarrow 1)) - Y(X \leftarrow 0) \tag{2.4}$$

Here:

- Running the model with $X \leftarrow 1$ to observe the mediator M.
- Re-running the model with $X \leftarrow 0$ while fixing M to its observed state from the previous step.

To better understand, consider that we want to evaluate the effect of noising the representation of a word in the input, such as "Sweden" in the context, on the probability of predicting "Stockholm" as the answer to the query. First, we run the model with a noised representation of "Sweden" ($X \leftarrow 1$) and observe the resulting intermediate hidden state h^l. Then, we re-run the model with the original, unnoised representation ($X \leftarrow 0$) but restore h^l to its state from the noised run. By comparing the probabilities of "Stockholm" in both scenarios, we can measure the indirect contribution of h^l to the outcome.

The design of X and Y is closely tied to the research question. For instance, in the study by Meng et al. (2022b), the authors set X as added noise to the subject representation within a factual prompt–For example, in the prompt "The Space Needle is in downtown," the subject is "The Space Needle." This setup measures the model's predicted completion of the prompt, specifically, the probability assigned to the correct object (e.g., "Seattle" in this example).

In LM applications, we can observe both outcomes ($X \leftarrow 0$ and $X \leftarrow 1$). However, in a general case (e.g., the medical example), we can observe only one of these conditions because patients either take the drug or do not. Similarly, computing the outcome $Y(X \leftarrow 0, M(X \leftarrow 1)) - Y(X \leftarrow 0)$ is feasible as we can explicitly set mediators in LMs, while it is impractical in general cases. This flexibility makes CMA particularly powerful for analyzing LMs.

2.4 Question-Answering Systems

Question-answering (QA) systems are a part of Natural Language Understanding (NLU), designed to automatically provide accurate and relevant answers to users' queries (Kwiatkowski et al., 2019). QA systems fetch information from external sources or recall information from internal knowledge to produce a response. This response can either be identified as a span from a passage (Extractive QA) (Tran and Kretchmar, 2024) or be generated directly using language models, potentially integrating a retrieval component (Generative QA) (Lála et al., 2023). QA systems usually are categorized into (Biancofiore et al., 2024):

- **Open-domain QA**: They aim to answer questions on various topics.
- **Factoid QA:** They focus on questions that have short and factual answers.
- **Visual QA:** These systems are built to produce responses based on visual inputs, such as images or videos.

In this thesis, we focus on a Generative QA system with access to world knowledge, leading us to examine an interesting behavior of the model: how it decides to generate a response when both internal information and external information as context are provided.

2.5 Conversational AI

Conversational AI (ConvAI) studies techniques that focus on building systems capable of interacting with humans (Ram et al., 2018). These systems should cover different ranges from simple chatbots to advanced dialogue agents and be able to manage complex tasks, engage in multi-turn conversations, and provide accurate and contextually aware responses (Gao et al., 2018).

The first generation of ConvAI focused on building task-specific dialogue systems (Task-Oriented Dialogue (TOD) systems). These modular systems could only be able to respond to simple and transactional tasks such as weather updates, music requests, and so on (Ram et al., 2018). TOD systems are limited to specific workflows and cannot address a wide range of topics and maximize long-term user engagement. These limitations direct us to another system (Open-Domain Dialogue (ODD) systems) that excels in handling unstructured, diverse, and dynamic conversations for having natural and engaging interactions (M. Huang et al., 2020).

Despite significant advancements, developing ODD systems comes with challenges, including as below (M. Huang et al., 2020):

- **Semantics**: ODD systems often struggle with understanding the deeper meaning behind user inputs, including their intent, sentiment, and contextual relevance.
- **Consistency**: Maintaining coherence throughout a conversation; in other words, the system must be aware of what has already been said. ODD systems may give conflicting responses to similar questions within the same interaction.
- **Interactiveness**: ODD systems should be able to engage in long and captivating conversations, but often, they fail when it comes to handling empathy, entertainment, or companionship.
- **Knowledge Integration**: These systems are often incapable of grounding knowledge in real-world knowledge, which leads to incorrect and irrelevant responses. Although one solution to this could involve providing access to external knowledge, challenges still remain in selecting and integrating the most relevant information.

These features of ODDs make them an interesting subject for investigation and were examined as the first step in this thesis. We focused on one specific challenge: "consistency." We aim to explore methods to improve the contextual consistency of ConvAI models.

2.6 Multi-Task Learning

Multi-task Learning (MTL) is a machine learning paradigm where the model is trained to perform multiple tasks parallelly, using shared representation to improve the generalization performance (Crawshaw, 2020). MTL has been widely applied in NLP, demonstrating success in language understanding and representation learning (Collobert and Weston, 2008; Søgaard and Goldberg, 2016).

This approach reflects the process humans use to learn knowledge across the domain and is efficient when tasks are related, as the information from one task can improve the others.

In this thesis, we adopt an MTL approach where the model is trained on a primary task (as our main task) alongside one or more auxiliary tasks to gain a richer representation. These auxiliary tasks are another objective for the models, providing additional information to enhance their ability to understand the context.

Chapter 3

Summary of Included Papers

This chapter summarizes the two research we have done exploring how LMs interpret and use context in their outputs. These studies help to focus more on whether LMs have a proper understanding of context and open new doors toward controlling this behavior of LMs.

3.1 Paper I

In this paper, we explore how auxiliary tasks can affect models' ability to comprehend context. In this particular study, we take context as the conversation history between the user and the agent, and also the agent's persona. We observe that models' responses during a conversation occasionally contradict or even violate the agent's established persona and prior responses. We introduce two pairs of auxiliary tasks, Utterance Masking (UM) and Utterances Permuation (UP), to improve contextual representation learning of a GPT family model on two datasets such as PersonaChat (S. Zhang et al., 2018) and DailyDialog (Y. Li et al., 2017).

Earlier efforts have been made using auxiliary tasks for other Transformer-based models (except for decoder-only models) in Conversational AI (Mehri et al., 2019; Y. Zhao et al., 2020). They demonstrate that adding auxiliary tasks related to context understanding alongside the main task can encourage the model to develop more robust general representations. For this reason, we introduce two pairs of auxiliary tasks targetting context to help LM be motivated to consider more of the context. UM is designed to help the model capture the semantics of dialogues by masking utterances in the context and training the model to predict them in two ways: detecting and recovering. Detecting approach helps the model identify the presence of masked utterances while recovering, generating the masked content in a semantically coherent manner. On the other hand, UP focuses on improving contextual coherence by permuting utterances in dialogue and asking the model to reconstruct the

logical sequences (detecting) or recover the correct order (recovering).

To evaluate the impact of these auxiliary tasks, we compare a standard GPT-2 model (baseline) with GPT-2 models trained with UM and UP tasks. We use several metrics for surface-level analysis, including Perplexity, BLEU (Papineni et al., 2002), ROUGE-L (J. Li et al., 2016), BERTScore (T. Zhang et al., 2020), MoverScore (W. Zhao et al., 2019), and Embedding-based metrics (Serban et al., 2017). Although most of these metrics compare the generated responses to a reference output, we assume that the reference inherently reflects some degree of contextual consistency. However, it does not indicate how well the response adheres to the dialogue history or aligns with the defined persona. We hypothesize that if these changes are significant, it indicates that the auxiliary tasks have affected the model, resulting in consistent responses. In this sense, the model has successfully accounted for the context.

Models trained with UM and UP showed a consistent reduction in perplexity and improvements in BLEU and ROUGE scores across both datasets. The results show that performing the auxiliary tasks results in a slight overall improvement.

These auxiliary tasks improve the model's ability to generate grammatically accurate responses and enhance the coherence and consistency of responses, as reflected in the evaluation metrics. Something still missing in this study is that, although we observe slight improvements, it does not explicitly address the model's comprehension of context. This motivates us to conduct a second study using a different toolset to investigate how the model understands context. However, we must first define what context entails.

Contribution

M. Farahani conducted the primary research and development, taking responsibility for the majority of the writing, with R. Johansson providing supervision and guidance throughout the process.

3.2 Paper II

In this study, we tackle a different approach instead of introducing new methods to train a model in order to understand context. We perform an analysis method to examine how the model interacts with the context. In this particular case, we define *context* as an external source of information provided by a retrieval-based LM, as introduced in Section 2.1.

As discussed in Section 2.1 and 2.2, LMs can memorize information during training by storing it in their weight parameters. For example, when we ask a simple factual question in a QA system, the LM attempts to recite the memorized information. However, when queries become more complex or require up-to-date or specialized knowledge, the LM can be supplemented with access to external knowledge via a retrieval-based LM (such as ATLAS). Assuming the retriever fetches relevant data, it becomes unclear whether the model relies on its internal, memorized knowledge or external information when generating a response. In simpler terms, if we treat external information as context, this raises the question of whether the model prioritizes the provided context or depends on the knowledge it has already learned during training when responding. We addressed two research questions on that observation:

1. Which aspect of the model representation impacts the output in copying mode?

2. What specific parts of the model trigger copying?

To answer these questions, we employ CMA (Section 2.3) to disentangle the contributions of parametric and non-parametric memory on two entity-centric question-answer pairs datasets, PopQA (Mallen et al., 2023) and PEQ (Sciavolino et al., 2021). The experiments apply the structure of factual triples (s, r, o), where s is the subject, r is the relation, and o is the object, such as this query "What is the capital of Sweden?" (o: "Stockholm", r: "capital of", s: "Sweden"). In this study, we address context in two distinct ways: external information in the form of real text chunks from Wikipedia or synthetic information, such as a template corresponding to each relation (e.g., for the relation "capital of," we define the template as "obj is the capital of subj."). We introduce two experiments:

Experiment 1 investigates how much the model relies on copying from the context versus recalling from parametric memory, as well as the components involved in making this decision.

To address this, we start by replacing object token representation in the context (a retrieved or synthetic document) with counterfactuals (e.g., "Stockholm" with "Milan"). We evaluate whether the model relies on copying from the context or ignores the context and considers its internal information. Then, as introduced in Equation 2.3 and Equation 2.4, we compute the Total Effect (TE) and the Indirect Effect (IE). Additionally, we compute the Path-Specific Effect (PSE), which is similar to the IE but provides insights into the impacts of individual components separately (Meng et al., 2022b).

The TE measures the extent to which the model shifts its output towards the counterfactuals when the context is altered. The IE and PSE also provide deeper insight into which parts of the input or model layers are involved in this decision-making process. The results show that object tokens are the most impactful part, and it is like the object tokens flow directly through the model, unaffected by the surrounding context. The results also demonstrate that MLP in mid-layers is important in translating object tokens and relation tokens representations from the encoder to the decoder, while attention mechanisms serve a supportive role.

Experiment 2 explores the factors that impact the model's decision to rely on non-parametric knowledge by focusing on subject and relation tokens. We refer to this impact factor as "context relevance."

Compared to the first experiment, we first substitute the object token with counterfactuals. Then, noise is added to subject and relation token representations to explore their impact on context relevance evaluation. As we have observed, the Atlas model predominantly adopts a copying behavior. By replacing the object token with a counterfactual and investigating the effect of noise on subject tokens and relation tokens, we can investigate which context components (external information) drive this behavior.

In this experiment, the TE is interpreted as a way to know the valuable part of the triple in context evaluation. Similarly, we compute the IE and PSE for this experiment as well. The results show that the MLP in the first layers is involved in computing the relevance of subject and relation tokens. This process determines whether the context is relevant enough for the model to rely on to generate the final answer from the context.

Contribution

M. Farahani conducted the primary research and development, taking responsibility for the majority of the writing, with R. Johansson providing supervision and guidance throughout the process.

Chapter 4

Discussion and Future Work

In this thesis, we sought answers to two core research questions: *How can LMs better incorporate context*, and *what do LMs know about context and how do they process it?* We addressed these questions in different domains and explored different behaviors of context.

In the first experiment, we demonstrated that introducing relevant auxiliary tasks can slightly improve response quality by looking into surface-level metrics. However, these improvements do not necessarily indicate that the model really incorporates context. Instead, the model might exploit statistical patterns in the data to optimize the metrics without understanding the underlying context. The term "understand" needs a more profound way, possibly a causal mechanism in which the model integrates dialogue histories and personas meaningfully into its decision-making. This led us to the second experiment, where we shifted focus to a different scenario–context as external knowledge in a factoid QA system–to investigate these mechanisms further under a controlled experimental environment.

In the second experiment, we used CMA to observe how Transformer-based LMs with a retriever (i.e., RAG) incorporate context and which parts of a model are responsible for this process. Specifically, we found that they evaluate the quality of context before generating a response. This evaluation involves measuring the alignment between question embeddings and context embeddings, a behavior we term context relevance. Our experiment involved retrieved documents–either text chunks from Wikipedia or synthetic data generated from templates–but left several aspects of context evaluation needing to be explored. For instance, beyond context relevance, qualities such as clarity and coherence, completeness, and correctness of context could also play vital roles in determining good context quality. Our findings also highlighted the dual role of the MLP. It not only supports the transition from context relevance to answer extraction but also aids in translating the most important contextual elements for response generation.

This thesis offers valuable insights and opens several directions for future

research. Still, there are areas in this research that need further investigation, which we will discuss in the following.

We examined the model's behavior in considering context and identified the mechanisms and parts of the model that are responsible for that decision-making. However, an important question arises: How can we control (or steer) this behavior to achieve the desired output? For example, if we ask the model a question following its context –"What is the capital of Sweden?" and "Gothenburg is the capital of Sweden"– we aim to have complete control over whether the model generates the output based on its internal knowledge ("Stockholm") or the external information ("Gothenburg").

Another area for future investigation in context relevance is understanding how the frequency of prompts affects the decision-making process we studied. For example, if we ask the model two questions –"What is Donald Trump's occupation?" and "What is Peter Murnoy's occupation?"– How does its decision-making change, considering the first question is more frequent than the second?

Temporal relevance is another interesting future direction. This involves how the model answers questions by accessing external knowledge tied to specific time periods. It helps to understand how the model deals with outdated information, chooses a specific time frame to answer a question, or identifies time dependencies. For example, if we ask the model, "Where have the Olympics been held in the USA?" and provide two temporal information –"In 1996, the USA hosted the Olympics for the first time in Atlanta, Georgia," and "In 2002, the Olympics were held in Salt Lake City, Utah"– How does the model decide which information to use?

So far, we have only discussed directions focused on LMs pre-trained jointly with a retriever. We have not explored whether the same behavior occurs in LLMs with access to a retriever (or in-context learning). Another interesting direction we can apply the methods used in this study to LLMs to determine whether similar behavior occurs. Additionally, we could compare the results with instruction-tuned variants of LLMs to assess how this tuning affects their decision-making.

Bibliography

Airenti, G., Cruciani, M., & Plebe, A. (2017). *Context in communication: A cognitive view* (G. Airenti, M. Cruciani & A. Plebe, Eds.) [Retrieved from the Library of Congress, https://www.loc.gov/item/2020394825/]. Frontiers Media SA. (Cit. on p. 3).

Bahdanau, D., Cho, K., & Bengio, Y. (2014). Neural machine translation by jointly learning to align and translate. *CoRR*, *abs/1409.0473* (cit. on p. 6).

Belinkov, Y., & Glass, J. (2019). Analysis methods in neural language processing: A survey. *Transactions of the Association for Computational Linguistics*, *7*, 49–72 (cit. on p. 9).

Bender, E. M., & Koller, A. (2020, July). Climbing towards NLU: On meaning, form, and understanding in the age of data. In D. Jurafsky, J. Chai, N. Schluter & J. Tetreault (Eds.), *Proceedings of the 58th annual meeting of the association for computational linguistics* (pp. 5185–5198). Association for Computational Linguistics. https://doi.org/10.18653/v1/2020.acl-main.463 (cit. on p. 3).

Bengio, Y., Ducharme, R., & Vincent, P. (2000). A neural probabilistic language model. *Advances in neural information processing systems*, *13* (cit. on pp. 5, 6).

Biancofiore, G. M., Deldjoo, Y., Noia, T. D., Di Sciascio, E., & Narducci, F. (2024). Interactive question answering systems: Literature review. *ACM Computing Surveys*, *56*(9), 1–38 (cit. on p. 12).

Chen, S. F., & Goodman, J. (1999). An empirical study of smoothing techniques for language modeling. *Computer Speech & Language*, *13*(4), 359–394 (cit. on p. 5).

Cheng, J., Dong, L., & Lapata, M. (2016, November). Long short-term memory-networks for machine reading. In J. Su, K. Duh & X. Carreras (Eds.), *Proceedings of the 2016 conference on empirical methods in natural language processing* (pp. 551–561). Association for Computational Linguistics. https://doi.org/10.18653/v1/D16-1053 (cit. on p. 6).

Collobert, R., & Weston, J. (2008). A unified architecture for natural language processing: Deep neural networks with multitask learning. *Proceedings of the 25th international conference on Machine learning*, 160–167 (cit. on p. 14).

Crawshaw, M. (2020). Multi-task learning with deep neural networks: A survey. *arXiv preprint arXiv:2009.09796* (cit. on p. 14).

Devlin, J., Chang, M.-W., Lee, K., & Toutanova, K. (2019, June). BERT: Pre-training of deep bidirectional transformers for language understanding. In J. Burstein, C. Doran & T. Solorio (Eds.), *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)* (pp. 4171–4186). Association for Computational Linguistics. https://doi.org/10.18653/v1/N19-1423 (cit. on p. 7).

Gao, J., Galley, M., & Li, L. (2018). Neural approaches to conversational ai. *The 41st international ACM SIGIR conference on research & development in information retrieval*, 1371–1374 (cit. on p. 13).

Geiger, A., Ibeling, D., Zur, A., Chaudhary, M., Chauhan, S., Huang, J., Arora, A., Wu, Z., Goodman, N., Potts, C., et al. (2024). Causal abstraction: A theoretical foundation for mechanistic interpretability. *Preprint* (cit. on p. 9).

Guu, K., Lee, K., Tung, Z., Pasupat, P., & Chang, M. (2020). Retrieval augmented language model pre-training. *International conference on machine learning*, 3929–3938 (cit. on p. 8).

Huang, J., Wu, Z., Potts, C., Geva, M., & Geiger, A. (2024, August). RAVEL: Evaluating interpretability methods on disentangling language model representations. In L.-W. Ku, A. Martins & V. Srikumar (Eds.), *Proceedings of the 62nd annual meeting of the association for computational linguistics (volume 1: Long papers)* (pp. 8669–8687). Association for Computational Linguistics. https://doi.org/10.18653/v1/2024.acl-long.470 (cit. on p. 11).

Huang, M., Zhu, X., & Gao, J. (2020). Challenges in building intelligent open-domain dialog systems. *ACM Transactions on Information Systems (TOIS)*, *38*(3), 1–32 (cit. on p. 13).

Huang, Y. (2015). *Pragmatics*. Oxford University Press, Incorporated. (Cit. on p. 3).

Izacard, G., Caron, M., Hosseini, L., Riedel, S., Bojanowski, P., Joulin, A., & Grave, E. (2021). Unsupervised dense information retrieval with contrastive learning. *arXiv:2112.09118* (cit. on p. 9).

Izacard, G., & Grave, E. (2020). Leveraging passage retrieval with generative models for open domain question answering. *arXiv preprint arXiv:2007.01282* (cit. on p. 9).

Izacard, G., Lewis, P., Lomeli, M., Hosseini, L., Petroni, F., Schick, T., Dwivedi-Yu, J., Joulin, A., Riedel, S., & Grave, E. (2023). Atlas: Few-shot learning with retrieval augmented language models. *J. Mach. Learn. Res.*, *24*(1) (cit. on p. 9).

Kaddour, J., Harris, J., Mozes, M., Bradley, H., Raileanu, R., & McHardy, R. (2023). Challenges and applications of large language models. *arXiv preprint arXiv:2307.10169* (cit. on p. 5).

Kandpal, N., Deng, H., Roberts, A., Wallace, E., & Raffel, C. (2023). Large language models struggle to learn long-tail knowledge. *Proceedings of the 40th International Conference on Machine Learning* (cit. on p. 8).

Karpukhin, V., Oguz, B., Min, S., Lewis, P., Wu, L., Edunov, S., Chen, D., & Yih, W.-t. (2020, November). Dense passage retrieval for open-domain

question answering. In B. Webber, T. Cohn, Y. He & Y. Liu (Eds.), *Proceedings of the 2020 conference on empirical methods in natural language processing (emnlp)* (pp. 6769–6781). Association for Computational Linguistics. https://doi.org/10.18653/v1/2020.emnlp-main.550 (cit. on p. 8).

Kwiatkowski, T., Palomaki, J., Redfield, O., Collins, M., Parikh, A., Alberti, C., Epstein, D., Polosukhin, I., Devlin, J., Lee, K., et al. (2019). Natural questions: A benchmark for question answering research. *Transactions of the Association for Computational Linguistics*, *7*, 453–466 (cit. on p. 12).

Lála, J., O'Donoghue, O., Shtedritski, A., Cox, S., Rodriques, S. G., & White, A. D. (2023). Paperqa: Retrieval-augmented generative agent for scientific research. *arXiv preprint arXiv:2312.07559* (cit. on p. 12).

Lewis, P., Perez, E., Piktus, A., Petroni, F., Karpukhin, V., Goyal, N., Küttler, H., Lewis, M., Yih, W.-t., Rocktäschel, T., Riedel, S., & Kiela, D. (2020). Retrieval-augmented generation for knowledge-intensive nlp tasks. In H. Larochelle, M. Ranzato, R. Hadsell, M. Balcan & H. Lin (Eds.), *Advances in neural information processing systems* (pp. 9459–9474, Vol. 33). Curran Associates, Inc. https://proceedings.neurips.cc/paper_files/paper/2020/file/6b493230205f780e1bc26945df7481e5-Paper.pdf (cit. on p. 8).

Li, J., Galley, M., Brockett, C., Gao, J., & Dolan, B. (2016). A diversity-promoting objective function for neural conversation models. *Proceedings of the 2016 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, 110–119. https://doi.org/10.18653/v1/N16-1014 (cit. on p. 16).

Li, Y., Su, H., Shen, X., Li, W., Cao, Z., & Niu, S. (2017, November). DailyDialog: A manually labelled multi-turn dialogue dataset. In G. Kondrak & T. Watanabe (Eds.), *Proceedings of the eighth international joint conference on natural language processing (volume 1: Long papers)* (pp. 986–995). Asian Federation of Natural Language Processing. https://aclanthology.org/I17-1099 (cit. on p. 15).

Liu, Y., Ott, M., Goyal, N., Du, J., Joshi, M., Chen, D., Levy, O., Lewis, M., Zettlemoyer, L., & Stoyanov, V. (2019). Roberta: A robustly optimized bert pretraining approach. *ArXiv*, *abs/1907.11692* (cit. on p. 10).

Mallen, A., Asai, A., Zhong, V., Das, R., Khashabi, D., & Hajishirzi, H. (2023, July). When not to trust language models: Investigating effectiveness of parametric and non-parametric memories. In A. Rogers, J. Boyd-Graber & N. Okazaki (Eds.), *Proceedings of the 61st annual meeting of the association for computational linguistics (volume 1: Long papers)* (pp. 9802–9822). Association for Computational Linguistics. https://doi.org/10.18653/v1/2023.acl-long.546 (cit. on p. 17).

Mann, B., Ryder, N., Subbiah, M., Kaplan, J., Dhariwal, P., Neelakantan, A., Shyam, P., Sastry, G., Askell, A., Agarwal, S., et al. (2020). Language models are few-shot learners. *arXiv preprint arXiv:2005.14165*, *1* (cit. on p. 8).

Marjanovic, S. V., Yu, H., Atanasova, P., Maistro, M., Lioma, C., & Augenstein, I. (2024, November). DYNAMICQA: Tracing internal knowledge conflicts in language models. In Y. Al-Onaizan, M. Bansal & Y.-N. Chen (Eds.), *Findings of the association for computational linguistics: Emnlp 2024* (pp. 14346–14360). Association for Computational Linguistics. https://doi.org/10.18653/v1/2024.findings-emnlp.838 (cit. on p. 10).

Mehri, S., Razumovskaia, E., Zhao, T., & Eskenazi, M. (2019, July). Pretraining methods for dialog context representation learning. In A. Korhonen, D. Traum & L. Màrquez (Eds.), *Proceedings of the 57th annual meeting of the association for computational linguistics* (pp. 3836–3845). Association for Computational Linguistics. https://doi.org/10.18653/v1/P19-1373 (cit. on p. 15).

Meng, K., Bau, D., Andonian, A., & Belinkov, Y. (2022a). Locating and editing factual associations in GPT. In S. Koyejo, S. Mohamed, A. Agarwal, D. Belgrave, K. Cho & A. Oh (Eds.), *Advances in neural information processing systems* (pp. 17359–17372, Vol. 35). Curran Associates, Inc. https://proceedings.neurips.cc/paper_files/paper/2022/file/6f1d43d5a82a37e89b0665b33bf3a182-Paper-Conference.pdf (cit. on p. 11).

Meng, K., Bau, D., Andonian, A., & Belinkov, Y. (2022b). Locating and editing factual associations in gpt. *Advances in Neural Information Processing Systems*, *35*, 17359–17372 (cit. on pp. 10–12, 17).

Papineni, K., Roukos, S., Ward, T., & Zhu, W.-J. (2002). Bleu: A method for automatic evaluation of machine translation. *Proceedings of the 40th Annual Meeting of the Association for Computational Linguistics*, 311–318. https://doi.org/10.3115/1073083.1073135 (cit. on p. 16).

Parikh, A. P., Täckström, O., Das, D., & Uszkoreit, J. (2016). A decomposable attention model for natural language inference. *arXiv preprint arXiv:1606.01933* (cit. on p. 6).

Pearl, J. (2001). Direct and indirect effects. *Proceedings of the Seventeenth Conference on Uncertainty in Artificial Intelligence*, 411–420 (cit. on pp. 11, 12).

Peña, J. M. (2023). Alternative measures of direct and indirect effects. *ArXiv*, *abs/2306.01292*. https://arxiv.org/pdf/2306.01292 (cit. on p. 11).

Petroni, F., Rocktäschel, T., Riedel, S., Lewis, P., Bakhtin, A., Wu, Y., & Miller, A. (2019, November). Language models as knowledge bases? In K. Inui, J. Jiang, V. Ng & X. Wan (Eds.), *Proceedings of the 2019 conference on empirical methods in natural language processing and the 9th international joint conference on natural language processing (emnlp-ijcnlp)* (pp. 2463–2473). Association for Computational Linguistics. https://doi.org/10.18653/v1/D19-1250 (cit. on p. 8).

Radford, A. (2018). Improving language understanding by generative pretraining (cit. on p. 8).

Radford, A., Wu, J., Child, R., Luan, D., Amodei, D., Sutskever, I., et al. (2019). Language models are unsupervised multitask learners. *OpenAI blog*, *1*(8), 9 (cit. on p. 8).

Raffel, C., Shazeer, N., Roberts, A., Lee, K., Narang, S., Matena, M., Zhou, Y., Li, W., & Liu, P. J. (2020). Exploring the limits of transfer learning with a unified text-to-text transformer. *Journal of machine learning research*, *21*(140), 1–67 (cit. on p. 8).

Ram, A., Prasad, R., Khatri, C., Venkatesh, A., Gabriel, R., Liu, Q., Nunn, J., Hedayatnia, B., Cheng, M., Nagar, A., et al. (2018). Conversational ai: The science behind the alexa prize. *arXiv preprint arXiv:1801.03604* (cit. on p. 13).

Ribeiro, M. T., Singh, S., & Guestrin, C. (2016). " why should i trust you?" explaining the predictions of any classifier. *Proceedings of the 22nd ACM SIGKDD international conference on knowledge discovery and data mining*, 1135–1144 (cit. on p. 10).

Ribeiro, M. T., Wu, T., Guestrin, C., & Singh, S. (2020, July). Beyond accuracy: Behavioral testing of NLP models with CheckList. In D. Jurafsky, J. Chai, N. Schluter & J. Tetreault (Eds.), *Proceedings of the 58th annual meeting of the association for computational linguistics* (pp. 4902–4912). Association for Computational Linguistics. https://doi.org/10.18653/v1/2020.acl-main.442 (cit. on p. 10).

Rogers, A., Kovaleva, O., & Rumshisky, A. (2020). A primer in BERTology: What we know about how BERT works (M. Johnson, B. Roark & A. Nenkova, Eds.). *Transactions of the Association for Computational Linguistics*, *8*, 842–866. https://doi.org/10.1162/tacl_a_00349 (cit. on p. 9).

Rosenfeld, R. (2000). Two decades of statistical language modeling: Where do we go from here? *Proceedings of the IEEE*, *88*(8), 1270–1278 (cit. on p. 5).

Sciavolino, C., Zhong, Z., Lee, J., & Chen, D. (2021, November). Simple entity-centric questions challenge dense retrievers. In M.-F. Moens, X. Huang, L. Specia & S. W.-t. Yih (Eds.), *Proceedings of the 2021 conference on empirical methods in natural language processing* (pp. 6138–6148). Association for Computational Linguistics. https://doi.org/10.18653/v1/2021.emnlp-main.496 (cit. on p. 17).

Serban, I. V., Sordoni, A., Lowe, R., Charlin, L., Pineau, J., Courville, A., & Bengio, Y. (2017). A hierarchical latent variable encoder-decoder model for generating dialogues. *Proceedings of the Thirty-First AAAI Conference on Artificial Intelligence*, 3295–3301 (cit. on p. 16).

Siriwardhana, S., Weerasekera, R., Wen, E., Kaluarachchi, T., Rana, R., & Nanayakkara, S. (2023). Improving the domain adaptation of retrieval augmented generation (rag) models for open domain question answering. *Transactions of the Association for Computational Linguistics*, *11*, 1–17 (cit. on p. 8).

Søgaard, A., & Goldberg, Y. (2016). Deep multi-task learning with low level tasks supervised at lower layers. *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers)*, 231–235 (cit. on p. 14).

Stolfo, A., Belinkov, Y., & Sachan, M. (2023, December). A mechanistic interpretation of arithmetic reasoning in language models using causal

mediation analysis. In H. Bouamor, J. Pino & K. Bali (Eds.), *Proceedings of the 2023 conference on empirical methods in natural language processing* (pp. 7035–7052). Association for Computational Linguistics. https://doi.org/10.18653/v1/2023.emnlp-main.435 (cit. on p. 11).

Tenney, I., Das, D., & Pavlick, E. (2019, July). BERT rediscovers the classical NLP pipeline. In A. Korhonen, D. Traum & L. Màrquez (Eds.), *Proceedings of the 57th annual meeting of the association for computational linguistics* (pp. 4593–4601). Association for Computational Linguistics. https://doi.org/10.18653/v1/P19-1452 (cit. on p. 10).

Tenney, I., Xia, P., Chen, B., Wang, A., Poliak, A., McCoy, R. T., Kim, N., Durme, B. V., Bowman, S. R., Das, D., & Pavlick, E. (2019). What do you learn from context? probing for sentence structure in contextualized word representations. https://arxiv.org/abs/1905.06316 (cit. on p. 10).

Tran, S. Q., & Kretchmar, M. (2024). Towards robust extractive question answering models: Rethinking the training methodology. *arXiv preprint arXiv:2409.19766* (cit. on p. 12).

Vaswani, A., Shazeer, N., Parmar, N., Uszkoreit, J., Jones, L., Gomez, A. N., Kaiser, Ł., & Polosukhin, I. (2017). Attention is all you need. *Proceedings of the 31st International Conference on Neural Information Processing Systems*, 6000–6010 (cit. on p. 6).

Vig, J., Gehrmann, S., Belinkov, Y., Qian, S., Nevo, D., Singer, Y., & Shieber, S. (2020). Investigating gender bias in language models using causal mediation analysis. In H. Larochelle, M. Ranzato, R. Hadsell, M. Balcan & H. Lin (Eds.), *Advances in neural information processing systems* (pp. 12388–12401, Vol. 33). Curran Associates, Inc. (Cit. on p. 11).

Yao, L., Chu, Z., Li, S., Li, Y., Gao, J., & Zhang, A. (2021). A survey on causal inference. *ACM Transactions on Knowledge Discovery from Data (TKDD)*, *15*(5), 1–46 (cit. on p. 11).

Yu, L., Cao, M., Cheung, J. C., & Dong, Y. (2024, November). Mechanistic understanding and mitigation of language model non-factual hallucinations. In Y. Al-Onaizan, M. Bansal & Y.-N. Chen (Eds.), *Findings of the association for computational linguistics: Emnlp 2024* (pp. 7943–7956). Association for Computational Linguistics. https://doi.org/10.18653/v1/2024.findings-emnlp.466 (cit. on pp. 10, 11).

Zhang, S., Dinan, E., Urbanek, J., Szlam, A., Kiela, D., & Weston, J. (2018, July). Personalizing dialogue agents: I have a dog, do you have pets too? In I. Gurevych & Y. Miyao (Eds.), *Proceedings of the 56th annual meeting of the association for computational linguistics (volume 1: Long papers)* (pp. 2204–2213). Association for Computational Linguistics. https://doi.org/10.18653/v1/P18-1205 (cit. on p. 15).

Zhang, T., Kishore, V., Wu, F., Weinberger, K. Q., & Artzi, Y. (2020). Bertscore: Evaluating text generation with bert. https://arxiv.org/abs/1904.09675 (cit. on p. 16).

Zhao, W., Peyrard, M., Liu, F., Gao, Y., Meyer, C. M., & Eger, S. (2019). MoverScore: Text generation evaluating with contextualized embeddings and earth mover distance. *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*, 563–578. https://doi.org/10.18653/v1/D19-1053 (cit. on p. 16).

Zhao, Y., Du, X., Hong, G., Gema, A. P., Devoto, A., Wang, H., He, X., Wong, K.-F., & Minervini, P. (2024). Analysing the residual stream of language models under knowledge conflicts. https://arxiv.org/abs/2410.16090 (cit. on p. 10).

Zhao, Y., Xu, C., & Wu, W. (2020, November). Learning a simple and effective model for multi-turn response generation with auxiliary tasks. In B. Webber, T. Cohn, Y. He & Y. Liu (Eds.), *Proceedings of the 2020 conference on empirical methods in natural language processing (emnlp)* (pp. 3472–3483). Association for Computational Linguistics. https://doi.org/10.18653/v1/2020.emnlp-main.279 (cit. on p. 15).