Evaluating the integration of pretrain-time and inference-time knowledge in large language model-based natural language understanding systems

Martin Pömsl

School of Computer Science
McGill University
Montreal, Canada

December 2023

A thesis submitted to McGill University in partial fulfillment of the requirements for the degree of Master of Science.

Abstract

Many state-of-the-art natural language understanding (NLU) systems are based on pretrained large language models (LLMs). These models make inferences using knowledge of various types observed at pretrain and inference time. However, the integration and reasoning abilities of NLU systems for different knowledge types from multiple knowledge sources have been largely understudied.

In order to evaluate these abilities systematically, we propose a test suite of coreference resolution tasks that require reasoning over multiple facts. We create three main dataset variants that vary in terms of which knowledge sources contain the relevant facts and evaluate state-of-the-art coreference resolution models on our dataset.

Our results show that with task-specific training and detailed annotations, some LLM-based NLU systems have the ability to reason on-the-fly over knowledge observed at pretrain and inference time. For the proposed task, the usefulness of knowledge in a source seems to depend on the knowledge type: background knowledge is more useful when drawn from pretrain-time parameters, while knowledge about specific entities seems to be better observed at inference time. However, performance generally is sensitive to a range of factors such as the underlying LLM architecture and annotation format.

Sommaire

De nombreux systèmes modernes de traitement automatique du language sont basés sur les modèles de langage massivement préentrainés. Les modèles de ce type font des prédictions en utilisant de l'information de différent types, obtenue au moment du pré-entrainement et de l'inférence. Cependant, les effets de l'origine et du type des informations sur l'intégration des connaissances et sur les capacités de raisonnement des modèles massifs de langages ont été sous-étudiées.

Afin d'évaluer systématiquement ces capacités, nous proposons une série de tests de résolution de coréférence qui nécessitent un raisonnement sur plusieurs faits. Nous créons trois variantes principales d'ensembles de données qui varient en fonction des sources de connaissances qui contiennent les faits pertinents et nous évaluons des modèles de résolution de coréférence de pointe sur notre ensemble de données.

Nos résultats montrent qu'avec une formation spécifique à cette tâche et des annotations détaillées, certains systèmes de compréhension du langage basés sur les modèles massivement préentrainés ont la capacité de raisonner à la volée sur les connaissances observées au moment du pré-entrainement et de l'inférence. Pour la tâche proposée, l'utilité de la connaissance dans une source semble dépendre du type de connaissance: les connaissances sur le contexte général sont plus utiles lorsqu'elles sont tirées des paramètres de pré-entrainement, tandis que les connaissance spécifiques à des entités semblent être mieux observée au moment de l'inférence. Cependant, la performance est généralement sensible à une série de facteurs tels que l'architecture des modèles massivement pré-entrainés sous-jacente et le format d'annotation.

Previously Published Material

The KITMUS test suite with experiments as motivated in Chapter 1 and described in Chapters 3 and 4 was previously published in the Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics as Arodi et al. (2023). As co-first author, Martin Pömsl contributed to all aspects of the dataset creation, experiment design, and evaluation.

All co-first authors have consented in writing to the use of the previously published material in this thesis. The PRETRAIN-TIME ENTITY-SPECIFIC variant as described in Section 3.2.3 and experiments using it are novel and the sole creation of Martin Pömsl. Together, the previously published KITMUS dataset and the new PRETRAIN-TIME ENTITY-SPECIFIC variant are referred to as KITMUS+ throughout this thesis.

Acknowledgments

First, I would like to thank my supervisor Jackie C. K. Cheung for his continued support and advice throughout my graduate studies. When I was an undergrad intern in his group, he gave me the amazing opportunity to come to Montreal and join his lab, for which I am very grateful. This work could not have happened without the many meetings in which he helped me plan a path through uncharted research waters.

I would like to thank my close research collaborator Akshatha Arodi. It was a joy to be working with her and her work ethic was always an example to me. I am also very grateful to my collaborators at Microsoft Research Kaheer Suleman, Adam Trischler, and Alexandra Olteanu, who helped shape the direction of my research projects through helpful advice and feedback.

In addition, I would like to thank my lab mates in Jackie's group for their constant support and all the fun we had during scrum meetings and at social events. A special thanks goes to Jules, who helped translate the abstract of this thesis to French.

I would also like to acknowledge the wider communities supporting me during my studies: the McGill Reasoning and Learning Lab, the McGill Computer Science Graduate Society, and the Mila Quebec AI Institute. Mila is where I spent most of my time and found many friends doing fun activities such as playing foosball, ultimate frisbee, chess, and Go. It's clear that without the compute resources and technical help provided by the extremely knowledgeable Mila IT staff, this research would not have been possible.

Last, I would like to thank my family for always being there for me throughout my studies.

Contents

List of Figures

List of Tables

Chapter 1

Introduction

Motivation

In recent years, advances in the pretraining of large language models (LLMs) have brought significant performance increases to a wide variety of downstream tasks for natural language understanding (NLU) systems (Raffel et al., 2020; Brown et al., 2020). With the increased availability of sophisticated NLU systems thanks to the public release of LLM weights (Touvron et al., 2023) and the provision of commercial APIs (OpenAI, 2023), the need for evaluating these systems' abilities and limitations is becoming more pronounced.

Many NLU tasks that these systems are deployed for require reasoning over knowledge (Guu et al., 2020; Petroni et al., 2021). This is true not only for inherently knowledge-intensive tasks such as the question answering and fact verification (Liu et al., 2021), but also for many other downstream tasks relevant for real-world applications (Piktus et al., 2022).

Given the large size of modern LLMs, one approach to provide NLU systems with the relevant knowledge is to exploit knowledge memorization during pretraining. This approach initially led to considerable success for downstream tasks such as question

answering (Roberts et al., 2020), but reaches its limits for tasks that require facts that have not been observed during pretraining because of domain or recency differences (Xu et al., 2023). An alternative to relying solely on pretrain-time knowledge is providing additional knowledge explicitly as part of the inference-time inputs, as implemented in the popular *retrieve-then-predict* paradigm (Lewis et al., 2020).

Consider the passage "John saw the newly elected president on TV." Pretrained parameters can conceivably contain information about what presidents do and what a TV is, but they cannot contain reliable knowledge about who John is (since "John" is an instance-specific identifier) or who the president is (since the president might have changed since pretraining). It follows that successful systems for knowledge-intensive NLU tasks require the ability to integrate both pretrain-time and inference-time knowledge.

To effectively use these two knowledge sources, models must (1) retrieve relevant information from both knowledge sources, (2) adjudicate between potentially conflicting information, and (3) integrate multiple units of information and reason over them on the fly. For example, pretrain-time parameters might contain the knowledge that Donald Trump is the president of the United States, but inference-time inputs might state that Joe Biden is the president. Based on the contextual information available in a task, models must infer the correct president.

Recent work by Longpre et al. (2021) examines the effects of knowledge conflicts across multiple knowledge sources. In this work, we aim to more broadly investigate the behaviour of NLU systems faced with tasks requiring both pretrain-time and inference-time knowledge. While Longpre et al. (2021) study how models handle conflicting facts, our goal is to evaluate whether models can combine complementary knowledge drawn from multiple sources rather than choose between sources.

In order to facilitate systematic evaluation of these knowledge integration capabilities, we propose a dataset for the task of knowledge-intensive coreference resolution.

Caplinger is a food preparation worker. **Berkowitz** is a labourer. **Eells** is a judge. **Thedford** is a secretary. At the improvisation class, **Berkowitz**, **Eells**, **Caplinger**, and **Thedford** started a conversation. The classes usually begin before work. **He** told anecdotes from a career of typing letters and keeping records for a company.
[Answer: **Thedford**]

Figure 1.1 Example from KITMUS+. To resolve the pronoun (in red), a model needs to draw on entity-specific knowledge about an entity's occupation as well as on background knowledge about what kind of work the occupation entails.

We select the coreference resolution task as an instance of NLU tasks since it is has an extensive history as a test bed for reasoning over knowledge (Levesque et al., 2012; Rahman and Ng, 2012; Durrett and Klein, 2013) and is a downstream task that is often approached with LLM-based NLU models such as BERT (Devlin et al., 2019) and ELMo (Peters et al., 2018).

We design the task such that the resolution of pronouns in our dataset requires two types of knowledge:

- Entity-specific knowledge such as "Ruth Bader Ginsburg is a judge."
- Background knowledge such as "Judges decide cases in courts of law."

Generally, background knowledge is learned during the pretraining of LLMs at pretrain-time, while entity-specific knowledge is typically observed at inference time. In our dataset, we vary the availability of both required knowledge types such that they may be available either as pretrain-time or inference-time knowledge. An example from our dataset where entity-specific knowledge is provided at inference time and background knowledge is obtained at pretrain time is shown in Figure 1.1.

Statement of Contributions

In this work, we propose a task and dataset for the evaluation of Knowledge InTegration from MUltiple Sources: the KITMUS+ test suite. We describe the rationale as well as the process for the creation of this dataset. We systematically choose and control the availability of knowledge of different types (background and entity-specific) in different sources (pretrain-time or inference-time). Using this dataset, we evaluate the ability of established LLM-based coreference resolution models to reason over knowledge available in different sources and report the results of several ablation experiments. The KITMUS+ test suite is publicly available on GitHub[1].

We find that with task-specific training and detailed annotations, some of the evaluated coreference resolution systems have the ability to reason on-the-fly over knowledge observed at pretrain and inference time. For the proposed task, the usefulness of knowledge in a knowledge source seems to depend on the knowledge type: background knowledge is more useful when drawn from pretrain-time parameters, while knowledge about specific entities seems to be better observed at inference time. However, performance generally is sensitive to a range of factors such as the annotation format and the underlying LLM's size and architecture. We do not find consistent benefits to providing knowledge redundantly both at pretrain and inference time.

Organisation of This Work

In the following, we first provide an overview of concepts and relevant strains of work in Chapter 2. Building on this, we propose a task and dataset for systematically evaluating knowledge integration in Chapter 3. We describe experiments that evaluate established models on the dataset in Chapter 4 and discuss the implications. Finally, we summarise our conclusions and provide an outlook on future work in Chapter 5.

[1] https://github.com/mpoemsl/kitmus/tree/kitmus-plus

Chapter 2

Background

In this chapter, we review literature related to two relevant strains of work in recent years: approaches to evaluating the knowledge integration capabilities of NLU systems and the use of the task of coreference resolution as a test bed for reasoning over knowledge. These strains are relevant to this work since it presents a novel approach to utilize the task of coreference resolution to evaluate the knowledge integration capabilities of NLU systems.

While presenting these strains, we also highlight concepts and works that were used in the experiments for this work such as the underlying architecture of the evaluated coreference resolution systems (Lee et al., 2017) and approaches to probing for pretrain-time knowledge in LLMs (Petroni et al., 2019).

2.1 Knowledge Integration in NLU Systems

Many NLU tasks require the integration of knowledge for successful completion. This is intuitively the case for downstream tasks like open-domain question answering (Roberts et al., 2020) that require the generation of true facts given only a query, but additional knowledge can also be required to successfully tackle tasks like fact ver-

ification (Thorne et al., 2018) and reading comprehension (Long et al., 2017). In the following we provide an overview of the dimensions along which knowledge integration is commonly analyzed as well as approaches to knowledge integration evaluation explored in previous literature.

2.1.1 Dimensions of Knowledge Integration

Knowledge integration is often analyzed along two dimensions: the source and the type of knowledge being integrated by a NLU system.

Knowledge Types

There are different types of knowledge that may be required for NLU tasks. While there is no clear consensus about a taxonomy or terminology of knowledge types for NLU tasks, previous works often make a distinction between entity-specific knowledge and background knowledge as described in the following (Petroni et al., 2019; Lauscher et al., 2020; Onoe et al., 2021).

Entity-specific knowledge conveys information about specific entities. It is typically presented in the form of `is-a` relations such as "Ruth Bader Ginsburg is a judge." This type of knowledge is variously also called "entity knowledge" or "factual knowledge". Resources like Wikipedia and Wikidata (Farda-Sarbas and Müller-Birn, 2019) attempt to capture this knowledge systematically in relational knowledge bases. With T-REx, Elsahar et al. (2018) present a resource that provides access to this kind of knowledge in a form amenable to NLU systems. Recent work emphasising the importance of entity-specific knowledge in NLU systems includes Chen et al. (2021) and Heinzerling and Inui (2021).

Background knowledge is information about relations that are commonly true in the world. It is typically presented in the form of rules such as "judges decides cases in courts of law" or hyponympy/hypernymy relations such as "every X is a Y." This type

of knowledge is variously also called “common sense knowledge” or “world knowledge”, but the latter term is sometimes also applied to entity-specific knowledge. Resources like ConceptNet (Speer et al., 2018) and ATOMIC (Sap et al., 2019) attempt to capture this type of knowledge. Recent work about the role of background knowledge in NLU systems includes Lin et al. (2020) and Porada et al. (2022).

Knowledge Sources

Progress on many NLU tasks has recently been driven by improvements in pretrained LLMs, which can be adapted to specific tasks via finetuning (Peters et al., 2018; Devlin et al., 2019; Raffel et al., 2020; Touvron et al., 2023; OpenAI, 2023). The fundamental idea of LLMs is to learn representations that are useful for predicting missing text (Jurafsky and Martin, 2023). These learned representations have been found to capture many aspects of the semantics of the input text (Tenney et al., 2019). Language modeling as an optimization objective is amenable to large-scale training since it does not require annotated data - the training data can be created from large collections of unstructured texts such as C4 (Raffel et al., 2020), The Pile (Gao et al., 2020), or ROOTS (Laurençon et al., 2022).

BERT (Devlin et al., 2019) and ELMo (Peters et al., 2018) were among the first influential LLMs to result in large gains on NLU tasks. While BERT is built using the Transformer architecture (Vaswani et al., 2017) , which is the underlying architecture of many modern successful LLMs (Brown et al., 2020; Touvron et al., 2023), ELMo is based on recurrent LSTMs (Hochreiter and Schmidhuber, 1997). Both are often incorporated into task-specific models for NLU tasks and finetuned with considerable success (Rogers et al., 2020).

LLM-based NLU systems must draw on a variety of knowledge sources to make successful inferences. These knowledge sources can be categorized into two classes based on the time of observation, which we call knowledge sources: pretrain-time

knowledge and inference-time knowledge.

Pretrain-time knowledge is knowledge acquired during language modeling pretraining and stored in parameters. This knowledge source is also called "parametric knowledge" in related literature. LLMs such as BERT (Devlin et al., 2019) and ELMo (Peters et al., 2018) memorize a considerable amount of knowledge in their parameters, which has been the subject of extensive studies such as Petroni et al. (2019) and Kassner and Schütze (2020). However, this pretrain-time knowledge is necessarily limited in scope by the parameter count of the LLM (Roberts et al., 2020) and may not be applicable to the domain of the task at hand (Xu et al., 2023). Additionally, since this knowledge was observed at the time of LLM pretraining, it can quickly become outdated. The updating of knowledge stored in a LLM's parameters is the subject of active work (De Cao et al., 2021; Meng et al., 2022), but not yet sufficiently robust to be useful for many applications (Hoelscher-Obermaier et al., 2023).

Inference-time knowledge is knowledge supplied at inference time as part of the textual inputs of a LLM. This knowledge source is also called "contextual knowledge" in related literature. In recent work, LLMs have been shown to benefit greatly from task-specific knowledge imbued through few-shot demonstrations at inference time (Brown et al., 2020; Wei et al., 2022a). Additionally, a common approach to address the shortcomings of pretrain-time knowledge is to complement it by retrieving up-to-date and relevant texts and providing them at inference time to LLMs (Guu et al., 2020; Lewis et al., 2020; Piktus et al., 2022).

2.1.2 Evaluating Knowledge Integration

Evaluating pretrain-time knowledge integration has been the subject of extensive study once the usefulness of LLMs such as BERT (Devlin et al., 2019) for downstream tasks became apparent.

One influential approach to evaluate LLMs' ability to retrieve memorized knowl-

edge is the LAMA probe (Petroni et al., 2019). They propose a "fill-in-the-blank" cloze infilling task that treats LLMs such as BERT and ELMo as knowledge bases for relational knowledge. In their work, a LLM is considered to have access to a `(subject, relation, object)` triple such as `(Dante, born-in, Florence)` if it can accurately predict the masked object in a cloze statement such as "Dante was born in ___."

To compute the completion of the blank in a way that is comparable across different model architectures, the authors follow the natural training objective of the LLMs. BERT is a masked language model (Devlin et al., 2019), which means its training objective is to predicted the correct replacement of a `[MASK]` token from the surrounding non-masked tokens in a text. Accordingly, the authors query for the completion of a blank by masking the corresponding token and following the decoding procedure of BERT's language modeling head. ELMo on the other hand is a bidirectional language model (Peters et al., 2018), meaning it separately attempts to predict a missing token from the left-hand context in the forward direction and the right-hand context in the backward direction. Following the objective defined by Peters et al. (2018), the LAMA authors average the forward and backward probabilities from the corresponding softmax layers to make a prediction for the blank.

One limitation of the LAMA probe is that it is restricted to single-token answers, since multi-token decoding would introduce additional confounding factors in the form of beam search hyperparameters obscuring the knowledge retrieval evaluation. The authors find that LLMs such as BERT are able to reproduce both background knowledge as captured in ConceptNet (Speer et al., 2018) and knowledge about specific entities as captured in T-REx (Elsahar et al., 2018).

In their follow-up work, Kassner and Schütze (2020) use the same methodology to infer completions for prompts regarding language use phenomena observed in humans. They find that in contrast to humans, the evaluated LLMs are prone to make

contradictory completions (e.g. predict "fly" as completion for both "Birds can ___" and "Birds cannot ___") and sensitive to misprimes such as "Talk? Birds can ___", for which LLMs predict the completion "talk". They found that the ability of LLMs to retrieve facts from pretrain-time parameters observed by Petroni et al. (2019) is sometimes brittle and not necessarily ideal for reasoning over all types of knowledge (Kassner and Schütze, 2020).

Evaluating inference-time knowledge integration has gained importance primarily in the context of retrieval-based methods, which address the shortcomings of pretrain-time knowledge (staleness, lack of coverage) by retrieving relevant texts for a query and providing it to the model at inference time (Guu et al., 2020; Lewis et al., 2020; Piktus et al., 2022).

A systematic approach to evaluating LLMs' ability to integrate inference-time knowledge is proposed in the KILT benchmark for knowledge-intensive language tasks (Petroni et al., 2021). They present a collection of NLU tasks grounded in a large Wikipedia corpus to facilitate research on models that must access specific information to solve a task. KILT is an "in-KB" resource in that the evidence required to answer each of the 3.2M queries is contained within the provided corpus. They evaluate a number of baselines, among them systems that rely only on pretrain-time knowledge as well as those that retrieve and provide knowledge from the corpus at inference time using methodology proposed by Karpukhin et al. (2020). The authors find that inference-time knowledge integration leads to much better downstream performance for the proposed tasks.

Evaluating knowledge integration from multiple sources is a largely understudied subject. One exception is the work by Longpre et al. (2021) on model behavior when faced with conflicting entity-specific knowledge from multiple sources. They achieve this by modifying existing question answering datasets that are commonly tackled using retrieval-based methods. The authors identify a subset of instances

that require entity-specific knowledge contained both in the pretrain-time parameters and inference-time inputs. For this subset, they substitute the relevant entity in the inference-time inputs, thereby creating a knowledge conflict between the two sources. The authors find that established LLM-based question answering systems tend to have an overreliance on memorized pretrain-time knowledge and propose finetuning on their dataset to mitigate the issue. Follow-up work further investigating conflicts in knowledge sources for question answering includes Chen et al. (2022) and Neeman et al. (2023). While this line of work focuses on model behavior when faced with conflicting facts from different sources, our work investigates systems' ability to reason over complementing information from pretrain-time and inference-time knowledge sources.

2.2 Coreference Resolution as Reasoning over Knowledge

Coreference resolution is a task that has an extensive history as a test bed for reasoning over knowledge. In the following, we first provide an overview of the task of coreference resolution and common annotation formats. We then describe established methods for coreference resolution and zoom in on previous work related to reasoning-based coreference resolution.

2.2.1 Task Definition

Coreference resolution is the task of determining whether two mentions refer to the same entity (Jurafsky and Martin, 2023). The mentions may be noun phrases, names, pronouns, or other referring expressions. In modern literature (and in this work as well), coreference resolution is often considered to be an end-to-end task that also includes the detection of mentions, e.g. whether a span of tokens refers to any entity at all. While mention detection is often considered a comparatively easy step, it can introduce an additional error source for end-to-end coreference resolution systems.

There are different types of task formulations in coreference resolution. In general coreference resolution, which is annotated for example in the canonical dataset OntoNotes (Hovy et al., 2006), coreferences between all types of mentions are considered. Querying for general coreference resolution is often realized as assigning clusters to token spans, since this allows for the annotation of multiple possibly nested coreferences. Pronoun coreference resolution on the other hand is often restricted to name-pronoun coreferences and can be posed as a binary classification task, as is the case in the Gendered Ambiguous Pronouns (GAP) dataset proposed by Webster et al. (2018). Other datasets such as Quoref (Dasigi et al., 2019) pose coreference resolution as a question answering task that requires span selection in response to a specific question.

2.2.2 Annotation Formats

The various task formulations are also reflected in different annotation types. The CoNLL 2012 format (Pradhan et al., 2012), which is used to annotate the coreference clusters in OntoNotes, contains token and sentence boundaries, Penn Treebank POS tags (Marcinkiewicz, 1994), and gold coreference clusters for all entity mentions. This means that all mentions of an entity are exhaustively annotated in a single cluster. Models that operate on the CoNLL format predict these clusters, which involves both detecting mentions and clustering them.

In contrast, GAP (Webster et al., 2018) uses a tab-separated value format which allows for the annotation of only two entities and only one mention per entity (excluding the pronoun), so mentions of other entities or additional mentions of the same entities remain un-annotated. Models that operate on the GAP format are presented with exactly two mentions and for each of them make a binary decision whether or not they are corefering with a pronoun. The GAP format task is more restricted in that models do not have to detect mentions and there are at most two entities per instance.

2.2.3 Methods for Coreference Resolution

Various methods have been proposed for coreference resolution, ranging from the rule-based algorithm proposed by Hobbs (1977) to machine learning methods with hand-engineered features by Ng and Cardie (2002) to the first neural end-to-end system by Lee et al. (2017).

End-to-end neural coreference resolution: In their influential work, Lee et al. (2017) propose to leverage neural text representations for coreference resolution using a mention ranking architecture. Their architecture considers all spans up to a maximum length as mention candidates. They prune the space of mentions using unary scoring based on learned span representations and then infer a distribution $P(y)$ over possible antecedents spans $y \in Y(x)$ for each mention span x.

This distribution $P(y)$ is defined as the softmax over pairwise coreference scores $s(x, y)$:

$$P(y) = \frac{e^{s(x,y)}}{\sum_{y' \in Y(x)} e^{s(x,y')}}$$

The pairwise coreference scoring function s consists of the unary mention scorer $s_m(\cdot)$ and the binary antecedent scorer $s_a(x, y)$. Using these scorers, the pairwise coreference score $s(x, y)$ is defined as:

$$s(x,y) = \begin{cases} 0 & \text{if } y = \epsilon \\ s_m(x) + s_m(y) + s_a(x,y) & \text{if } y \neq \epsilon \end{cases}$$

ϵ is here a dummy antecedent which signifies that a mention does not have a corresponding antecedent. By fixing $s(x, \epsilon) = 0$, the authors ensure that coreference is only predicted if at least one non-dummy score is higher than zero.

The scoreres s_m and s_a are non-linear mappings from span representation vectors to scalar scores with learned weights.

For any vector span representation x, $s_m(x)$ is defined as follows:

$$s_m(x) = w_m \cdot \text{FFNN}_m(x)$$

Here $\cdot$ denotes the dot product and w_m is a learned weight vector. FFNN denotes a feedforward neural network with ReLU (Glorot et al., 2011) activation and a learned weights matrix.

For any two vector span representations x and y, $s_a(x, y)$ is defined as follows:

$$s_a(x,y) = w_a \cdot \text{FFNN}_a([x, y, x \odot y, \phi(x,y)])$$

Here $\odot$ denotes the element-wise product and w_a is a learned weight vector. $\phi(x, y)$

is a feature vector encoding metadata such as speaker, genre, and position distance between spans x and y in the text.

The system predicts the most likely clustering given the inferred antecedent distributions for all mentions in a text. Using supervision of gold coreference clusters, the model weights are learned by optimizing the marginal log-likelihood of the correct antecedents.The authors find that the system can successfully learn to generate useful mention candidates and determine their antecedents (Lee et al., 2017).

Higher-order coreference resolution: In a later refinement, Lee et al. (2018) introduce an iterative procedure to improve span representations by including information about the current expected antecedent for each mention a_x^t at time t. This is implemented via an attention-like sum of antecedent representations scaled by the current inferred antecedent distribution $P^t(y)$:

$$a_x^t = \sum_{y \in Y(X)} P^t(y) \cdot x^t$$

The mention representation for the next iteration x^{t+1} is then computed via weighted interpolation between a_x^t and x^t. This aims to ensure that the predicted clusters are not only locally but also globally consistent by propagating information from each antecedent prediction to all others, thereby making the prediction process higher-order. In their experiments, Lee et al. (2018) find that while second-order coreference with $t \in (1, 2)$ yields improvements, the performance increases become diminishingly small for $t > 2$.

Coarse-to-fine pruning: Since the iterative higher order process is much more computationally intensive, Lee et al. (2018) also introduce a mention pruning mechanism by including a computationally less demanding pairwise score $s_c(x, y)$ as an additional summand in the definition of $s(x, y)$. The reduction in resource usage comes from computing the full sum $s(x, y)$ lazily only for those pairs (x, y) which already show high

scores in the summands $s_m(x)$, $s_m(y)$, and $s_c(x, y)$. Through the use of this pruning heuristic, the higher-order inference procedure becomes computationally feasible.

Modern coreference resolution systems: Today's state-of-the-art coreference resolution systems are still largely built on the mention ranking architecture proposed in Lee et al. (2017) and refined in (Lee et al., 2018), but make use of better span representations derived from LLMs as showcased in C2F (Lee et al., 2018) which uses ELMo (Peters et al., 2018) and BERT4Coref (Joshi et al., 2019) which uses BERT (Devlin et al., 2019). As more powerful LLMs come available, their representations also lead to better coreference resolution performance, as exemplified by the recent adaptation of T5 (Raffel et al., 2020) for coreference resolution by Porada et al. (2023). This shows that coreference resolution is a NLU task to which task-specific models based on LLMs are well-suited.

External knowledge for coreference resolution: Prior work has shown that integrating world knowledge can lead to improvements in coreference solvers, emphasising the need for knowledge beyond that contained in the pretrained parameters. Bean and Riloff (2004) learn caseframe co-occurrence statistics, which they use to predict coreference. Rahman and Ng (2012); Zhang et al. (2019); Aralikatte et al. (2019); Emami et al. (2019) showed improved results using external knowledge supervision.

2.2.4 Reasoning for Coreference Resolution

There is a large body of work studying the exploitation of linguistic knowledge about shallow cues such as gender, position, and number cues for naturally occurring coreference resolution as annotated in OntoNotes (Durrett and Klein, 2013). Other approaches incorporate additional properties like semantic roles as features on the unrestricted task of general coreference resolution (Baker et al., 1998; Chambers and Jurafsky, 2009).

In a departure from this, the Winograd Schema Challenge (WSC) (Levesque et al., 2012; Rahman and Ng, 2012) posed coreference resolution as a semantic reasoning

task and inspired a number of specialized datasets such as GAP (Webster et al., 2018) and Winogrande (Sakaguchi et al., 2020) where coreference resolution is used as a test bed for reasoning over knowledge and cases cannot be solved with shallow features (Emami et al., 2019; Rahman and Ng, 2012). WSC is considered an established format for testing reasoning abilities, as demonstrated by its inclusion in the standard benchmarks GLUE (Wang et al., 2018) and BigBench (Srivastava et al., 2023) recast as natural language inference.

Recent work has shown that the mention ranking architecture established by Lee et al. (2017) paired with modern LLMs is well suited for both general coreference resolution as required for OntoNotes and reasoning-based coreference resolution as required for WSC (Toshniwal et al., 2021). However, generalization from one task to the other is generally poor (Porada et al., 2023), which emphasises the need for task-specific finetuning to effectively address reasoning-based tasks such as the one presented in this work.

Chapter 3

Evaluating Knowledge Integration

In order to facilitate systematic evaluation of knowledge integration capabilities, we propose a dataset for the task of knowledge-intensive coreference resolution. We choose coreference resolution as an instance of NLU tasks since it is has an extensive history of use as a test bed for reasoning over knowledge and is a downstream task that is often approached with models based on LLMs such as BERT (Devlin et al., 2019) and ELMo (Peters et al., 2018), for which the notion of pretrain-time knowledge is well defined.

In this chapter, we describe the design and implementation of the proposed test suite as well as its three main variants. Finally, we describe the procedures we used to validate the resulting dataset.

3.1 The KITMUS Test Suite

3.1.1 Overview

We evaluate the knowledge integration capability of coreference resolution models from two different knowledge sources as described in Section 2.1.1:

- **Pretrain-time**: knowledge accumulated in the parameters during LLM pretraining

- **Inference-time**: knowledge observed as part of the input text or prompt

To design KITMUS, we formulate a reasoning-based coreference resolution task which requires access to two facts. This can be viewed as an instance of two-hop reasoning. We systematically vary the presence of these facts across the knowledge sources to evaluate the models.

As an instantiation of the idea of presenting two facts, we experiment with the following two knowledge types as described in Section 2.1.1:

- **Entity-specific**: occupation of an entity such as "Rosenow is an architect."
- **Background**: situation typical for an occupation such as "architects design buildings and houses."

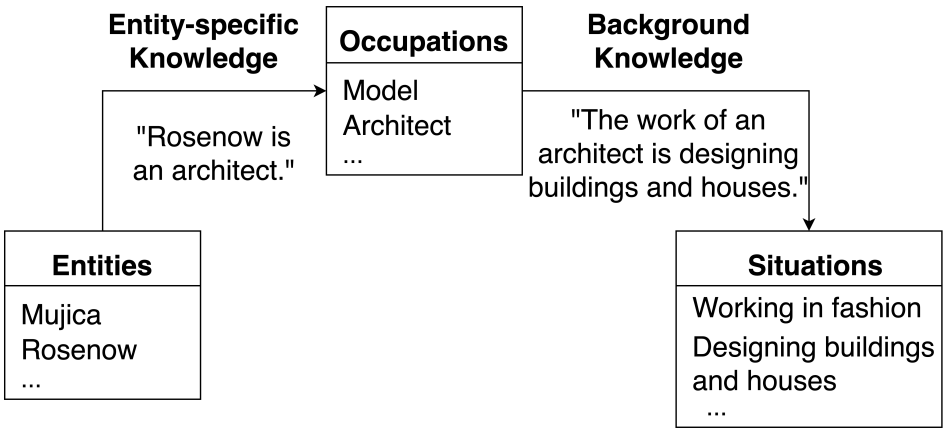


Figure 3.1 Schema of knowledge types in KITMUS+.

For example, consider the following task to predict whether Mujica or Rosenow is the correct antecedent of the pronoun "he."

> Mujica is a model. Rosenow is an architect. At the bus station, **Mujica** and **Rosenow** connected. Public transports are eco-friendly. **He** shared experiences from a career of designing buildings and houses. [Answer: **Rosenow**]

Here, the occupations are *model* and *architect*, and the situational cue is *designing building and houses*. Both knowledge types are required in order to resolve this coreference. An illustration of this knowledge schema can be found in Figure 3.1.

Each instance of the KITMUS task consists of two fragments of text that are concatenated: 1) a knowledge text—containing the inference-time knowledge that models are given access to—and 2) a task text—consisting of the coreference task that models solve.

3.1.2 Creation

To construct KITMUS, we manipulate which entities are mentioned in each instance, what occupations those entities have, what situations those occupations pertain to, what contexts they are mentioned in, and whether noise is present. Each entry is structured to first (1) introduce the entities, (2) then place them in the same location, and (3) finally, place one of them in a situation related to their occupation. If additional knowledge is given at inference-time, this is preceded by a knowledge text containing that knowledge.

The dataset entries are generated using hand-crafted English-language templates and sampling from a variety of resource pools to fill the template slots. The use of templates facilitates control over the source a certain type of knowledge is stored in, which may not be possible to do with a natural dataset.

We aim to minimize the likelihood of models learning to exploit any spurious correlations in the templates or resources and to promote data diversity using the following methods:

- We use multiple templates for each sentence. Examples are shown in Table 3.1.
- We sample from diverse resource pools to fill template slots as detailed in Section 3.1.3.

- We include location-dependent noise statements that act as distractors and serve to vary the distance between entities.
- We create canonical train, validation, and test splits for each variant that are generated using disjunct subsets of templates and resources.

With these measures, we ensure that all entity names, occupations, situations, locations, templates, and noise statements that occur in the test instances do not occur in the train instances.

At {location}, {mentions} met.
At {location}, {mentions} ran into each other.
At {location}, {mentions} started a conversation.
At {location}, {mentions} came across each other.
At {location}, {mentions} encountered each other.
At {location}, {mentions} bumped into each other.
At {location}, {mentions} connected.
{mentions} met at {location}.
{mentions} ran into each other at {location}.
{mentions} started a conversation at {location}.
{mentions} came across each other at {location}.
{mentions} encountered each other at {location}.
{mentions} bumped into each other at {location}.
{mentions} connected at {location}.

(a) Meet Sentence Templates

After a long day at work {situation}, {pronoun} was happy to relax.
{pronoun} told anecdotes from a career of {situation}.
{pronoun} reflected on whether {situation} for a living was a good career choice.
When a question related to {situation} arose, {pronoun} offered a professional opinion.
{pronoun} was relieved to unwind after a demanding day at work {situation}.
{pronoun} was glad to unwind after a long day at work {situation}.
{pronoun} shared experiences from a career of {situation}.
{pronoun} pondered whether choosing {situation} as a career was a wise decision.

(b) Pronoun Sentence Templates

Table 3.1 Templates used to introduce ("Meet Sentence") and refer to ("Pronoun Sentence") entities in KITMUS+.

3.1.3 Resources

We collect $20{,}000$ last names as entities, 60 common occupations and their associated job descriptions as situations and 112 common meet-up places as locations from a mix

of governmental and other publicly available resources.

Pronouns are sampled randomly from both the gendered pronouns `he` and `she` as well as gender-indefinite pronouns such as singular `they` and the neopronouns `ey` and `ze`. In doing so, we aim to follow the practices established by the gender-inclusive coreference resolution dataset `GICoref` (Cao and Daumé III, 2020). Ideally, we would want the distribution of pronouns to approximate the frequency in naturally occurring text, but few reliable statistics exist to estimate them. We include 40% `he`, 40% `she`, 10% `they`, and 10% neopronouns.

Noise statements are sampled randomly from a collection of statements based on the selected location in order to maintain the natural flow of the text. Each location is associated with 25 noise sentences. These sentences are automatically generated using GPT-2 (Radford et al., 2019) and then manually verified by the authors not to include cues related to any entity or occupation.

Entity Names are sampled from a pool of the $20{,}000$ most frequent last names in the 2010 U.S. census.[1] We use last names as entity names in order to avoid introducing gender-related cues. We discard those last names that are also first names. The order of occurrence of entity names within a template is also randomized. We assume that there is no confounding pretrain-time knowledge based on common entity last names in the models.

Occupations consist of a curated list of 60 common occupations compiled by scraping a career website[2] and the US Labor census data.[3] Following Cao and Daumé III (2020), we remove referential gender cues from occupations such as "man" in "fireman." Jobs pertaining to very specific domains or related to one of the locations where entities meet are removed from the list.

Situations are assembled using the occupation descriptions of the scraped occupa-

[1] `https://www.census.gov/topics/population/genealogy/data/2010_surnames.html`

[2] `https://ca.indeed.com/career-advice/finding-a-job/common-jobs`

[3] `https://www.bls.gov/emp/tables/emp-by-detailed-occupation.htm`

tions. We manually filter the pairs of descriptions that are semantically similar, such as descriptions of an accountant and an analyst.

Locations are derived from a curated list of 112 locations scraped from a website of common meet-up places.[4] We manually filter out locations that could provide inadvertent surface cues related to the entities' occupation, nationality, or gender.

3.1.4 Format

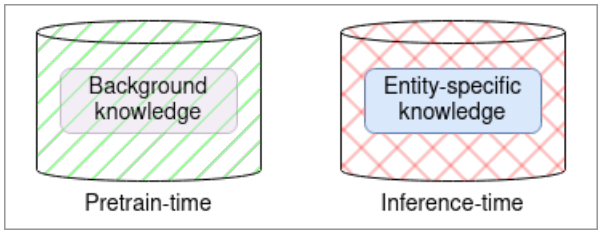
Each variant in KITMUS consists of a train, validation, and test split with 2000, 400, and 2000 examples respectively. The size of KITMUS is comparable to that of the GAP dataset (Webster et al., 2018), which similarly tests for a specific phenomenon in ambiguous pronoun coreference resolution. Ablation experiments with a larger train set size can be found in Section 4.3.1.

We create subtasks with two, three, and four entities for each variant. In this work, we mostly consider the four entity subtask by default, since it is the most challenging subtask. Ablation experiments with fewer entities can be found in Section 4.3.1.

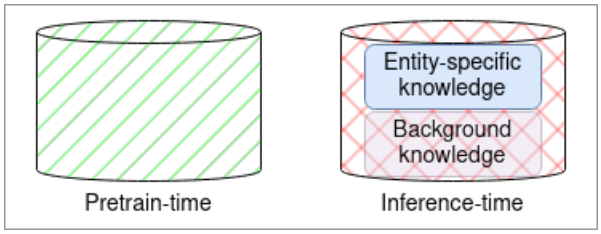
The test suite is provided in two different formats which are commonly used by state-of-the-art coreference solvers: the CoNLL 2012 format (Pradhan et al., 2012) and the GAP format (Webster et al., 2018). The CoNLL 2012 format allows for the comprehensive annotation of all mentions of an entity including in the knowledge text. The GAP format, however, allows for the annotation of only two entities and only one mention per entity. In this work, we use the CoNLL 2012 format by default, since it has the most informative annotations. Ablation experiments with the GAP format can be found in Section 4.3.3.

[4] https://www.happierhuman.com/meet-new-people/

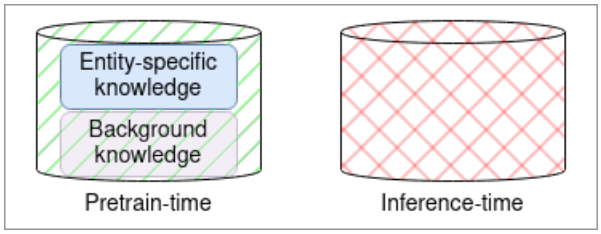
3.2 KITMUS+ Variants



(a) BASE



(b) INFERENCE-TIME BACKGROUND



(c) PRETRAIN-TIME ENTITY-SPECIFIC

Figure 3.2 Variants of KITMUS+ by mapping of knowledge types to knowledge sources.

The three main variants of KITMUS+ represent three mappings of knowledge types to knowledge sources.

- BASE: Background knowledge is pretrain-time and entity-specific knowledge is inference-time
- INFERENCE-TIME BACKGROUND: Both background and entity-specific knowledge are inference-time

- PRETRAIN-TIME ENTITY-SPECIFIC: Both background an entity-specific knowledge are pretrain-time

The BASE and INFERENCE-TIME BACKGROUND variants are part of the previously published KITMUS test suite. Together with the new PRETRAIN-TIME ENTITY-SPECIFIC variant, they constitute the KITMUS+ test suite. An illustration of the three main variants of KITMUS+ is shown in Figure 3.2.

Given that the models evaluated in this work incorporate canonical weights of pretrained LLMs, it is not possible to create a variant where background knowledge is inference-time and entity-specific knowledge is pretrain-time, since that would require the pretraining data to contain real-world entities with fictional occupations. However, we believe these three variants to be sufficient to shed light on the behavior of the evaluated models in a variety of settings.

3.2.1 Base

In the BASE variant, entity-specific knowledge is provided at inference time and background knowledge about occupations is assumed to be pretrain-time knowledge. This is the setting most in line with the language modeling training of LLMs, since there are far more entities than occupations in the real-world and the probability of encountering a new named entity at inference time is much higher than encountering a new occupation.

The entity-specific knowledge is provided in the knowledge text via with a template mapping entities to their respective occupations using the phrase "is a." The entities are fictional and only identified via their last name, which ensures that LLMs cannot have observed their occupation during pretraining. An example from this variant:

> Fresquez is a secretary. Frates is a politician. Horner is a newsreader. Barlett is a bookkeeper. At the power yoga class, **Barlett**, **Frates**, **Fresquez**, and **Horner** came across each

> other. A yoga class helps live a happier life. **She** told anecdotes from a career of seeking an elected seat in government. [Answer: **Frates**]

With this variant, we aim to evaluate whether models have the ability to integrate and reason over both pretrain-time and inference-time knowledge effectively.

3.2.2 Inference-Time Background (ITB)

In order to evaluate whether a model can solve the proposed task using exclusively inference-time knowledge (i.e., in the absence of pretrain-time knowledge), we introduce fictional "knowledge." Fictional knowledge such as "the work of a mornisdeiver is gupegaing advaily" is unlikely to have been observed during pretraining, in contrast to real-world knowledge which is likely to have been observed. As in the BASE variant, the entities are fictional, ensuring that entity-specific knowledge about them was not observed at pretrain time. Thus, in this variant, both knowledge types are fictional and not contained in the pretrained parameters. An example from this variant:

> The work of a towcer is lopening ackly. The work of a vangiwer is aughuing ominly. Yoshimura is a contaker. Rhoads is a towcer. The work of an agovember is rethiling orsuly. Cobian is a vangiwer. Kutz is an agovember. The work of a contaker is acmastatting rigeorly. **Yoshimura**, **Kutz**, **Cobian**, and **Rhoads** ran into each other at the communal dining restaurant. The coffee cake was quite good. After a long day at work acmastatting rigeorly, **he** was happy to relax. [Answer: **Yoshimura**]

Background knowledge about occupations maps occupations to situations that are typical for the occupation, such as "astronomer" and "studying the stars and the universe." To make background knowledge fictional, either the occupation, the situation, or both have to be fictional. For situations, we furthermore distinguish between levels of fictionality: 1) character-level fictional situations that use novel words and 2) word-level fictional situations that use existing words but describe novel occupations.

Example texts resulting from different forms of fictionality can be seen in Table 3.2.

Occupation	Situation	Example
Real	CharFict	The work of a *politician* is *ehemting smorbtly*. Chichester is a politician[...]
Real	WordFict	The work of a *politician* is *controlling the pool of an aircraft by using its directional flight controls*. Chichester is a politician[...]
CharFict	Real	The work of a *mirituer* is *seeking an elected seat in government*. Chichester is a mirituer[...]
CharFict	CharFict	The work of a *mirituer* is *ehemting smorbtly*. Chichester is a mirituer[...]
CharFict	WordFict	The work of a *mirituer* is *controlling the pool of an aircraft by using its directional flight controls*. Chichester is a mirituer. [...]

Table 3.2 Different combinations of fictional occupations and situations in the INFERENCE-TIME BACKGROUND variant.

Creation: To create fictional background knowledge that maps occupations to situations, we create fictional occupations and fictional situations. Following the methodology of Malkin et al. (2021), we generate 60 names of fictional occupation by sampling from a character-level LSTM language model.

We generally follow the methodology of Malkin et al. (2021) in creating fictional occupations and situations. To bias the model towards strings that can be used as occupation names, we train it on a reversed sequence of characters and prompt with the suffix `er`. We manually filter the words to eliminate unpronounceable or pre-existing English words.

We employ the following two methodologies to generate fictional situations: 1) character-level fictional situations—like the fictional occupations—are generated with the suffix prompts `ing` and `ly`, and 2) word-level fictional situations are generated by randomly shuffling existing words with the same POS tags across real situation descriptions followed by manual filtering based on semantic plausibility.

Since there are no obvious advantages to either combination of fictionality, we report results for the INFERENCE-TIME BACKGROUND variant as a range over all five subvariants.

3.2.3 Pretrain-Time Entity-Specific (PTES)

In addition to the variants provided in the original KITMUS test suite, we create a PRETRAIN-TIME ENTITY-SPECIFIC variant where the entity-specific knowledge is not provided at inference time, but already contained in the parameters at pretrain-time. We achieve this by using the names of well-known real-world entities along with their true occupations and pronouns that LLMs are likely to have observed during pretraining. As in the BASE variant, the background knowledge is assumed to be contained in the pretrain-time parameters as well. Thus, in this variant, both knowledge types are pretrain-time and no knowledge text is given. An example from this variant:

> **Mamo Clark**, **Stephanie Beatriz**, **Pat Nixon**, and **Patricia Anthony** started a conversation at the dog park. The dogs here are lovely. When a question related to writing books or novels professionally arose, **she** offered a professional opinion.
> [Answer: **Patricia Anthony**, a well-known science-fiction author]

Creation: We turn to the Wikidata/Wikipedia ecosystem (Farda-Sarbas and Müller-Birn, 2019) as a source of knowledge about entities that are likely to have been observed by LLMs during pretraining, since many LLM pretraining corpora include Wikipedia (Gao et al., 2020; Laurençon et al., 2022).

We retrieve an initial list of candidate entities by querying the Wikidata SPARQL endpoint[5] for entities that have the `occupation` ("Property:P106") property for each occupation in the KITMUS dataset. As a proxy for being well-known, we limit the search to those entities that have a corresponding article in English Wikipedia and keep only the 200 entities with the most sitelinks. In order to mitigate the bias that

[5] `https://query.wikidata.org/`

there are more English Wikipedia articles about men than any other gender, we repeat this query for all possible `sex or gender` ("Property:P21") values.

This process resulted in 3948 candidate entities, most of which belonged to a handful of occupations that are likely to have an Wikipedia article such as actor, astronomer, or politician. Other occupations such as janitor or cashier had no or only very few Wikidata entries. We drop entities that were associated with multiple occupations, which is quite often the case for some occupation pairs such as `actor` and `model` or `politician` and `lawyer`.

For the variant to fulfill its purpose, the entities' occupations must be known to the LLMs evaluated in this work. While for real-world occupations such as "politician", one can reasonably assume that LLMs like BERT (Devlin et al., 2019) and ELMo (Peters et al., 2018) observed instances during training, however even for well-known entities it is not necessarily guaranteed that a specific LLM has explicitly been exposed to their occupation.

For that reason, we filter the candidate entities based on a LAMA probe (Petroni et al., 2019) to ensure that both BERT and ELMo can predict the correct occupation for this entity. A LAMA probe uses LLMs' ability to solve "fill-in-the-blank" cloze statements to test whether a model can fill in a certain `[MASK]` token that requires specific knowledge. The idea is that the model can only select the correct token among a large number of possibilities if the knowledge is stored in its parameters, which were determined at pretrain-time. We use the template `{name} is a [MASK].` and compare the probabilities of `[MASK]` being filled with the correct profession for the entity with `{name}`. We only keep those candidate entities for which both LLMs can accurately predict the occupation.

This procedure left entities with the following five occupations and associated frequencies:

- `author`: 291

- actor: 79
- model: 8
- politician: 6
- painter: 1

To increase the count of the underrepresented occupations model[6], politician[7], painter[8], we retrieved additional candidates for these occupations from the corresponding English Wikipedia categories sorted by number of pageviews as an alternate proxy for entities' popularity.

After another round of filtering based on the LAMA probe applied to BERT and ELMo, 348 entities[9] remained. The occupation count was high enough to create a sufficient number of non-repeating data samples through random slot filling. To have a full complement of name, occupation, and pronoun for each real-world entity, Wikidata was queried for the personal pronoun ("Property:P6553") property, with a heuristic based on the sex or gender value as a fallback for those entities without the personal pronoun property.

Bias: Both the Wikipedia/Wikidata ecosystem (Callahan and Herring, 2011; Hube, 2017) and the training data of BERT/ELMo (Ahn and Oh, 2021; Jentzsch and Turan, 2022) are human-curated and have biases. Through the collection and filtering process,

[6] https://pageviews.wmcloud.org/massviews/?platform=all-access&agent=user&source=category&range=latest-20&subjectpage=0&subcategories=1&sort=views&direction=1&view=list&target=https://en.wikipedia.org/wiki/Category:Models_(profession)

[7] https://pageviews.wmcloud.org/massviews/?platform=all-access&agent=user&source=category&range=latest-20&subjectpage=0&subcategories=1&sort=views&direction=1&view=list&target=https://en.wikipedia.org/wiki/Category:20th-century_politicians

[8] https://pageviews.wmcloud.org/massviews/?platform=all-access&agent=user&source=category&range=latest-20&subjectpage=0&subcategories=1&sort=views&direction=1&view=list&target=https://en.wikipedia.org/wiki/Category:Painters

[9] https://github.com/mpoemsl/kitmus/blob/kitmus-plus/resources/wiki_entities.csv

these biases are reflected in the selection of entities. Therefore this new variant does not conform to the standards of the fictional entities in the original KITMUS dataset in terms of pronoun and occupation balance.

To name a few biases, the majority of `model` entities use the pronoun "she", while the majority of `politician` entities use the pronoun "he". Additionally, due to the use of popularity proxies in the filtering process, many occupations such as carpenter are not represented in the final set of entities. Similarly, none of the entities that remained after the LAMA probe uses gender-indefinite pronouns, so they are not represented in this variant. However, at the level of individual instances, these biases should not allow models to learn any shortcuts, since all entities in a text use the same pronoun but have distinct occupations. For more details on imbalances in this variant, see Section 3.3.

3.3 KITMUS+ Validation

In order to ensure the suitability of the proposed test suite for its intended purpose, we validate the created dataset variants both quantitatively and qualitatively.

3.3.1 Descriptive Statistics

The instances of KITMUS+ are generated by randomly sampling from resource pools which are distinct for each split, thus ensuring that there is no overlap between train, validation, and test splits. For the BASE and INFERENCE-TIME BACKGROUND variants, that means any occupation and name occurring in one split cannot occur in another. For the PRETRAIN-TIME ENTITY-SPECIFIC variant, that means any entity occurring in one split cannot occur in another.

Weighted random sampling ensures that for the BASE and INFERENCE-TIME BACKGROUND variant, all occupations and pronoun combinations occur with a frequency approaching the predefined pronoun sampling ratio of 40% `he`, 40% `she`, 10% `they`, and 10% neopronouns such as `ey` and `ze`. Since the entities in the PRETRAIN-TIME ENTITY-SPECIFIC variant are drawn from the real world as depicted by Wikidata, this is not the case for this variant.

Table 3.3 shows the most, median, and least frequent combinations of occupation and pronoun per variant. While in the BASE and INFERENCE-TIME BACKGROUND variants, the most frequent combinations make up only a small share among the 20 occupations and six pronouns per split, the PRETRAIN-TIME ENTITY-SPECIFIC variant features only five occupations and two pronouns.

Additionally, the PRETRAIN-TIME ENTITY-SPECIFIC variant is reflective of various pronoun-occupation biases as described in Section 3.2.3. This should not affect the difficulty of solving an individual instance, but notably decreases the diversity of training data which may affect generalization behavior.

Variant	Frequency		Combination	
	Descriptor	Share	Occupation	Pronoun
BASE	most	2.42%	labourer	he
	median	0.45%	plumber	they
	least	0.14%	judge	ze
INFERENCE-TIME BACKGROUND	most	2.27%	administrative assistant	she
	median	0.45%	janitor	they
	least	0.15%	firefighter	ey
PRETRAIN-TIME ENTITY-SPECIFIC	most	12.54%	model	she
	median	10.20%	painter	he
	least	9.75%	author	he

Table 3.3 Most, median, and least frequent combinations of occupation and pronoun for entities in the test split for each variant in KITMUS+.

In comparison with other datasets for coreference resolution, KITMUS is most similar in size to analysis datasets such as GAP (Webster et al., 2018). In the BASE and INFERENCE-TIME BACKGROUND variants, each instance contains eight annotated entity mentions: one for each of the four entities conveying entity-specific knowledge such as "Urbanek is an architect" and one placing the entity in a situation with the other entities, such as "Urbanek, Petterson, Bertucci, and Klem met at the bus station." In addition to the entity mentions, there is a pronoun mention triggering the coreference, bringing the total number of annotated mentions to nine for the BASE and INFERENCE-TIME BACKGROUND variants. A comparison of size statistics with other coreference resolution datasets is shown in Table 3.4.

In terms of average words and mentions per document, KITMUS+ is comparable with other datasets drawn from natural text. OntoNotes has a higher relative density of annotated mentions since it annotates not only pronoun coreference, but coreferences between all types of mentions (Hovy et al., 2006). Like KITMUS+, GAP, WSC, and

Dataset	Number of Documents			Average per Document	
	Train	Validation	Test	Words	Mentions
OntoNotes	2802	343	348	467	56
GAP	2000	400	2000	95	3
WSC	0	0	271	16	3
Quoref	3771	454	477	384	5
KITMUS+	2000	400	2000	115	9†

Table 3.4 Size statistics of KITMUS+ in comparison with other coreference resolution datasets. Statistics for other datasets adapted from Dasigi et al. (2019) and Toshniwal et al. (2021). †The PRETRAIN-TIME ENTITY-SPECIFIC variant only has five annotated mentions since entity-specific knowledge is not provided at inference time.

Quoref present pronoun coreference resolution tasks and have a comparable mention density (Rahman and Ng, 2012; Webster et al., 2018; Dasigi et al., 2019).

3.3.2 Human Validation Study

To investigate whether human assessors agree on the resolution of our test cases and whether this resolution is in agreement with the automatically generated labels, we conduct a human validation study. We also investigate whether our assumption that both background and entity-specific knowledge are required to resolve the cases by including instances where the knowledge text is not provided to human participants.

For the validation study, we created a multiple-choice questionnaire by randomly selecting instances from the BASE and INFERENCE-TIME BACKGROUND variants with differing number of entities from each split (e.g., validation). Additionally, we included one instance from each variant and with each number of entities where the participants were only given the task text and not the accompanying knowledge text. A total of 96 sampled instances were presented to six different participants in random

order.

Variant	Occupation	Situation	With Knowledge	Without Knowledge
BASE	Real	Real	0.93	0.00
BASE without noise†			0.91	0.00
BASE with redundant knowledge†			1.00	0.00
INFERENCE-TIME BACKGROUND	Real	CharFict	1.00	0.00
		WordFict	0.98	0.00
INFERENCE-TIME BACKGROUND	CharFict	Real	0.98	0.00
		CharFict	0.98	0.00
		WordFict	0.96	0.06

Table 3.5 Accuracy on all variants aggregated over subtasks, splits, and participants. Random performance is 0.25. Human participants could select "can't say," which is never in agreement with the automatically generated labels. Experiments marked with † are from the ablation experiments in Section 4.3.

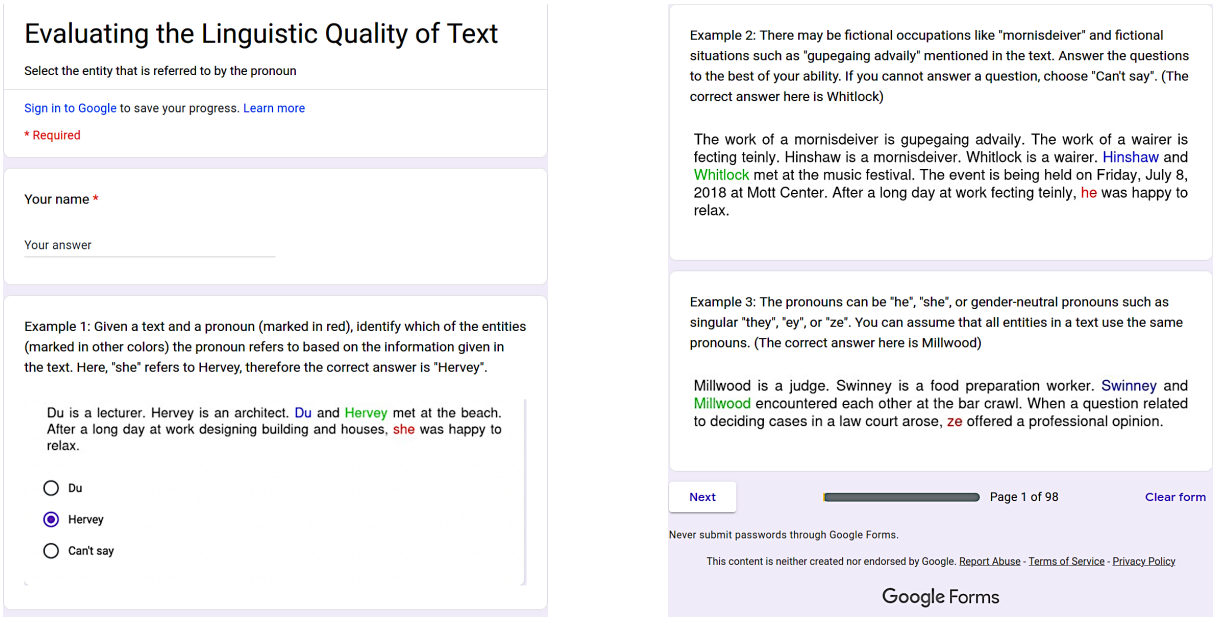
A high inter-annotator agreement of 0.938 as measured by Fleiss' Kappa (Fleiss et al., 2003) leads us to believe that human participants agree on the resolution of KITMUS test cases. We use accuracy as a measure of agreement with the automatically generated labels and find that mean accuracy aggregated over all participants and subtasks is higher than 0.9 for all variants when the knowledge text is given. As expected, when neither background nor entity-specific knowledge are given, accuracy is below 0.1 for all variants, since most participants indicate that the question cannot be answered. This suggests that there are no inadvertent cues that can be exploited by humans to solve the task without having access to the entity-specific knowledge and background knowledge contained in the knowledge text.

The study participants were undergraduate and graduate students with fluency in English which were recruited via an institution-wide open call. The participants were compensated with the equivalent of 12 USD[10] for their participation. The study was approved by the institution's ethics review board and the participants gave their written consent via a form.

[10]Matches the minimum wage in the participants' demographic

The participants were tasked to resolve the coreferences in a randomly sampled subset of KITMUS texts. The task is presented to the participants as a multiple choice questionnaire. The participants are given gold mentions and have to select the antecedent that is referred to by the pronoun. The answer options include the names of all mentioned entities and a "can't say" option to indicate that the question is not answerable. The questionnaire contained 96 questions to be completed in 60 minutes, which was generous for most participants.

The human validation was conducted using Google forms. The participants are introduced to the task with examples as shown in Figure 3.3.



(a) Top Half (b) Bottom Half

Figure 3.3 Introduction of the questionnaire used in the human validation study.

This is followed by 96 questions where the participants have to choose one option among all entity names and the option "can't say," which indicates that the task cannot be solved for this instance. The aggregated results of the validation study are shown in Table 3.5.

3.3.3 Pretrain-Time Knowledge Availability

Controlling the availability of knowledge from pretrain-time sources is inherently unreliable for pretrained large language models such as BERT and ELMo, since the training corpus is large and unstructured (Devlin et al., 2019). A useful tool for determining whether a LLM has access to a certain fact is the LAMA probe as proposed by Petroni et al. (2019) and described in Section 2.1.2.

A LAMA probe uses LLMs' ability to solve "fill-in-the-blank" cloze statements to test whether a model can fill in a certain `[MASK]` token that requires specific knowledge. The idea is that the model can only select the correct token among a large number of possibilities if the knowledge is stored in its parameters, which were determined at pretrain-time.

For the real-world entities in the PRETRAIN-TIME ENTITY-SPECIFIC variant, we filtered based on a LAMA probe (Petroni et al., 2019) and can therefore be certain that the knowledge is available in the parameters. However, the background knowledge about occupations that we assume to be pretrain-time knowledge did not go through such a filtering process.

To verify that the pretrained LLMs evaluated in this work contain background knowledge mapping occupations to situations, we run a LAMA probe on BERT and ELMo with the template `The work of a ___ is {situation}.`, where `{situation}` is a occupation description such as "acting in a play or movie". We compare the probabilities the LLMs assigned to all single-token occupation names used in KITMUS+ (probing for multi-token words is not supported by LAMA).

BERT assigned higher probabilities to the correct occupation than to any other occupation for 90% of occupations, suggesting that it is reasonable to assume background knowledge about occupations to be pretrain-time knowledge for BERT and other similar-sized Transformer (Vaswani et al., 2017) language models.

ELMo assigned the highest probability to the correct occupation for only 45% oc-

cupations. This might indicate that ELMo has memorized background knowledge about occupations to a lesser degree due to its smaller parameter count (93.6 million to `bert-large`'s 360 million parameters). The findings are consistent with prior work which reported ELMo to be worse at recalling facts than BERT (Petroni et al., 2019).

Chapter 4

Experiments

We distinguish two sets of experiments: the main experiments, which are conducted on the three main variants of the KITMUS+ test suite (BASE, INFERENCE-TIME BACKGROUND, PRETRAIN-TIME ENTITY-SPECIFIC) and ablation experiments, which investigate alternatives to the design choices made in the creation of this dataset.

In this chapter, we first describe the experimental setup including evaluated models. Then, we display the results of the main experiments and their implications. Finally, we present a range of ablation experiments and contextualize their results.

4.1 Experimental Setup

4.1.1 Model Selection

For the main experiments, we evaluate two state-of-the-art coreference resolution models using pretrained LLMs on the KITMUS+ test suite. We choose among a pool of large models that are intended for the task of general coreference resolution, which is commonly trained and evaluated on the large OntoNotes corpus (Hovy et al., 2006) in the CoNLL 2012 format (Pradhan et al., 2012). Among these, we include BERT4Coref (Joshi et al., 2019) as an example of a state-of-the-art models on OntoNotes as well as

C2F (Lee et al., 2018), which is the direct successor to the first end-to-end neural coreference resolution model (Lee et al., 2017). BERT4Coref uses BERT (Devlin et al., 2019) as part of its architecture and C2F uses ELMo (Peters et al., 2018) as its base LLM.

For the ablation experiments, we also consider models that are specialized for pronoun coreference resolution and adapted to the GAP format (Webster et al., 2018), which contains less detailed annotations than the CoNLL format. For more details on the CoNLL and GAP format, see Section 2.2.2.

Among the GAP pronoun resolution models, we include GREP (Attree, 2019), the winner of the GAP Kaggle competition as well as PeTra (Toshniwal et al., 2020), a memory-augmented model. Both GAP format models use BERT as part of their architecture.

Table 4.1 shows an overview of the evaluated models, their LLMs, and parameter counts.

Model	Proposed by	Format	LLM	Parameters
BERT4Coref	Joshi et al. (2019)	CoNLL	`bert-large`	340M
C2F	Lee et al. (2018)		`elmo-original`	93.6M
GREP	Attree (2019)	GAP	`bert-large`	340M
PeTra	Toshniwal et al. (2020)		`bert-large`	340M

Table 4.1 Evaluated models and LLMs with annotation format and parameter count in million (M).

4.1.2 Training

We conduct task-specific training with all models on the train split of each KITMUS+ variants using their best reported hyperparameters, which are displayed in Table 4.2.

The larger general coreference models BERT4Coref and C2F are conventionally not trained on datasets with just 2000 train instances such as GAP or KITMUS+, but rather trained on OntoNotes and then evaluated on smaller datasets (Joshi et al., 2019). How-

Model	Optimizer	Learning Rate	FFNN Size	Dropout Rate
BERT4Coref	Adam	$2 \cdot 10^{-4}$	1000	0.3
C2F	Adam	$1 \cdot 10^{-3}$	150	0.2
GREP	Adam	$4 \cdot 10^{-6}$	1024	0.1
PeTra	Adam	$1 \cdot 10^{-3}$	300	0.5

Table 4.2 Best reported hyperaparameters for evaluated models. Adam is the optimizer proposed by Kingma and Ba (2015). Dropout is implemented as proposed by Srivastava et al. (2014).

ever, since coreference cases in KITMUS+ diverge significantly from those in OntoNotes, training on OntoNotes not necessarily effect for our task. We still include an ablation study with OntoNotes-trained models in Section 4.3.2.

Since training with different seeds can induce variance into the results, we report mean metrics over six runs for all trained models. We train the GAP format models models—PeTra and GREP—only on the KITMUS+ version with two entities following the constraints of the GAP format.

We train our models in a compute cluster infrastructure on `Nvidia Quadro RTX 8000` GPUs. For BERT4Coref, training on the train split of one KITMUS+ variant took about 8 hours per run. For C2F it took about 16 hours. The training of GREP took 18 hours. The training of smaller models and inference on pretrained models took about 4 hours per run.

4.1.3 Evaluation

We evaluate all models on the KITMUS+ test split of each variant. During inference, CoNLL format models predict coreference clusters over all tokens in a text.

Pronoun Accuracy: The main metric of evaluation is pronoun accuracy: A sample was predicted correctly if the pronoun was assigned to the same coreference cluster as

the correct antecedent and to no other coreference clusters. Accuracy is then the ratio e of correctly predicted samples.

Antecedent F1: In the ablation experiments, we also consider the antecedent classification F1 metric, which is typically used for pronoun coreference resolution datasets such as GAP (Webster et al., 2018). It considers the coreference between each candidate antecedent mention and the pronoun as a binary classification decision i.e., for a text with n entities, it evaluates the correctness of n binary predictions.

Precision, recall, and F1 score are determined by separating all binary predictions into four categories based on the gold labels (Jurafsky and Martin, 2023): a prediction can be either count towards the true positives (tp), false positives (fp), true negatives (tn), or false negatives (fn). Given that, the following definitions hold:

$$\text{precision} = \frac{tp}{tp + fp}$$

$$\text{recall} = \frac{tp}{tp + fn}$$

$$\text{F1} = \frac{2 \cdot \text{precision} \cdot \text{recall}}{\text{precision} + \text{recall}}$$

In other words, F1 score is defined as the harmonic mean of precision and recall.

Random baseline: We compare against a random baseline, which is implemented as random choice among gold candidate mentions. This baseline accuracy is usually 0.25 given a choice among four entities. Note that it is still possible for a trained model to be worse than this random baseline, since the random choice presupposes access to gold mentions, while a model would have to do perfectly accurate mention detection to achieve a similar performance.

4.2 Main Experiments

Variant	Knowledge Source by Type		Accuracy	
	Background	Entity-specific	C2F	BERT4Coref
Base	Pretrain	Inference	0.48	0.94
ITB	Inference	Inference	0.08 - 0.18	0.25 - 0.43
PTES	Pretrain	Pretrain	0.45	0.75

Table 4.3 Mean accuracy by model and variant aggregated over six training runs. ITB results are range over fictional subvariants. Standard deviation is ≤ 0.06 for all values. Random baseline accuracy for this four entity variant is 0.25 assuming gold mention detection.

Main experiment results for all three variants are displayed in Table 4.3. On the BASE variant, both BERT4Coref and C2F demonstrate the ability to reason over knowledge observed at pretrain and inference time. However, differing performances on the INFERENCE-TIME BACKGROUND and PRETRAIN-TIME ENTITY-SPECIFIC variants indicate that the usefulness of knowledge in a source seems to depend on the knowledge type: background knowledge is more useful when drawn from pretrain-time parameters, while knowledge about entities seems to be better observed at inference time. In the following sections, we provide a detailed breakdown of observations and interpretations of the main experiments.

Observations

BASE variant performance: Both BERT4Coref and C2F clearly outperform the random baseline of 0.25, with BERT4Coref reaching a near perfect accuracy on the BASE variant. This suggests that both models have the ability to draw background knowledge from their parameters, entity-specific knowledge from the inference-time inputs, and reason over them on-the-fly with task-specific training.

INFERENCE-TIME BACKGROUND variant performance: On the INFERENCE-TIME

BACKGROUND variant, both models fail to reliably outperform the random baseline, with C2F falling belong it. Both models' performances seem to depend on the kind of fictionality employed (word-level or character-level fictional occupations). A detailed breakdown of the range values reported here for the INFERENCE-TIME BACKGROUND variant can be found in Table 4.4.

PRETRAIN-TIME ENTITY-SPECIFIC variant performance: Performance using only pretrain-time knowledge about well-known entities is slightly worse than BASE variant performance for both models. The performance difference seems to be larger for BERT4Coref than for C2F.

Cross-model performance comparison: BERT4Coref seems to consistently perform better than C2F on all variants.

Cross-variant performance comparison: Among all the mappings of background and entity-specific knowledge to different knowledge sources, the BASE variant configuration of pretrain-time background knowledge and inference-time entity-specific knowledge seems to result in the best performance for both models.

Interpretations

Underlying pretrained LLMs: BERT4Coref seems to consistently outperform C2F. This might be due to the difference in their underlying pretrained LLMs: BERT4Coref uses the Transformer architecture (Vaswani et al., 2017), which has been shown to be effective at reasoning tasks presented in natural language form (Clark et al., 2021) and utilizing information presented in inference-time contexts (Petroni et al., 2020), while C2F uses ELMo (Peters et al., 2018), which is much smaller.

Additionally, BERT and ELMo might differ in their memorization of background knowledge about occupations. A LAMA probe (Petroni et al., 2019) ran for model validation showed that BERT is more likely to contain the background knowledge compared to ELMo (see Section 3.3.3). This might contribute to the better performance of

BERT-based on knowledge intensive tasks such as KITMUS+, however, it cannot explain the worse performance of C2F on the INFERENCE-TIME BACKGROUND variant, where all knowledge is inference-time.

Fictional occupations in INFERENCE-TIME BACKGROUND: Both models perform consistently poorly on the INFERENCE-TIME BACKGROUND variant with fictional occupations and situations. An example of character-level fictional occupation knowledge erroneously answered by both models is shown below:

> The work of a vangiwer is aughuing ominly. Pirkle is a peeptoer. Alspaugh is a vangiwer. McCants is a towcer. The work of a peeptoer is mepuing cevely. Ostrander is a culfaer. The work of a culfaer is gholicoring intairly. The work of a towcer is lopening ackly. **McCants**, **Ostrander**, **Alspaugh**, and **Pirkle** started a conversation at the high intensity class. The classes usually begin before work. When a question related to lopening ackly arose, **he** offered a professional opinion. [Correct answer: **McCants**; BERT4Coref: **Alspaugh**; C2F: pronoun not part of any cluster]

One possible reason for particularly bad performance of BERT on character-level fictional situations could be BERT's tokenization strategy, which involves pooling subword representations (Devlin et al., 2019). In character-level fictional words, the subwords are meaningless, rendering their representations unhelpful. This is consistent with previous work showing that representations of LLMs for character-level fictional "Jabberwocky" words are less useful (Kasai and Frank, 2019) and that the presence of out-of-vocabulary (OOV) tokens decreases performance of neural models for NLU tasks (Schick and Schütze, 2020; Moon and Okazaki, 2020; He et al., 2021). C2F's lower than random baseline performance might similarly be explained with mention detection difficulties induced by the character-level fictional words.

Despite the character-fictional occupations and situations, it is still possible for models to resolve the coreferences successfully in this setting. In the given example, the pronoun "he" can be resolved by matching the situation "lopening ackly" to the occupation "towcer" (using the word overlap between the situations and the occupation

Occupation	Situation	C2F	BERT4Coref
Real	CharFict	0.18	0.25
	WordFict	0.08	0.48
CharFict	Real	0.08	0.43
	CharFict	0.18	0.26
	WordFict	0.11	0.38

Table 4.4 KITMUS+-trained accuracy on INFERENCE-TIME BACKGROUND subvariants with four entities by fictionality. Random baseline performance is 0.25.

descriptions) and identifying the correct entity associated with the occupation.

Humans can successfully make these inferences by matching fictional occupations and situations. However, the evaluated systems do not perform better than a random baseline in this setting. Our hope is that eventually, models should be able to handle even knowledge presented in previously unknown terms. Given that languages are forever growing, robustness to neologisms is crucial, considering that OOV words such as new occupations like "TikToker" develop constantly.

Preferred mapping from type to sources: While apple-to-apple comparisons between the variants might not be possible due to confounding factors like fictional words in the INFERENCE-TIME BACKGROUND variant and bias in the real-world entities of the PRETRAIN-TIME ENTITY-SPECIFIC variant, cross-variant result differences seem to suggest the same trend for both models: the usefulness of a knowledge source seems to depend on the knowledge type. Background knowledge is more useful when drawn from pretrain-time parameters, while knowledge about entities seems to be better observed at inference time.

One possible explanation could be that LLMs observed different frequencies of unseen entities and occupations during language modeling pretraining, which result in a difference in their ability to adapt to novel instances of those categories. This would be

in line with the initial intuition behind creating the BASE variant described in Section 3.2.1.

4.3 Ablation Experiments

In order to shed light on the effects of design decisions in the dataset creation process and experimental setup, we run a series of ablation experiments on the data configuration, evaluation protocol, and task format. We find that while a range of factors seems to influence system performance on the proposed task, the trends observed in the main experiments are robust to different design choices.

In particular, task-specific training on KITMUS+ with detailed annotations in the CoNLL 2012 (Pradhan et al., 2012) seem to be a necessary preconditions for success on our task. Results on BERT4Coref seem to be robust to dataset design choices such as number of entities, presence of noise, and train set size, while C2F performance is sensitive to small changes. We do not find consistent benefits to providing knowledge redundantly in both sources. Our results are echoed by a different evaluation metric and cannot be explained away through exploitation of root word overlap between occupations and situations. In the following, we provide a more detailed breakdown of the different ablation experiments.

4.3.1 Data Ablation

Modification to Base Variant	Accuracy	
	C2F	BERT4Coref
(none)	0.48	0.94
3 instead of 4 entities	0.28	0.98
2 instead of 4 entities	0.52	0.99
no noise	0.24	0.92
5k instead of 2k train examples	0.24	0.94
Redundant background knowledge	0.09	0.96

Table 4.5 Base variant modifications mean accuracy by model aggregated over six training runs. Standard deviation is ≤ 0.08 for all values. Random baseline performance is $\frac{1}{n}$ where n is the number of entities ($n = 4$ except where specified otherwise).

Ablation experiments using modifications to the BASE variant dataset are displayed in Table 4.5.

Number of Entities and Noise

We run ablation experiments with number of entities different than four (the default in this work) and without noise statements, which are by default part of the task text.

The accuracy of both models generally increases as the number of entities decreases, which is unsurprising since the more candidate entities there are, the less likely the accidental selection of the correct entity becomes. However, C2F specifically does not seem to reliably follow this pattern: performance with three instead of four entities is below random baseline performance of 0.33, but with other entity counts it is generally above.

In order to explore the effect of noise statements, we conduct additional experiments on the BASE variant without noise. The removal of noise does not result in a significant performance change for BERT4Coref, indicating that the model learns to ignore the noise during finetuning. However, C2F performance seems to be sensitive to the removal of noise statements.

The unintuitive performance drops of C2F might be attributed to a comparatively high variance across the six training runs. Alternatively, C2F might be more sensitive to sequence length changes, which affects the computation budget. There exists some evidence (Wei et al., 2022b) that auto-regressive language models can benefit from an increased sequence length and computation budget for reasoning tasks. Both decreasing the number of entities and removing noise result in a smaller sequence length and computation budget, which might contribute to the performance drop.

Train Set Size

The size of the train set for KITMUS+, 2000, was chosen to mirror that of GAP (Webster et al., 2018), another coreference resolution dataset that tests for a specific capability. To evaluate whether this choice of train set size affected results significantly, we run a ablation experiments on the BASE variant with a higher number of train samples.

Consistent with previous results, BERT4Coref performs well on this larger setting. C2F seems to perform worse with more train data, which might indicate overfitting on the train set. We release the KITMUS+ generation code to enable experimentation with other train set sizes in future work.

Redundant Knowledge

In order to shed more light on the knowledge integration behavior, we run an ablation experiment where background knowledge about occupations is made redundant by being supplied both at pretrain-time and inference-time. BERT4Coref shows a slight performance increase compared to the BASE variant, indicating that it might benefit from this redundant information. However, C2F performance drops sharply to a level below the random baseline.

One explanation for the performance drop of C2F might be that the background knowledge explicitly provided at inference time conflicts with the knowledge contained in the underlying LLM's parameters. While we are confident in the validity of the occupation descriptions used as resources, they are not exhaustive of the types of activities a certain occupation entails. LLM behavior when confronted with knowledge conflicts is the subject of active work (Longpre et al., 2021; Xie et al., 2023), but many suggest that performance becomes unstable, which might explain the performance drop. ELMo having memorized differing pretrain-time background knowledge would also be consistent with the findings of the LAMA probe, which suggest that ELMo's pretrain-time knowledge base only agrees with 45% of the occupation

description in KITMUS+ (see Section 3.3.3 for more details).

4.3.2 Evaluation Ablation

Ablation experiments on the BASE variant with modifications to the training and evaluation process are displayed in Table 4.6.

Train Data	Evaluation Data	Accuracy		Antecedent F1	
		C2F	BERT4Coref	C2F	BERT4Coref
Train Split	Test Split	0.48	0.94	0.48	0.94
Train Split	Train Split	1.00	1.00	1.00	1.00
OntoNotes	Test Split	0.13	0.14	0.25	0.19
Train Split	Test Split w/o RWO	0.46	0.92	0.46	0.92

Table 4.6 Mean metric by model aggregated over six training runs on KITMUS base variant unless specified otherwise. RWO is short for Root Word Overlap. Standard deviation is ≤ 0.08 for all values.

Train Set Evaluation

We find that evaluation on the train set yields perfect scores with all models, which validates that models learn as intended during finetuning on the train split. The high performance might even indicate overfitting and memorization, however given the large parameter count of the models, this seems unlikely to adversely affect generalization (an overview of the parameter counts of the LLMs was shown in Table 4.1).

Generalisation from OntoNotes

Coreference resolution models are often used off-the-shelf with training on OntoNotes (Hovy et al., 2006), since OntoNotes is considered to include most relevant coreference phenomena due to its size and generality. We therefore run an ablation with OntoNotes-trained versions of BERT4Coref and C2F.

The performance of OntoNotes-trained models is generally poor. This suggests that when trained on general coreference resolution datasets, models learn to exploit surface cues, which does not help when testing on KITMUS+ where such cues are removed. Another factor might be the structure of the texts in KITMUS+, which are designed to place knowledge in specific knowledge sources. This might affect models' abilities to form useful representations resulting in poor performance of OntoNotes-trained models. Given that the performance is even below random choice given gold mentions, the mention detection abilities acquired on OntoNotes might not transfer to KITMUS+.

These failures suggest that training on "general" datasets is not necessarily enough to induce knowledge integration from multiple knowledge sources. We conclude that task-specific training is required to solve the KITMUS+ task for the evaluated coreference models. This is also in line with recent work which suggests most coreference resolution models do not generalize well beyond their intended training domain (Toshniwal et al., 2021; Porada et al., 2023).

Root Word Overlap

One potential limitation of the two-hop task that KITMUS+ poses is that non-fictional background knowledge like "firefighters put out fires" can often be inferred by a simple string matching heuristic. In this case, the natural occurrence of the root word "fire" in both occupation and situation might enable models to solve the task without having access to background knowledge. For other background knowledge such as "judges preside over courts of law", no obvious string-matching shortcuts exist.

An analysis of trigram overlaps in all occupation-situation pairs shows that 45% of non-fictional occupation descriptions have at least one overlapping root word with the associated occupation. As an ablation, we compute performance on the subset of test instances that do not have any root word overlap. We find that while the subset performance is slightly worse than the overall performance, the magnitude of the difference

(± 0.02 accuracy) is small enough to not affect the validity of the observed results.

Evaluation Metric

We report results with the alternative antecedent F1 metric described in Section 4.1.3 as opposed to pronoun accuracy. We find that both metrics are mostly in agreement. In fact, the metric values only differ when model performance is so poor that the same pronoun is predicted to be corefering with multiple separate named entities, which is the case for some of the OntoNotes-trained model predictions.

As an example, if a model predicts both "Horner" and "Barlett" to be corefering with the pronoun "she", but in fact only "Barlett" corefers with "she", pronoun accuracy would count this as a single incorrect pronoun prediction, while antecedent F1 score would count it as one incorrect antecedent prediction ("Horner") and one correct antecedent prediction ("Barlett").

4.3.3 Format Ablation

Ablation experiments on the BASE variant provided in the GAP format are displayed in Table 4.7.

Entities	CoNLL Format		GAP Format		
	C2F	BERT4Coref	PeTra	GREP	Random
4 entities	0.48	0.94			0.25
3 entities	0.28	0.98			0.33
2 entities	0.52	0.99	0.01	0.49	0.50

Table 4.7 Mean accuracy by model aggregated over six training runs. Random is short for random choice among gold mentions. Standard deviation is ≤ 0.08 for all values.

In order to evaluate the effect of the choice of the CoNLL format (Pradhan et al., 2012), we run ablation experiments with comparable BERT-based models that accept

the GAP format instead (Webster et al., 2018). The GAP format only allows for the annotation of two entities, so we only report results with two entities for these models.

We find that performance is generally at random level or even worse for GAP format models. This might indicate that mention annotations in the knowledge text, which are present in CoNLL format but absent in the GAP format, are important for absorbing the entity-specific knowledge provided in the knowledge text of the BASE variant. For more details on differences between CoNLL and GAP format, see Section 2.2.2.

Chapter 5

Conclusion

Summary

In this work, we investigated the ability of LLM-based NLU systems to use knowledge observed at pretrain and inference time to solve a coreference resolution task that requires reasoning over knowledge of different types. For this purpose, we created the KITMUS+ test suite, a collection of coreference resolution tasks with different mappings of knowledge types to sources. We evaluated established LLM-based coreference resolution systems on the three main variants of the dataset and reported the results of several ablation experiments.

Findings

Our results show that with task-specific training and detailed annotations, some LLM-based NLU systems have the ability to reason over knowledge observed at pretrain and inference time. For the proposed task, the usefulness of knowledge in a source seems to depend on the knowledge type: background knowledge is more useful when drawn from pretrain-time parameters, while knowledge about entities seems to be bet-

ter observed at inference time. However, performance generally is sensitive to a range of factors such as the task format and the underlying LLM's size and architecture. We do not find consistent benefits to providing knowledge redundantly both at pretrain and inference time.

While these results represent experiments conducted on coreference resolution systems only, we believe they may have implications for LLM-based NLU systems in general due to the reliance of coreference resolution systems on their underlying LLMs. If different knowledge types have different preferred mappings to knowledge sources in LLMs, a careful analysis of the knowledge types required for a task might be beneficial for the design of future LLM-based NLU systems.

Limitations

Data diversity: As a template-generated dataset, KITMUS+ does not reflect the full diversity of natural data. However, we do not attempt to emulate the diversity of natural datasets. Instead, we believe that the advantages of using synthetic data for diagnostic purposes outweigh the disadvantages. Templates facilitate control over the source of each knowledge type, which would not be possible with natural datasets. This allows us to isolate the model behavior we want to probe. We also take several steps to add diversity, such as using multiple templates, sampling from large resource pools, random shuffling of entities, addition of noise sentences, and canonical data splits with non-overlapping templates and resources.

Result robustness: We report results averaged over six training runs for all settings and report standard deviation values. However, while results for BERT4Coref are stable, C2F's performance appears to be sensitive to small changes in the experimental setup as demonstrated in the ablation experiments. This may be due to the fact that C2F was originally intended to be trained on larger datasets such as OntoNotes,

rendering training on the smaller KITMUS+ dataset more susceptible to random variations. Nevertheless, we believe that the larger trends observed for both systems in this work allow us to draw reliable conclusions about models' behavior with respect to the integration of pretrain-time and inference-time knowledge.

Outlook

With KITMUS+, we have proposed a resource that could be used in future work to explore the knowledge integration abilities of more advanced NLU models and their underlying LLMs. Models can also be finetuned on our dataset to encourage knowledge integration across different sources. Finally, we hope our results can serve as guidance for architects of future NLU systems for design decisions such as the choice of whether to provide knowledge relevant for a given task as part of the pretraining data or as part of inference-time inputs.

Bibliography

Jaimeen Ahn and Alice Oh. 2021. Mitigating language-dependent ethnic bias in BERT. In *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing*, pages 533–549, Online and Punta Cana, Dominican Republic. Association for Computational Linguistics.

Rahul Aralikatte, Heather Lent, Ana Valeria Gonzalez, Daniel Hershcovich, Chen Qiu, Anders Sandholm, Michael Ringaard, and Anders Søgaard. 2019. Rewarding coreference resolvers for being consistent with world knowledge. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*, pages 1229–1235, Hong Kong, China. Association for Computational Linguistics.

Akshatha Arodi, Martin Pömsl, Kaheer Suleman, Adam Trischler, Alexandra Olteanu, and Jackie Chi Kit Cheung. 2023. The KITMUS test: Evaluating knowledge integration from multiple sources. In *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 15088–15108, Toronto, Canada. Association for Computational Linguistics.

Sandeep Attree. 2019. Gendered ambiguous pronouns shared task: Boosting model confidence by evidence pooling. In *Proceedings of the First Workshop on Gender Bias in Natural Language Processing*, pages 134–146, Florence, Italy. Association for Computational Linguistics.

Collin F. Baker, Charles J. Fillmore, and John B. Lowe. 1998. The Berkeley FrameNet project. In *36th Annual Meeting of the Association for Computational Linguistics and 17th International Conference on Computational Linguistics, Volume 1*, pages 86–90, Montreal, Quebec, Canada. Association for Computational Linguistics.

David Bean and Ellen Riloff. 2004. Unsupervised learning of contextual role knowledge for coreference resolution. In *Proceedings of the Human Language Technology Conference of the North American Chapter of the Association for Computational Linguistics: HLT-NAACL 2004*, pages 297–304, Boston, Massachusetts, USA. Association for Computational Linguistics.

Tom B. Brown, Benjamin Mann, Nick Ryder, Melanie Subbiah, Jared Kaplan, Prafulla Dhariwal, Arvind Neelakantan, Pranav Shyam, Girish Sastry, Amanda Askell, Sandhini Agarwal, Ariel Herbert-Voss, Gretchen Krueger, Tom Henighan, Rewon Child, Aditya Ramesh, Daniel M. Ziegler, Jeffrey Wu, Clemens Winter, Christopher Hesse, Mark Chen, Eric Sigler, Mateusz Litwin, Scott Gray, Benjamin Chess, Jack Clark, Christopher Berner, Sam McCandlish, Alec Radford, Ilya Sutskever, and Dario Amodei. 2020. Language models are few-shot learners.

Ewa S Callahan and Susan C Herring. 2011. Cultural bias in wikipedia content on famous persons. *Journal of the American society for information science and technology*, 62(10):1899–1915.

Yang Trista Cao and Hal Daumé III. 2020. Toward gender-inclusive coreference resolution. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 4568–4595, Online. Association for Computational Linguistics.

Nathanael Chambers and Dan Jurafsky. 2009. Unsupervised learning of narrative schemas and their participants. In *Proceedings of the Joint Conference of the 47th Annual Meeting of the ACL and the 4th International Joint Conference on Natural Language Processing of the AFNLP*, pages 602–610, Suntec, Singapore. Association for Computational Linguistics.

Anthony Chen, Pallavi Gudipati, Shayne Longpre, Xiao Ling, and Sameer Singh. 2021. Evaluating entity disambiguation and the role of popularity in retrieval-based NLP. In *Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing (Volume 1: Long Papers)*, pages 4472–4485, Online. Association for Computational Linguistics.

Hung-Ting Chen, Michael Zhang, and Eunsol Choi. 2022. Rich knowledge sources bring complex knowledge conflicts: Recalibrating models to reflect conflicting evidence. In *Proceedings of the 2022 Conference on Empirical Methods in Natural Language Processing*, pages 2292–2307, Abu Dhabi, United Arab Emirates. Association for Computational Linguistics.

Peter Clark, Oyvind Tafjord, and Kyle Richardson. 2021. Transformers as soft reasoners over language. In *Proceedings of the Twenty-Ninth International Joint Conference on Artificial Intelligence*, IJCAI'20.

Pradeep Dasigi, Nelson F. Liu, Ana Marasović, Noah A. Smith, and Matt Gardner. 2019. Quoref: A reading comprehension dataset with questions requiring coreferential reasoning. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language*

Processing (EMNLP-IJCNLP), pages 5925–5932, Hong Kong, China. Association for Computational Linguistics.

Nicola De Cao, Wilker Aziz, and Ivan Titov. 2021. Editing factual knowledge in language models. In *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing*, pages 6491–6506, Online and Punta Cana, Dominican Republic. Association for Computational Linguistics.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2019. 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)*, pages 4171–4186, Minneapolis, Minnesota. Association for Computational Linguistics.

Greg Durrett and Dan Klein. 2013. Easy victories and uphill battles in coreference resolution. In *Proceedings of the 2013 Conference on Empirical Methods in Natural Language Processing*, pages 1971–1982, Seattle, Washington, USA. Association for Computational Linguistics.

Hady Elsahar, Pavlos Vougiouklis, Arslen Remaci, Christophe Gravier, Jonathon Hare, Frederique Laforest, and Elena Simperl. 2018. T-REx: A large scale alignment of natural language with knowledge base triples. In *Proceedings of the Eleventh International Conference on Language Resources and Evaluation (LREC 2018)*, Miyazaki, Japan. European Language Resources Association (ELRA).

Ali Emami, Paul Trichelair, Adam Trischler, Kaheer Suleman, Hannes Schulz, and Jackie Chi Kit Cheung. 2019. The KnowRef coreference corpus: Removing gender and number cues for difficult pronominal anaphora resolution. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 3952–3961, Florence, Italy. Association for Computational Linguistics.

Mariam Farda-Sarbas and Claudia Müller-Birn. 2019. Wikidata from a research perspective – a systematic mapping study of wikidata.

Joseph L. Fleiss, Bruce Levin, and Myunghee Cho Paik. 2003. *The Measurement of Interrater Agreement*, chapter 18. John Wiley and Sons, Ltd.

Leo Gao, Stella Biderman, Sid Black, Laurence Golding, Travis Hoppe, Charles Foster, Jason Phang, Horace He, Anish Thite, Noa Nabeshima, Shawn Presser, and Connor Leahy. 2020. The Pile: An 800gb dataset of diverse text for language modeling. *arXiv preprint arXiv:2101.00027*.

Xavier Glorot, Antoine Bordes, and Yoshua Bengio. 2011. Deep sparse rectifier neural networks. In *Proceedings of the Fourteenth International Conference on Artificial Intelligence and Statistics*, volume 15 of *Proceedings of Machine Learning Research*, pages 315–323, Fort Lauderdale, FL, USA. PMLR.

Kelvin Guu, Kenton Lee, Zora Tung, Panupong Pasupat, and Ming-Wei Chang. 2020. Realm: Retrieval-augmented language model pre-training.

Keqing He, Yuanmeng Yan, and Weiran Xu. 2021. From context-aware to knowledge-aware: Boosting oov tokens recognition in slot tagging with background knowledge. *Neurocomputing*, 445:267–275.

Benjamin Heinzerling and Kentaro Inui. 2021. Language models as knowledge bases: On entity representations, storage capacity, and paraphrased queries. In *Proceedings of the 16th Conference of the European Chapter of the Association for Computational Linguistics: Main Volume*, pages 1772–1791, Online. Association for Computational Linguistics.

Jerry R. Hobbs. 1977. Pronoun resolution. *SIGART Bull.*, (61):28.

Sepp Hochreiter and Jürgen Schmidhuber. 1997. Long short-term memory. *Neural Comput.*, 9(8):1735–1780.

Jason Hoelscher-Obermaier, Julia Persson, Esben Kran, Ioannis Konstas, and Fazl Barez. 2023. Detecting edit failures in large language models: An improved specificity benchmark. In *Findings of the Association for Computational Linguistics: ACL 2023*, pages 11548–11559, Toronto, Canada. Association for Computational Linguistics.

Eduard Hovy, Mitchell Marcus, Martha Palmer, Lance Ramshaw, and Ralph Weischedel. 2006. OntoNotes: The 90% solution. In *Proceedings of the Human Language Technology Conference of the NAACL, Companion Volume: Short Papers*, pages 57–60, New York City, USA. Association for Computational Linguistics.

Christoph Hube. 2017. Bias in wikipedia. In *Proceedings of the 26th International Conference on World Wide Web Companion*, pages 717–721.

Sophie Jentzsch and Cigdem Turan. 2022. Gender bias in BERT - measuring and analysing biases through sentiment rating in a realistic downstream classification task. In *Proceedings of the 4th Workshop on Gender Bias in Natural Language Processing (GeBNLP)*, pages 184–199, Seattle, Washington. Association for Computational Linguistics.

Mandar Joshi, Omer Levy, Luke Zettlemoyer, and Daniel Weld. 2019. BERT for coreference resolution: Baselines and analysis. In *Proceedings of the 2019 Conference on*

Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP), pages 5803–5808, Hong Kong, China. Association for Computational Linguistics.

Daniel Jurafsky and James Martin. 2023. *Speech and Language Processing: An Introduction to Natural Language Processing, Computational Linguistics, and Speech Recognition (3rd Edition)*.

Vladimir Karpukhin, Barlas Oguz, Sewon Min, Patrick Lewis, Ledell Wu, Sergey Edunov, Danqi Chen, and Wen-tau Yih. 2020. Dense passage retrieval for open-domain question answering. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 6769–6781, Online. Association for Computational Linguistics.

Jungo Kasai and Robert Frank. 2019. Jabberwocky parsing: Dependency parsing with lexical noise. In *Proceedings of the Society for Computation in Linguistics (SCiL) 2019*, pages 113–123.

Nora Kassner and Hinrich Schütze. 2020. Negated and misprimed probes for pretrained language models: Birds can talk, but cannot fly. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 7811–7818, Online. Association for Computational Linguistics.

Diederik P. Kingma and Jimmy Ba. 2015. Adam: A method for stochastic optimization. In *3rd International Conference on Learning Representations, ICLR 2015, San Diego, CA, USA, May 7-9, 2015, Conference Track Proceedings*.

Hugo Laurençon, Lucile Saulnier, Thomas Wang, Christopher Akiki, Albert Villanova del Moral, Teven Le Scao, Leandro Von Werra, Chenghao Mou, Eduardo González Ponferrada, Huu Nguyen, Jörg Frohberg, Mario Šaško, Quentin Lhoest, Angelina McMillan-Major, Gerard Dupont, Stella Biderman, Anna Rogers, Loubna Ben allal, Francesco De Toni, Giada Pistilli, Olivier Nguyen, Somaieh Nikpoor, Maraim Masoud, Pierre Colombo, Javier de la Rosa, Paulo Villegas, Tristan Thrush, Shayne Longpre, Sebastian Nagel, Leon Weber, Manuel Muñoz, Jian Zhu, Daniel Van Strien, Zaid Alyafeai, Khalid Almubarak, Minh Chien Vu, Itziar Gonzalez-Dios, Aitor Soroa, Kyle Lo, Manan Dey, Pedro Ortiz Suarez, Aaron Gokaslan, Shamik Bose, David Adelani, Long Phan, Hieu Tran, Ian Yu, Suhas Pai, Jenny Chim, Violette Lepercq, Suzana Ilic, Margaret Mitchell, Sasha Alexandra Luccioni, and Yacine Jernite. 2022. The bigscience roots corpus: A 1.6tb composite multilingual dataset. In *Advances in Neural Information Processing Systems*, volume 35, pages 31809–31826. Curran Associates, Inc.

Anne Lauscher, Olga Majewska, Leonardo F. R. Ribeiro, Iryna Gurevych, Nikolai Rozanov, and Goran Glavaš. 2020. Common sense or world knowledge? investigating adapter-based knowledge injection into pretrained transformers. In *Proceedings of Deep Learning Inside Out (DeeLIO): The First Workshop on Knowledge Extraction and Integration for Deep Learning Architectures*, pages 43–49, Online. Association for Computational Linguistics.

Kenton Lee, Luheng He, Mike Lewis, and Luke Zettlemoyer. 2017. End-to-end neural coreference resolution. In *Proceedings of the 2017 Conference on Empirical Methods in Natural Language Processing*, pages 188–197, Copenhagen, Denmark. Association for Computational Linguistics.

Kenton Lee, Luheng He, and Luke Zettlemoyer. 2018. Higher-order coreference resolution with coarse-to-fine inference. In *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 2 (Short Papers)*, pages 687–692, New Orleans, Louisiana. Association for Computational Linguistics.

Hector Levesque, Ernest Davis, and Leora Morgenstern. 2012. The winograd schema challenge. In *Thirteenth International Conference on the Principles of Knowledge Representation and Reasoning*.

Patrick Lewis, Ethan Perez, Aleksandra Piktus, Fabio Petroni, Vladimir Karpukhin, Naman Goyal, Heinrich Küttler, Mike Lewis, Wen-tau Yih, Tim Rocktäschel, Sebastian Riedel, and Douwe Kiela. 2020. Retrieval-augmented generation for knowledge-intensive nlp tasks. In *Advances in Neural Information Processing Systems*, volume 33, pages 9459–9474. Curran Associates, Inc.

Bill Yuchen Lin, Seyeon Lee, Rahul Khanna, and Xiang Ren. 2020. Birds have four legs?! NumerSense: Probing Numerical Commonsense Knowledge of Pre-Trained Language Models. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 6862–6868, Online. Association for Computational Linguistics.

Yang Liu, Chenguang Zhu, and Michael Zeng. 2021. Modeling entity knowledge for fact verification. In *Proceedings of the Fourth Workshop on Fact Extraction and VERification (FEVER)*, pages 50–59, Dominican Republic. Association for Computational Linguistics.

Teng Long, Emmanuel Bengio, Ryan Lowe, Jackie Chi Kit Cheung, and Doina Precup. 2017. World knowledge for reading comprehension: Rare entity prediction with hierarchical LSTMs using external descriptions. In *Proceedings of the 2017 Conference on*

Empirical Methods in Natural Language Processing, pages 825–834, Copenhagen, Denmark. Association for Computational Linguistics.

Shayne Longpre, Kartik Perisetla, Anthony Chen, Nikhil Ramesh, Chris DuBois, and Sameer Singh. 2021. Entity-based knowledge conflicts in question answering. In *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing*, pages 7052–7063, Online and Punta Cana, Dominican Republic. Association for Computational Linguistics.

Nikolay Malkin, Sameera Lanka, Pranav Goel, Sudha Rao, and Nebojsa Jojic. 2021. GPT perdetry test: Generating new meanings for new words. In *Proceedings of the 2021 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 5542–5553, Online. Association for Computational Linguistics.

Mary Ann Marcinkiewicz. 1994. Building a large annotated corpus of english: The penn treebank. *Using Large Corpora*, page 273.

Kevin Meng, Arnab Sen Sharma, Alex Andonian, Yonatan Belinkov, and David Bau. 2022. Mass editing memory in a transformer. *arXiv preprint arXiv:2210.07229*.

Sangwhan Moon and Naoaki Okazaki. 2020. PatchBERT: Just-in-time, out-of-vocabulary patching. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 7846–7852, Online. Association for Computational Linguistics.

Ella Neeman, Roee Aharoni, Or Honovich, Leshem Choshen, Idan Szpektor, and Omri Abend. 2023. DisentQA: Disentangling parametric and contextual knowledge with counterfactual question answering. In *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 10056–10070, Toronto, Canada. Association for Computational Linguistics.

Vincent Ng and Claire Cardie. 2002. Identifying anaphoric and non-anaphoric noun phrases to improve coreference resolution. In *COLING 2002: The 19th International Conference on Computational Linguistics*.

Yasumasa Onoe, Michael J. Q. Zhang, Eunsol Choi, and Greg Durrett. 2021. Creak: A dataset for commonsense reasoning over entity knowledge.

OpenAI. 2023. Gpt-4 technical report.

Matthew E. Peters, Mark Neumann, Mohit Iyyer, Matt Gardner, Christopher Clark, Kenton Lee, and Luke Zettlemoyer. 2018. Deep contextualized word representations. In *Proceedings of the 2018 Conference of the North American Chapter of the Association for*

Computational Linguistics: Human Language Technologies, Volume 1 (Long Papers), pages 2227–2237, New Orleans, Louisiana. Association for Computational Linguistics.

Fabio Petroni, Patrick Lewis, Aleksandra Piktus, Tim Rocktäschel, Yuxiang Wu, Alexander H. Miller, and Sebastian Riedel. 2020. How context affects language models' factual predictions. In *Automated Knowledge Base Construction*.

Fabio Petroni, Aleksandra Piktus, Angela Fan, Patrick Lewis, Majid Yazdani, Nicola De Cao, James Thorne, Yacine Jernite, Vladimir Karpukhin, Jean Maillard, Vassilis Plachouras, Tim Rocktäschel, and Sebastian Riedel. 2021. KILT: a benchmark for knowledge intensive language tasks. In *Proceedings of the 2021 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 2523–2544, Online. Association for Computational Linguistics.

Fabio Petroni, Tim Rocktäschel, Sebastian Riedel, Patrick Lewis, Anton Bakhtin, Yuxiang Wu, and Alexander Miller. 2019. Language models as knowledge bases? In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*, pages 2463–2473, Hong Kong, China. Association for Computational Linguistics.

Aleksandra Piktus, Fabio Petroni, Vladimir Karpukhin, Dmytro Okhonko, Samuel Broscheit, Gautier Izacard, Patrick Lewis, Barlas Oğuz, Edouard Grave, Wen tau Yih, and Sebastian Riedel. 2022. The web is your oyster - knowledge-intensive nlp against a very large web corpus.

Ian Porada, Alexandra Olteanu, Kaheer Suleman, Adam Trischler, and Jackie Chi Kit Cheung. 2023. Investigating failures to generalize for coreference resolution models.

Ian Porada, Alessandro Sordoni, and Jackie Cheung. 2022. Does pre-training induce systematic inference? how masked language models acquire commonsense knowledge. In *Proceedings of the 2022 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 4550–4557, Seattle, United States. Association for Computational Linguistics.

Sameer Pradhan, Alessandro Moschitti, Nianwen Xue, Olga Uryupina, and Yuchen Zhang. 2012. CoNLL-2012 shared task: Modeling multilingual unrestricted coreference in OntoNotes. In *Joint Conference on EMNLP and CoNLL - Shared Task*, pages 1–40, Jeju Island, Korea. Association for Computational Linguistics.

Alec Radford, Jeffrey Wu, Rewon Child, David Luan, Dario Amodei, Ilya Sutskever, et al. 2019. Language models are unsupervised multitask learners. *OpenAI blog*, 1(8):9.

Colin Raffel, Noam Shazeer, Adam Roberts, Katherine Lee, Sharan Narang, Michael Matena, Yanqi Zhou, Wei Li, and Peter J. Liu. 2020. Exploring the limits of transfer learning with a unified text-to-text transformer. *Journal of Machine Learning Research*, 21(140):1–67.

Altaf Rahman and Vincent Ng. 2012. Resolving complex cases of definite pronouns: The Winograd schema challenge. In *Proceedings of the 2012 Joint Conference on Empirical Methods in Natural Language Processing and Computational Natural Language Learning*, pages 777–789, Jeju Island, Korea. Association for Computational Linguistics.

Adam Roberts, Colin Raffel, and Noam Shazeer. 2020. How much knowledge can you pack into the parameters of a language model? In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 5418–5426, Online. Association for Computational Linguistics.

Anna Rogers, Olga Kovaleva, and Anna Rumshisky. 2020. A primer in BERTology: What we know about how BERT works. *Transactions of the Association for Computational Linguistics*, 8:842–866.

Keisuke Sakaguchi, Ronan Le Bras, Chandra Bhagavatula, and Yejin Choi. 2020. Winogrande: An adversarial winograd schema challenge at scale. In *Proc. of AAAI*, volume 34, pages 8732–8740.

Maarten Sap, Ronan Le Bras, Emily Allaway, Chandra Bhagavatula, Nicholas Lourie, Hannah Rashkin, Brendan Roof, Noah A. Smith, and Yejin Choi. 2019. Atomic: An atlas of machine commonsense for if-then reasoning. *ArXiv*, abs/1811.00146.

Timo Schick and Hinrich Schütze. 2020. Rare words: A major problem for contextualized embeddings and how to fix it by attentive mimicking. In *Proc. of AAAI*, volume 34, pages 8766–8774.

Robyn Speer, Joshua Chin, and Catherine Havasi. 2018. Conceptnet 5.5: An open multilingual graph of general knowledge.

Aarohi Srivastava, Abhinav Rastogi, Abhishek Rao, Abu Awal Md Shoeb, Abubakar Abid, Adam Fisch, Adam R. Brown, Adam Santoro, Aditya Gupta, Adrià Garriga-Alonso, Agnieszka Kluska, Aitor Lewkowycz, Akshat Agarwal, Alethea Power, Alex Ray, Alex Warstadt, Alexander W. Kocurek, Ali Safaya, Ali Tazarv, Alice Xiang, Alicia Parrish, Allen Nie, Aman Hussain, Amanda Askell, Amanda Dsouza, Ambrose Slone, Ameet Rahane, Anantharaman S. Iyer, Anders Andreassen, Andrea Madotto, Andrea Santilli, Andreas Stuhlmüller, Andrew Dai, Andrew La, Andrew Lampinen, Andy Zou, Angela Jiang, Angelica Chen, Anh Vuong, Animesh Gupta, Anna Gottardi, Antonio Norelli, Anu Venkatesh, Arash Gholamidavoodi, Arfa Tabassum, Arul Menezes, Arun Kirubarajan, Asher Mullokandov,

Ashish Sabharwal, Austin Herrick, Avia Efrat, Aykut Erdem, Ayla Karakaş, B. Ryan Roberts, Bao Sheng Loe, Barret Zoph, Bartłomiej Bojanowski, Batuhan Özyurt, Behnam Hedayatnia, Behnam Neyshabur, Benjamin Inden, Benno Stein, Berk Ekmekci, Bill Yuchen Lin, Blake Howald, Bryan Orinion, Cameron Diao, Cameron Dour, Catherine Stinson, Cedrick Argueta, César Ferri Ramírez, Chandan Singh, Charles Rathkopf, Chenlin Meng, Chitta Baral, Chiyu Wu, Chris Callison-Burch, Chris Waites, Christian Voigt, Christopher D. Manning, Christopher Potts, Cindy Ramirez, Clara E. Rivera, Clemencia Siro, Colin Raffel, Courtney Ashcraft, Cristina Garbacea, Damien Sileo, Dan Garrette, Dan Hendrycks, Dan Kilman, Dan Roth, Daniel Freeman, Daniel Khashabi, Daniel Levy, Daniel Moseguí González, Danielle Perszyk, Danny Hernandez, Danqi Chen, Daphne Ippolito, Dar Gilboa, David Dohan, David Drakard, David Jurgens, Debajyoti Datta, Deep Ganguli, Denis Emelin, Denis Kleyko, Deniz Yuret, Derek Chen, Derek Tam, Dieuwke Hupkes, Diganta Misra, Dilyar Buzan, Dimitri Coelho Mollo, Diyi Yang, Dong-Ho Lee, Dylan Schrader, Ekaterina Shutova, Ekin Dogus Cubuk, Elad Segal, Eleanor Hagerman, Elizabeth Barnes, Elizabeth Donoway, Ellie Pavlick, Emanuele Rodola, Emma Lam, Eric Chu, Eric Tang, Erkut Erdem, Ernie Chang, Ethan A. Chi, Ethan Dyer, Ethan Jerzak, Ethan Kim, Eunice Engefu Manyasi, Evgenii Zheltonozhskii, Fanyue Xia, Fatemeh Siar, Fernando Martínez-Plumed, Francesca Happé, Francois Chollet, Frieda Rong, Gaurav Mishra, Genta Indra Winata, Gerard de Melo, Germán Kruszewski, Giambattista Parascandolo, Giorgio Mariani, Gloria Wang, Gonzalo Jaimovitch-López, Gregor Betz, Guy Gur-Ari, Hana Galijasevic, Hannah Kim, Hannah Rashkin, Hannaneh Hajishirzi, Harsh Mehta, Hayden Bogar, Henry Shevlin, Hinrich Schütze, Hiromu Yakura, Hongming Zhang, Hugh Mee Wong, Ian Ng, Isaac Noble, Jaap Jumelet, Jack Geissinger, Jackson Kernion, Jacob Hilton, Jaehoon Lee, Jaime Fernández Fisac, James B. Simon, James Koppel, James Zheng, James Zou, Jan Kocoń, Jana Thompson, Janelle Wingfield, Jared Kaplan, Jarema Radom, Jascha Sohl-Dickstein, Jason Phang, Jason Wei, Jason Yosinski, Jekaterina Novikova, Jelle Bosscher, Jennifer Marsh, Jeremy Kim, Jeroen Taal, Jesse Engel, Jesujoba Alabi, Jiacheng Xu, Jiaming Song, Jillian Tang, Joan Waweru, John Burden, John Miller, John U. Balis, Jonathan Batchelder, Jonathan Berant, Jörg Frohberg, Jos Rozen, Jose Hernandez-Orallo, Joseph Boudeman, Joseph Guerr, Joseph Jones, Joshua B. Tenenbaum, Joshua S. Rule, Joyce Chua, Kamil Kanclerz, Karen Livescu, Karl Krauth, Karthik Gopalakrishnan, Katerina Ignatyeva, Katja Markert, Kaustubh D. Dhole, Kevin Gimpel, Kevin Omondi, Kory Mathewson, Kristen Chiafullo, Ksenia Shkaruta, Kumar Shridhar, Kyle McDonell, Kyle Richardson, Laria Reynolds, Leo Gao, Li Zhang, Liam Dugan, Lianhui Qin, Lidia Contreras-Ochando, Louis-Philippe Morency, Luca Moschella, Lucas Lam, Lucy Noble, Ludwig Schmidt, Luheng He, Luis Oliveros Colón, Luke Metz, Lütfi Kerem Şenel, Maarten Bosma, Maarten Sap, Maartje ter Hoeve, Maheen Farooqi, Manaal Faruqui, Mantas Mazeika,

Marco Baturan, Marco Marelli, Marco Maru, Maria Jose Ramírez Quintana, Marie Tolkiehn, Mario Giulianelli, Martha Lewis, Martin Potthast, Matthew L. Leavitt, Matthias Hagen, Mátyás Schubert, Medina Orduna Baitemirova, Melody Arnaud, Melvin McElrath, Michael A. Yee, Michael Cohen, Michael Gu, Michael Ivanitskiy, Michael Starritt, Michael Strube, Michał Swędrowski, Michele Bevilacqua, Michihiro Yasunaga, Mihir Kale, Mike Cain, Mimee Xu, Mirac Suzgun, Mitch Walker, Mo Tiwari, Mohit Bansal, Moin Aminnaseri, Mor Geva, Mozhdeh Gheini, Mukund Varma T, Nanyun Peng, Nathan A. Chi, Nayeon Lee, Neta Gur-Ari Krakover, Nicholas Cameron, Nicholas Roberts, Nick Doiron, Nicole Martinez, Nikita Nangia, Niklas Deckers, Niklas Muennighoff, Nitish Shirish Keskar, Niveditha S. Iyer, Noah Constant, Noah Fiedel, Nuan Wen, Oliver Zhang, Omar Agha, Omar Elbaghdadi, Omer Levy, Owain Evans, Pablo Antonio Moreno Casares, Parth Doshi, Pascale Fung, Paul Pu Liang, Paul Vicol, Pegah Alipoormolabashi, Peiyuan Liao, Percy Liang, Peter Chang, Peter Eckersley, Phu Mon Htut, Pinyu Hwang, Piotr Miłkowski, Piyush Patil, Pouya Pezeshkpour, Priti Oli, Qiaozhu Mei, Qing Lyu, Qinlang Chen, Rabin Banjade, Rachel Etta Rudolph, Raefer Gabriel, Rahel Habacker, Ramon Risco, Raphaël Millière, Rhythm Garg, Richard Barnes, Rif A. Saurous, Riku Arakawa, Robbe Raymaekers, Robert Frank, Rohan Sikand, Roman Novak, Roman Sitelew, Ronan LeBras, Rosanne Liu, Rowan Jacobs, Rui Zhang, Ruslan Salakhutdinov, Ryan Chi, Ryan Lee, Ryan Stovall, Ryan Teehan, Rylan Yang, Sahib Singh, Saif M. Mohammad, Sajant Anand, Sam Dillavou, Sam Shleifer, Sam Wiseman, Samuel Gruetter, Samuel R. Bowman, Samuel S. Schoenholz, Sanghyun Han, Sanjeev Kwatra, Sarah A. Rous, Sarik Ghazarian, Sayan Ghosh, Sean Casey, Sebastian Bischoff, Sebastian Gehrmann, Sebastian Schuster, Sepideh Sadeghi, Shadi Hamdan, Sharon Zhou, Shashank Srivastava, Sherry Shi, Shikhar Singh, Shima Asaadi, Shixiang Shane Gu, Shubh Pachchigar, Shubham Toshniwal, Shyam Upadhyay, Shyamolima, Debnath, Siamak Shakeri, Simon Thormeyer, Simone Melzi, Siva Reddy, Sneha Priscilla Makini, Soo-Hwan Lee, Spencer Torene, Sriharsha Hatwar, Stanislas Dehaene, Stefan Divic, Stefano Ermon, Stella Biderman, Stephanie Lin, Stephen Prasad, Steven T. Piantadosi, Stuart M. Shieber, Summer Misherghi, Svetlana Kiritchenko, Swaroop Mishra, Tal Linzen, Tal Schuster, Tao Li, Tao Yu, Tariq Ali, Tatsu Hashimoto, Te-Lin Wu, Théo Desbordes, Theodore Rothschild, Thomas Phan, Tianle Wang, Tiberius Nkinyili, Timo Schick, Timofei Kornev, Titus Tunduny, Tobias Gerstenberg, Trenton Chang, Trishala Neeraj, Tushar Khot, Tyler Shultz, Uri Shaham, Vedant Misra, Vera Demberg, Victoria Nyamai, Vikas Raunak, Vinay Ramasesh, Vinay Uday Prabhu, Vishakh Padmakumar, Vivek Srikumar, William Fedus, William Saunders, William Zhang, Wout Vossen, Xiang Ren, Xiaoyu Tong, Xinran Zhao, Xinyi Wu, Xudong Shen, Yadollah Yaghoobzadeh, Yair Lakretz, Yangqiu Song, Yasaman Bahri, Yejin Choi, Yichi Yang, Yiding Hao, Yifu Chen, Yonatan Belinkov, Yu Hou, Yufang Hou, Yuntao Bai, Zachary Seid, Zhuoye Zhao, Zijian Wang, Zijie J. Wang, Zirui Wang,

and Ziyi Wu. 2023. Beyond the imitation game: Quantifying and extrapolating the capabilities of language models.

Nitish Srivastava, Geoffrey Hinton, Alex Krizhevsky, Ilya Sutskever, and Ruslan Salakhutdinov. 2014. Dropout: A simple way to prevent neural networks from overfitting. *J. Mach. Learn. Res.*, 15(1):1929–1958.

Ian Tenney, Dipanjan Das, and Ellie Pavlick. 2019. BERT rediscovers the classical NLP pipeline. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 4593–4601, Florence, Italy. Association for Computational Linguistics.

James Thorne, Andreas Vlachos, Christos Christodoulopoulos, and Arpit Mittal. 2018. FEVER: a large-scale dataset for fact extraction and VERification. In *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long Papers)*, pages 809–819, New Orleans, Louisiana. Association for Computational Linguistics.

Shubham Toshniwal, Allyson Ettinger, Kevin Gimpel, and Karen Livescu. 2020. PeTra: A Sparsely Supervised Memory Model for People Tracking. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 5415–5428, Online. Association for Computational Linguistics.

Shubham Toshniwal, Patrick Xia, Sam Wiseman, Karen Livescu, and Kevin Gimpel. 2021. On generalization in coreference resolution. In *Proceedings of the Fourth Workshop on Computational Models of Reference, Anaphora and Coreference*, pages 111–120, Punta Cana, Dominican Republic. Association for Computational Linguistics.

Hugo Touvron, Louis Martin, Kevin Stone, Peter Albert, Amjad Almahairi, Yasmine Babaei, Nikolay Bashlykov, Soumya Batra, Prajjwal Bhargava, Shruti Bhosale, Dan Bikel, Lukas Blecher, Cristian Canton Ferrer, Moya Chen, Guillem Cucurull, David Esiobu, Jude Fernandes, Jeremy Fu, Wenyin Fu, Brian Fuller, Cynthia Gao, Vedanuj Goswami, Naman Goyal, Anthony Hartshorn, Saghar Hosseini, Rui Hou, Hakan Inan, Marcin Kardas, Viktor Kerkez, Madian Khabsa, Isabel Kloumann, Artem Korenev, Punit Singh Koura, Marie-Anne Lachaux, Thibaut Lavril, Jenya Lee, Diana Liskovich, Yinghai Lu, Yuning Mao, Xavier Martinet, Todor Mihaylov, Pushkar Mishra, Igor Molybog, Yixin Nie, Andrew Poulton, Jeremy Reizenstein, Rashi Rungta, Kalyan Saladi, Alan Schelten, Ruan Silva, Eric Michael Smith, Ranjan Subramanian, Xiaoqing Ellen Tan, Binh Tang, Ross Taylor, Adina Williams, Jian Xiang Kuan, Puxin Xu, Zheng Yan, Iliyan Zarov, Yuchen Zhang, Angela Fan, Melanie Kambadur, Sharan Narang, Aurelien Rodriguez, Robert Stojnic, Sergey Edunov, and Thomas Scialom. 2023. Llama 2: Open foundation and fine-tuned chat models.

Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Ł ukasz Kaiser, and Illia Polosukhin. 2017. Attention is all you need. In *In NeurIPS*, volume 30. Curran Associates, Inc.

Alex Wang, Amanpreet Singh, Julian Michael, Felix Hill, Omer Levy, and Samuel Bowman. 2018. GLUE: A multi-task benchmark and analysis platform for natural language understanding. In *Proceedings of the 2018 EMNLP Workshop BlackboxNLP: Analyzing and Interpreting Neural Networks for NLP*, pages 353–355, Brussels, Belgium. Association for Computational Linguistics.

Kellie Webster, Marta Recasens, Vera Axelrod, and Jason Baldridge. 2018. Mind the GAP: A balanced corpus of gendered ambiguous pronouns. *Transactions of the Association for Computational Linguistics*, 6:605–617.

Jason Wei, Maarten Paul Bosma, Vincent Zhao, Kelvin Guu, Adams Wei Yu, Brian Lester, Nan Du, Andrew Mingbo Dai, and Quoc V. Le. 2022a. Finetuned language models are zero-shot learners. In *International Conference on Learning Representations*.

Jason Wei, Xuezhi Wang, Dale Schuurmans, Maarten Bosma, brian ichter, Fei Xia, Ed Chi, Quoc V Le, and Denny Zhou. 2022b. Chain-of-thought prompting elicits reasoning in large language models. In *Advances in Neural Information Processing Systems*, volume 35, pages 24824–24837. Curran Associates, Inc.

Jian Xie, Kai Zhang, Jiangjie Chen, Renze Lou, and Yu Su. 2023. Adaptive chameleon or stubborn sloth: Unraveling the behavior of large language models in knowledge clashes.

Benfeng Xu, Chunxu Zhao, Wenbin Jiang, PengFei Zhu, Songtai Dai, Chao Pang, Zhuo Sun, Shuohuan Wang, and Yu Sun. 2023. Retrieval-augmented domain adaptation of language models. In *Proceedings of the 8th Workshop on Representation Learning for NLP (RepL4NLP 2023)*, pages 54–64, Toronto, Canada. Association for Computational Linguistics.

Hongming Zhang, Yan Song, Yangqiu Song, and Dong Yu. 2019. Knowledge-aware pronoun coreference resolution. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 867–876, Florence, Italy. Association for Computational Linguistics.