Applying characteristics of human reasoning to zero-shot reasoning in GPT models

Loïs Dona

August 3, 2023

Abstract

This work builds upon an analogy between human and artificial reasoning. Large Transformer based language models have achieved state-of-the-art in many tasks, and recently have even been able to do this without requiring task specific fine-tuning, by deploying either in-context or zero-shot learning. However, reasoning remains to be a difficult task for these models, especially in a zero-shot setting. On the contrary, humans are good at reasoning without being explicitly trained for it like models are, hinting that properties of human reasoning might be of help to boost the performance of models. We explore this intuition in two ways: (1) using human-like linguistic input for fine-tuning and (2) prompting models to "imagine", a technique that has shown to help humans reason better. Our results show that our approach was fruitful for reasoning about fantastical scenarios, which is in line with previous research on humans, confirming that making an analogy between human and artificial reasoning can be helpful. This research opens many doors for future research on zero-shot reasoning, also using smaller models, which is a desirable development towards human-like general intelligence.

Acknowledgements

I want to express my gratitude to my main supervisor dr. Denis Paperno for guiding me in this research, from coming up with the concept of this research, to writing the final thesis. Second, I want to thank my second supervisor dr. Meaghan Fowlie, for taking the time to give me useful feedback on the research proposal, and for evaluating the final result. Lastly, I want to thank Marcello Eiermann for proofreading the final thesis.

Contents

1 Introduction

One of the overarching goals of Artificial Intelligence is to achieve human-like general intelligence, meaning that one model can do any task with human-like performance. Most recently, large language models, for example originating from BERT (Devlin, Chang, Lee, & Toutanova, 2018) and GPT (Radford et al., 2019; Brown et al., 2020), have achieved state-of-the-art in tasks involving natural language, such as question answering, summarization or translation. This great performance is largely due to fine-tuning approaches (Devlin et al., 2018; Radford, Narasimhan, Salimans, Sutskever, et al., 2018; Howard & Ruder, 2018), where a pre-trained model is trained further on large amounts of task specific data.

Recently, more attention has been given to eliminating the need for these large amounts of task specific data by relying on approaches commonly referred to as in-context and zero-shot learning. This is desirable for two main reasons. First of all, dismissing the need for fine-tuning reduces computational cost. Second, it allows one to use a model for many different downstream tasks, which gets us a step closer to general intelligence. In-context refers to the ability to perform a task after observing just a few examples representative of the task. Zero-shot refers to the ability to perform a task without observing any examples of the task beforehand. Most literature refers to these abilities as in-context or zero-shot *learning*. Strictly speaking, learning is not the right term to describe this behaviour, as the presentation of task examples does not update the model's weights. Still, we will continue to refer to these phenomena using *learning*, to stay consistent with existing literature.

These learning paradigms are advantageous, because they allow for easy transfer across a wide range of tasks without the need for as much task specific training data as would be needed when training a task specific model. These approaches have shown promising results, but still do not compete with fine-tuning approaches or human performance on a range of tasks, especially in the context of zero-shot learning (Brown et al., 2020). Furthermore, the capacity to perform in-context or zero-shot learning is a function of model size (Brown et al., 2020; Rae et al., 2021; Srivastava et al., 2022; Smith et al., 2022; Wei, Tay, et al., 2022). One of the advantages of in-context and zero-shot learning is not needing large amounts of task specific data and costly fine-tuning for each task, but this is compromised by a higher cost caused by the requirement of larger models. For this reason, it is important to still explore possibilities of using smaller models in a zero-shot and in-context setting, which the current work does.

Large Transformer based language models still tend to struggle with reasoning in an in-context and zero-shot setup (Brown et al., 2020; Rae et al., 2021; Smith et al., 2022; Srivastava et al., 2022), demonstrated by worse performance than supervised task specific models, fine-tuned pre-trained models and most importantly, humans. On the contrary, humans reason effortlessly, being a skill that is already acquired at a young age (Keenan, Ruffman, & Olson, 1994; Piaget & Inhelder, 2008; Gazes, Hampton, & Lourenco, 2017). This is supported by three observations. First, our ability to navigate everyday life, which requires a lot of implicit reasoning. Second, the fact that language models are not able to outperform humans in reasoning tasks (Srivastava et al., 2022). Last and most importantly, reasoning datasets that are used to evaluate language models are often annotated by humans, demonstrating that we have no difficulty with doing such tasks. In this research, we make an analogy between how children learn to reason and artificial learning; we take inspiration from factors that might facilitate the learning of reasoning by children and apply them to zero-shot reasoning in GPT models.

Out of in-context and zero-shot learning, in-context learning has shown most promising performance, but the underlying processes of in-context learning are poorly understood. Much research has been conducted to understand how in-context learning works (S. M. Xie, Raghunathan, Liang, & Ma, 2021; von Oswald et al., 2022), and what the mechanisms are that drive in-context learning (Min et al., 2022; Chan et al., 2022). Chan et al. (2022) has identified characteristics of data that models are originally trained on that facilitate in-context learning in Transformers. Initial analysis indicates that child directed speech shows less of these properties compared to types of language normally used to train language models. This leads to the intuition that children often do not perform tasks in an in-context fashion. Thinking about it, humans learn most tasks involving natural language implicitly from our daily linguistic input, being able to perform many different tasks without the need for task-specific examples beforehand. This resembles zero-shot learning. So, perhaps it is better to focus on improving zero-shot learning, rather than in-context learning, in order to approach human-like intelligence. Additionally, in-context learning is an emergent property that only arises in large models (in terms of parameters), as opposed to zero-shot learning, which has shown to be possible in smaller models as well (Mikolov, Chen, Corrado, & Dean, 2013). Thus, the other advantage of focusing on zero-shot as opposed to in-context is that there is no requirement of having large models. Previous research proposed that human acquisition of reasoning is partly facilitated by our linguistic input, providing latent information about how to perform reasoning tasks (Falmagne, 1990). This leads to our first research question:

Q1: Can zero-shot inference in Transformer based language models be improved by fine-tuning on human-like data?

We investigate this using a computational experiment in which we fine-tune GPT-2 on child directed speech and subtitles data. These represent the two kinds of input readily available to a child; the first represents speech that is directed to the child by for example the parents and the second represents conversations between adults that could be picked up by the child, regardless if the child was meant to hear them or not. This experiment is described in more detail in section 4. In this way, we make an analogy between input from which children learn and input from which models learn.

Moreover, previous research has also found that children are better at reasoning when they tackle the problem from an imaginary perspective (Dias &

Harris, 1988; Richards & Sanderson, 1999). Possibly, this finding can be applied to enhance zero-shot reasoning in language models by prompting the model to “imagine”. Second, previous research has shown that a technique called chain-of-thought (COT) prompting can substantially improve zero-shot performance on reasoning tasks. This technique has been shown to be fruitful in an in-context (Wei, Wang, et al., 2022) and zero-shot setting (Kojima, Gu, Reid, Matsuo, & Iwasawa, 2022). COT prompting relies on teaching the model to reason step by step, instead of going from problem to solution in one step. Using these findings, we can make an analogy between the way a reasoning task is presented to humans versus models. Experiment 2 takes inspiration from Betz, Richardson, and Voigt (2021) and Kojima et al. (2022), who prompt language models to elaborate on a presented problem, after which their output is used as additional input to solve the problem, simulating step-by-step reasoning. Similarly, we first prompt the model to imagine a premise or set of premises and use it’s generated “imagination” as further input to assess the correctness of a conclusion. Hence, our second research question:

Q2: Can zero-shot reasoning in Transformer based language models be improved by prompting the model to “imagine”?

Additionally, humans struggle more with reasoning when the presented scenario is not in line with their world knowledge (Dias & Harris, 1988; Richards & Sanderson, 1999; Gazes et al., 2017), a pattern also found in language models (Dasgupta et al., 2022). We seek to evaluate what effect both fine-tuning or prompting to imagine has on this pattern by including a reasoning task with fantastical scenarios. Lastly, the two described experiments are combined to see how fine-tuning and imagination prompting interact with each other. This final experiment is described in section 6.

In section (2), the motivation for this research will be explained and situated in academic context, starting with giving a general overview of Transformer language models in section 2.1. Then, we contrast in-context and zero-shot learning in section 2.2, and motivate the focus on zero-shot learning. Lastly, the underlying context for the two experiments is explained further in sections 2.3 and 2.4.

2 Theoretical Background

2.1 Transformer Language Models

Language models are machine learning models that are capable of predicting word probabilities given some context after being trained on large amounts of text data. Recently, large language models originating from GPT and BERT have become state-of-the-art in a wide range of tasks, such as text summarization, translation and question answering. These Transformer based language models use part of the Transformer neural network architecture that was originally proposed for machine translation.

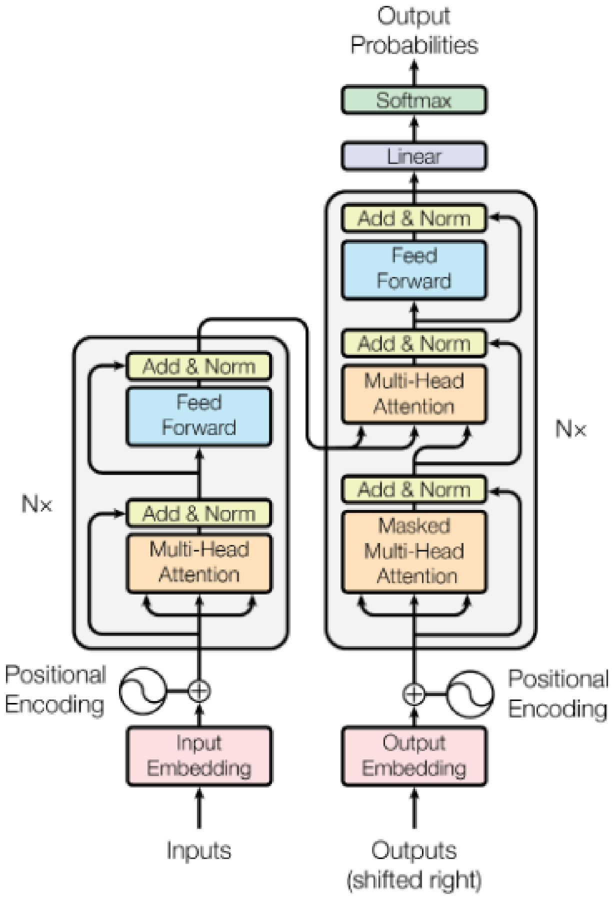


Figure 1: Transformer architecture (Vaswani et al., 2017)

Transformers consist of an encoder stack and a decoder stack (Vaswani et al., 2017). In general, the encoder produces an encoded representation of the input using multiple layers of self-attention and the decoder uses that representation to produce an output sequence symbol by symbol, taking the previously generated output symbols into account when producing the next output symbol. Figure 1 visualizes this process in more detail. The process starts with converting the inputs to input embeddings, adding positional encoding to retain information about the order of the sequence. The first element of the encoder layer is a multi-head self-attention mechanism, followed by a fully connected feed forward network. A full architecture consists of a stack of multiple of these encoder layers. The output of the encoder is fed into the decoder, which is also made out of a stack of multiple layers. Each layer consists of two multi-head attention mechanisms, followed by a feed forward network. The decoder layer takes the output of the encoder layer, together with previously generated output predictions, and uses those to predict the next output in the sequence. The use of an self-attention mechanism and the ability to process all the input words in parallel is what makes the architecture so successful. Self-attention works as relating every word in the input to every other word; some other words in the input might relate more to the word than others. By doing this, the model tracks relationships between words, which can sometimes, but not necessarily, be related to semantic relationships.

GPT models use decoder blocks from the architecture and is trained using the classic language modelling objective, namely to predict the next token in a sequence of text. GPT models thus process their inputs from left to right, making them particularly suitable for text generation. In contrast, BERT uses encoder blocks and is trained using a masked language modelling objective. This means that it is trained to predict a masked word anywhere in the sentence. Thus, BERT processes its input in both directions, making it able to encode the meaning of a word based on context on both sides.

In this research, we make use of GPT-2 and GPT-3. The decision of GPT over BERT is motivated by the fact that BERT models are not suitable for the generation of text, while GPT models are; we need to generate text in order to investigate imagination prompting in experiment 2. We use both GPT-2 and GPT-3, because they're publicly available and within the scope of resources, while still being able to show a contrast between smaller and larger models.

2.2 Emergent Generalization Capacities

Both in-context and zero-shot learning are among emergent generalization capacities of large Transformer based language models. An emergent generalization capacity is a capacity of a model that emerges only with sufficient scale (Wei, Tay, et al., 2022). In this section, I contrast zero-shot and in-context learning and motivate why the focus of this work is on zero-shot learning. Lastly, I discuss the domain that this work focuses on: reasoning.

2.2.1 In-context Learning

Large Transformer based language models possess the capability to perform in-context learning without being explicitly designed for it (Brown et al., 2020). In-context learning is the ability to generalize quickly from just a few examples of a new concept on which they have not been previously trained, without updating any weights in the model. For in-context learning, the model is prompted with a number of examples of the task followed by a query, after which the model yields the output by generating a sequence of tokens. Figure 2 shows this in more detail. In 1, some data from pre-training is shown. GPT models are trained to predict the next token given a passage of text. After pre-training, the network can be fed a context and query of any task. For example, predicting the nationality of a person, as shown in 2. Finally, 3 shows how context and query is concatenated and fed to the network. All context examples are separated by a delimiter and the query item is realized by omitting the last position, for which an answer is desired. The language model will then predict the most likely token for that omitted position. As opposed to fine-tuning, in-context learning has the advantage that it does not require a large dataset for the desired task; a few examples as context suffice.

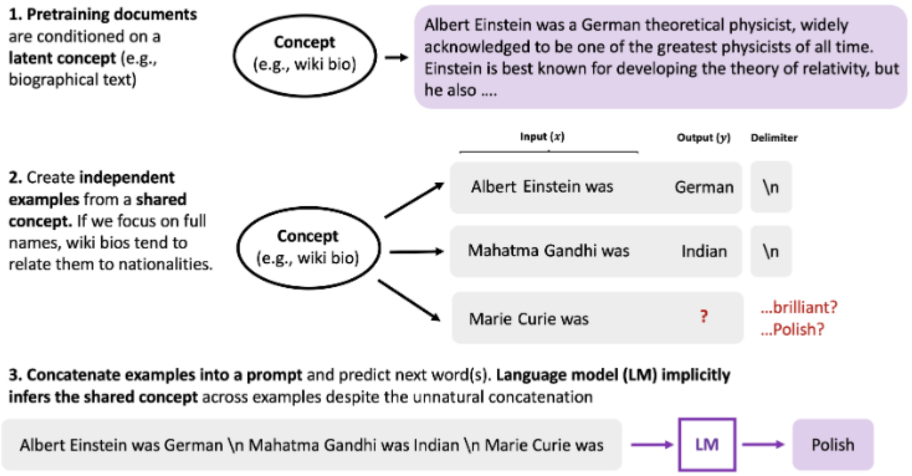

1. Pretraining documents are conditioned on a latent concept (e.g., biographical text)
Concept (e.g., wiki bio)
Albert Einstein was a German theoretical physicist, widely acknowledged to be one of the greatest physicists of all time. Einstein is best known for developing the theory of relativity, but he also ....
Input (x)
Output (y)
Delimiter
2. Create independent examples from a shared concept. If we focus on full names, wiki bios tend to relate them to nationalities.
Concept (e.g., wiki bio)
Albert Einstein was
German
\n
Mahatma Gandhi was
Indian
\n
Marie Curie was
?
...brilliant?
...Polish?
3. Concatenate examples into a prompt and predict next word(s). Language model (LM) implicitly infers the shared concept across examples despite the unnatural concatenation
Albert Einstein was German \n Mahatma Gandhi was Indian \n Marie Curie was
LM
Polish

Figure 2: Demonstration of in-context learning (S. M. Xie et al., 2021)

The mechanisms that drive in-context learning are largely unknown. Many efforts have been made to clarify the underlying workings of in-context learning, but there are conflicting results. The following paragraphs present previous research set out to investigate the underlying mechanisms of in-context learning from characteristics of both context and training regime. S. M. Xie et al. (2021) propose that a language model implicitly applies Bayesian inference when performing in-context learning. Namely, the model infers a latent concept given a set of examples and uses that concept to infer the most likely output token. The model has learned to do this during pre-training, where it predicts the next token given previous tokens by means of learning the underlying concept of the document. They prove this intuition using their GINC dataset, generated with a clear concept structure in mind. They show that encountering a large set of tokens during pre-training is not sufficient for in-context learning and that the presence of a concepts structure is a must. Moreover, they show that the model is not capable of in-context learning for concepts that are not seen during training. Lastly, they also prove that the length, number and ordering of examples matters for in-context learning accuracy.

On the other hand, von Oswald et al. (2022) claim that Transformer in-context learning mirrors gradient descent. They demonstrate that the output of a single self-attention layer with trained weights is similar to the output of a self-attention layer with weights constructed to mirror a single step of gradient descent. To verify that the underlying algorithm leading to these results resemble each other as well, authors investigate similarities between multiple steps of gradient descent and multiple self-attention layers and visualize loss over training steps.

Kirsch, Harrison, Sohl-Dickstein, and Metz (2022) show that Transformers can be meta-trained to act as general-purpose in-context learners. Meta-learning is the process of discovering new learning algorithms that are not manually designed into the model. In other words, these models learn to learn. One particularly ambitious goal of meta-learning is to train general-purpose in-context learning models. Such a model takes in training data, and produces test-set predictions across a wide range of problems, without any explicit definition of an inference model, training loss, or optimization algorithm. They find that for Transformers to be able to meta-learn, they need to encounter a large number of different tasks during training. Furthermore, they show the learned behavior evolves from memorization, to task identification, to learning-to-learn as number of tasks increase. The last of their findings is that Transformers benefit more from their large state size due to self-attention than a high number of parameters for learning-to-learn. State size refers to the amount of information about the input sequence that can be stored. This demonstrates that the Transformer model structure is crucial for achieving general-purpose in-context learning.

In contrast to these attempts of characterizing the inference process happening during in-context learning, Min et al. (2022) investigate the role of context characteristics. Interestingly, they show that true input-label contexts are not required. They prove this by replacing true context labels by random labels and point out that this only lowers performance slightly. In addition, changing proportions of incorrect versus correct labels does not affect performance gains significantly. However, having some input-label pairing does impact performance, shown by a drop in performance when omitting labels entirely. Increasing the number of input-label pairs presented to the model does not improve performance past eight pairings, which contradicts previous findings (S. M. Xie et al., 2021). However, the format of pairing inputs to labels does improve performance over having no pairings, but just labels or just input. Furthermore, they substitute labels with random English words to investigate the role of the label space. Performance decreases when using random English words, showing that conditioning on a label space contributes to performance gains. Relatedly, inputs have been replaced by out of distribution inputs to assess the influence of the underlying distribution of input examples. Results suggest that inputs from the same underlying distribution as the training data contribute to performance gains.

Finally, Chan et al. (2022) show that certain data-distributional properties of the training data that could lead to the ability to perform in-context learning. They used the Omniglot dataset, an dataset consisting of images with handwritten characters and labels. The model was fed with a sequence of 8 image-label pairs, after which the task was to find the label for another image. In particular, they identify burstiness as a property promoting in-context learning. They found this by varying levels of burstiness in the training data by changing the balance between bursty and non-bursty sequences. In the bursty sequences, the query class appeared 3 times in the context. For the non-bursty sequences, the image-label pairs were drawn randomly and uniformly from the full Omniglot set. Models displayed better in-context learning when there was more burstiness in the training data. This might explain the success of in-context learning in natural language processing, since burstiness is a prevalent phenomenon in natural language.

Burstiness is a characteristic of many natural phenomena, such as traffic or natural disasters. For example, traffic jams tend to occur all in a close time frame, instead of steadily throughout the day. They tend to occur in "bursts", hence the term burstiness. In this work, we focus on burstiness in language.

The finding that burstiness is an important factor for emergent in-context learning together with the claim that children's linguistic input plays a role in learning (Falmagne, 1990) leads to the question if burstiness in children's linguistic input is one of the driving forces behind learning, and if this could be leveraged to train Transformer language models more efficiently. However, an initial analysis strongly indicates that child directed speech is much less bursty than other types of corpora that large language models are normally trained on such as Wikipedia, OpenSubtitles, or Brown. Therefore, we suspect that burstiness is not the driving force behind learning for children.

Apart from burstiness, (1) a large number of rarely occurring classes, (2) multiplicity of labels and (3) within-class variation were identified as properties driving emergent in-context learning in Transformers. It is likely that child directed speech also lacks some of these properties. Starting with (1), child directed speech does not contain as many rare words as other corpora according to our analysis. Moving on to (2), multiplicity of labels refers to the assignment of multiple different labels to an image class. In the context of language modelling, the image class would be a specific sentence and the label would correspond to the word that follows it. Although our analysis does not test for this property, it is known that children use simpler language and adults adapt their linguistic complexity accordingly (Kunert, Fernández, & Zuidema, 2011). This indicates that child directed speech probably shows less variation in continuations of a specific sentence than adult directed language. Similarly, (3) refers to variation of images from a specific class, which corresponds to variations in sentences with the same meaning in language modelling. Again, the lower degree of linguistic variation in child directed speech indicates leads to the conclusion that sentences with the same continuations are more constant than in adult directed language.

2.2.2 Zero-shot Reasoning in Large Language Models

We believe that properties important for in-context learning are not as prevalent in the linguistic input of a child, because humans don't use in-context learning. For many daily tasks, humans learn implicitly from all the sensory input they receive and are able to make generalizations and solve tasks without explicitly being primed with a number of examples of the task at hand. For example, when presenting a human with the premise "Daniel owns a dog", it is straightforward to infer that "Daniel is a pet-owner", even though they didn't see any examples of such a reasoning task immediately before.

In the context of language modelling, this ability is referred to as zero-shot learning, where a model is asked to perform a task in the absence of any examples (Palatucci, Pomerleau, Hinton, & Mitchell, 2009). Contrary to in-context learning shown in figure 2, the model would be only prompted with the query (i.e. only "Marie Curie was ..."). This is also an emergent property of large language models, like in-context learning. Work has been done to test different language models in a zero-shot setting on a range of tasks, like summarization, translation and traditional language modelling and show that the size of a language model is crucial for the success of zero-shot learning (Radford et al., 2019; Brown et al., 2020). Zero shot has shown to be a promising avenue, but currently performance is often still lower than state-of-the-art, human performance or in-context learning results (Brown et al., 2020; Rae et al., 2021; Smith et al., 2022; Srivastava et al., 2022). Nevertheless, it shows that these models are able to learn from naturally occurring examples (pre-training data) rather than manually constructed (fine-tuning) datasets, which is something humans do as well. Still, models struggled with many tasks, achieving performance much lower than state-of-the-art. This indicates that there is still a lot of work to be done to approach human performance on tasks in a zero-shot setting, but eliminating the need for large task specific datasets is a big step towards general intelligence. Additionally, zero-shot generalization is a desirable skill not only because it reflects how humans solve tasks, but also because it emerges already in smaller models, which are less computationally costly. For example, work by Mikolov et al. (2013) shows that a small model like Word2vec proves to perform well at tasks in a zero-shot setting. In particular, it performed well at SemEval-2012 task 2, a task aimed at evaluating semantic similarity of relationships. All in all, this highlights the importance of focusing on zero-shot solving of tasks instead of in-context in order to strive for general human-like artificial intelligence, one of the overarching goals of AI.

Specifically, large language models have shown relatively good zero-shot performance in tasks that are intuitive and straightforward, but struggle with tasks requiring multi-step reasoning (Brown et al., 2020; Rae et al., 2021; Smith et al., 2022; Srivastava et al., 2022). To clarify, these models are skilled at answering factual questions, such as "Who was Alan Turing?" or text summarization, but struggle more with tasks such as the following: "Mary is baking a cake. The recipe calls for 12 cups of flour 14 cups of sugar and 7 cups of salt. She already put in 2 cups of flour. How many more cups of flour does she need to add now?" The latter requires doing multiple steps of arithmetic reasoning, whereas the first is essentially recalling stored information. Examples of these difficulties are shown in figures for GPT-2 (124M parameters). It can be seen that different issues are at play, ranging from not understanding the type of answer that needs to be given (last two options in figure 3b), to giving an appropriate but incorrect answer (first two options in figure 3a). However, figure 3c shows that the model has no problem understanding and answering a question that only requires recalling facts.

GPT-3 actually answered all these questions correctly, supporting that zero-shot reasoning performance indeed scales with model size (Brown et al., 2020;

Rae et al., 2021; Srivastava et al., 2022; Smith et al., 2022; Wei, Tay, et al., 2022). Still, these larger models do make mistakes.

Contrastingly, reasoning comes very naturally to humans, shown by humans consistently outperforming language models on several reasoning benchmarks (Srivastava et al., 2022), the fact that most reasoning datasets are annotated by humans and our ability to navigate everyday life which is filled with latent reasoning tasks. Due to this sharp performance contrast between humans and machines regarding reasoning, we suspect models might benefit from applying knowledge about human reasoning to artificial reasoning. This is why we focus on reasoning specifically in this work, and leverage the knowledge about human reasoning and apply it to zero-shot reasoning in machines.

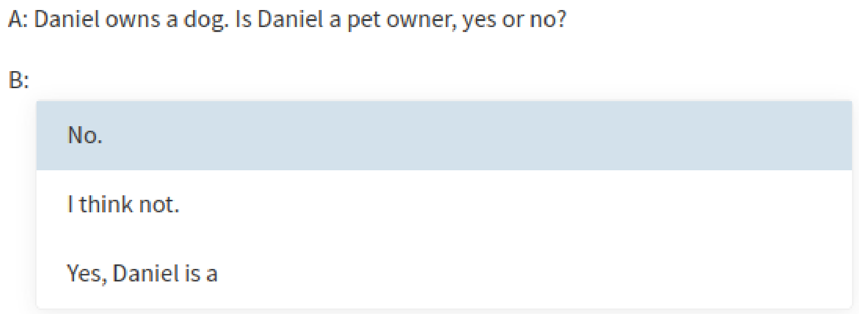
One might wonder why we do not focus on meta-learning, as was discussed in the context of work by Kirsch et al. (2022). The decision to not focus on this technique is that this requires enormous amounts of training data and very large models, which is outside the scope of resources.

The task framing of reasoning can be done in different ways. One option is to frame it as question answering, where the problem is formulated as a question and the model needs to generate an answer, as shown in figure 3. The seemingly poor performance on reasoning questions of GPT-2 using this strategy motivates using another strategy in this work. Possibly, using another strategy will already improve performance, even for a small model like GPT-2. Specifically, we will rely on classification using embedding similarity, as has been used by others before (Veeranna, Nam, Mencıa, & Fürnkranz, 2016; H. Xie & Virtanen, 2021) and was suggested in documentations of both GPT-2 and GPT-3 model libraries. Section 3.1.3 discusses a more in depth explanation of this strategy.

2.3 Towards a Human-like Linguistic Experience

2.3.1 Fine-tuning in Machines

Much research on how to improve performance on zero-shot reasoning has been done, with methods like fine-tuning or COT. First, we will focus on research using fine-tuning to improve reasoning of language models. Firstly, Wei et al. (2021) show that instruction fine-tuning improves zero-shot performance on unseen tasks, including reasoning. For a wide range of tasks they constructed ten instruction templates and fine-tuned the model using various examples from datasets formatted conforming these templates. The training objective of this fine-tuning was to predict the label, given an input in the instruction format of one of the constructed templates. For the reasoning task, this label would correspond to “entailment” or “no entailment”. It improved zero-shot performance on tasks usually formulated as an instruction, but is less successful at improving zero-shot performance on tasks that are formulated according to the language modelling objective (commonsense reasoning and coreference resolution formulated as sentence completion tasks). Moreover, models that are trained using this method do not generalize well to tasks with unseen labels.



A: Daniel owns a dog. Is Daniel a pet owner, yes or no?

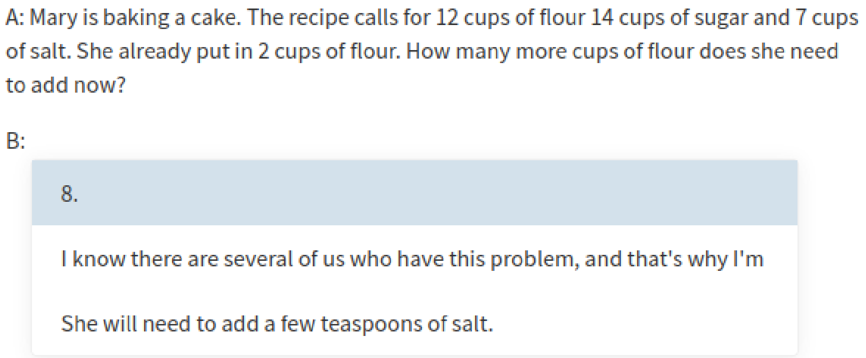
B:

No.

I think not.

Yes, Daniel is a

(a) Output of GPT-2 small for "Daniel owns a dog. Is Daniel a pet owner, yes or no?"



A: Mary is baking a cake. The recipe calls for 12 cups of flour 14 cups of sugar and 7 cups of salt. She already put in 2 cups of flour. How many more cups of flour does she need to add now?

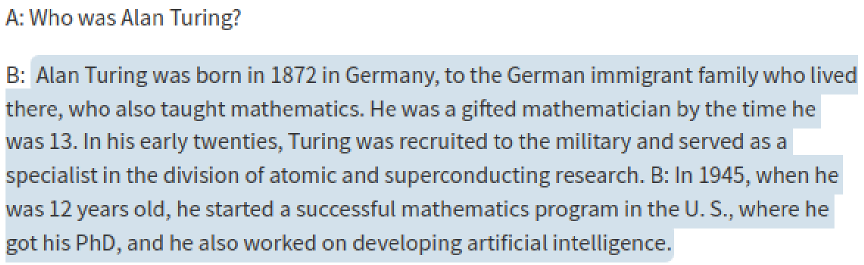
B:

8.

I know there are several of us who have this problem, and that's why I'm

She will need to add a few teaspoons of salt.

(b) Output of GPT-2 small for "Mary is baking a cake. The recipe calls for 12 cups of flour 14 cups of sugar and 7 cups of salt. She already put in 2 cups of flour. How many more cups of flour does she need to add now?"



A: Who was Alan Turing?

B: Alan Turing was born in 1872 in Germany, to the German immigrant family who lived there, who also taught mathematics. He was a gifted mathematician by the time he was 13. In his early twenties, Turing was recruited to the military and served as a specialist in the division of atomic and superconducting research. B: In 1945, when he was 12 years old, he started a successful mathematics program in the U. S., where he got his PhD, and he also worked on developing artificial intelligence.

(c) Output of GPT-2 small for "Who was Alan Turing?" after continuously choosing the first continuation option

Figure 3: Some example outputs of GPT-2 small

Ye, Kim, Jang, Shin, and Seo (2022) introduce flipped learning to improve this. With flipped learning, the model learns to predict a task description given an input-label pair, instead of outputting labels given a description. Conditioning on the labels instead of outputting them avoids label overfitting and yields better zero-shot generalization to tasks with new labels. Even though this is a great advancement, it requires specific fine-tuning data for all possible tasks.

2.3.2 Linguistic Input in Humans

As has been mentioned before, humans generally reason with ease, even young children. Piaget's theory of cognitive development poses that cognitive development of a child is composed of four stages, of which the concrete operational stage is the third (Piaget & Inhelder, 2008). This stage takes place from seven to eleven years old and it is where reasoning is acquired. However, there is little agreement on the exact age that this ability is acquired. Keenan et al. (1994) show through a series of experiments that children understand that inference is a source of knowledge around the age of five. They presented children with a doll and instructed them to think of the doll as a real person. The child was then presented with a red and green marble, which were shown to the doll and the child and then placed into a covered dish. The doll was placed behind the screen and the experimenter told the child that the doll could no longer see what the child and experimenter were doing, but could still hear them. One of the marbles was moves into a bag while the experimenter spoke aloud to the doll about what they were doing. The doll was then moved from behind the screen and the child was asked if the doll knew the color of the marble in the bag. The experiment had four conditions (1) inference salient, where the marbles were both the same color and the child was reminded by the experimenter about this and the fact the doll knew this as well; (2) inference unsalient, which is the same as salient but without the reminders; (3) uninference, where the marbles were a different color; (4) false-belief, where children were shown a Smarties box and asked what they thought was inside, after which the experimenter opened the box revealing crayons instead of Smarties. The child is then asked what another child would think is in the box.

More recently, Gazes et al. (2017) have shown evidence of transitive inference in infants, although no linguistic inputs were used in their experiments. They used a play setting where they used dolls to act out scenarios indicating dominance relationships between the dolls. For example, they showed infants a video of two dominance interactions between three puppets (bear dominates elephant and hippo dominates bear) meaning that through reasoning one can infer that the hippo is most dominant, followed by the bear and elephant. Infants then viewed interactions between the hippo and the elephant in scenarios that were either consistent or inconsistent with the inferred dominance hierarchy. Infants looked longer to inconsistent than consistent interactions. The authors claim this difference was influenced by their inferred knowledge about the dominance hierarchy.

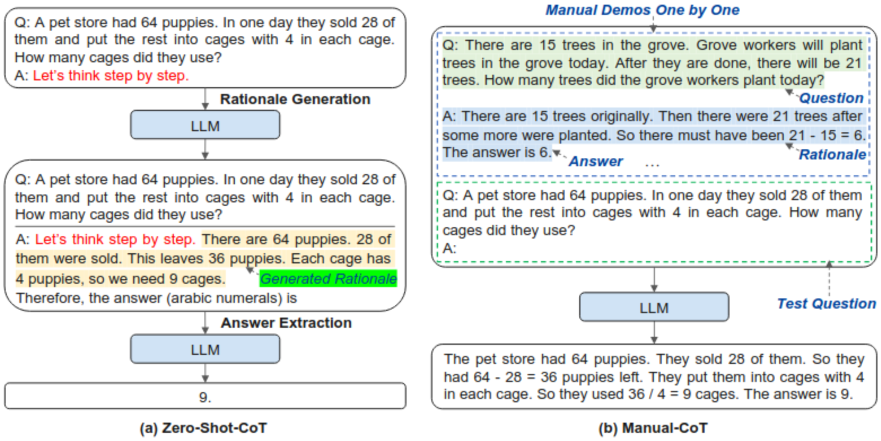
Thus, it is clear that children acquire capabilities of reasoning at a young age. Falmagne (1990) claims that linguistic input of children plays a role in the acquisition of logical reasoning. Specifically, the claim is that although origins of logical reasoning may be acquired through sensorimotor experience, higher forms (deductive reasoning in particular) are acquired through linguistic input once the child is able to process this input. This is motivated by two main arguments: (1) the prominent role of logic in linguistic theory and the resemblance between logic and syntax and (2) because of this resemblance and the fact that children acquire syntax of their language, it is psychologically plausible that logic is also acquired through language. If this is true, then linguistic input of children might be useful for language models to develop reasoning skills as well. We will also investigate if this analogy holds for other types of reasoning than logical deduction. This leads to our first research question: **Can zero-shot reasoning in Transformer based language models be improved by fine-tuning on human-like data?**

2.4 Towards Imagination-based Reasoning

2.4.1 Chain of Thought Prompting for Zero-shot Reasoning in Machines

Research on prompt design for language models has shown that the way a model is presented with a problem has major effects on its ability to perform zero-shot reasoning. A major advantage of using prompting to elicit reasoning in language models as opposed to fine-tuning is that it reduces the need for large amounts of task specific data. Betz et al. (2021) have shown that GPT-2 performs better in the context of deductive reasoning tasks when it is fed with a self-generated elaboration of the problem before generating a final answer. Similarly, Wei, Wang, et al. (2022) introduce a concept they name “chain-of-thought (COT) prompting”, where a language model is first prompted with some examples of reasoning tasks and answers in which the answers are preceded by a multi-step thought process, as shown in figure 4. This approach results in performance gains in models with around 100B parameters for arithmetic, commonsense and symbolic reasoning tasks. Interestingly, Kojima et al. (2022) take this approach a step further, by only preceding the task description by “let’s think step by step” in a zero-shot setting, having no examples of task-answer pairs. This eliminates the workload of manually creating a set of examples. By doing this, the language model generates an elaboration on the problem, which can be presented to the language model again to obtain the final answer (see figure 4). Remarkably, adding this one sentence to the prompt resulted in significant performance gains on zero-shot reasoning. However, performance gains were less than those obtained by Wei, Wang, et al. (2022). Zhang, Zhang, Li, and Smola (2022) introduce automatic chain-of-thought prompting, which eliminates the need to manually construct examples. It works by sampling questions that are representative for the task at hand from existing data and letting the model generate the reasoning chain for those. These questions and their reasoning chains are used as examples for the chain-of-thought prompting. This method matched or exceeded performance of manual chain-of-thought prompting.

Even small models can benefit from the COT by fine-tuning them on COT outputs generated by larger models, which has led to performance gains on several kinds of reasoning datasets (Magister, Mallinson, Adamek, Malmi, & Severyn, 2022). This shows the potential of these small models for reasoning tasks, but this approach does not match the goals of this research, as we want to test the limits of smaller models without relying on larger models entirely. In a sense, this fine-tuning on COT outputs of larger models can still be seen as task specific training, as these outputs resemble examples of reasoning. This makes the approach not purely zero-shot, whereas the goal of this research was to focus on zero-shot reasoning. The work of Kojima et al. (2022) matches our purpose best, which is why our imagination prompting approach will inherit from their approach.


Q: A pet store had 64 puppies. In one day they sold 28 of them and put the rest into cages with 4 in each cage. How many cages did they use?
A: Let's think step by step.
Rationale Generation
LLM
Q: A pet store had 64 puppies. In one day they sold 28 of them and put the rest into cages with 4 in each cage. How many cages did they use?
A: Let's think step by step. There are 64 puppies. 28 of them were sold. This leaves 36 puppies. Each cage has 4 puppies, so we need 9 cages.
Generated Rationale
Therefore, the answer (arabic numerals) is
Answer Extraction
LLM
9.
Manual Demos One by One
Q: There are 15 trees in the grove. Grove workers will plant trees in the grove today. After they are done, there will be 21 trees. How many trees did the grove workers plant today?
Question
A: There are 15 trees originally. Then there were 21 trees after some more were planted. So there must have been 21 - 15 = 6. The answer is 6.
Answer
...
Rationale
Q: A pet store had 64 puppies. In one day they sold 28 of them and put the rest into cages with 4 in each cage. How many cages did they use?
A:
LLM
Test Question
The pet store had 64 puppies. They sold 28 of them. So they had 64 - 28 = 36 puppies left. They put them into cages with 4 in each cage. So they used 36 / 4 = 9 cages. The answer is 9.

(a) Zero-Shot-CoT

(b) Manual-CoT

Figure 4: Visualizations of the workings of chain-of-thought prompting as done by (Kojima et al., 2022) (a) and by (Wei, Wang, et al., 2022) (b)

2.4.2 Imagination for Zero-shot Reasoning in Humans

It seems that "thinking in steps" is an important aspect of good reasoning performance in machines and that it is important to instruct models to do this explicitly. Similarly, the way tasks are presented to humans is also crucial for their performance on those tasks. Research on logical inference in children shows that children are able to reason better when they are asked to imagine the scenario they are reasoning about, especially when the presented problems are not in line with their knowledge of the world (Dias & Harris, 1988; Richards & Sanderson, 1999). Richards and Sanderson (1999) found that when 2-, 3- and 4-year-olds were encouraged to use their imagination when reasoning, they reached logically correct conclusions even when the premises not being in line with the real world. The children also gave more theoretical justifications for their responses. These findings also suggest that children are able to reason before the concrete operational stage defined by Piaget. Dias and Harris (1988) show similar results. In their experiment there were two groups: one with children that are presented with a reasoning task in a make-believe play setting and thus was led to use imagination, and another with children that are presented with the reasoning task in a factual manner. The group of children which were presented with the reasoning task in a make-believe play setting performed better at the task than the other group, especially when premises contained contradictory facts. Possibly, imagination allows children to let go of their world knowledge more, which is needed to perform reasoning on a more abstract level. Interestingly, these content effects also show up in language models (Dasgupta et al., 2022). Like humans, models zero-shot reason more poorly about situations that are inconsistent with world knowledge or highly abstract.

This research shows that language models and humans have common characteristics when it comes to reasoning. Namely, better reasoning performance when the situation to be reasoned about is consistent with world knowledge and more importantly, making a mental representation about the problem before reaching a final answer. This leads to the expectation that like children, language models might become better reasoners when we prompt them to "imagine" the situation they have to reason about, especially if that situation clashes with world knowledge or is abstract. The second research question focuses on applying knowledge about fruitful problem representation with children to zero-shot chain of thought prompting (as in Kojima et al. (2022)) in models: **Q2: Can zero-shot reasoning in Transformer based language models be improved by prompting the model to "imagine"?**

3 Baseline Experiment

The first step is to establish a baseline to compare the results of the experiments to. For this, we used GPT-2 small (124M parameters) from the Huggingface library and GPT-3 (175B parameters) from the OpenAI API without any finetuning or COT prompting applied. Both of these models are also used as a starting point in all the following experiments. We used a zero-shot classification strategy that utilizes similarity between embeddings of items and label descriptions to classify statements as being a correct reasoning or incorrect reasoning (true/false), which was used in all experiments. This classification strategy has been used succesfully in other classification tasks before, such as topic classification (Veeranna et al., 2016) or audio classification (H. Xie & Virtanen, 2021). In addition, documentation of the Huggingface library and OpenAI API suggest this method for zero-shot classification. [1] We made use of four different task data sets to evaluate performance across different kinds of reasoning.

[1] `https://platform.openai.com/docs/guides/embeddings/use-cases`
`https://joeddav.github.io/blog/2020/05/29/ZSL.html`

3.1 Method

3.1.1 Data

To evaluate the models in terms of their reasoning capabilities, four different types of reasoning datasets have been used. These include general reasoning and more specialized types of reasoning such as, common sense reasoning, logical deduction and reasoning about fantastical scenarios. These datasets together form a diverse set of reasoning tasks. In this way, we can contrast different types of reasoning when drawing conclusions from the experiments. Tables 10, 11, 12 and 13 show what the task data look like in their original state.

The Stanford Natural Language Inference (SNLI) dataset consists of 570152 pairs of premises and hypotheses, in which the relationship between these is either an entailment, contradiction or neutral (Bowman, Angeli, Potts, & Manning, 2015). This dataset is a famous benchmark for natural language inference (NLI) and it tests for finding true entailment relations, where some implicit knowledge of the world is sometimes necessary. It is a suitable dataset to represent general reasoning. The rest of the datasets are used to represent more specialized types of reasoning. An example of an item from the dataset is:

- *Premise*: A soccer game with multiple males playing.
- *Hypothesis*: Some men are playing a sport.
- *Label*: Entailment

The com2sense dataset, from the BIG-bench [2] (Srivastava et al., 2022), has 1874 statements testing common sense reasoning. Statements are labelled as true or false. This dataset is different from SNLI, in that the items do not test for true entailment relations, but instead rely on an intuition that is heavily dependent on world knowledge, experience and "common sense" that comes effortlessly to humans.

- *Statement*: A knife could be used in place of a screwdriver.
- *Label*: True

The logical deduction dataset, from BIG-bench as well, consists of 8300 premises that are mapped to a set of possible hypotheses. One of these hypotheses is labelled as the correct one (false) and the others are labelled as false. As the same suggests, this dataset focuses on reasoning in the form of logical deduction. This contrasts the previously mentioned data sets by focusing purely on logical reasoning, where just the context provided by the premise is sufficient to reach the conclusion. In particular, this dataset focuses on ordering relations.

- *Premise*: On a shelf, there are three books: a black book, an orange book, and a blue book. The blue book is to the right of the orange book. The orange book is to the right of the black book.

[2] `https://github.com/google/BIG-bench`

- *Hypotheses*: [The black book is the leftmost., The orange book is the leftmost., The blue book is the leftmost.]
- *Labels*: [1, 0, 0]

The fantasy dataset also originating from the BIG-bench, is a relatively small dataset consisting of 201 scenarios followed by a question, labelled as either yes or no. This dataset was chosen to investigate how the model behaves when presented with fantastical scenarios and eventually compare this to previous research on how humans reason under such conditions.

- *Statement*: After millennia of waiting, an evil alien science corporation is brought down. The galactic government moves to remove an abandoned island laboratory that the company had put their lethal creatures on, only to find the island had been named Australia by a sentient species who now lives there. Animals appeared in Australia because they were created by God.
- *Label*: No.

Thus, each dataset serves a different purpose. The fantasy dataset is meant to specifically evaluate models on their ability of reasoning where it is necessary to abstract away from world knowledge and to rely on logical reasoning patterns to arrive at the hypothesis based on the premises. The com2sense dataset is the opposite, with the purpose to target intuitive reasoning, where knowledge about the world is necessary, but logical reasoning will not be of any help. Third, the purpose of the logical deduction dataset is to test models on deductive reasoning, where the conclusion can be reached using just the premises. Lastly, the SNLI dataset is a compromise between using world knowledge and logical reasoning and can be viewed as a more general type of reasoning. These four different datasets allow us to evaluate a diverse set of reasoning types.

Data preprocessing has been performed to make sure that all data is in the same format for seamless processing: one column with a statement containing a premise and hypothesis conjoined using the phrase "it follows that", one column with the reverse of that statement and two more columns with the label for the original and reversed statement being either 1 (true) or 0 (false). The addition of "It follows that" is done to create a statement structure that clearly represents a reasoning pattern, and that this pattern is the same across different datasets. It is important that it is the same across datasets, because this mitigates possible confounding effects on performance due to differences in statement structure. See tables Tables 10, 11, 12 and 13 for clarification on what premises and hypotheses might look like. The following paragraph describes in more detail how this preprocessing has been realized for each data set.

In case of SNLI, the phrase "it follows that" has been prepended to the hypothesis. Then, the premise and altered hypothesis have been concatenated into one statement. For the logical deduction dataset, the process is similar. However, we have multiple hypotheses for each premise in this case, as can be seen in table 13. Separate statements are made using each of these hypotheses, using the same strategy as has been used for SNLI. For the fantasy reasoning dataset, items already contain a concatenated premise and hypothesis. The hypothesis is always the last sentence, so the phrase "It follows that" has been prepended to the last sentence of each item. The com2sense data is of a different nature, because items are not consistently structured as having a premise and hypothesis, but are statements of any structure that can be evaluated using our common sense. For this reason, the data has been preprocessed differently. The phrase "it is straightforward that" has been added to each item wherever this naturally fits. For a comparison of what the transformed data looks like compared to the original data, see tables 14, 15, 16 and 17.

Once all items had been consistently transformed, reversed versions of all items were made to control for noise due to random variation in the data. This is clarified further in section 3.1.3. For each datapoint in the dataset, a negation has been added to the dataset by conjoining the premise and hypothesis with "it does not follow that" instead of "it follows that". Reversed versions of statements from the com2sense dataset have been created by including "it is not straightforward that" instead. All reversed statements were assigned to the class label that is the inverse of the class label belonging to the original, non-negated statement. So, if the original statement was labelled as 1, the negated version would be labelled as 0.

After transforming the data, 600 items from each dataset have been selected randomly, with the exception of the fantasy dataset due to its small size, where 140 items have been randomly selected. This downsampling is done for two reasons, namely to reduce computational load and to have extra unseen data for possible additional experimentation. These selections have been divided into 70% development and 30% test set. Class labels have been balanced in both development and test set. The reason for having a test set in addition to a development set is to spot potential overfitting.

3.1.2 Creating Embeddings

In order to perform classification, the data described in the previous paragraph needed to be transformed into sentence embeddings. First, all statements, including their corresponding negated versions, were tokenized using the GPT2Tokenizer. The tokenized items were then fed to the GPT2Model that returns the hidden states. The hidden state outputs were used to extract the embeddings. The model outputs the last hidden layer for each item in the dataset with the shape $(1, j, k)$, with j being the number of tokens in the item and k being the size of the hidden layer. From this matrix, we take the vector of the last token in the sequence. This is because GPT-2 uses self-attention, processing its input from left to right. Thus, the last token embedding contains information about all the preceding tokens as well. In case of GPT-3, the items could be fed to the embedding API without the need for tokenizing first or extracting embeddings from hidden states by hand [3]. For this, the text-

[3] However, it is not clear from documentation what kind of pooling is used to extract these embeddings from the hidden state outputs of the model.

embedding-ada-002 has been used. In the same way, GPT-2 and GPT-3 were used to create embeddings for three different label descriptions, which can be seen in table 1. These label descriptions correspond to the positive label (1). From now, we refer to these as “label descriptions” and will use the term “class label” or “label” when referring to the binary class labels 0 (false) and 1 (true), that are assigned to each item in a task dataset. Creating embeddings for the label descriptions is necessary for our classification strategy, which is explained in detail in the next section.

Letter	Label description
a	This is an entailment
b	This statement makes sense
c	This statement is logical

Table 1: Different label descriptions of the true class that were used in all the experiments

3.1.3 Evaluation

For each original statement and its negated counterpart, the cosine similarity of the embedding with respect to an embedding of a label description of the true class was calculated. A visualization of this can be seen in figure 5. This was done for all three label descriptions.

This strategy of classification is motivated by the intuition that an embedding of a correct reasoning statement should be more semantically similar (than an embedding of an incorrect reasoning) to the embedding of a description of the true label (such as label descriptions in table 1). By using original statements and their negated version we minimized noise in the data that could influence results. If we would only compare similarity scores given to positively labelled examples to negatively labelled examples, there could be random variation in the data giving a distorted image of model performance, making results more difficult to interpret.

Then the question remains why we did not compare the embedding of the original statement to an embedding of a positive label description versus an embedding of a negative label description, assigning the statement to the label corresponding to the embedding which is most similar. There is one disadvantage of this approach, that is solved by the current approach. Namely, such a method does not check whether the model assigns two contradicting statements (the original and negated version) to have the both true or false, which is not correct reasoning. The current approach avoids this limitation by comparing similarities of original and negated statements, choosing the one with the highest similarity as the correct reasoning, automatically deeming the other statement to be incorrect reasoning.

Computing cosine similarity between an original and negated statement and a label description yielded two similarity values for each label description. The similarity value of the negated statement was subtracted from that of the original item yielding one score according to equation 1, referred to as the difference score. This score was used to classify the original statement as either 1 (true) or 0 (false), meaning that the reasoning present in that statement is correct or not. So, in a situation where the model reasons very well, it would always yield a higher similarity score for the (negated) statement labelled as true (1) than for its (negated) counterpart that is labelled as false (0). Our classification strategy is motivated by the intuition that an embedding of a correct reasoning statement should be more semantically similar than that of an incorrect statement to the embedding of a description of the true label. So, in a situation where the model reasons very well, it would always yield a higher similarity score for the (negated) statement labelled as true (1) than for its (negated) counterpart that is labelled as false (0).

$$\mathit{diff} = \cos\theta_a - \cos\theta_b \tag{1}$$

$$\mathit{predict}(\mathit{diff}, \mathit{threshold}) = \begin{cases} 1, & \text{if } \mathit{diff} \geq \mathit{threshold} \\ 0, & \text{otherwise} \end{cases} \tag{2}$$

Thus, each item in a dataset resulted in three scores, corresponding to the different kinds of label descriptions. For each score, the accuracy, precision, recall and F1 were calculated using a threshold of 0. If the difference score exceeded 0, the original item was classified as 1 (true), otherwise as 0 (false). We will discuss results in light of this definition of "reasoning very well", which is tied to this threshold of 0.0 that was discussed in the previous paragraph. This first definition of reasoning will be referred to as *definition 1* from now

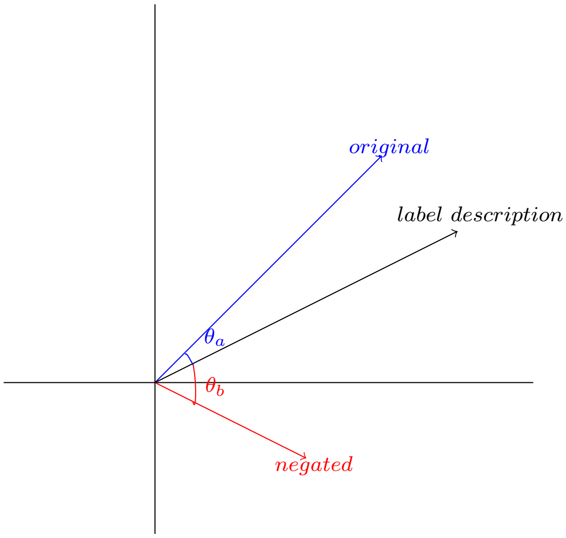
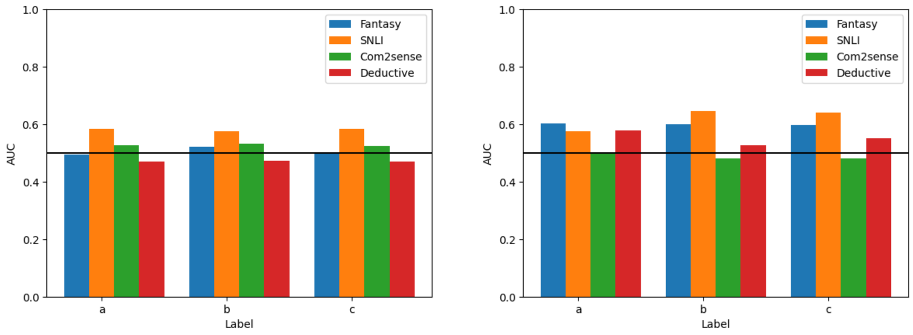


Figure 5: Visualization of classification strategy using embedding similarity

on. However, this is not the only way to define good performance. Another way is to define good reasoning is in terms of how well the model is able to classify statements correctly, irrespective of threshold. This second definition of reasoning will be referred to as *definition 2*. This is why the Area Under the Curve (AUC) and Pearson correlation between the difference scores and ground truth class labels have been calculated as well. These measures give a more complete picture of performance, independent of a predefined threshold. This is important, since we have no information about what the optimal threshold would be. The choice of 0.0 for the other measures was chosen with our firstly discussed notion of good reasoning in mind. However, there is no evidence that this would indeed be the optimal threshold to use for a good classification performance. AUC has been used as the metric for finding the best choice of label description to use for evaluation on the test set. The best label description is the one with the highest average AUC score on the development set over all different datasets. For the test sets, we repeated this process, but only using one label description that was selected as the best one.

3.2 Results



(a) GPT-2

(b) GPT-3

Figure 6: AUC on development sets for different label descriptions.

A general overview of AUC scores on the development sets can be seen in figure 6. The black line indicates what would be chance level AUC. A more extensive overview of results (including accuracy, precision, recall, F1-score and Pearson correlation) of different configurations on the development set for GPT-2 and GPT-3 are shown in table 18 and 19 in appendix B.1. It was a deliberate decision to only put F1-score and AUC in the main text tables in order to reduce the large table size. F1-score and AUC have been chosen as most informative to be presented here, as F1-score reflects precision and recall at the same time and AUC gives an idea of performance independent of a classification threshold (similar to correlation). Precision, recall and correlation have been stated in the appendix, but are also mentioned throughout the text for convenience.

It can be seen that GPT-2 has the highest AUC score on the development portion of SNLI irrespective of the label description, with the logical deduction dataset scoring the lowest in terms of AUC. It can be seen that AUC scores and correlations are in agreement; SNLI displays significantly positive correlation (0.12), which is in agreement with the relatively high AUC score. On top of that, AUC scores are similar across different label descriptions. Only SNLI and com2sense consistently exceed chance AUC.

GPT-3 shows slightly more variation across label descriptions. GPT-3 shows the highest AUC score for the SNLI dataset using label description b. Contrary to the logical deduction dataset scoring the lowest for GPT-2, the com2sense dataset shows the lowest AUC scores for GPT-3. The logical deduction and the fantasy datasets show a fair increase in AUC compared to GPT-2. Almost all datasets exceed chance level AUC, except for the com2sense dataset, which shows a decrease in AUC compared to GPT-2. Based on these reported development AUC scores, label descriptions have been chosen that will be used for testing.

Another observation is that precision and recall (can be found in appendix B.1 table 18) are often 0.0 in case of GPT-2, whereas this is not the case for GPT-3 (see table 19); precision ranges from 0.50 to 0.68 and recall from 0.05 to 98. The SNLI dataset shows a different pattern for GPT-2, namely a fairly high precision across different label descriptions (ranging from 0.80 to 1.00), and a fairly low recall (ranging from 0.005 to 0.04). Furthermore, the logical deduction dataset shows some variation with relatively higher values of precision and recall in case of GPT-2 (in the ranges of 0.00-0.50 and 0.00-0.37 respectively).

		F1	**AUC**
SNLI	**Overall best**	0.13	0.54
	Task best	0.0	0.55
Fantasy	**Overall best**	0.0	0.31
	Task best	0.0	0.31
Com2sense	**Overall best**	0.0	0.54
	Task best	0.0	0.54
Deductive	**Overall best**	0.49	0.55
	Task best	0.49	0.55

Table 2: Baseline results on the test set for GPT-2. A threshold of 0.0 has been used to calculate precision and recall (resulting in above F1 scores).

		F1	AUC
SNLI	**Overall best**	0.56	0.59
	Task best	0.55	0.61
Fantasy	**Overall best**	0.29	0.49
	Task best	0.70	0.66
Com2sense	**Overall best**	0.66	0.50
	Task best	0.55	0.46
Deductive	**Overall best**	0.18	0.45
	Task best	0.37	0.49

Table 3: Baseline results on the test set for GPT-3. A threshold of 0.0 has been used to calculate precision and recall (resulting in above F1 scores).

Results of the best configurations according to development AUC on the test sets is shown in tables 2 and 3. See tables 20 and 21 in appendix B.1 for a more extensive overview of measures including accuracy, precision, recall, F1-score and Pearson correlation. Here, similar patterns are visible again. GPT-2 shows generally higher AUC values and Pearson correlation (0.08 and 0.12, although not significant this time) for SNLI than for the other datasets, but this time the fantasy dataset shows the lowest AUC scores instead of the logical deduction dataset. All datasets except for the fantasy dataset exceed chance level AUC this time. This low AUC score on the fantasy dataset is confirmed by a significantly negative correlation (-0.31) as well. This performance is worse than what has been reported for the development set. Lastly, the described observations about precision and recall from the development set are also visible for the test set: precision and recall are often 0.0, with SNLI showing patterns of high precision (1.0 for the overall best configuration) and low recall (0.0 and 0.07) and the logical deduction dataset showing relatively higher precision (0.54) and recall (0.46).

GPT-3 also generally shows highest AUC score for the task specific best configuration on the fantasy dataset, followed by the SNLI dataset. Here, the logical deduction dataset scores the lowest. The most apparent increase in test AUC when comparing GPT-2 to GPT-3 is for the fantasy dataset, the task specific best configuration in particular. Other than that, SNLI also shows a fair increase, but the rest shows a decrease in AUC compared to GPT-2. However, precision and recall always show an increase compared to GPT-2 (0.36-0.64 and 0.12-0.97 versus lots of 0.0 occurrences for GPT-2), except for the logical deduction dataset (0.37-0.48 precision and 0.12-0.30 recall versus 0.54 precision and 0.46 recall for GPT-2). Precision and recall scores are more variable instead of being consistently 0.0, which was often the case for GPT-2.

3.3 Discussion

GPT-2 is able to reason better as defined by definition 2 (recall the contrast between defition 1 and 2 that was introduced in section 3.1.3) about scenarios such as those in SNLI than scenarios such as those in the logical deduction dataset. The high AUC scores on the SNLI dataset and the low AUC scores on the logical deduction dataset relating to GPT-2 indicate that GPT-2 is better able to separate correct from incorrect reasoning for SNLI than for the logical deduction dataset across a larger range of thresholds, meaning that similarity scores given to those two classes are further apart.

GPT-2 tends to classify everything as incorrect reasoning, except for the SNLI and logical deduction dataset. The fact that precision and recall are often 0.0 when using GPT-2 means that the classification using GPT-2 embeddings tends to always lead to classifying statements as 0 (false) when using a threshold of 0.0. Note again that precision, recall and accuracy are heavily dependent on the chosen threshold (in this case 0.0) and that there might be another threshold yielding better performance metrics. This means that the negated statement is consistently assigned a higher cosine similarity than the original statement, irrespective of the ground truth class labels of those statements.

A different pattern emerged for the SNLI dataset, namely high precision and low recall. This means that our model is unlikely to classify an item as 1 (true), but when it does it is usually a correct classification. The more varied precision and recall for the logical deduction dataset using GPT-2 displays a more varied classification instead of mostly classifying statements as 0 (false).

In addition, the similar results across label descriptions show that this classification method is not sensitive to exact label descriptions when using GPT-2, possibly because the model is already performing poorly regardless. The worse performance on the test set compared to the development set of the fantasy dataset by GPT-2 indicates possible overfitting on the label description. This is likely due to the small size of the dataset.

On the other hand, GPT-3 shows more variation in AUC across different label descriptions, meaning that the choice of label description can make a considerable difference in performance using this classification method together with GPT-3. The high AUC scores on the SNLI dataset (using label b) and the low AUC scores on the com2sense dataset relating to GPT-3 indicate that GPT-3 is better able to separate correct from incorrect reasoning for SNLI (using label b) than for the com2sense dataset across a larger range of thresholds, meaning that similarity scores given to those two classes are further apart. This can be interpreted as GPT-3 being able to reason better about scenarios such as those in SNLI than about common sense reasoning scenarios (at least under definition 2 of reasoning). Also here we can see some possible overfitting, but this time for the logical deduction dataset, since AUC is somewhat lower for the test set than for the development set.

Looking more closely to precision and recall, we can see that the model is not blindly classifying all items as 0 (false) anymore when using a 0.0 classification threshold. Whether this is an improvement is up to debate. When precision and recall are around 0.5, it means that classification performance is close to chance, which is not an improvement over making the same classification all the time, even though the numbers are higher. It is also important to remember that accuracy, precision and recall are dependent on a specific threshold of 0.0, which conceptually would represent reasoning as finding a correct reasoning more similar to a label description of correct reasoning than an incorrect reasoning. This means that the values higher than chance of precision and recall present in the results of GPT-3 do indicate an improvement of reasoning under definition 1. However, it could be that another threshold works better in practice for classification. It is up to debate whether definitions 1 and 2 of reasoning can both be viewed as true reasoning.

The increases in AUC for the logical deduction, fantasy and SNLI dataset show that GPT-3 is better at discriminating correct from incorrect reasoning statements than GPT-2 for many types of reasoning tasks. However, the decrease visible for com2sense indicates that GPT-3 struggles with this discrimination in the context of common sense reasoning, even more than GPT-2 does. Interestingly, this indicates that zero-shot classification performance on some reasoning tasks does not necessarily scale with model size, which is in disagreement with existing literature (Brown et al., 2020; Rae et al., 2021; Srivastava et al., 2022; Smith et al., 2022; Wei, Tay, et al., 2022).

This experiment is meant as a baseline for comparison of the following experiments. To summarize, two important results are found here. The most important result to take away from this experiment is that results are in conflict with existing literature proposing that zero-shot reasoning is an emergent property, of which performance scales with model size. Second, the experiment reveals that choosing the right label description is crucial for optimizing performance, at least in case of GPT-3.

4 Experiment 1: Fine-tuning

In this experiment, we repeat the baseline experiment, but instead of using "bare" models, we fine-tune them first. We evaluate two different fine-tuning datasets, namely one that represents linguistic input of adults and one representing linguistic input of children. In this way, we can contrast the effects of fine-tuning on human like linguistic input of adults versus children. Due to resource limitations, this experiment has been conducted using GPT-2 only.

4.1 Method

4.1.1 Data

There are two different datasets that have been used for fine-tuning, (1) the English portion of the OpenSubtitles dataset (Lison & Tiedemann, 2016) and (2) the Manchester and Manchester-EVA corpora in CHILDES (Theakston, Lieven, Pine, & Rowland, 2001; Lieven, Salomo, & Tomasello, 2009). For CHILDES, utterances of mother, father and child have been extracted and each conversation serves as one data point [4].

[4] Each .cha file is treated as one conversation, see `https://sla.talkbank.org/TBB/childes/Eng-UK` for more details on the structure of the dataset

All data points have been collected in a single plain text file. The GPT2Tokenizer from the Transformers library has been used to tokenize all items in the data and all of them have been padded to a maximum length of 1024: the maximum length the model can handle. Data points that were longer than this limit were truncated. Finally, this has resulted in a total of 941 separate documents having a total of 20676788 words for OpenSubtitles. For CHILDES, it resulted in 1116 documents with 38502626 words. Both datasets have been divided into 80% training and 20% test set. Even though this experiment is not focused on how well the model can predict text from CHILDES or OpenSubtitles, it is useful to evaluate whether there is overfitting, since this might be able to explain patterns in performance on the reasoning tasks. This is why data used for fine-tuning is divided into training and test data.

The task data is another component in this experiment. The procedure for preprocessing this data is the same as what has been described in section 3.1.1.

4.1.2 Fine-tuning

In this experiment we will evaluate **two** fine-tuned models on a set of reasoning tasks. The first model is fine-tuned on the training portions of the English part of the OpenSubtitles dataset, and will be referred to as the *Subtitles* model. The second model is fine-tuned on the training portion of the Manchester and Manchester-EVA data sets from CHILDES, and will be referred to as the *CHILDES* model.

A GPT2LMHead model from the Huggingface Transformers library has been fine-tuned on OpenSubtitles data and CHILDES training data for thirty epochs, with a learning rate of 0.0002 and weight decay of 0.01. The GPT2LMHead model is a bare GPT2 model, the stack of decoder blocks outputting raw hidden states from the Transformer architecture previously discussed in 2.1, with a language modeling head on top, meaning that there is an extra layer on top of the base model allowing for language modeling. These parameter values were chosen by means of comparing convergence rates of different combinations of values, with values differing by factors of ten. These reported values had a sufficiently fast convergence rate and minimally different loss on training and evaluation set. Fine-tuning has been done for three runs for each data set to investigate consistency of results under random shuffling of the data. Models are trained using the classic language modelling objective. The Subtitles and CHILDES training process yields model checkpoints after each epoch, which are the models with updated weights up to that epoch.

4.1.3 Evaluation

All model checkpoints (thirty, namely one for each epoch) have been evaluated using cross entropy loss for both training and test set of the Subtitles or CHILDES data, depending on what data it was fine-tuned on. The development sets of the reasoning tasks have also been evaluated at each checkpoint, using the same classification strategy as described in sections 3.1.3. Though, there is one important difference regarding the creation of embeddings. The creation of embeddings as described in section 3.1.2 used the GPT2Model without any finetuning applied. In this experiment, the model used for creating the embeddings corresponds to the model checkpoint that is being evaluated at that point. We chose AUC to represent performance over the course of checkpoints, since this measure is threshold independent and thus most informative and robust.

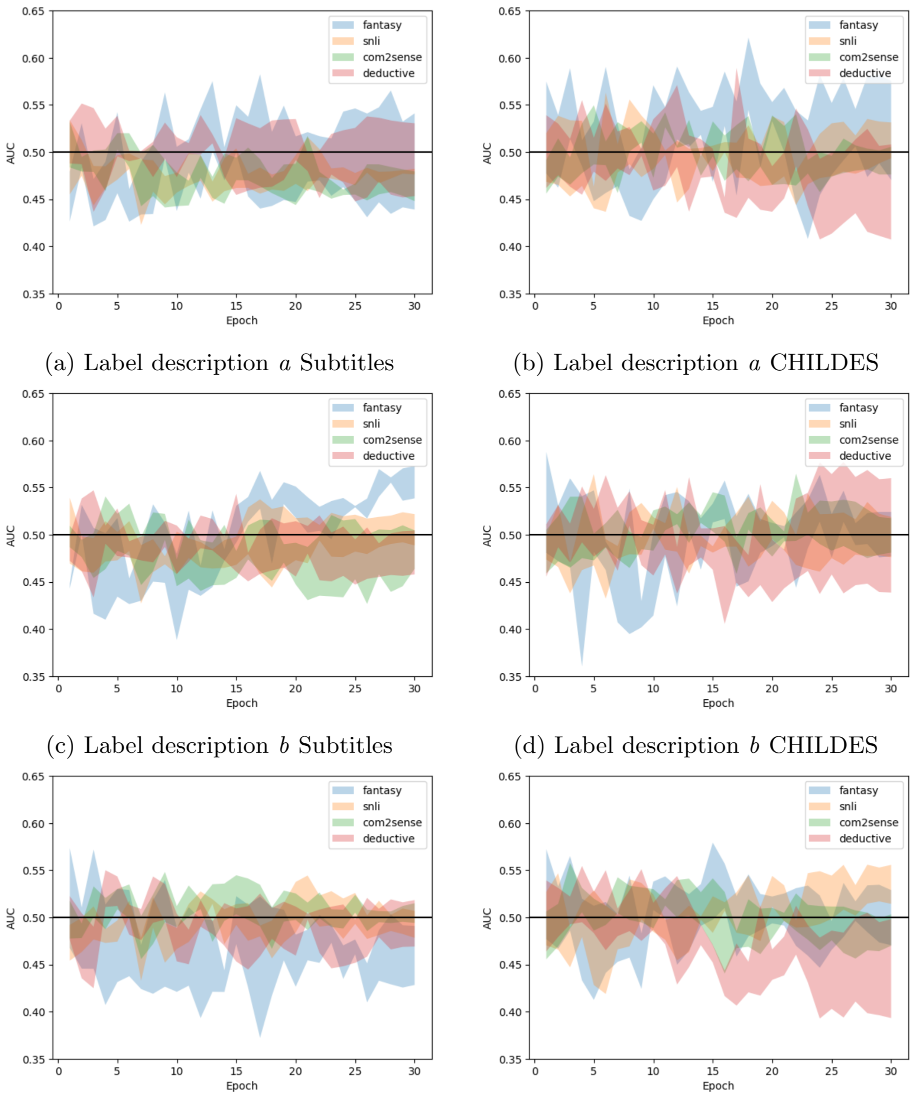
This evaluation has been done for all different checkpoints of both the Subtitles and CHILDES model, runs and label descriptions. Results of different runs have been averaged, meaning that there are three different performance metrics for each checkpoint, corresponding to the different label descriptions that were evaluated. The combination of checkpoint and label description that yielded the highest overall average of AUC was chosen as the best overall configuration. Moreover, the combination of checkpoint and label description yielding the highest average AUC for each dataset has been selected as well as task specific best configurations. Finally, the selected overall optimal model configuration has been used to give predictions for the test set of each reasoning task dataset. In addition, the selected optimal task specific model configuration has been used to give predictions for the corresponding reasoning test set. We compared results to the baseline model.

4.2 Results

Figure 7 shows how AUC progresses over the course of fine-tuning for both Subtitles and CHILDES models. The outer edges of each "line" represent the standard deviation over three runs. Runs differ in the order in which the training data is shuffled. The black lines again indicate the chance level AUC value.

It can be seen that label description *a* yields generally higher AUC values, as the lines reach above the chance AUC line the most for both Subtitles and CHILDES. Moreover, the models fine-tuned on CHILDES tend to exhibit an even slightly higher AUC for label a. Moreover, all datasets show high and low spikes in AUC across epochs. More specifically, the fantasy and logical deduction datasets often show more variation across runs than others.

Over the course of epochs, it seems that AUC is staying relatively constant around chance level in most cases. The only clear increase in AUC that can be seen is for the fantasy reasoning task using the Subtitles model and label description b. Due to the fantasy dataset being quite small, an additional 5-fold cross validation has been performed to ensure that this result is consistent across different development splits. Results of this can be seen in figure 8. It is visible that are results are consistent; figure 8b shows an increase in AUC across all different development splits. On the other hand, AUC declines for the logical deduction dataset using the CHILDES model with label descriptions a and c, but this trend displays much more variation.



(a) Label description *a* Subtitles

(b) Label description *a* CHILDES

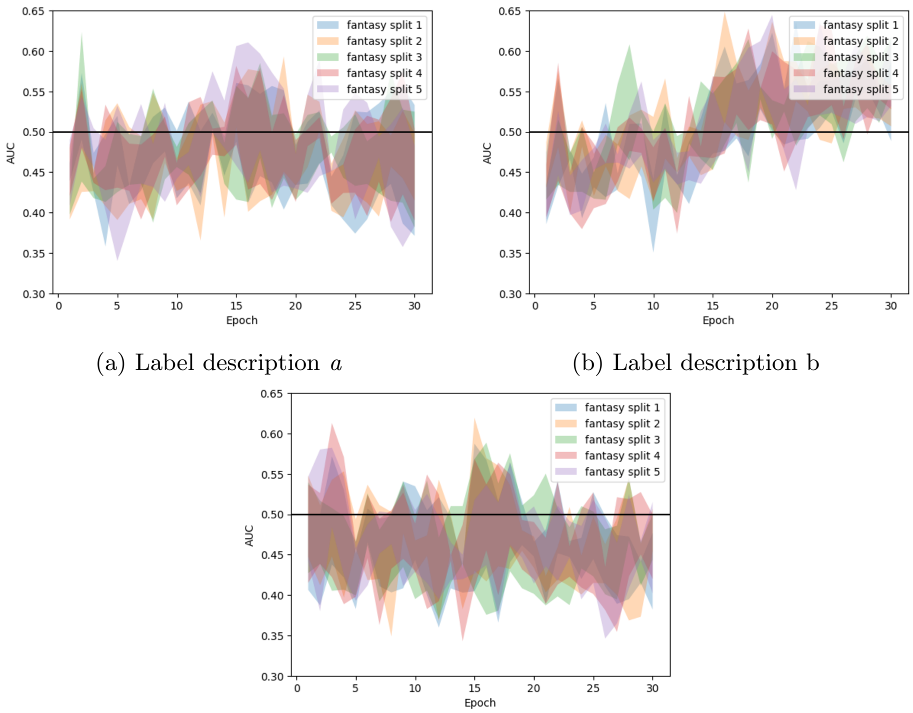
(c) Label description *b* Subtitles

(d) Label description *b* CHILDES

(e) Label description *c* Subtitles

(f) Label description *c* CHILDES

Figure 7: AUC over the course of fine-tuning



(a) Label description *a*

(b) Label description b

(c) Label description c

Figure 8: Results of 5-fold cross-validation on the fantasy dataset using the Subtitles model

Based on the development AUC values, optimal label descriptions and model checkpoints have been selected for testing. Tables 4 and 5 show F1 and AUC on the test set using the optimal configurations of model checkpoints and label descriptions. Tables 22 and 23 in appendix B.2 show a more complete overview of performance metrics. AUC has only increased for the fantasy dataset compared to the baseline test results. This holds for both the Subtitles and CHILDES models, but is more apparent for the Subtitles model. This is supported by the Pearson correlation being positive for the Subtitles model (0.09), but negative for the CHILDES model (-0.11). The correlation of Subtitles is a considerable improvement from the baseline experiment, which showed a negative value (-0.31).

AUC dropped or stayed similar for the rest of the datasets relative to the corresponding GPT-2 baseline AUC, and a drop is especially apparent for the com2sense dataset. This low AUC is supported by the Pearson correlation becoming significantly negative in the case of the Subtitles model (-0.003 and -0.10). AUC and correlation stayed relatively equal to and sometimes slightly lower (in case of logical deduction) than the baseline after fine-tuning for the rest of the datasets.

There are no occurrences of precision and recall being 0.0 anymore after fine-tuning on both CHILDES or OpenSubtitles. Precision and recall increased for most datasets (resulting in F1 also increasing) compared to the baseline (0.41-0.53 precision and 0.06-0.60 recall now versus many 0.0 occurrences in the baseline), whereas accuracy stayed similar (0.44-0.53). But, the opposite happened for the logical deduction dataset, where precision and recall decreased compared to the baseline (0.28-0.50 precision and 0.24-0.47 recall now versus 0.54 precision and 0.46 recall). Moreover, precision and recall is higher for the CHILDES model than the Subtitles model for all datasets, as reflected by F1 scores. The last observation is that precision, recall and conversely F1-score display relatively high standard deviation regarding the Subtitles model. For the CHILDES model, only recall displays high standard deviation (reflected by F1-score), but less than what is observed for the Subtitles model.

		Subtitles		**Baseline**	
		F1	**AUC**	**F1**	**AUC**
SNLI	**Overall best**	(0.24, 0.29)	(0.52, 0.04)	0.13	0.54
	Task best	(0.31, 0.27)	(0.50, 0.01)	0.0	0.55
Fantasy	**Overall best**	(0.21, 0.24)	(0.55, 0.09)	0.0	0.31
	Task best	(0.22, 0.31)	(0.56, 0.09)	0.0	0.31
Com2sense	**Overall best**	(0.24, 0.27)	(0.44, 0.05)	0.0	0.54
	Task best	(0.09, 0.08)	(0.51, 0.04)	0.0	0.54
Deductive	**Overall best**	(0.21, 0.27)	(0.51, 0.02)	0.49	0.55
	Task best	(0.37, 0.24)	(0.50, 0.03)	0.49	0.55

Table 4: Results for the Subtitles models on the test sets. A threshold of 0.0 has been used to calculate precision and recall (resulting in above F1 scores). Results are reported as (x, y), with x indicating mean and y standard deviation over three runs.

		CHILDES		**Baseline**	
		F1	**AUC**	**F1**	**AUC**
SNLI	**Overall best**	(0.56, 0.04)	(0.54, 0.01)	0.13	0.54
	Task best	(0.32, 0.10)	(0.51, 0.05)	0.0	0.55
Fantasy	**Overall best**	(0.36, 0.13)	(0.45, 0.02)	0.0	0.31
	Task best	(0.36, 0.13)	(0.45, 0.02)	0.0	0.31
Com2sense	**Overall best**	(0.41, 0.09)	(0.48, 0.03)	0.0	0.54
	Task best	(0.35, 0.07)	(0.49, 0.03)	0.0	0.54
Deductive	**Overall best**	(0.47, 0.11)	(0.52, 0.04)	0.49	0.55
	Task best	(0.40, 0.13)	(0.52, 0.01)	0.49	0.55

Table 5: Results for the CHILDES models on the test sets. A threshold of 0.0 has been used to calculate precision and recall (resulting in above F1 scores). Results are reported as (x, y), with x indicating mean and y standard deviation over three runs.

4.3 Discussion

The high and low spikes and variation in AUC during development indicate an instability in classification performance across model checkpoints, label descriptions and runs. Apparently, the classification strategy is highly sensitive to slight changes in model parameters, such as label description, and model weights. This is not just the case for AUC, but also for other performance metrics, shown by high variation in precision and recall on the test sets.

Fine-tuning on OpenSubtitles improves reasoning about fantastical scenarios. The fact that AUC stays relatively constant over epochs in most cases indicates that the fine-tuning is not affecting performance most of the time. However, there is a steep increase in AUC for the fantasy dataset using the Subtitles model with label b. Though, the lower AUC on the fantasy dataset for the test set compared to the development set when using the CHILDES model hints at overfitting. Luckily, the improvement that is observed when using the Subtitles model is persistent through cross validation and testing, which shows that these results are generalizable and that fine-tuning indeed can have an effect. It also shows that whether there will be an effect is heavily dependent on the task, having the right label description and the fine-tuning data.

A possible explanation for why the Subtitles model shows improvement for the fantasy dataset is that the OpenSubtitles fine-tuning data contains conversations originating from movies and tv shows. A portion of these movies and tv shows contains fantastical scenarios, for example genres such as as horror or sci-fi. These fantastical stories bear similarity to the scenarios that are present in the fantasy dataset. By having similar fantastical scenarios included in the fine-tuning data, the model is more familiar with such scenarios and it is likely that this helped the model reason about other fantastical scenarios. It would be similar as world knowledge being of help for common sense reasoning, but this time it is knowledge about fantastical worlds being of help for fantastical reasoning.

Another explanation could be that this improvement is due to the dialogue structure of the OpenSubtitles data. Speech acts often contains implicit meaning and intention (Grice, 1975), and a listener first infer this implicit meaning before answering back. The answer they give reveals more about the meaning of the initial speech act, and so on. This feedback loop between interlocutors thus reflects reasoning about eachothers intentions. It could be that this structure helps teach the model to reason. Even though CHILDES data also contains dialogues, these dialogues are between parents and toddlers and are much less informative because of that. Furthermore, the fact that the fantasy dataset requires the model to abstract away from world knowledge and rely on logical patterns of reasoning also hints at the structure and patterns in the OpenSubtitles data being of help for this task.

It seems that fine-tuning on CHILDES hurts logical deduction performance slightly. The reason for the slight decline in AUC for the logical deduction dataset that is seen for label description *a* and *c* using the CHILDES model could be that the linguistic content of the CHILDES fine-tuning data is hurting the model's ability to understand logical deduction. The scenarios in the logical deduction dataset are among the longest of all datasets, and longterm dependencies are important to understand them. Longterm dependencies means that you need to look back in a passage of text to understand what is said later in the text. In child directed speech, conversations consist of very short utterances and contain very little examples of longterm dependencies. This discrepancy between the fine-tuning data and the task data could be an explanation for the drop in performance over time.

Results on the test set also indicate that fine-tuning does not improve reasoning performance when comparing to the baseline experiment, except for the fantasy dataset. The CHILDES model also shows an improvement on the fantasy test dataset, but less of an improvement than the Subtitles model, which reflects the outstanding improvement for the Subtitles model with label description *b* that was seen during the development stage. The fact that AUC is already higher after the first fine-tuning epoch than it was in the baseline experiment means that just one epoch of fine-tuning with either CHILDES or Subtitles already yields a substantial performance boost.

Results on the test set also point out that fine-tuning on both CHILDES and OpenSubtitles hurts performance for the com2sense dataset in particular. This could be due to characteristics of the fine-tuning data not reflecting knowledge needed for common sense reasoning. Possibly, it contains knowledge that would go against real world knowledge needed for common sense reasoning, which is likely in case of the OpenSubtitles dataset for the same reason it would be helping the fantasy reasoning task. Namely, the fantastical scenarios in that dataset could hurt the model's ability to reason about scenarios where real world knowledge is needed, like for common sense reasoning. Moreover, children do not have a lot of experience in the world yet. This world knowledge is important for common sense reasoning and the lack of this knowledge in children might also be reflected in their linguistic input. The lack of content that is needed for common sense reasoning in the fine-tuning data might lead to the performance drops that were seen.

Moreover, the observation that precision and recall are never 0.0 anymore after fine-tuning on either CHILDES or OpenSubtitles indicates that fine-tuning helps prevent the model from always classifying everything as 0 (false). But again, this increase is not necessarily a large improvement, as was also explained in section 3.3, especially since both measures are far below 0.50 most of the time. This means that classification is wrong more than it is right, which could be a result of the threshold not being suitable. At least this shows us that fine-tuning does not improve reasoning (even for the fantasy dataset) when reasoning is defined as definition 1.

Lastly, it seems that fine-tuning on CHILDES data has a less negative impact than OpenSubtitles data according to definition 1, shown by the increase in precision and recall compared to the Subtitles model. This and the fact that there is no noticeable difference in accuracy indicates that the CHILDES model assigns more statements to the positive class than the Subtitles model. It is important to note that this result is likely dependent on the threshold that has been used here. Another difference is that precision and recall have more variation for the Subtitles model than the CHILDES model, whereas the rest of the metrics display little variation in both model, showing that performance is less stable when using the Subtitles model.

To summarize, this experiment relied on the analogy between human and artificial reasoning in the sense that we use input that is claimed to help children reason to fine-tune language models. Results show that this is rarely of any help, but in the case of reasoning about fantastical scenarios, fine-tuning on the OpenSubtitles dataset has shown to the ability improve performance over the course of fine-tuning. This can be explained by there being a significant amount of fantastical content in the OpenSubtitles data, or even the dialogue structure of this dataset.

5 Experiment 2: Imagination Prompting

Experiment 2 takes big inspiration from zero-shot chain of thought prompting, where the prompt specifically mentions to “think step by step”. We altered this method and instead prompt the model to “imagine” the scenario that it has to reason about. We investigated how this kind of prompting (imagination prompting) affects performance on various kinds of reasoning tasks. This experiment is motivated by early observations that children can benefit from a problem being framed in a make-believe setting or when specifically being asked to “imagine” it (Dias & Harris, 1988; Richards & Sanderson, 1999). This experiments transfers this technique to large language models, viewing them as a model reflecting a learning child. According to Wei, Wang, et al. (2022), chain of thought prompting can have no or even an adverse effect on models with less than 100B parameters (such as GPT-2) and only has substantial benefit for sufficiently large models (more than 100B parameters). To test this observation in our domain, the experiment was conducted using both GPT-2 and GPT-3, with them having 124M and 175B parameters respectively.

5.1 Method

5.1.1 Data

For this experiment, we again used the same task data sets and the same development and test splits. However, the preprocessing was different from what has been described in section 3.1.1. This time, the end goal of preprocessing was to let each data set have one column with the premise and a separate column with the hypothesis, and one column with the reverse of the hypothesis and columns for both the original and reversed case with a class label that is either 1 (true) or 0 (false).

In case of SNLI, the phrase “it follows that” has been prepended to the hypothesis. Since the premise and hypothesis are already separate, nothing else was done. For the logical deduction dataset, the process is similar. However, there were multiple hypotheses for each premise in this case, as can be seen in table **??**. Separate data points were made using each of these hypotheses, using the same strategy as has been used for SNLI. For the fantasy reasoning dataset, items were concatenated premises and hypotheses. The hypothesis is always the last sentence, so the phrase "It follows that" has been prepended to the last sentence of each item and the last sentence is separated as the hypothesis. The com2sense data is of a different nature, because items are not consistently structured as having a premise and hypothesis, but are statements of any structure that can be evaluated using our common sense. For this reason, the data has been preprocessed differently. The phrase "it is straightforward that" has been added to each item where this naturally fits. In order to keep the same structure for easy further processing of the dataset, a separate empty column representing the hypothesis has been added to this data set.

As a next step, reversed versions of all items were made again to control for noise due to random variation in the data. For each datapoint in the dataset, a negation has been added to the dataset by starting the hypothesis with "it does not follow that" instead of "it follows that". Reversed versions of items from the com2sense dataset have been created by including "it is not straightforward that" in the premises instead. All reversed statements were assigned to the label that is the inverse of the label belonging to the original, non-negated statement. In order to prepare for the prompting experiment, the clause "imagine the scenario" has been prepended to all premises and negated premises. This clause was chosen among a set of similar clauses (e.g. "Imagine this situation", "Imagine this", etc.) that have been experimented with to investigate differences in generation quality, but no noticable differences were found. Whatever comes after this clause was enclosed in quotes. Thus, in the end, all data sets consisted of five columns, of which two represent the premise and negated premise, one represents the hypothesis and the rest represents the class labels corresponding to the original and negated premise respectively.

5.1.2 Generating Elaborations

Now that data has been preprocessed, elaborations based on the premises of each data point could be generated. All premises were tokenized using the GPT2Tokenizer from the Huggingface library, to prepare for the generation of elaborations by GPT-2.

Elaborations of each premise have been generated by GPT-2, using top-p sampling with $p = 0.9$. This value has been selected out of values ranging from 0 to 1, using a step size of 0.1. The chosen value was found to display the least repetition, most naturally sounding sentences, but still displayed sufficient variation across runs. Three different variations of each item have been made, namely cutting off the elaboration after the first, second or third sentence, yielding three different datasets.

For GPT-3, a similar approach has been taken. The only difference is that no preset cut-off points have been used, since GPT-3 has it's own stopping criterion. This means that the model can stop generating when it has reached a "natural" end point, which was not the case for GPT-2.

Note that the com2sense dataset has no separate premises and hypotheses. Due to this different nature, elaborations have been generated based on the full statements and also for their negated counterpart.

After generating an elaboration of the premise, the hypothesis has been appended to the premise and elaboration, which was the final step in the construction of the data. Conclusions in com2sense were not separate, so in that case statement and elaboration are kept as is. Thus, in the end we had created a dataset with four columns for each reasoning task (leaving out hypothesis in the case of com2sense), namely the premise + elaboration + hypothesis, the corresponding negated version and the class label corresponding to that negated version.

5.1.3 Evaluation

Classification and evaluation has been done in the same way as described in section 3.1.2 and 3.1.3, but this time using the data that includes generated elaborations. For GPT-2, different combinations of label description and cut-off point have been evaluated using the development set, whereas for GPT-3 only different label descriptions have been evaluated. Again, AUC has been used to select the best configuration to use for evaluating performance on the reasoning task test sets.

5.2 Results

First, here are examples of data points including generated elaborations by GPT-2 (marked in bold) with a cut-off point of three for each dataset.

- **Fantasy**: Imagine the scenario 'they say you die twice. one time when you stop breathing and a second time, a bit later on, when somebody says your name for the last time. what they don't say is that in between those deaths, you get stuck in purgatory with all the great philosophers and authors - all just waiting to die.' **That is what the philosopher Jacques Baudrillard has discovered. The paradox that is the case with the modern philosophy is that the 'people who are going to die' are going to live forever. To find out what is going to happen to them depends on some fundamental question.** It follows that you will be stuck forever in purgatory if nobody says your name for the last time.

- **SNLI**: Imagine the scenario 'the brown and white dogs run through the field.' **That would be the story of a little-known group of men with the vision of having a dream world. If they were a race, they would be brown. If they were a race, they would be black.** It follows that there are dogs outdoors.

- **Com2sense**: Imagine the scenario 'because he can push a cart, it is straightforward that pushing a stroller should not be difficult.' **difficult.'**

In this case, you are trying to get a car on the track and you're trying to get a car on the track, so the wheel stops. You can do this with any other situation, and with a car with brakes and braking assist.

- **Logical deduction**: Imagine the scenario 'a fruit stand sells seven fruits: loquats, peaches, watermelons, plums, kiwis, mangoes, and pears. the pears are more expensive than the kiwis. the watermelons are less expensive than the peaches. the mangoes are the third-cheapest. the watermelons are the third-most expensive. the plums are the second-most expensive. the loquats are the second-cheapest.' **It should be said that this is not the same as any of the other 'chances' I have used in the past. the actual reason is that the more money that you can make without being caught, the less likely you are to fall prey to the bad guys. There's no need to worry too much about the money, because the amount that you will ever receive is simply the amount you will ever spend.** It follows that the peaches are the most expensive.

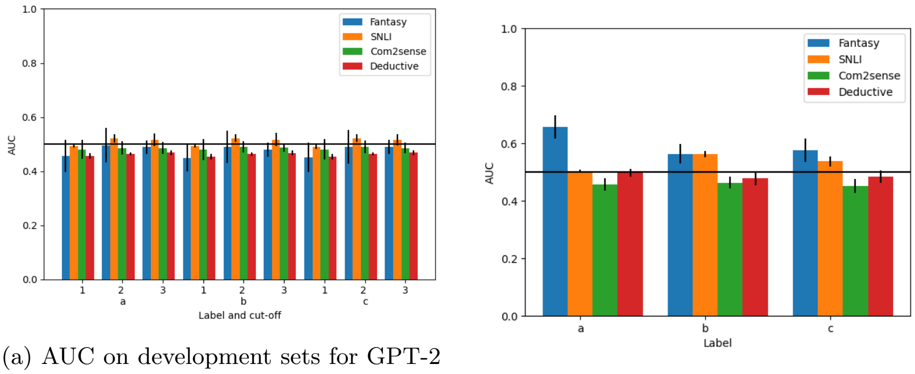
Below are the same examples, but with elaborations generated by GPT-3 instead.

- **Fantasy**: Imagine the scenario 'they say you die twice. one time when you stop breathing and a second time, a bit later on, when somebody says your name for the last time. what they don't say is that in between those deaths, you get stuck in purgatory with all the great philosophers and authors - all just waiting to die.' **In this scenario, the person is stuck in purgatory with all the great philosophers and authors. They are all just waiting to die.** It follows that you will be stuck forever in purgatory if nobody says your name for the last time.

- **SNLI**: Imagine the scenario 'the brown and white dogs run through the field.' **In this scenario, the brown and white dogs are running through a field. They are playful and happy, enjoying the freedom to run and explore. The grass is tall and the sun is shining. They are having a great time!** It follows that there are dogs outdoors.

- **Com2sense**: Imagine the scenario 'because he can push a cart, it is straightforward that pushing a stroller should not be difficult.' **A man is pushing a grocery cart through a grocery store. He sees a woman with a stroller trying to maneuver through the narrow aisles. The man thinks to himself, "because he can push a cart, it is straightforward that pushing a stroller should not be difficult." The man offers to help the woman with her stroller.**

- **Logical deduction**: Imagine the scenario 'a fruit stand sells seven fruits: loquats, peaches, watermelons, plums, kiwis, mangoes, and pears. the pears are more expensive than the kiwis. the watermelons are less expensive than the peaches. the mangoes are the third-cheapest. the watermelons are the third-most expensive. the plums are the second-most expensive. the loquats are the second-cheapest.’ **If someone were to ask for the price of the plums, you would tell them that the plums are the second-most expensive fruit at the stand.** It follows that the peaches are the most expensive.



(a) AUC on development sets for GPT-2 using different label descriptions and cut-off points

(b) AUC on development sets for GPT-3 using different label descriptions

Figure 9: AUC on development sets

Figure 9 shows mean AUC values over three different runs on the development sets for both GPT-2 and GPT-3 using different configurations. Different runs differ in their generations, since generation is not deterministic due to the use of top-p sampling. The black line indicates chance level AUC. Tables 24 and 25 in appendix B.3 give a complete overview of all performance metrics.

For GPT-2, AUC and other performance metrics for the development set are relatively consistent across different label descriptions and cut-off points for each dataset. GPT-2 has the highest AUC value for the SNLI dataset for all label descriptions and cut-off points. Accuracy on SNLI is also the highest among datasets (0.50-0.52). Still, measures are at chance level and do not show a major difference compared to other datasets. The fantasy dataset shows visibly more variation than other datasets in all metrics across runs, shown by higher standard deviations. No dataset except for the SNLI dataset ever exceeds chance AUC when using GPT-2. Nevertheless, recall sometimes exceeds 0.50 for datasets other than SNLI (recall ranges from 0.09 to 0.75).

Also important to note is that com2sense has a remarkably low recall compared to the other datasets (0.09-0.22 versus 0.59-0.75) and this effect persists across different configurations of label and cut-off point as shown in appendix B.3 table 24. This effect is only observed in the current experiment. This pattern is strengthened by an occurrence of significantly negative Pearson correlation (-0.04). For other datasets, recall is consistently higher than precision (0.59-0.75

versus 0.48-0.51). Accuracy is around chance level at all times (0.48-0.52).

When comparing these development results to the GPT-2 baseline development results, we see that precision and recall has increased for all datasets (0.42-0.51 precision 0.09-0.22 recall versus lots of 0.0 occurrences in the baseline). Moreover, accuracy remained similar (0.48-0.52), but AUC generally displays a slight drop compared to the baseline.

Figure 9b shows that in case of GPT-3, the fantasy dataset always exceeds chance level AUC and displays the highest AUC of all datasets. Also, the AUC score on fantasy for label description *a* is considerably higher than when other label descriptions were used. Similar patterns show up for accuracy, precision and recall, but the picture is more nuanced. Accuracy is highest in all cases (0.54-0.59), but precision is highest in most cases (0.58 for label description *a* and 0.62 for *b*), except when using label description *c* (0.56). Recall is only highest when using label description *a* (0.67). The success of label description *a* is also shown in a significantly positive Pearson correlation (0.27). These observations are different than what has been observed for GPT-2, where SNLI was scoring the highest. Next highest AUC is on SNLI, which now only exceeds chance AUC when label descriptions *b* and *c* are used. The other datasets do not exceed chance AUC and the com2sense dataset consistently shows the lowest AUC of all, similar to what we observed for GPT-2. This observation is accompanied by significantly negative correlations (-0.11 and -0.10).

All datasets, except for fantasy (0.54-0.59), display an accuracy around chance. Recall is always higher than precision for the com2sense dataset (0.57-0.75 recall versus 0.48-0.49 precision). For the rest of the datasets, recall (0.03-0.50) is usually lower than precision (0.50-0.58). The exception to this is the fantasy dataset combined with label description *a* (0.67 recall and 0.58 precision). When comparing these GPT-3 development results to those of the baseline, it becomes apparent that the pattern in precision versus recall is similar. Both AUC and accuracy have dropped or stayed similar compared to the baseline (0.48-0.59 accuracy now versus 0.50-0.61 for the baseline). Another important observation is that there is more variation in metrics across label descriptions now than there was in the baseline experiment.

When comparing GPT-2 and GPT-3 development results to each other, the most obvious differences relate to the fantasy dataset. GPT-3 displays higher AUC values than GPT-2. This difference is especially apparent when using label description *a*, where also a significantly positive Pearson correlation is observed (0.27). Moreover, precision is generally higher for the fantasy dataset when using GPT-3 (0.56-0.62) than when using GPT-2 (0.48-0.51), as well as accuracy (0.54-0.59 for GPT-3 versus 0.48-0.51 for GPT-2).

Lastly, recall shows a distinctive pattern for the SNLI, fantasy and logical deduction datasets. Recall is less in GPT-3, especially for label descriptions *b* and *c* (0.03-0.33 for GPT-3 versus 0.59-0.75 for GPT-2). This difference in recall between GPT-2 and GPT-3 is more pronounced for label descriptions *b* and *c* (0.03-0.33 for GPT-3 versus 0.59-0.75 for GPT-2) than for label description *a* (0.16-0.67 versus 0.65-0.75). On top of that, this contrast between label descriptions is also observed for com2sense, but recall values are higher in general than those of the other datasets (0.57-0.75 for com2sense versus 0.03-0.67 for the other datasets), which contrast the development results of GPT-2. This specific contrast in recall between label descriptions was also observed in the baseline experiment.

More importantly, the contrast in recall between GPT-2 and GPT-3 observed for most datasets was not observed for com2sense. Recall shows a steep increase for all label descriptions (0.57-0.75 for GPT-3 and 0.09-0.22 for GPT-2), accompanied by a slight increase in precision (0.48-0.49 for GPT-3 versus 0.42-0.48 for GPT-2). AUC for GPT-3 on com2sense even shows a slight decrease compared to GPT-2. The low AUC scores on com2sense are supported by significantly negative Pearson correlations (-0.10 and -0.11), which were also sometimes present when using GPT-2.

		COT		**Baseline**	
		F1	**AUC**	**F1**	**AUC**
SNLI	**Overall best**	(0.58, 0.01)	(0.50, 0.03)	0.13	0.54
	Task best	(0.58, 0.01)	(0.50, 0.02)	0.0	0.55
Fantasy	**Overall best**	(0.58, 0.05)	(0.47, 0.06)	0.0	0.31
	Task best	(0.58, 0.05)	(0.47, 0.06)	0.0	0.31
Com2sense	**Overall best**	(0.17, 0.03)	(0.53, 0.04)	0.0	0.54
	Task best	(0.25, 0.02)	(0.53, 0.04)	0.0	0.54
Deductive	**Overall best**	(0.55, 0.05)	(0.45, 0.04)	0.49	0.55
	Task best	(0.55, 0.04)	(0.45, 0.05)	0.49	0.55

Table 6: COT results on the test set for GPT-2. A threshold of 0.0 has been used to calculate precision and recall (resulting in above F1 scores). Results are reported as (x, y), with x indicating mean and y standard deviation over three runs.

		COT		**Baseline**	
		F1	**AUC**	**F1**	**AUC**
SNLI	**Overall best**	(0.58, 0.01)	(0.50, 0.02)	0.56	0.59
	Task best	(0.28, 0.03)	(0.56, 0.01)	0.55	0.61
Fantasy	**Overall best**	(0.55, 0.06)	(0.47, 0.07)	0.29	0.49
	Task best	(0.61, 0.04)	(0.66, 0.04)	0.70	0.66
Com2sense	**Overall best**	(0.57, 0.02)	(0.48, 0.02)	0.66	0.50
	Task best	(0.59, 0.01)	(0.46, 0.02)	0.55	0.46
Deductive	**Overall best**	(0.54, 0.05)	(0.45, 0.04)	0.18	0.45
	Task best	(0.12, 0.01)	(0.48, 0.02)	0.37	0.49

Table 7: COT results on the test set for GPT-3. A threshold of 0.0 has been used to calculate precision and recall (resulting in above F1 scores). Results are reported as (x, y), with x indicating mean and y standard deviation over three runs.

Based on these development AUC values, best configurations of label description (and cut-off point in case of GPT-2) have been selected for testing. Results of testing are reported in tables 6 and 7 and a complete overview of metrics has been given in appendix B.3 tables 26 and 27. When comparing test results for GPT-2 to the baseline test results, it can be seen that this experiment yields lower (or similar in case of com2sense) AUC values for all datasets except for the fantasy dataset, which has increased in terms of AUC. On top of that, precision and recall is not 0.0 anymore like it often was in the baseline experiment; they increased for most datasets (0.47-0.63 precision and 0.10-0.71 recall) except for the logical deduction dataset, where precision remained similar. Accuracy remained similar for all datasets compared to the baseline (0.49-0.52).

For GPT-3, AUC values on the test set are comparable to those of the baseline experiment for the fantasy, com2sense and logical deduction datasets. For the SNLI dataset AUC decreased compared to the baseline, but there is still an occurrence of a positive correlation that is significant (0.12). Although com2sense is similar in AUC to the baseline, significantly negative Pearson correlations are observed now (-0.11), which were not present in the baseline test results. The fantasy dataset displays significantly positive Pearson correlation now (0.27), as opposed to the baseline. Regarding precision and recall, there is a general increase compared to the baseline (0.47-0.64 precision and 0.07-0.75 recall now versus 0.36-0.56 precision and 0.12-0.97 recall for the baseline). However, in some cases there is a decline, namely recall for the overall best model on the com2sense datasets (0.70 versus 0.97 for the baseline) and lastly, recall for the task best model on the logical deduction dataset (0.07 now versus 0.46 for the baseline). Again, accuracy is similar to the baseline in all cases (0.46-0.52).

If we compare these test results to the development results, we see that they are mostly similar. This even holds for the occurrences of low recall that were seen during development, where com2sense and SNLI, fantasy and logical deduction displayed low recall compared to other datasets for GPT-2 and GPT-3 respectively. To be more precise, recall is low for the com2sense dataset in case of GPT-2 (0.10-0.17 versus 0.65-0.71 for other datasets) and low for the SNLI and logical deduction dataset in case of GPT-3 (0.07-0.68 versus 0.62-0.75). Precision is similar in most cases, with a few exceptions: it is higher on the com2sense dataset using the overall best model when using GPT-2 (0.63 for GPT-2 and 0.49 for GPT-3) and it is higher on the logical deduction dataset using the task best model when using GPT-3 (0.64 for GPT-3 versus 0.48 for GPT-2). It can also be seen that AUC of the task best model is generally higher when using GPT-3. Lastly, accuracy is similar between GPT-2 (0.46-0.52) and GPT-3 (0.46-0.58).

5.3 Discussion

Generations of GPT-3 are better than those of GPT-2. Generations by GPT-2 are somewhat related to the prompt, by having similar words as in the prompt. However, the continuations are not related to the prompt semantically and are often nonsensical. Continuations do not explain or elaborate on what is said in the prompt, which is the underlying concept of COT prompting. On the other hand, continuations generated by GPT-3 are much more sensical and actually elaborate on the scenario that is present in the prompt. Still, the content from the prompt is sometimes repeated in different words, instead of being elaborated on.

The consistency of performance across label descriptions and cut-off points for GPT-2 indicates that performance is not sensitive to exact label descriptions and cut-off points when using GPT-2. Possibly, the model is already performing poorly to a point where label descriptions are not able to make a difference and it is also possible that the length of continuations are not making a difference to performance, because they are nonsensical regardless of the cut-off point.

Moreover, GPT-2 tends to classify many statements as false when performing common sense reasoning compared to other tasks, which could be explained by the poor continuations. The persistently low recall of com2sense compared to the other datasets when using GPT-2 means that GPT-2 is very likely to assign a statement to the negative class when performing common sense reasoning, resulting in more false negatives. For other datasets recall is higher than precision, meaning that there are more false positives than false negatives. So, contrary to common sense reasoning, GPT-2 is more likely to assign an item to the positive class for other types of reasoning. This result could be due to the poor quality continuations having a larger effect on the com2sense dataset, leading GPT-2 to deem many statements to be unlikely to be correct reasoning in the com2sense dataset than in other datasets. This intuition is further confirmed when looking at performance of GPT-3 on common sense reasoning, which has improved recall considerably compared to GPT-2. This is likely due to the difference in continuations, which are not nonsensical anymore, eliminating this negative effect that was observed when using nonsensical continuations of GPT-2.

However, Pearson correlation being negative and AUC being similar for both GPT-2 and GPT-3 hints at a different picture. It shows that even though continuations of GPT-3 are better than those of GPT-2, this does not mean reasoning according to definition 2 is better for GPT-3.

Such a contrast between precision and recall was not seen in the baseline experiment, where precision and recall were often 0.0. This difference indicates that COT leads GPT-2 to make more nuanced predictions, instead of assigning all statements to the negative class. It indicates that reasoning according to definition 1 improved. In other words, it is more often the case that GPT-2 assigns a higher similarity to a label description of correct reasoning for a correct statement than for an incorrect statement. Nevertheless, the drop of AUC in most datasets does indicate that this does not necessarily mean that reasoning improved according to definition 2. It means that it is harder for the model to discriminate incorrect from correct reasoning, according to similarity to a label description. AUC when using GPT-2 increased for the fantasy dataset regarding the test set, but not for the development set. For this reason, it can not be said that there was an improvement. can be explained by the continuations generated by GPT-2 being nonsensical and semantically unrelated to the scenarios presented.

Reasoning according to definition 1 improved for GPT-3. Even though no apparent improvements in AUC were found, an overall improved precision and recall compared to the baseline for GPT-3 indicate that GPT-3 was able to reason better for different tasks according to definition 1 of reasoning. The fact that we see some improvement for GPT-3 but not for GPT-2, can be explained by the continuations of GPT-2 being nonsensical, while those of GPT-3 were understandable and relevant to the prompt. Again, label descriptions prove to be important factor in determining performance when using GPT-3, even more than in the baseline. This is shown by fluctuations in not only AUC, but also accuracy, precision and recall across label descriptions.

GPT-3 reasons better than GPT-2 in the context of fantastical scenarios when reasoning is defined as definition 2. It is apparent that GPT-3 shows a higher AUC compared to GPT-2 for the fantasy development dataset, and this boost is the highest when using label description a. This again shows the importance of choosing your label description carefully.

However, precision and recall tell a different story. The decrease in recall and increase in precision show that more statements are being assigned to the negative class (incorrect reasoning), resulting in more false negatives and less false positives. Since both precision and recall are equally important, we can conclude that GPT-3 is not better than GPT-2 in reasoning about fantastical scenarios according to definition 1. A similar observation can be made about logical deduction and SNLI, but for these datasets reasoning according to definition 2 was equally good for GPT-2 and GPT-3.

Com2sense shows a different pattern. Namely, the decrease in AUC shows that common sense reasoning under definition 2 is worse for GPT-3 than GPT-2, whereas the increase in precision and recall shows that common sense reasoning under definition 1 is better when using GPT-3 as opposed to GPT-2. This is similar to what was observed in the baseline experiment. This can be explained by the fact that continuations of GPT-3 did not contain new information and were merely a rewording of the content that was in the prompt. As a result, GPT-3 did not get any extra information to help improve performance on common sense reasoning.

To summarize, this experiment relied on research on human reasoning, where it was claimed that humans reason better when they are primed to imagine the scenario they have to reason about. Through our analogy between humans and machines, we apply this technique to models by performing zero-shot chain of thought prompting. Zero-shot imagination prompting seems to have no effect on reasoning performance in the sense of making correct classifications, but an overall increase in precision and recall for GPT-3 do indicate a slight improvement of zero-shot reasoning for GPT-3 when reasoning is defined as definition 1.

6 Experiment 3: Combining Fine-tuning and Imagination Prompting

The last experiment aimed to combine experiment 1 and 2. Using the model configurations found to be the best performing in experiment 1, we performed imagination prompting. We only use GPT-2 in this experiment due to resource constraints.

6.1 Method

For this experiment, the same task data sets have been used again, using the same preprocessing as described in section 5.1.1. The process for generating elaborations is slightly different from the explanation given in section 5.1.2. Again, all premises were tokenized using the GPT2Tokenizer from the Huggingface library, to prepare for the generation of elaborations by selected model checkpoints.

The model checkpoint that was found to be the best overall has been used to generate elaborations for all datasets. In addition, elaborations have been generated for each task data set using the model checkpoint that was found to be the best for that task specifically. This has been done with both CHILDES and Subtitles models. Like before, top-p sampling with $p = 0.9$ was used and using three different preset cut-off points for the elaborations. The rest of the process remained the same as the process described in section 5.1.2. This resulted in a dataset with four columns for each reasoning task, namely the premise + elaboration + hypothesis, corresponding class label, the corresponding negated version and the class label corresponding to that negated version.

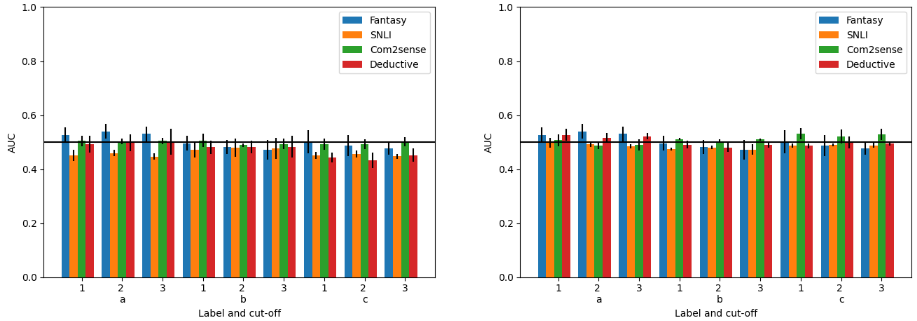
Evaluation has been done like stated before in section 5.1.3, using AUC to choose the best combinations of label description and cut-off point for the test set. Again, an overall best configuration was chosen based on AUC. On top of that, configurations that led to the highest AUC value for each reasoning task specifically were selected for evaluation on the test set of the corresponding reasoning task.

6.2 Results

Below are listed examples of the elaborations generated by the CHILDES models. As you can see, this specific example from com2sense did not yield a continuation. This issue of the models not generating text sometimes showed up in this experiment, but much much more for the Subtitles than for the CHILDES model. The Subtitles models rarely generated any continuations, so those are not reported. Since this means that results for the Subtitles model will be highly similar to those of experiment 1, the results are not reported here either. They can be found in appendix B.4 tables 32, 28 and 29.

- **Fantasy**: Imagine the scenario 'they say you die twice. one time when you stop breathing and a second time, a bit later on, when somebody says your name for the last time. what they don't say is that in between those deaths, you get stuck in purgatory with all the great philosophers and authors - all just waiting to die.' **this is what look theyve done.. www. oh. whats that then?** It does not follow that you will be stuck forever in purgatory if nobody says your name for the last time.

- **SNLI**: Imagine the scenario 'the brown and white dogs run through the field.' **other up. thats the right. lets look at the same color.** It follows that there are dogs outdoors.
- **Com2sense**: Imagine the scenario 'because he can push a cart, it is straightforward that pushing a stroller should not be difficult.'
- **Logical deduction**: Imagine the scenario 'a fruit stand sells seven fruits: loquats, peaches, watermelons, plums, kiwis, mangoes, and pears. the pears are more expensive than the kiwis. the watermelons are less expensive than the peaches. the mangoes are the third-cheapest. the watermelons are the third-most expensive. the plums are the second-most expensive. the loquats are the second-cheapest.' **the third. one. to three.** It does not follow that the peaches are the most expensive.



(a) AUC on development sets for CHILDES model using different label descriptions and cut-off points, using the overall best model checkpoint found in experiment 1

(b) AUC on development sets for CHILDES model using different label descriptions and cut-off points, using the task specific best model checkpoints found in experiment 1

Figure 10: AUC on development sets

		COT		Baseline	
		F1	**AUC**	**F1**	**AUC**
SNLI	**Overall best**	(0.30, 0.18)	(0.26, 0.27)	0.13	0.54
	Task best	(0.22, 0.05)	(0.51, 0.06)	0.0	0.55
Fantasy	**Overall best**	(0.36, 0.22)	(0.50, 0.06)	0.0	0.31
	Task best	(0.35, 0.25)	(0.54, 0.06)	0.0	0.31
Com2sense	**Overall best**	(0.54, 0.11)	(0.48, 0.05)	0.0	0.54
	Task best	(0.62, 0.02)	(0.50, 0.03)	0.0	0.54
Deductive	**Overall best**	(0.29, 0.14)	(0.50, 0.05)	0.49	0.55
	Task best	(0.29, 0.14)	(0.50, 0.05)	0.49	0.55

Table 8: COT results of the overall best CHILDES model checkpoint on the test set using the best configurations found during development. A threshold of 0.0 has been used to calculate precision and recall (resulting in above F1 scores). Results are reported as (x, y), with x indicating mean and y standard deviation over three runs.

		COT		Baseline	
		F1	**AUC**	**F1**	**AUC**
SNLI	**Overall best**	(0.44, 0.09)	(0.44, 0.04)	0.56	0.59
	Task best	(0.49, 0.07)	(0.46, 0.05)	0.55	0.61
Fantasy	**Overall best**	(0.29, 0.21)	(0.51, 0.03)	0.29	0.49
	Task best	(0.35, 0.25)	(0.54, 0.06)	0.70	0.66
Com2sense	**Overall best**	(0.58, 0.11)	(0.54, 0.03)	0.66	0.50
	Task best	(0.55, 0.11)	(0.51, 0.04)	0.55	0.46
Deductive	**Overall best**	(0.40, 0.28)	(0.52, 0.03)	0.18	0.45
	Task best	(0.34, 0.13)	(0.46, 0.02)	0.37	0.49

Table 9: COT results of the task specific best CHILDES model checkpoints on the test set using the best configurations found during development. A threshold of 0.0 has been used to calculate precision and recall (resulting in above F1 scores). Results are reported as (x, y), with x indicating mean and y standard deviation over three runs.

Figure 10 shows mean AUC on the development sets across three runs for CHILDES models using different configurations of label description and cut-off points. Figure 10a shows results using the overall best checkpoint that was found in experiment 1. Figure 10b shows results using the task specific best checkpoints that were found in experiment 1. Tables 8 and 9 show results on the test sets using the combinations of label description and cut-off point that were found to be best during development. This has been done for the overall and task specific checkpoint models. In appendix B.4, a more extensive overview of performance metrics can be found (tables 33, 30 and 31). Again, the black line represents chance performance.

In general, development results of using the CHILDES model show little to no variation in AUC, and chance level is often not exceeded. This holds for both overall best and task specific best checkpoints. Compared to the baseline, AUC increased only slightly for the fantasy dataset, but decreased for all other datasets. There was no increase in AUC compared to experiment 1 or 2. The same was observed for other performance metrics: precision and recall show similar values (0.28-0.59 precision and 0.16-0.63 recall) as in experiment 1. On top of that, precision and recall are generally lower than what is observed in the results of experiment 2. Also Pearson correlation is similar to both experiment 1 and 2, with values being very close to 0.0. Test results confirm this picture.

6.3 Discussion

Firstly, the generation examples show that the Subtitles model rarely generates an elaboration. The CHILDES model displays this behavior sometimes as well, but much less than the Subtitles model. This leads to the performance metrics being almost identical to those in experiment 1 in case of Subtitles. In addition, this explains the observation that cut-off point does not seem to cause variation in AUC, since generations are often not present.

Concerning CHILDES continuations, we see that they are very representative of child directed speech: sentences consisting of one word, nonsensical and repetitive.

Results again show that carefully choosing a label description to work with is crucial to get optimal performance. In this experiment, it became apparent that using CHILDES model does not help performance, even hurting it most of the time when using definition 2 of reasoning. Moreover, this strategy of combining experiment 1 and 2 yielded no advantage over either experiment 1 or 2 in terms of reasoning defined as definition 1.

The observation that using CHILDES models tends to hurt performance becomes clearer when looking at the continuations that were generated, which were always nonsensical and almost gibberish. It is likely that these elaborations distract the model from the content of the statement that matters for classification. This, together with the observation that the Subtitles model does not generate elaborations, shows that combining our imagination prompting with fine-tuning is not a fruitful approach the way it was done now. Possibly, fine-tuning was too much here, and results would be more promising when fine-tuning for with a smaller learning rate.

To summarize, this experiment aimed to investigate the effect of combining our two previous experiments. Unfortunately, results show that this tends to hurt zero-shot reasoning performance more than it helps, which can be explained by the nonsensical (or non-existent in case of the Subtitles models) continuations that the fine-tuned models generate. Possibly, future research could attempt to do less fine-tuning than was done here to improve these continuations.

7 Conclusion

This section brings together the results from all experiments and conclude with suggestions for future work. Results from the baseline experiment confirmed that both GPT-2 and GPT-3 struggle with zero-shot reasoning. It also became apparent that GPT-2 performs worse than GPT-3 for tasks involving fantastical scenarios or more general reasoning (SNLI), but performs better than GPT-3 on common sense reasoning and logical deduction tasks. However, both GPT-2 and GPT-3 do not exceed chance level performance for all types of reasoning tasks, showing that both still have a long way to go to reach human performance.

The first striking observation was already made in the baseline experiment, where results hinted at zero-shot reasoning performance does not scale with model size for all types of reasoning, especially when using definition 2 of reasoning. This is in conflict with previous research (Brown et al., 2020; Rae et al., 2021; Srivastava et al., 2022; Smith et al., 2022; Wei, Tay, et al., 2022). However, their research did not evaluate on the same task datasets that were used here. Thus, whether zero-shot learning is an emergent property seems highly dependent on the specific task that is being evaluated. Either way, further experiments are needed to get to the bottom of this observation.

The first attempt to make an analogy between human reasoning and artificial reasoning was through fine-tuning on linguistic data that children receive on a daily basis. Results of this experiments point to that this is only fruitful in a specific setting. Namely, fine-tuning on OpenSubtitles data while using the label description “This statement makes sense” for classification improves reasoning about fantastical scenarios. This can be explained by the substantial amounts of fantastical content in the OpenSubtitles data, that could guide the model to reason better about such situations. Horror or sci-fi movies and tv-shows are contained in the dataset, which are genres where fantastical scenarios occur in the stories. Possibly, there are also other characteristics of this data that help the model to reason better, such as its dialogue structure, but we believe the content to be the most plausible explanation. Actually, this finding can also be related to previous findings about imagination helping children to reason (Dias & Harris, 1988; Richards & Sanderson, 1999). The fact that OpenSubtitles contains fantastical contains possibly triggers the model into an imaginary state, where it is more inclined to imagine upon seeing a statement.

The second attempt to draw a parallel between human and artificial reasoning is to apply results from psychological research indicating that children reason better when being presented a problem in a make-believe setting or being instructed to imagine the scenario to zero-shot chain of thought prompting. This experiment showed that this strategy did not improve reasoning, and sometimes even hurts it. The expectation is that this is caused by continuations generated by the models not being informative enough to guide the model’s reasoning; elaborations of GPT-2 were often nonsensical and those of GPT-3 were often a rephrasing of the prompt. We believe that this approach would have been more succesfull if continuations were able make latent information from the prompt more explicit.

In the last experiment, we investigated how experiment 1 and 2 interact by performing zero-shot chain of thought prompting using fine-tuned models. These results show that performance is not improved in case of models fine-tuned on OpenSubtitles, due to elaborations not being generated by the models. In case of models fine-tuned on CHILDES, performance is even hurt, which can be attributed to elaborations generated being nonsensical and not anywhere close to regular human language, but more like gibberish of a child. Future research could continue upon this result, by replicating the experiment, but using a smaller learning rate to make the fine-tuning less steep and generations more understandable and useful. Possibly, the fine-tuning was too strong in the current work, leading to generations that resemble the fine-tuning data too closely.

Imagination prompting seems to have no effect on zero-shot reasoning performance under definition 1. However, the overall increase in precision and recall for GPT-3 compared to the baseline indicates an overall improvement across different types of reasoning when using GPT-3 under our second definition of reasoning. This would further strengthen the validity of our analogy between humans and models, because it was found that triggering children to use their imagination while reasoning affected their performance in a positive way. In any case, we believe performance could still be improved further, since we found that continuations of GPT-3 were mostly a rephrasing of the prompt. Therefore, future research to investigate strategies to let GPT-3 generate elaborations that uncover implicit information from the prompt instead of mere rephrasing could give the zero-shot reasoning performance a better boost.

For example, one could search for other prompting strategies than the “Imagine the scenario” prefix that has been attempted here. Even though the current initial search did not reveal any noticable differences in continuations, a search in a larger prompt space could still reveal prompts yielding continuations of a better quality. If this indeed proved to be fruitful, then additional experiments could be performed on the fantasy reasoning dataset using GPT-3 Subtitles models combined with imagination prompting to discover a possible additional performance boost on top of the one we found using solely fine-tuning. In general, fine-tuning GPT-3 is a direction for future research that remains to be fulfilled due to resource constraints.

It would also be fruitful to investigate a larger space of label descriptions, as results in this work have pointed out that differences in label descriptions can have a large impact on performance despite them being semantically similar. An important part in that would be research on efficient and possibly unsupervised ways of searching for optimal label descriptions for a given task.

Next to that, exploring outside this classification strategy is a possible way to expand on this work. Examples 3 showed that GPT-2 does not do well at reasoning using a question answering approach, which motivated our choice of exploring the current strategy. However, there are more strategies that could be explored. The first that comes to mind is a language modeling strategy, which is similar to a question answering strategy, but more controlled. Specifically, one could feed the model a reasoning task in the form of a question and evaluate whether “yes” or “no” has the highest probability of being the next word (the answer to the question). Possibly, our experiments yield different results when using such a strategy.

Generally, performance on the fantasy dataset is often among the highest across all experiments, even the baseline. This is in conflict with results from Dasgupta et al. (2022), who observe a struggle for reasoning about scenarios not in line with world knowledge in language models, similar to the struggle observed in humans. Thinking about the nature of this dataset and how it is different from the others leads to a plausible explanation. It is related to a distinction between a strong and weak notion of reasoning: strong reasoning is being able to produce intermediate steps in going from premises to hypothesis, whereas merely distinguishing good from bad reasoning is a weaker notion of reasoning. Statements he fantasy dataset contain many of the intermediate reasoning steps that are needed to go from premise to hypothesis. So, the model does not need to actively discover these steps, meaning that only the weaker notion of reasoning is being tested. On the other hand, the other datasets do not contain these intermediate steps. So in that case, the model must discover these steps in order to arrive at the correct hypothesis given a premise, meaning that a stronger notion of reasoning is being tested. This is a likely explanation of the consistently good performance of models on the fantasy dataset.

Falmagne (1990) claimed that higher forms of logic, logical deduction in particular, are acquired through linguistic input. Results from experiment 1 indicate that this claim does not extend to models, since fine-tuning on linguistic input that a child is likely to encounter did not result in a performance boost for the logical deduction dataset. However, positive results on the fantasy dataset show that fine-tuning can be helpful for other types of reasoning, in this case reasoning about fantastical scenarios.

There was a visible improvement in performance for the fantasy dataset when fine-tuning on OpenSubtitles data for label description b when using GPT-2. Additional cross validation has already been performed to confirm that this result was not due to characteristics specific to the used development and test split. Our discussion of weaker versus stronger notions of reasoning gives rise to the expectation that fine-tuning causes improvement regarding the weaker notion of reasoning. Nevertheless, this result is a promising start for making zero-shot generalization accessible for smaller models as well.

This result, being the most promising in this work, should be explored further. The first thing to do is to investigate if this results remains similar when fine-tuning GPT-3 and other larger models, including larger versions of GPT-2. Another important aspect of this result to investigate is the cause of improvement. This result could be due to fantastical content being present in the fine-tuning data that can serve as “world-knowledge” for fantastical scenario’s, but also due to the dialogue like structure being helpful for teaching the model to apply logical reasoning structures, which is an important aspect of this dataset. To explore these possibilities further, additional experiments need to be performed. Possibly, two datasets originating from OpenSubtitles can be constructed based on genres: one containing only fantastical tv-shows and movies and the other containing reality tv-shows and movies about real life. These datasets can be used to repeat experiment 1. If the dataset containing realistic content does not yield a performance increase for the fantasy dataset and the one containing fantastical scenarios does, it is a confirmation that the fantastical content is driving the improvement. If it is the other way around, it could be the structure of the fine-tuning data helping the model to learn about logical patterns that are important for solving the task. If both show an improvement, it is likely to be a combination of both explanations.

Our positive result on the fantasy dataset has shown that drawing a parallel between human reasoning and reasoning in machines is worth pursuing further to go in the direction of general human-like artificial intelligence. It has provided us with a more nuanced view on our first research question, namely that fine-tuning on human data can be helpful, but that this is very sensitive to the task and parameters. This finding also provides a promising outlook on being able to use smaller models for zero-shot reasoning. The conclusion on our second research question is that we need to investigate more generation strategies in order to fully explore the potential of imagination prompting. All in all, this research opens many doors for future research on zero-shot reasoning.

References

Betz, G., Richardson, K., & Voigt, C. (2021). Thinking aloud: Dynamic context generation improves zero-shot reasoning performance of gpt-2. *arXiv preprint arXiv:2103.13033*.

Bowman, S. R., Angeli, G., Potts, C., & Manning, C. D. (2015). A large annotated corpus for learning natural language inference. *arXiv preprint arXiv:1508.05326*.

Brown, T., Mann, B., Ryder, N., Subbiah, M., Kaplan, J. D., Dhariwal, P., ... others (2020). Language models are few-shot learners. *Advances in neural information processing systems*, *33*, 1877–1901.

Chan, S. C., Santoro, A., Lampinen, A. K., Wang, J. X., Singh, A. K., Richemond, P. H., ... Hill, F. (2022). Data distributional properties drive emergent in-context learning in transformers. In *Advances in neural information processing systems.*

Dasgupta, I., Lampinen, A. K., Chan, S. C., Creswell, A., Kumaran, D., McClelland, J. L., & Hill, F. (2022). Language models show human-like content effects on reasoning. *arXiv preprint arXiv:2207.07051*.

Devlin, J., Chang, M.-W., Lee, K., & Toutanova, K. (2018). Bert: Pre-training of deep bidirectional transformers for language understanding. *arXiv preprint arXiv:1810.04805*.

Dias, M. d. G., & Harris, P. L. (1988). The effect of make-believe play on deductive reasoning. *British journal of developmental psychology*, *6*(3), 207–221.

Falmagne, R. J. (1990). Language and the acquisition of logical knowledge. *Reasoning, necessity, and logic: Developmental perspectives*, 111–131.

Gazes, R. P., Hampton, R. R., & Lourenco, S. F. (2017). Transitive inference of social dominance by human infants. *Developmental science*, *20*(2), e12367.

Grice, H. P. (1975). Logic and conversation. In *Speech acts* (pp. 41–58). Brill.

Howard, J., & Ruder, S. (2018). Universal language model fine-tuning for text classification. *arXiv preprint arXiv:1801.06146*.

Keenan, T., Ruffman, T., & Olson, D. R. (1994). When do children begin to understand logical inference as a source of knowledge? *Cognitive development*, *9*(3), 331–353.

Kirsch, L., Harrison, J., Sohl-Dickstein, J., & Metz, L. (2022). General-purpose in-context learning by meta-learning transformers. *arXiv preprint arXiv:2212.04458*.

Kojima, T., Gu, S. S., Reid, M., Matsuo, Y., & Iwasawa, Y. (2022). Large language models are zero-shot reasoners. *arXiv preprint arXiv:2205.11916*.

Kunert, R., Fernández, R., & Zuidema, W. (2011). Adaptation in child directed speech: Evidence from corpora. *Proc. SemDial*, 112–119.

Lieven, E., Salomo, D., & Tomasello, M. (2009). Two-year-old children's production of multiword utterances: A usage-based analysis.

Lison, P., & Tiedemann, J. (2016). Opensubtitles2016: Extracting large parallel corpora from movie and tv subtitles.

Magister, L. C., Mallinson, J., Adamek, J., Malmi, E., & Severyn, A. (2022). Teaching small language models to reason. *arXiv preprint arXiv:2212.08410*.

Mikolov, T., Chen, K., Corrado, G., & Dean, J. (2013). Efficient estimation of word representations in vector space. *arXiv preprint arXiv:1301.3781*.

Min, S., Lyu, X., Holtzman, A., Artetxe, M., Lewis, M., Hajishirzi, H., & Zettlemoyer, L. (2022). Rethinking the role of demonstrations: What makes in-context learning work? *arXiv preprint arXiv:2202.12837*.

Palatucci, M., Pomerleau, D., Hinton, G. E., & Mitchell, T. M. (2009). Zero-shot learning with semantic output codes. *Advances in neural information processing systems*, *22*.

Piaget, J., & Inhelder, B. (2008). *The psychology of the child.* Basic books.

Radford, A., Narasimhan, K., Salimans, T., Sutskever, I., et al. (2018). Improving language understanding by generative pre-training.

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

Rae, J. W., Borgeaud, S., Cai, T., Millican, K., Hoffmann, J., Song, F., ... others (2021). Scaling language models: Methods, analysis & insights from training gopher. *arXiv preprint arXiv:2112.11446*.

Richards, C. A., & Sanderson, J. A. (1999). The role of imagination in facilitating deductive reasoning in 2-, 3-and 4-year-olds. *Cognition*, *72*(2), B1–B9.

Smith, S., Patwary, M., Norick, B., LeGresley, P., Rajbhandari, S., Casper, J., ... others (2022). Using deepspeed and megatron to train megatron-turing nlg 530b, a large-scale generative language model. *arXiv preprint arXiv:2201.11990*.

Srivastava, A., Rastogi, A., Rao, A., Shoeb, A. A. M., Abid, A., Fisch, A., ... others (2022). Beyond the imitation game: Quantifying and extrapolating the capabilities of language models. *arXiv preprint arXiv:2206.04615*.

Theakston, A. L., Lieven, E. V., Pine, J. M., & Rowland, C. F. (2001). The role of performance limitations in the acquisition of verb-argument structure: An alternative account. *Journal of child language*, *28*(1), 127–152.

Vaswani, A., Shazeer, N., Parmar, N., Uszkoreit, J., Jones, L., Gomez, A. N., ... Polosukhin, I. (2017). Attention is all you need. *Advances in neural information processing systems*, *30*.

Veeranna, S. P., Nam, J., Mencıa, E. L., & Fürnkranz, J. (2016). Using semantic similarity for multi-label zero-shot classification of text documents. In *Proceeding of european symposium on artificial neural networks, computational intelligence and machine learning. bruges, belgium: Elsevier* (pp. 423–428).

von Oswald, J., Niklasson, E., Randazzo, E., Sacramento, J., Mordvintsev, A., Zhmoginov, A., & Vladymyrov, M. (2022). Transformers learn in-context by gradient descent. *arXiv preprint arXiv:2212.07677*.

Wei, J., Bosma, M., Zhao, V. Y., Guu, K., Yu, A. W., Lester, B., ... Le, Q. V. (2021). Finetuned language models are zero-shot learners. *arXiv preprint arXiv:2109.01652*.

Wei, J., Tay, Y., Bommasani, R., Raffel, C., Zoph, B., Borgeaud, S., ... others (2022). Emergent abilities of large language models. *arXiv preprint arXiv:2206.07682*.

Wei, J., Wang, X., Schuurmans, D., Bosma, M., Chi, E., Le, Q., & Zhou, D. (2022). Chain of thought prompting elicits reasoning in large language models. *arXiv preprint arXiv:2201.11903*.

Xie, H., & Virtanen, T. (2021). Zero-shot audio classification via semantic embeddings. *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, *29*, 1233–1242.

Xie, S. M., Raghunathan, A., Liang, P., & Ma, T. (2021). An explanation of in-context learning as implicit bayesian inference. *arXiv preprint arXiv:2111.02080*.

Ye, S., Kim, D., Jang, J., Shin, J., & Seo, M. (2022). Guess the instruction! making language models stronger zero-shot learners. *arXiv preprint arXiv:2210.02969*.

Zhang, Z., Zhang, A., Li, M., & Smola, A. (2022). Automatic chain of thought prompting in large language models. *arXiv preprint arXiv:2210.03493*.

A Data

A.1 Baseline

Premise	**Hypothesis**	**label**
A soccer game with multiple males playing.	Some men are playing a sport.	Entailment
A man inspects the uniform of a figure in some East Asian country.	The man is sleeping.	Contradiction
An older and younger man smiling.	Two men are smiling and laughing at the cats playing on the floor.	Neutral

Table 10: Examples from the SNLI dataset

Item	**Label**
After millennia of waiting, an evil alien science corporation is brought down. The galactic government moves to remove an abandoned island laboratory that the company had put their lethal creatures on, only to find the island had been named Australia by a sentient species who now lives there. Animals appeared in Australia because they were created by God.	No
When you die, you appear in a cinema with a number of other people who look like you. You find out that they are your previous reincarnations, and soon you all begin watching your next life on the big screen. The people in the cinema are linked to you because they are you in your past lives.	Yes

Table 11: Examples from the fantasy dataset

Item	**Label**
A knife could be used in place of a screwdriver.	True
Since my last exam is today, I booked a flight home from school that departed yesterday.	False

Table 12: Examples from the com2sense dataset

Premise	**Hypotheses**	**Labels**
On a shelf, there are three books: a black book, an orange book, and a blue book. The blue book is to the right of the orange book. The orange book is to the right of the black book.	[The black book is the leftmost., The orange book is the leftmost., The blue book is the leftmost.]	[1, 0, 0]
In an antique car show, there are five vehicles: a station wagon, a tractor, a truck, a hatchback, and a minivan. The station wagon is newer than the tractor. The truck is older than the tractor. The minivan is newer than the hatchback. The hatchback is the second-newest.	[The station wagon is the newest., The tractor is the newest., The truck is the newest., The hatchback is the newest., The minivan is the newest.]	[0, 0, 0, 0, 1]
A fruit stand sells seven fruits: watermelons, oranges, mangoes, cantaloupes, kiwis, pears, and peaches. The pears are the second-cheapest. The peaches are more expensive than the cantaloupes. The peaches are less expensive than the mangoes. The cantaloupes are more expensive than the kiwis. The oranges are the fourth-most expensive. The watermelons are the second-most expensive.	[The watermelons are the second-cheapest, The oranges are the second-cheapest., The mangoes are the second-cheapest., The cantaloupes are the second-cheapest., The kiwis are the second-cheapest., The pears are the second-cheapest., The peaches are the second-cheapest.]	[0, 0, 0, 0, 0, 1, 0]

Table 13: Examples from the logical deduction dataset

Statement	Label	Negated statement	Label
A soccer game with multiple males playing. It follows that some men are playing a sport.	1	A soccer game with multiple males playing. It does not follow that some men are playing a sport.	0
A man inspects the uniform of a figure in some East Asian country. It follows that the man is sleeping.	1	A man inspects the uniform of a figure in some East Asian country. It does not follow that the man is sleeping.	0
An older and younger man smiling. It follows that two men are smiling and laughing at the cats playing on the floor.	0	An older and younger man smiling. It does not follow that two men are smiling and laughing at the cats playing on the floor.	1

Table 14: Examples from the transformed SNLI dataset

Statement	Label	Negated statement	Label
After millennia of waiting, an evil alien science corporation is brought down. The galactic government moves to remove an abandoned island laboratory that the company had put their lethal creatures on, only to find the island had been named Australia by a sentient species who now lives there. It follows that animals appeared in Australia because they were created by God.	0	After millennia of waiting, an evil alien science corporation is brought down. The galactic government moves to remove an abandoned island laboratory that the company had put their lethal creatures on, only to find the island had been named Australia by a sentient species who now lives there. It does not follow that animals appeared in Australia because they were created by God.	1
When you die, you appear in a cinema with a number of other people who look like you. You find out that they are your previous reincarnations, and soon you all begin watching your next life on the big screen. It follows that the people in the cinema are linked to you because they are you in your past lives.	1	When you die, you appear in a cinema with a number of other people who look like you. You find out that they are your previous reincarnations, and soon you all begin watching your next life on the big screen. It does not follow that the people in the cinema are linked to you because they are you in your past lives.	0

Table 15: Examples from the transformed fantasy dataset

Statement	Label	Negated statement	Label
It is straightforward that a knife could be used in place of a screwdriver.	1	It is not straightforward that a knife could be used in place of a screwdriver.	0
It is straightforward that since my last exam is today, I booked a flight home from school that departed yesterday.	0	It is not straightforward that since my last exam is today, I booked a flight home from school that departed yesterday.	1

Table 16: Examples from the transformed com2sense dataset

Statement	Label	Negated statement	Label
On a shelf, there are three books: a black book, an orange book, and a blue book. The blue book is to the right of the orange book. The orange book is to the right of the black book. It follows that the black book is the leftmost.	1	On a shelf, there are three books: a black book, an orange book, and a blue book. The blue book is to the right of the orange book. The orange book is to the right of the black book. It does not follow that the black book is the leftmost.	0
On a shelf, there are three books: a black book, an orange book, and a blue book. The blue book is to the right of the orange book. The orange book is to the right of the black book. It follows that the orange book is the leftmost.	0	On a shelf, there are three books: a black book, an orange book, and a blue book. The blue book is to the right of the orange book. The orange book is to the right of the black book. It does not follow that the orange book is the leftmost.	1
On a shelf, there are three books: a black book, an orange book, and a blue book. The blue book is to the right of the orange book. The orange book is to the right of the black book. It follows that the blue book is the leftmost.	0	On a shelf, there are three books: a black book, an orange book, and a blue book. The blue book is to the right of the orange book. The orange book is to the right of the black book. It does not follow that the blue book is the leftmost.	1

Table 17: Examples from the transformed logical deduction dataset

B Results

B.1 Baseline

		Acc	**Prec**	**Recall**	**F1**	**AUC**	**Corr**
a	**SNLI**	0.50	1.0	0.005	0.01	0.58	0.14
	Fantasy	0.50	0.0	0.0	0.0	0.49	-0.005
	Com2sense	0.50	0.0	0.0	0.0	0.53	0.04
	Deductive	0.50	0.0	0.0	0.0	0.47	-0.05
b	**SNLI**	0.51	0.8	0.04	0.07	0.58	**0.12**
	Fantasy	0.50	0.0	0.0	0.0	0.52	0.02
	Com2sense	0.50	0.0	0.0	0.0	0.53	0.07
	Deductive	0.47	0.46	0.37	0.41	0.47	-0.05
c	**SNLI**	0.50	1.0	0.005	0.01	0.58	**0.12**
	Fantasy	0.50	0.0	0.0	0.0	0.50	0.02
	Com2sense	0.50	0.0	0.0	0.0	0.52	0.05
	Deductive	0.50	0.50	0.01	0.02	0.47	-0.06

Table 18: Baseline results on the development set for GPT-2 for three different label descriptions. A threshold of 0.0 has been used to calculate accuracy, precision, recall and F1 score.

		Acc	**Prec**	**Recall**	**F1**	**AUC**	**Corr**
a	**SNLI**	0.53	0.52	0.82	0.64	0.58	**0.14**
	Fantasy	0.56	0.57	0.49	0.53	0.60	0.13
	Com2sense	0.50	0.50	0.57	0.53	0.50	-0.005
	Deductive	0.56	0.60	0.34	0.44	0.58	**0.13**
b	**SNLI**	0.61	0.62	0.59	0.60	0.65	**0.26**
	Fantasy	0.57	0.68	0.27	0.38	0.60	0.14
	Com2sense	0.51	0.50	0.98	0.67	0.48	-0.01
	Deductive	0.51	0.63	0.05	0.09	0.53	0.05
c	**SNLI**	0.59	0.57	0.69	0.63	0.64	**0.26**
	Fantasy	0.55	0.60	0.31	0.41	0.60	0.12
	Com2sense	0.50	0.50	0.97	0.66	0.48	-0.02
	Deductive	0.54	0.62	0.19	0.29	0.55	**0.10**

Table 19: Baseline results on the development set for GPT-3 for three different label descriptions. A threshold of 0.0 has been used to calculate accuracy, precision, recall and F1 score.

		Acc	**Prec**	**Recall**	**F1**	**AUC**	**Corr**
SNLI	**Overall best**	0.53	1.0	0.07	0.13	0.54	0.12
	Task best	0.50	0.0	0.0	0.0	0.55	0.08
Fantasy	**Overall best**	0.50	0.0	0.0	0.0	0.31	**-0.31**
	Task best	0.50	0.0	0.0	0.0	0.31	**-0.31**
Com2sense	**Overall best**	0.50	0.0	0.0	0.0	0.54	0.06
	Task best	0.50	0.0	0.0	0.0	0.54	0.06
Deductive	**Overall best**	0.53	0.54	0.46	0.49	0.55	0.08
	Task best	0.53	0.54	0.46	0.49	0.55	0.08

Table 20: Baseline results on the test set for GPT-2. A threshold of 0.0 has been used to calculate accuracy, precision, recall and F1 score.

		Acc	**Prec**	**Recall**	**F1**	**AUC**	**Corr**
SNLI	**Overall best**	0.52	0.51	0.61	0.56	0.59	**0.18**
	Task best	0.56	0.56	0.54	0.55	0.61	**0.19**
Fantasy	**Overall best**	0.41	0.36	0.24	0.29	0.49	-0.005
	Task best	0.67	0.64	0.76	0.70	0.66	0.27
Com2sense	**Overall best**	0.50	0.50	0.97	0.66	0.50	-0.002
	Task best	0.48	0.49	0.63	0.55	0.46	-0.08
Deductive	**Overall best**	0.46	0.37	0.12	0.18	0.45	-0.07
	Task best	0.49	0.48	0.30	0.37	0.49	-0.03

Table 21: Baseline results on the test set for GPT-3. A threshold of 0.0 has been used to calculate accuracy, precision, recall and F1 score.

B.2 Experiment 1

		Acc	**Prec**	**Recall**	**F1**	**AUC**	**Corr**
SNLI	**Overall best**	(0.50, 0.02)	(0.28, 0.21)	(0.30, 0.40)	(0.24, 0.29)	(0.52, 0.04)	(0.03, 0.07)
	Task best	(0.50, 0.01)	(0.33, 0.23)	(0.39, 0.43)	(0.31, 0.27)	(0.50, 0.01)	(-0.001, 0.03)
Fantasy	**Overall best**	(0.48, 0.05)	(0.48, 0.41)	(0.25, 0.33)	(0.21, 0.24)	(0.55, 0.09)	(0.09, 0.16)
	Task best	(0.50, 0.0)	(0.17, 0.24)	(0.33, 0.47)	(0.22, 0.31)	(0.56, 0.09)	(0.09, 0.16)
Com2sense	**Overall best**	(0.49, 0.01)	(0.33, 0.23)	(0.30, 0.38)	(0.24, 0.27)	(0.44, 0.05)	(-0.10, 0.07)**
	Task best	(0.50, 0.02)	(0.30, 0.22)	(0.06, 0.05)	(0.09, 0.08)	(0.51, 0.04)	(-0.003, 0.06)
Deductive	**Overall best**	(0.50, 0.01)	(0.28, 0.21)	(0.24, 0.33)	(0.21, 0.27)	(0.51, 0.02)	(0.03, 0.03)
	Task best	(0.50, 0.02)	(0.47, 0.08)	(0.44, 0.41)	(0.37, 0.24)	(0.50, 0.03)	(-0.02, 0.04)

Table 22: Results for the Subtitles models on the test sets. A threshold of 0.0 has been used to calculate accuracy, precision, recall and F1 score. The amount of * indicates amount of runs that the correlation was significant (p <0.05). Results are reported as (x, y), with x indicating mean and y standard deviation over three runs.

		Acc	**Prec**	**Recall**	**F1**	**AUC**	**Corr**
SNLI	**Overall best**	(0.53, 0.01)	(0.53, 0.01)	(0.60, 0.11)	(0.56, 0.04)	(0.54, 0.01)	(0.08, 0.05)
	Task best	(0.50, 0.03)	(0.50, 0.09)	(0.24, 0.11)	(0.32, 0.10)	(0.51, 0.05)	(0.03, 0.09)*
Fantasy	**Overall best**	(0.44, 0.02)	(0.41, 0.05)	(0.37, 0.21)	(0.36, 0.13)	(0.45, 0.02)	(-0.11, 0.02)
	Task best	(0.44, 0.02)	(0.41, 0.05)	(0.37, 0.21)	(0.36, 0.13)	(0.45, 0.02)	(-0.11, 0.02)
Com2sense	**Overall best**	(0.49, 0.03)	(0.52, 0.05)	(0.38, 0.16)	(0.41, 0.09)	(0.48, 0.03)	(-0.03, 0.07)
	Task best	(0.50, 0.02)	(0.49, 0.04)	(0.27, 0.08)	(0.35, 0.07)	(0.49, 0.03)	(0.03, 0.05)
Deductive	**Overall best**	(0.50, 0.04)	(0.50, 0.05)	(0.47, 0.18)	(0.47, 0.11)	(0.52, 0.04)	(0.02, 0.07)
	Task best	(0.49, 0.01)	(0.48, 0.02)	(0.38, 0.17)	(0.40, 0.13)	(0.52, 0.01)	(0.05, 0.01)

Table 23: Results for the CHILDES models on the test sets. A threshold of 0.0 has been used to calculate accuracy, precision, recall and F1 score. The amount of * indicates amount of runs that the correlation was significant (p <0.05). Results are reported as (x, y), with x indicating mean and y standard deviation over three runs.

B.3 Experiment 2

			Acc	Prec	Recall	F1	AUC	Corr
1	**a**	**SNLI**	(0.50, 0.01)	(0.50, 0.01)	(0.66, 0.02)	(0.57, 0.01)	(0.49, 0.01)	(0.02, 0.04)
		Fantasy	(0.49, 0.06)	(0.49, 0.04)	(0.67, 0.08)	(0.57, 0.06)	(0.46, 0.06)	(-0.1, 0.1)
		Com2sense	(0.49, 0.01)	(0.45, 0.05)	(0.09, 0.01)	(0.14, 0.01)	(0.48, 0.04)	(-0.04, 0.07)*
		Deductive	(0.48, 0.01)	(0.49, 0.01)	(0.68, 0.03)	(0.57, 0.02)	(0.46, 0.01)	(-0.03, 0.04)
	b	**SNLI**	(0.50, 0.01)	(0.50, 0.004)	(0.59, 0.02)	(0.54, 0.01)	(0.49, 0.01)	(0.02, 0.04)
		Fantasy	(0.48, 0.05)	(0.48, 0.04)	(0.65, 0.06)	(0.55, 0.05)	(0.45, 0.05)	(-0.10, 0.09)
		Com2sense	(0.48, 0.02)	(0.42, 0.08)	(0.12, 0.02)	(0.19, 0.03)	(0.48, 0.04)	(-0.04, 0.07)*
		Deductive	(0.48, 0.01)	(0.49, 0.01)	(0.67, 0.03)	(0.57, 0.02)	(0.45, 0.01)	(-0.03, 0.04)
	c	**SNLI**	(0.49, 0.01)	(0.49, 0.01)	(0.60, 0.02)	(0.54, 0.01)	(0.49, 0.01)	(0.02, 0.04)
		Fantasy	(0.48, 0.05)	(0.48, 0.04)	(0.65, 0.06)	(0.55, 0.05)	(0.45, 0.06)	(-0.10, 0.09)
		Com2sense	(0.49, 0.01)	(0.46, 0.05)	(0.11, 0.01)	(0.17, 0.02)	(0.48, 0.04)	(-0.04, 0.07)*
		Deductive	(0.48, 0.01)	(0.49, 0.01)	(0.68, 0.03)	(0.57, 0.02)	(0.45, 0.01)	(-0.03, 0.04)
2	**a**	**SNLI**	(0.51, 0.01)	(0.51, 0.01)	(0.72, 0.02)	(0.59, 0.01)	(0.52, 0.02)	(0.03, 0.04)
		Fantasy	(0.50, 0.04)	(0.50, 0.03)	(0.69, 0.04)	(0.58, 0.03)	(0.50, 0.06)	(-0.09, 0.07)
		Com2sense	(0.49, 0.01)	(0.44, 0.05)	(0.10, 0.02)	(0.16, 0.02)	(0.49, 0.02)	(-0.04, 0.04)
		Deductive	(0.48, 0.01)	(0.49, 0.01)	(0.67, 0.03)	(0.56, 0.01)	(0.46, 0.01)	(-0.02, 0.06)
	b	**SNLI**	(0.50, 0.004)	(0.50, 0.003)	(0.67, 0.01)	(0.57, 0.01)	(0.52, 0.01)	(0.03, 0.04)
		Fantasy	(0.50, 0.02)	(0.50, 0.01)	(0.66, 0.03)	(0.57, 0.02)	(0.49, 0.06)	(-0.10, 0.06)
		Com2sense	(0.49, 0.01)	(0.47, 0.04)	(0.18, 0.01)	(0.25, 0.01)	(0.49, 0.02)	(-0.03, 0.02)
		Deductive	(0.48, 0.01)	(0.49, 0.01)	(0.67, 0.03)	(0.56, 0.01)	(0.46, 0.01)	(-0.02, 0.06)
	c	**SNLI**	(0.52, 0.003)	(0.51, 0.002)	(0.71, 0.01)	(0.59, 0.004)	(0.52, 0.01)	(0.03, 0.04)
		Fantasy	(0.50, 0.03)	(0.50, 0.02)	(0.69, 0.02)	(0.58, 0.02)	(0.49, 0.06)	(-0.09, 0.07)
		Com2sense	(0.49, 0.01)	(0.46, 0.05)	(0.12, 0.01)	(0.19, 0.02)	(0.49, 0.02)	(-0.04, 0.03)
		Deductive	(0.48, 0.01)	(0.49, 0.01)	(0.68, 0.03)	(0.57, 0.01)	(0.46, 0.01)	(-0.02, 0.06)
3	**a**	**SNLI**	(0.51, 0.01)	(0.51, 0.004)	(0.75, 0.02)	(0.61, 0.01)	(0.52, 0.02)	(0.01, 0.04)
		Fantasy	(0.50, 0.02)	(0.50, 0.02)	(0.70, 0.02)	(0.58, 0.004)	(0.49, 0.03)	(-0.05, 0.04)
		Com2sense	(0.49, 0.02)	(0.46, 0.07)	(0.13, 0.02)	(0.21, 0.03)	(0.49, 0.02)	(-0.04, 0.03)
		Deductive	(0.48, 0.01)	(0.48, 0.01)	(0.65, 0.03)	(0.55, 0.01)	(0.47, 0.01)	(-0.02, 0.06)
	b	**SNLI**	(0.51, 0.004)	(0.51, 0.003)	(0.71, 0.02)	(0.59, 0.004)	(0.52, 0.02)	(0.01, 0.04)
		Fantasy	(0.50, 0.02)	(0.50, 0.01)	(0.69, 0.03)	(0.58, 0.01)	(0.48, 0.03)	(-0.06, 0.04)
		Com2sense	(0.48, 0.01)	(0.47, 0.03)	(0.22, 0.01)	(0.30, 0.02)	(0.49, 0.02)	(-0.03, 0.03)
		Deductive	(0.48, 0.01)	(0.49, 0.01)	(0.65, 0.03)	(0.56, 0.01)	(0.47, 0.01)	(-0.02, 0.06)
	c	**SNLI**	(0.51, 0.003)	(0.51, 0.002)	(0.75, 0.02)	(0.61, 0.01)	(0.52, 0.02)	(0.01, 0.04)
		Fantasy	(0.51, 0.01)	(0.51, 0.01)	(0.73, 0.03)	(0.60, 0.01)	(0.49, 0.03)	(-0.05, 0.04)
		Com2sense	(0.49, 0.01)	(0.48, 0.04)	(0.16, 0.02)	(0.24, 0.02)	(0.49, 0.02)	(-0.04, 0.03)
		Deductive	(0.48, 0.01)	(0.49, 0.01)	(0.66, 0.03)	(0.56, 0.01)	(0.47, 0.01)	(-0.02, 0.06)

Table 24: COT results of GPT-2 on the development set for different configurations. A threshold of 0.0 has been used to calculate accuracy, precision, recall and F1 score. The amount of * indicates amount of runs that the correlation was significant (p <0.05). Results are reported as (x, y), with x indicating mean and y standard deviation over three runs.

		Acc	Prec	Recall	F1	AUC	Corr
a	**SNLI**	(0.52, 0.003)	(0.52, 0.003)	(0.50, 0.01)	(0.51, 0.004)	(0.50, 0.01)	(-0.001, 0.01)
	Fantasy	(0.59, 0.03)	(0.58, 0.03)	(0.67, 0.04)	(0.62, 0.06)	(0.66, 0.04)	(0.27, 0.06)*
	Com2sense	(0.48, 0.02)	(0.48, 0.02)	(0.57, 0.02)	(0.52, 0.03)	(0.46, 0.02)	(-0.11, 0.06)**
	Deductive	(0.52, 0.02)	(0.58, 0.08)	(0.16, 0.03)	(0.25, 0.04)	(0.50, 0.01)	(-0.003, 0.01)
b	**SNLI**	(0.51, 0.02)	(0.53, 0.06)	(0.17, 0.01)	(0.26, 0.02)	(0.56, 0.01)	(0.12, 0.03)*
	Fantasy	(0.56, 0.04)	(0.62, 0.06)	(0.32, 0.08)	(0.42, 0.08)	(0.56, 0.03)	(0.14, 0.05)
	Com2sense	(0.49, 0.01)	(0.49, 0.01)	(0.75, 0.04)	(0.59, 0.01)	(0.46, 0.02)	(-0.09, 0.06)
	Deductive	(0.51, 0.01)	(0.58, 0.18)	(0.03, 0.01)	(0.06, 0.03)	(0.48, 0.03)	(-0.03, 0.03)
c	**SNLI**	(0.50, 0.01)	(0.50, 0.01)	(0.23, 0.01)	(0.31, 0.01)	(0.54, 0.02)	(0.09, 0.04)*
	Fantasy	(0.54, 0.04)	(0.56, 0.05)	(0.33, 0.10)	(0.41, 0.09)	(0.57, 0.04)	(0.14, 0.06)
	Com2sense	(0.49, 0.04)	(0.49, 0.02)	(0.74, 0.01)	(0.59, 0.02)	(0.45, 0.02)	(-0.10, 0.07)*
	Deductive	(0.51, 0.01)	(0.60, 0.09)	(0.07, 0.01)	(0.12, 0.01)	(0.48, 0.02)	(-0.03, 0.03)

Table 25: COT results of GPT-3 on the development set for different label description configurations. A threshold of 0.0 has been used to calculate accuracy, precision, recall and F1 score. The amount of * indicates amount of runs that the correlation was significant ($p < 0.05$). Results are reported as (x, y), with x indicating mean and y standard deviation over three runs.

		Acc	Prec	Recall	F1	AUC	Corr
SNLI	**Overall best**	(0.49, 0.02)	(0.49, 0.01)	(0.71, 0.01)	(0.58, 0.01)	(0.50, 0.03)	(-0.03, 0.04)
	Task best	(0.49, 0.01)	(0.49, 0.01)	(0.69, 0.01)	(0.58, 0.01)	(0.50, 0.02)	(-0.03, 0.04)
Fantasy	**Overall best**	(0.52, 0.04)	(0.52, 0.03)	(0.65, 0.10)	(0.58, 0.05)	(0.47, 0.06)	(0.08, 0.14)
	Task best	(0.52, 0.04)	(0.52, 0.03)	(0.65, 0.10)	(0.58, 0.05)	(0.47, 0.06)	(0.08, 0.14)
Com2sense	**Overall best**	(0.52, 0.01)	(0.63, 0.08)	(0.10, 0.02)	(0.17, 0.03)	(0.53, 0.04)	(0.06, 0.05)
	Task best	(0.51, 0.02)	(0.54, 0.06)	(0.17, 0.02)	(0.25, 0.02)	(0.53, 0.04)	(0.06, 0.06)
Deductive	**Overall best**	(0.46, 0.05)	(0.47, 0.04)	(0.65, 0.06)	(0.55, 0.05)	(0.45, 0.04)	(0.04, 0.03)
	Task best	(0.47, 0.04)	(0.48, 0.03)	(0.65, 0.06)	(0.55, 0.04)	(0.45, 0.05)	(0.04, 0.04)

Table 26: COT results on the test set for GPT-2. A threshold of 0.0 has been used to calculate accuracy, precision, recall and F1 score. The amount of * indicates amount of runs that the correlation was significant ($p < 0.05$). Results are reported as (x, y), with x indicating mean and y standard deviation over three runs.

		Acc	Prec	Recall	F1	AUC	Corr
SNLI	**Overall best**	(0.50, 0.02)	(0.50, 0.01)	(0.68, 0.01)	(0.58, 0.01)	(0.50, 0.02)	(-0.003, 0.01)
	Task best	(0.52, 0.02)	(0.55, 0.07)	(0.19, 0.02)	(0.28, 0.03)	(0.56, 0.01)	(0.12, 0.03)*
Fantasy	**Overall best**	(0.51, 0.04)	(0.50, 0.04)	(0.62, 0.12)	(0.55, 0.06)	(0.47, 0.07)	(0.27, 0.07)*
	Task best	(0.58, 0.04)	(0.57, 0.04)	(0.65, 0.06)	(0.61, 0.04)	(0.66, 0.04)	(0.27, 0.07)*
Com2sense	**Overall best**	(0.48, 0.03)	(0.49, 0.02)	(0.70, 0.02)	(0.57, 0.02)	(0.48, 0.02)	(-0.11, 0.06)**
	Task best	(0.49, 0.02)	(0.49, 0.01)	(0.75, 0.04)	(0.59, 0.01)	(0.46, 0.02)	(-0.09, 0.06)
Deductive	**Overall best**	(0.46, 0.05)	(0.47, 0.04)	(0.64, 0.06)	(0.54, 0.05)	(0.45, 0.04)	(0.0003, 0.01)
	Task best	(0.51, 0.01)	(0.64, 0.14)	(0.07, 0.01)	(0.12, 0.01)	(0.48, 0.02)	(-0.03, 0.03)

Table 27: COT results on the test set for GPT-3. A threshold of 0.0 has been used to calculate accuracy, precision, recall and F1 score. The amount of * indicates amount of runs that the correlation was significant ($p < 0.05$). Results are reported as (x, y), with x indicating mean and y standard deviation over three runs.

B.4 Experiment 3

		Acc	**Prec**	**Recall**	**F1**	**AUC**	**Corr**
SNLI	**Overall best**	(0.50, 0.01)	(0.45, 0.04)	(0.31, 0.35)	(0.28, 0.25)	(0.52, 0.04)	(0.04, 0.04)
	Task best	(0.50, 0.01)	(0.50, 0.06)	(0.29, 0.31)	(0.28, 0.23)	(0.52, 0.03)	(0.04, 0.02)
Fantasy	**Overall best**	(0.53, 0.08)	(0.41, 0.31)	(0.35, 0.26)	(0.36, 0.25)	(0.56, 0.04)	(0.08, 0.07)
	Task best	(0.51, 0.01)	(0.34, 0.24)	(0.32, 0.24)	(0.33, 0.24)	(0.56, 0.02)	(0.13, 0.05)
Com2sense	**Overall best**	(0.49, 0.01)	(0.50, 0.01)	(0.68, 0.26)	(0.55, 0.11)	(0.47, 0.03)	(-0.06, 0.06)
	Task best	(0.52, 0.01)	(0.51, 0.003)	(0.82, 0.10)	(0.63, 0.03)	(0.51, 0.03)	(0.03, 0.02)
Deductive	**Overall best**	(0.51, 0.01)	(0.34, 0.24)	(0.31, 0.24)	(0.32, 0.23)	(0.50, 0.03)	(0.002, 0.05)
	Task best	(0.51, 0.02)	(0.52, 0.04)	(0.43, 0.28)	(0.43, 0.15)	(0.50, 0.03)	(0.003, 0.06)

Table 28: COT results of the overall best Subtitles model checkpoint on the test set using the best configurations found during development. A threshold of 0.0 has been used to calculate accuracy, precision, recall and F1 score. The amount of * indicates amount of runs that the correlation was significant (p <0.05). Results are reported as (x, y), with x indicating mean and y standard deviation over three runs.

		Acc	**Prec**	**Recall**	**F1**	**AUC**	**Corr**
SNLI	**Overall best**	(0.49, 0.02)	(0.65, 0.25)	(0.33, 0.29)	(0.31, 0.21)	(0.47, 0.01)	(-0.03, 0.01)
	Task best	(0.20, 0.16)	(0.22, 0.18)	(0.29, 0.24)	(0.25, 0.20)	(0.14, 0.15)	(-0.61, 0.33)***
Fantasy	**Overall best**	(0.54, 0.06)	(0.56, 0.08)	(0.54, 0.35)	(0.48, 0.19)	(0.53, 0.05)	(0.10, 0.11)
	Task best	(0.51, 0.05)	(0.35, 0.26)	(0.30, 0.21)	(0.32, 0.23)	(0.54, 0.05)	(0.12, 0.08)
Com2sense	**Overall best**	(0.48, 0.01)	(0.46, 0.05)	(0.37, 0.37)	(0.32, 0.22)	(0.45, 0.04)	(-0.09, 0.09)
	Task best	(0.50, 0.01)	(0.50, 0.003)	(0.89, 0.03)	(0.64, 0.01)	(0.49, 0.003)	(0.01, 0.01)
Deductive	**Overall best**	(0.51, 0.01)	(0.35, 0.25)	(0.37, 0.44)	(0.29, 0.28)	(0.49, 0.02)	(-0.03, 0.03)
	Task best	(0.51, 0.01)	(0.35, 0.25)	(0.37, 0.44)	(0.29, 0.28)	(0.49, 0.02)	(-0.03, 0.03)

Table 29: COT results of the task specific best Subtitles model checkpoints on the test set using the best configurations found during development. A threshold of 0.0 has been used to calculate accuracy, precision, recall and F1 score. The amount of * indicates amount of runs that the correlation was significant (p <0.05). Results are reported as (x, y), with x indicating mean and y standard deviation over three runs.

		Acc	Prec	Recall	F1	AUC	Corr
SNLI	**Overall best**	(0.27, 0.19)	(0.29, 0.18)	(0.30, 0.18)	(0.30, 0.18)	(0.26, 0.27)	(-0.39, 0.50)***
	Task best	(0.49, 0.01)	(0.46, 0.02)	(0.15, 0.05)	(0.22, 0.05)	(0.51, 0.06)	(0.02, 0.11)*
Fantasy	**Overall best**	(0.50, 0.06)	(0.49, 0.06)	(0.37, 0.27)	(0.36, 0.22)	(0.50, 0.06)	(-0.02, 0.11)
	Task best	(0.49, 0.02)	(0.33, 0.24)	(0.37, 0.28)	(0.35, 0.25)	(0.54, 0.06)	(0.02, 0.07)
Com2sense	**Overall best**	(0.51, 0.02)	(0.51, 0.02)	(0.63, 0.25)	(0.54, 0.11)	(0.48, 0.05)	(-0.04, 0.09)
	Task best	(0.51, 0.03)	(0.51, 0.02)	(0.80, 0.02)	(0.62, 0.02)	(0.50, 0.03)	(-0.01, 0.06)
Deductive	**Overall best**	(0.50, 0.03)	(0.53, 0.06)	(0.24, 0.16)	(0.29, 0.14)	(0.50, 0.05)	(-0.02, 0.10)*
	Task best	(0.50, 0.03)	(0.53, 0.06)	(0.24, 0.16)	(0.29, 0.14)	(0.50, 0.05)	(-0.02, 0.10)*

Table 30: COT results of the overall best CHILDES model checkpoint on the test set using the best configurations found during development. A threshold of 0.0 has been used to calculate accuracy, precision, recall and F1 score. The amount of * indicates amount of runs that the correlation was significant (p <0.05). Results are reported as (x, y), with x indicating mean and y standard deviation over three runs.

		Acc	Prec	Recall	F1	AUC	Corr
SNLI	**Overall best**	(0.45, 0.04)	(0.44, 0.05)	(0.45, 0.13)	(0.44, 0.09)	(0.44, 0.04)	(-0.11, 0.07)*
	Task best	(0.49, 0.03)	(0.49, 0.03)	(0.50, 0.12)	(0.49, 0.07)	(0.46, 0.05)	(-0.08, 0.09)*
Fantasy	**Overall best**	(0.52, 0.02)	(0.37, 0.26)	(0.24, 0.17)	(0.29, 0.21)	(0.51, 0.03)	(0.02, 0.03)
	Task best	(0.49, 0.02)	(0.33, 0.24)	(0.37, 0.28)	(0.35, 0.25)	(0.54, 0.06)	(0.02, 0.07)
Com2sense	**Overall best**	(0.53, 0.01)	(0.53, 0.02)	(0.72, 0.26)	(0.58, 0.11)	(0.54, 0.03)	(0.05, 0.05)
	Task best	(0.49, 0.03)	(0.49, 0.02)	(0.69, 0.25)	(0.55, 0.11)	(0.51, 0.04)	(0.02, 0.06)
Deductive	**Overall best**	(0.51, 0.01)	(0.34, 0.24)	(0.48, 0.34)	(0.40, 0.28)	(0.52, 0.03)	(0.05, 0.05)
	Task best	(0.48, 0.003)	(0.45, 0.02)	(0.31, 0.17)	(0.34, 0.13)	(0.46, 0.02)	(-0.08, 0.03)

Table 31: COT results of the task specific best CHILDES model checkpoints on the test set using the best configurations found during development. A threshold of 0.0 has been used to calculate accuracy, precision, recall and F1 score. The amount of * indicates amount of runs that the correlation was significant (p <0.05). Results are reported as (x, y), with x indicating mean and y standard deviation over three runs.

				Acc	Prec	Recall	F1	AUC	Corr
1	**a**	**SNLI**	**Overall best**	(0.47, 0.03)	(0.47, 0.03)	(0.43, 0.31)	(0.38, 0.19)	(0.46, 0.03)	(-0.07, 0.04)*
			Task best	(0.50, 0.01)	(0.34, 0.24)	(0.13, 0.10)	(0.18, 0.14)	(0.52, 0.02)	(0.04, 0.03)
		Fantasy	**Overall best**	(0.47, 0.02)	(0.33, 0.23)	(0.45, 0.34)	(0.37, 0.27)	(0.46, 0.01)	(-0.09, 0.02)
			Task best	(0.48, 0.01)	(0.40, 0.11)	(0.43, 0.32)	(0.37, 0.24)	(0.46, 0.01)	(-0.09, 0.01)
		Com2sense	**Overall best**	(0.51, 0.01)	(0.51, 0.01)	(0.35, 0.24)	(0.35, 0.22)	(0.49, 0.01)	(-0.03, 0.01)
			Task best	(0.50, 0.003)	(0.50, 0.002)	(0.58, 0.41)	(0.43, 0.29)	(0.49, 0.02)	(-0.03, 0.03)
		Deductive	**Overall best**	(0.51, 0.02)	(0.42, 0.16)	(0.28, 0.21)	(0.31, 0.22)	(0.52, 0.02)	(0.04, 0.03)
			Task best	(0.50, 0.003)	(0.34, 0.24)	(0.41, 0.43)	(0.33, 0.27)	(0.52, 0.03)	(0.03, 0.05)*
	b	**SNLI**	**Overall best**	(0.50, 0.04)	(0.47, 0.07)	(0.25, 0.18)	(0.29, 0.19)	(0.49, 0.05)	(-0.01, 0.07)
			Task best	(0.49, 0.03)	(0.50, 0.02)	(0.66, 0.08)	(0.56, 0.02)	(0.49, 0.04)	(-0.01, 0.07)
		Fantasy	**Overall best**	(0.51, 0.02)	(0.36, 0.26)	(0.27, 0.22)	(0.30, 0.22)	(0.55, 0.06)	(0.07, 0.08)
			Task best	(0.49, 0.02)	(0.34, 0.24)	(0.20, 0.20)	(0.23, 0.19)	(0.54, 0.05)	(0.07, 0.09)
		Com2sense	**Overall best**	(0.51, 0.01)	(0.51, 0.01)	(0.42, 0.34)	(0.37, 0.26)	(0.49, 0.03)	(-0.001, 0.02)
			Task best	(0.50, 0.004)	(0.45, 0.08)	(0.35, 0.45)	(0.27, 0.29)	(0.49, 0.001)	(-0.02, 0.02)
		Deductive	**Overall best**	(0.49, 0.01)	(0.32, 0.23)	(0.26, 0.21)	(0.28, 0.21)	(0.50, 0.02)	(-0.02, 0.03)
			Task best	(0.50, 0.01)	(0.50, 0.003)	(0.92, 0.08)	(0.65, 0.02)	(0.52, 0.02)	(0.05, 0.04)*
	c	**SNLI**	**Overall best**	(0.50, 0.003)	(0.67, 0.24)	(0.05, 0.03)	(0.08, 0.05)	(0.46, 0.04)	(-0.06, 0.06)*
			Task best	(0.50, 0.004)	(0.17, 0.23)	(0.32, 0.46)	(0.22, 0.31)	(0.47, 0.06)	(-0.04, 0.08)*
		Fantasy	**Overall best**	(0.52, 0.02)	(0.42, 0.30)	(0.11, 0.13)	(0.15, 0.17)	(0.48, 0.01)	(-0.06, 0.03)
			Task best	(0.51, 0.02)	(0.35, 0.25)	(0.10, 0.13)	(0.14, 0.17)	(0.48, 0.01)	(-0.05, 0.03)
		Com2sense	**Overall best**	(0.50, 0.01)	(0.33, 0.23)	(0.34, 0.26)	(0.33, 0.24)	(0.50, 0.03)	(-0.01, 0.04)
			Task best	(0.50, 0.002)	(0.32, 0.23)	(0.08, 0.08)	(0.12, 0.11)	(0.49, 0.01)	(-0.002, 0.01)
		Deductive	**Overall best**	(0.48, 0.02)	(0.29, 0.20)	(0.09, 0.07)	(0.13, 0.10)	(0.45, 0.02)	(-0.08, 0.03)*
			Task best	(0.50, 0.02)	(0.67, 0.23)	(0.55, 0.39)	(0.42, 0.29)	(0.53, 0.02)	(0.05, 0.04)
2	**a**	**SNLI**	**Overall best**	(0.47, 0.03)	(0.47, 0.03)	(0.43, 0.31)	(0.38, 0.19)	(0.46, 0.03)	(-0.07, 0.04)*
			Task best	(0.50, 0.01)	(0.35, 0.24)	(0.13, 0.10)	(0.19, 0.14)	(0.52, 0.02)	(0.04, 0.03)
		Fantasy	**Overall best**	(0.48, 0.02)	(0.33, 0.23)	(0.45, 0.34)	(0.37, 0.27)	(0.46, 0.01)	(-0.08, 0.02)
			Task best	(0.48, 0.01)	(0.40, 0.11)	(0.43, 0.32)	(0.37, 0.24)	(0.46, 0.01)	(-0.08, 0.01)
		Com2sense	**Overall best**	(0.50, 0.01)	(0.43, 0.11)	(0.36, 0.25)	(0.36, 0.24)	(0.49, 0.01)	(-0.03, 0.01)
			Task best	(0.51, 0.01)	(0.50, 0.003)	(0.58, 0.41)	(0.43, 0.29)	(0.49, 0.02)	(-0.02, 0.03)
		Deductive	**Overall best**	(0.51, 0.02)	(0.42, 0.16)	(0.28, 0.21)	(0.31, 0.22)	(0.53, 0.02)	(0.04, 0.04)
			Task best	(0.50, 0.003)	(0.34, 0.24)	(0.41, 0.43)	(0.33, 0.27)	(0.52, 0.03)	(0.03, 0.05)*
	b	**SNLI**	**Overall best**	(0.50, 0.04)	(0.47, 0.07)	(0.25, 0.18)	(0.29, 0.19)	(0.49, 0.05)	(-0.01, 0.07)
			Task best	(0.49, 0.03)	(0.50, 0.02)	(0.67, 0.09)	(0.57, 0.02)	(0.49, 0.04)	(-0.01, 0.07)
		Fantasy	**Overall best**	(0.51, 0.03)	(0.36, 0.26)	(0.28, 0.22)	(0.31, 0.22)	(0.56, 0.06)	(0.08, 0.08)
			Task best	(0.50, 0.02)	(0.35, 0.25)	(0.21, 0.20)	(0.24, 0.19)	(0.55, 0.05)	(0.07, 0.09)
		Com2sense	**Overall best**	(0.51, 0.01)	(0.51, 0.02)	(0.42, 0.34)	(0.37, 0.26)	(0.49, 0.03)	(-0.01, 0.04)
			Task best	(0.50, 0.01)	(0.45, 0.08)	(0.35, 0.45)	(0.27, 0.29)	(0.49, 0.001)	(-0.02, 0.02)
		Deductive	**Overall best**	(0.49, 0.01)	(0.32, 0.23)	(0.26, 0.20)	(0.28, 0.20)	(0.50, 0.02)	(-0.02, 0.03)
			Task best	(0.50, 0.01)	(0.50, 0.003)	(0.92, 0.08)	(0.65, 0.02)	(0.52, 0.02)	(0.05, 0.04)*
	c	**SNLI**	**Overall best**	(0.50, 0.003)	(0.67, 0.24)	(0.05, 0.03)	(0.08, 0.05)	(0.46, 0.04)	(-0.06, 0.06)*
			Task best	(0.50, 0.004)	(0.17, 0.23)	(0.32, 0.46)	(0.22, 0.31)	(0.47, 0.06)	(-0.04, 0.08)*
		Fantasy	**Overall best**	(0.52, 0.02)	(0.42, 0.30)	(0.11, 0.13)	(0.15, 0.17)	(0.48, 0.01)	(-0.05, 0.02)
			Task best	(0.51, 0.02)	(0.35, 0.25)	(0.10, 0.13)	(0.14, 0.17)	(0.48, 0.01)	(-0.05, 0.03)
		Com2sense	**Overall best**	(0.49, 0.02)	(0.32, 0.23)	(0.33, 0.25)	(0.32, 0.24)	(0.49, 0.03)	(-0.02, 0.05)
			Task best	(0.50, 0.003)	(0.31, 0.22)	(0.08, 0.07)	(0.12, 0.11)	(0.49, 0.01)	(-0.002, 0.01)
		Deductive	**Overall best**	(0.48, 0.02)	(0.29, 0.20)	(0.09, 0.07)	(0.13, 0.10)	(0.45, 0.02)	(-0.09, 0.03)*
			Task best	(0.50, 0.02)	(0.67, 0.23)	(0.55, 0.39)	(0.42, 0.29)	(0.53, 0.02)	(0.05, 0.04)
3	**a**	**SNLI**	**Overall best**	(0.47, 0.03)	(0.47, 0.03)	(0.43, 0.31)	(0.38, 0.19)	(0.46, 0.03)	(-0.07, 0.04)*
			Task best	(0.50, 0.01)	(0.34, 0.24)	(0.13, 0.10)	(0.18, 0.14)	(0.52, 0.02)	(0.04, 0.03)
		Fantasy	**Overall best**	(0.48, 0.02)	(0.41, 0.11)	(0.46, 0.34)	(0.38, 0.25)	(0.46, 0.02)	(-0.08, 0.02)
			Task best	(0.47, 0.01)	(0.40, 0.11)	(0.42, 0.32)	(0.37, 0.24)	(0.46, 0.004)	(-0.09, 0.01)
		Com2sense	**Overall best**	(0.50, 0.01)	(0.46, 0.05)	(0.36, 0.25)	(0.35, 0.24)	(0.49, 0.01)	(-0.03, 0.02)
			Task best	(0.51, 0.01)	(0.50, 0.003)	(0.58, 0.41)	(0.43, 0.29)	(0.49, 0.02)	(-0.03, 0.03)
		Deductive	**Overall best**	(0.51, 0.02)	(0.42, 0.16)	(0.28, 0.21)	(0.31, 0.22)	(0.53, 0.02)	(0.04, 0.04)
			Task best	(0.50, 0.003)	(0.34, 0.24)	(0.41, 0.43)	(0.33, 0.27)	(0.52, 0.03)	(0.03, 0.05)
	b	**SNLI**	**Overall best**	(0.50, 0.04)	(0.47, 0.07)	(0.25, 0.18)	(0.29, 0.19)	(0.49, 0.05)	(-0.01, 0.07)
			Task best	(0.49, 0.03)	(0.50, 0.02)	(0.67, 0.08)	(0.57, 0.02)	(0.49, 0.04)	(-0.01, 0.07)
		Fantasy	**Overall best**	(0.51, 0.03)	(0.36, 0.26)	(0.28, 0.22)	(0.31, 0.22)	(0.56, 0.06)	(0.08, 0.08)
			Task best	(0.50, 0.02)	(0.35, 0.25)	(0.22, 0.20)	(0.25, 0.20)	(0.55, 0.05)	(0.08, 0.09)
		Com2sense	**Overall best**	(0.50, 0.01)	(0.41, 0.15)	(0.41, 0.34)	(0.36, 0.26)	(0.49, 0.02)	(-0.02, 0.04)
			Task best	(0.50, 0.006)	(0.45, 0.08)	(0.35, 0.45)	(0.27, 0.29)	(0.49, 0.001)	(-0.02, 0.02)
		Deductive	**Overall best**	(0.49, 0.01)	(0.32, 0.22)	(0.26, 0.20)	(0.28, 0.20)	(0.50, 0.02)	(-0.02, 0.03)
			Task best	(0.50, 0.01)	(0.50, 0.003)	(0.92, 0.08)	(0.65, 0.02)	(0.52, 0.02)	(0.05, 0.04)*
	c	**SNLI**	**Overall best**	(0.50, 0.003)	(0.67, 0.24)	(0.05, 0.03)	(0.08, 0.05)	(0.46, 0.04)	(-0.06, 0.06)*
			Task best	(0.50, 0.004)	(0.17, 0.23)	(0.32, 0.46)	(0.22, 0.31)	(0.47, 0.06)	(-0.04, 0.08)*
		Fantasy	**Overall best**	(0.52, 0.02)	(0.45, 0.32)	(0.12, 0.12)	(0.17, 0.16)	(0.48, 0.01)	(-0.05, 0.03)
			Task best	(0.50, 0.01)	(0.35, 0.25)	(0.10, 0.12)	(0.13, 0.16)	(0.47, 0.01)	(-0.06, 0.03)
		Com2sense	**Overall best**	(0.49, 0.02)	(0.32, 0.23)	(0.33, 0.25)	(0.33, 0.24)	(0.49, 0.03)	(-0.01, 0.04)
			Task best	(0.50, 0.004)	(0.30, 0.22)	(0.08, 0.07)	(0.11, 0.11)	(0.49, 0.01)	(-0.01, 0.01)
		Deductive	**Overall best**	(0.48, 0.02)	(0.29, 0.20)	(0.09, 0.07)	(0.13, 0.10)	(0.45, 0.02)	(-0.09, 0.04)*
			Task best	(0.50, 0.02)	(0.67, 0.23)	(0.55, 0.39)	(0.42, 0.29)	(0.53, 0.02)	(0.05, 0.04)

Table 32: COT results of the Subtitles model on the development set for different configurations. A threshold of 0.0 has been used to calculate accuracy, precision, recall and F1 score. The amount of * indicates amount of runs that the correlation was significant (p <0.05). Results are reported as (x, y), with x indicating mean and y standard deviation over three runs.

				Acc	Prec	Recall	F1	AUC	Corr
1	**a**	**SNLI**	**Overall best**	(0.48, 0.01)	(0.47, 0.02)	(0.38, 0.23)	(0.38, 0.15)	(0.50, 0.02)	(-0.02, 0.02)
			Task best	(0.48, 0.03)	(0.48, 0.03)	(0.60, 0.11)	(0.53, 0.06)	(0.45, 0.02)	(-0.06, 0.03)*
		Fantasy	**Overall best**	(0.52, 0.03)	(0.48, 0.11)	(0.38, 0.30)	(0.37, 0.24)	(0.53, 0.03)	(0.05, 0.07)
			Task best	(0.52, 0.03)	(0.48, 0.11)	(0.38, 0.30)	(0.37, 0.24)	(0.53, 0.03)	(0.05, 0.07)
		Com2sense	**Overall best**	(0.51, 0.02)	(0.51, 0.02)	(0.46, 0.08)	(0.48, 0.03)	(0.51, 0.02)	(0.03, 0.05)
			Task best	(0.50, 0.01)	(0.50, 0.01)	(0.43, 0.23)	(0.43, 0.16)	(0.50, 0.02)	(-0.0002, 0.02)
		Deductive	**Overall best**	(0.51, 0.003)	(0.54, 0.03)	(0.26, 0.21)	(0.31, 0.16)	(0.53, 0.02)	(0.03, 0.05)
			Task best	(0.50, 0.03)	(0.56, 0.08)	(0.18, 0.10)	(0.25, 0.11)	(0.49, 0.03)	(-0.02, 0.06)*
	b	**SNLI**	**Overall best**	(0.49, 0.01)	(0.42, 0.08)	(0.25, 0.26)	(0.26, 0.21)	(0.48, 0.01)	(-0.05, 0.01)
			Task best	(0.47, 0.02)	(0.48, 0.02)	(0.57, 0.06)	(0.52, 0.03)	(0.47, 0.03)	(-0.03, 0.05)
		Fantasy	**Overall best**	(0.52, 0.04)	(0.53, 0.10)	(0.20, 0.11)	(0.29, 0.14)	(0.50, 0.03)	(-0.02, 0.08)
			Task best	(0.52, 0.04)	(0.53, 0.10)	(0.20, 0.11)	(0.29, 0.14)	(0.50, 0.03)	(-0.02, 0.08)
		Com2sense	**Overall best**	(0.51, 0.01)	(0.52, 0.02)	(0.36, 0.10)	(0.41, 0.06)	(0.51, 0.004)	(0.03, 0.02)
			Task best	(0.50, 0.01)	(0.50, 0.01)	(0.32, 0.18)	(0.36, 0.12)	(0.51, 0.02)	(0.01, 0.02)
		Deductive	**Overall best**	(0.50, 0.004)	(0.49, 0.02)	(0.20, 0.13)	(0.26, 0.13)	(0.49, 0.02)	(-0.02, 0.03)
			Task best	(0.48, 0.01)	(0.32, 0.23)	(0.53, 0.38)	(0.40, 0.28)	(0.48, 0.03)	(-0.05, 0.06)
	c	**SNLI**	**Overall best**	(0.50, 0.01)	(0.50, 0.02)	(0.32, 0.10)	(0.38, 0.07)	(0.49, 0.01)	(-0.01, 0.02)
			Task best	(0.46, 0.01)	(0.47, 0.01)	(0.54, 0.04)	(0.50, 0.02)	(0.45, 0.01)	(-0.07, 0.03)*
		Fantasy	**Overall best**	(0.50, 0.01)	(0.34, 0.24)	(0.16, 0.17)	(0.19, 0.18)	(0.50, 0.04)	(0.01, 0.09)
			Task best	(0.50, 0.01)	(0.34, 0.24)	(0.16, 0.17)	(0.19, 0.18)	(0.50, 0.04)	(0.01, 0.09)
		Com2sense	**Overall best**	(0.51, 0.01)	(0.59, 0.09)	(0.24, 0.17)	(0.30, 0.15)	(0.53, 0.02)	(0.07, 0.04)*
			Task best	(0.49, 0.01)	(0.49, 0.01)	(0.48, 0.27)	(0.45, 0.15)	(0.49, 0.02)	(-0.004, 0.03)
		Deductive	**Overall best**	(0.50, 0.01)	(0.50, 0.01)	(0.17, 0.11)	(0.23, 0.13)	(0.49, 0.01)	(-0.03, 0.01)
			Task best	(0.48, 0.01)	(0.45, 0.05)	(0.32, 0.32)	(0.30, 0.21)	(0.44, 0.02)	(-0.11, 0.03)**
2	**a**	**SNLI**	**Overall best**	(0.49, 0.01)	(0.47, 0.03)	(0.41, 0.28)	(0.39, 0.18)	(0.49, 0.01)	(-0.02, 0.02)
			Task best	(0.48, 0.01)	(0.48, 0.01)	(0.63, 0.06)	(0.55, 0.03)	(0.46, 0.01)	(-0.05, 0.03)
		Fantasy	**Overall best**	(0.50, 0.03)	(0.44, 0.14)	(0.40, 0.35)	(0.36, 0.24)	(0.54, 0.03)	(0.07, 0.08)
			Task best	(0.50, 0.03)	(0.44, 0.14)	(0.40, 0.35)	(0.36, 0.24)	(0.54, 0.03)	(0.07, 0.08)
		Com2sense	**Overall best**	(0.49, 0.01)	(0.49, 0.01)	(0.51, 0.06)	(0.50, 0.03)	(0.49, 0.02)	(-0.02, 0.03)
			Task best	(0.52, 0.02)	(0.53, 0.03)	(0.44, 0.24)	(0.44, 0.15)	(0.50, 0.01)	(0.01, 0.01)
		Deductive	**Overall best**	(0.51, 0.01)	(0.51, 0.05)	(0.29, 0.26)	(0.31, 0.19)	(0.52, 0.02)	(0.02, 0.05)
			Task best	(0.50, 0.01)	(0.51, 0.02)	(0.19, 0.13)	(0.25, 0.15)	(0.50, 0.03)	(-0.02, 0.06)*
	b	**SNLI**	**Overall best**	(0.49, 0.01)	(0.50, 0.02)	(0.28, 0.23)	(0.31, 0.18)	(0.48, 0.01)	(-0.03, 0.01)
			Task best	(0.48, 0.04)	(0.48, 0.03)	(0.56, 0.06)	(0.52, 0.04)	(0.48, 0.03)	(-0.02, 0.05)
		Fantasy	**Overall best**	(0.50, 0.07)	(0.45, 0.17)	(0.25, 0.14)	(0.32, 0.16)	(0.48, 0.03)	(-0.06, 0.08)
			Task best	(0.50, 0.07)	(0.45, 0.17)	(0.25, 0.14)	(0.32, 0.16)	(0.48, 0.03)	(-0.06, 0.08)
		Com2sense	**Overall best**	(0.50, 0.02)	(0.51, 0.02)	(0.45, 0.16)	(0.46, 0.08)	(0.50, 0.01)	(0.004, 0.01)
			Task best	(0.50, 0.01)	(0.50, 0.01)	(0.31, 0.12)	(0.37, 0.08)	(0.49, 0.01)	(0.002, 0.002)
		Deductive	**Overall best**	(0.49, 0.01)	(0.48, 0.01)	(0.22, 0.11)	(0.28, 0.10)	(0.48, 0.02)	(-0.04, 0.03)
			Task best	(0.50, 0.01)	(0.33, 0.24)	(0.56, 0.40)	(0.41, 0.29)	(0.48, 0.02)	(-0.04, 0.04)
	c	**SNLI**	**Overall best**	(0.50, 0.01)	(0.50, 0.02)	(0.37, 0.09)	(0.42, 0.05)	(0.49, 0.01)	(-0.01, 0.02)
			Task best	(0.46, 0.01)	(0.47, 0.01)	(0.56, 0.02)	(0.51, 0.02)	(0.46, 0.01)	(-0.06, 0.02)
		Fantasy	**Overall best**	(0.48, 0.03)	(0.28, 0.21)	(0.20, 0.22)	(0.22, 0.21)	(0.49, 0.04)	(-0.04, 0.09)
			Task best	(0.48, 0.03)	(0.28, 0.21)	(0.20, 0.22)	(0.22, 0.21)	(0.49, 0.04)	(-0.04, 0.09)
		Com2sense	**Overall best**	(0.50, 0.02)	(0.55, 0.07)	(0.33, 0.22)	(0.35, 0.18)	(0.52, 0.03)	(0.04, 0.05)*
			Task best	(0.50, 0.02)	(0.49, 0.02)	(0.46, 0.24)	(0.45, 0.14)	(0.49, 0.02)	(-0.01, 0.01)
		Deductive	**Overall best**	(0.51, 0.01)	(0.54, 0.07)	(0.23, 0.09)	(0.30, 0.09)	(0.50, 0.02)	(-0.01, 0.04)
			Task best	(0.47, 0.02)	(0.43, 0.02)	(0.29, 0.29)	(0.27, 0.21)	(0.43, 0.03)	(-0.11, 0.04)**
3	**a**	**SNLI**	**Overall best**	(0.48, 0.02)	(0.44, 0.06)	(0.42, 0.34)	(0.38, 0.21)	(0.48, 0.01)	(-0.03, 0.02)
			Task best	(0.47, 0.02)	(0.47, 0.01)	(0.62, 0.06)	(0.54, 0.03)	(0.45, 0.01)	(-0.07, 0.03)*
		Fantasy	**Overall best**	(0.51, 0.02)	(0.54, 0.03)	(0.42, 0.34)	(0.39, 0.20)	(0.53, 0.03)	(0.06, 0.07)
			Task best	(0.51, 0.02)	(0.54, 0.03)	(0.42, 0.34)	(0.39, 0.20)	(0.53, 0.03)	(0.06, 0.07)
		Com2sense	**Overall best**	(0.49, 0.02)	(0.49, 0.01)	(0.55, 0.10)	(0.52, 0.04)	(0.49, 0.02)	(-0.01, 0.04)
			Task best	(0.50, 0.01)	(0.52, 0.04)	(0.43, 0.23)	(0.43, 0.14)	(0.50, 0.01)	(0.02, 0.02)
		Deductive	**Overall best**	(0.53, 0.02)	(0.55, 0.02)	(0.33, 0.29)	(0.34, 0.20)	(0.52, 0.01)	(0.03, 0.03)
			Task best	(0.49, 0.02)	(0.52, 0.06)	(0.21, 0.12)	(0.27, 0.12)	(0.50, 0.05)	(0.01, 0.06)
	b	**SNLI**	**Overall best**	(0.49, 0.01)	(0.47, 0.03)	(0.30, 0.24)	(0.31, 0.19)	(0.47, 0.02)	(-0.05, 0.03)
			Task best	(0.47, 0.04)	(0.48, 0.04)	(0.54, 0.05)	(0.50, 0.03)	(0.48, 0.04)	(-0.02, 0.06)
		Fantasy	**Overall best**	(0.48, 0.06)	(0.47, 0.10)	(0.25, 0.09)	(0.31, 0.10)	(0.47, 0.04)	(-0.07, 0.08)
			Task best	(0.48, 0.06)	(0.47, 0.10)	(0.25, 0.09)	(0.31, 0.10)	(0.47, 0.04)	(-0.07, 0.08)
		Com2sense	**Overall best**	(0.51, 0.01)	(0.55, 0.04)	(0.37, 0.26)	(0.37, 0.20)	(0.53, 0.02)	(0.04, 0.05)
			Task best	(0.49, 0.01)	(0.49, 0.02)	(0.30, 0.12)	(0.35, 0.08)	(0.49, 0.02)	(0.01, 0.03)
		Deductive	**Overall best**	(0.49, 0.01)	(0.48, 0.03)	(0.24, 0.10)	(0.30, 0.08)	(0.49, 0.01)	(-0.01, 0.02)
			Task best	(0.48, 0.02)	(0.66, 0.24)	(0.53, 0.38)	(0.41, 0.28)	(0.48, 0.04)	(-0.03, 0.07)*
	c	**SNLI**	**Overall best**	(0.49, 0.02)	(0.50, 0.03)	(0.42, 0.11)	(0.44, 0.07)	(0.49, 0.01)	(-0.02, 0.02)
			Task best	(0.46, 0.02)	(0.47, 0.01)	(0.58, 0.02)	(0.52, 0.02)	(0.45, 0.01)	(-0.08, 0.02)*
		Fantasy	**Overall best**	(0.50, 0.01)	(0.51, 0.02)	(0.27, 0.25)	(0.29, 0.21)	(0.48, 0.02)	(-0.05, 0.04)
			Task best	(0.50, 0.01)	(0.51, 0.02)	(0.27, 0.25)	(0.29, 0.21)	(0.48, 0.02)	(-0.05, 0.04)
		Com2sense	**Overall best**	(0.50, 0.01)	(0.51, 0.02)	(0.30, 0.04)	(0.38, 0.02)	(0.51, 0.04)	(0.003, 0.04)*
			Task best	(0.50, 0.01)	(0.50, 0.01)	(0.46, 0.23)	(0.45, 0.13)	(0.50, 0.02)	(0.02, 0.01)
		Deductive	**Overall best**	(0.50, 0.02)	(0.51, 0.05)	(0.24, 0.09)	(0.31, 0.08)	(0.50, 0.01)	(-0.01, 0.01)
			Task best	(0.48, 0.02)	(0.48, 0.04)	(0.29, 0.29)	(0.28, 0.20)	(0.45, 0.02)	(-0.08, 0.04)*

Table 33: COT results of the CHILDES model on the development set for different configurations. A threshold of 0.0 has been used to calculate accuracy, precision, recall and F1 score. The amount of * indicates amount of runs that the correlation was significant (p <0.05). Results are reported as (x, y), with x indicating mean and y standard deviation over three runs.