AN EXPLANATION OF IN-CONTEXT LEARNING AS IMPLICIT BAYESIAN INFERENCE

Sang Michael Xie, Aditi Raghunathan, Percy Liang, Tengyu Ma
Stanford University
{xie,aditir,pliang,tengyuma}@cs.stanford.edu

ABSTRACT

Large language models (LMs) such as GPT-3 have the surprising ability to do in-context learning, where the model learns to do a downstream task simply by conditioning on a prompt consisting of input-output examples. The LM learns from these examples *without being explicitly pretrained to learn*. Thus, it is unclear what enables in-context learning. In this paper, we study how in-context learning can emerge when pretraining documents have long-range coherence. Here, the LM must infer a latent document-level concept to generate coherent next tokens during pretraining. At test time, in-context learning occurs when the LM also infers a shared latent concept between examples in a prompt. We prove when this occurs despite a distribution mismatch between prompts and pretraining data in a setting where the pretraining distribution is a mixture of HMMs. In contrast to messy large-scale datasets used to train LMs capable of in-context learning, we generate a small-scale synthetic dataset (GINC) where Transformers and LSTMs both exhibit in-context learning[1]. Beyond the theory, experiments on GINC mirror real-world phenomena including improved in-context performance with model scaling, sensitivity to example order, and instances where zero-shot is better than few-shot in-context learning.

1 INTRODUCTION

Large language models (LMs) such as GPT-3 (Brown et al., 2020; Lieber et al., 2021; Wang & Komatsuzaki, 2021; Radford et al., 2019) are pretrained on massive text corpora to predict the next word given previous words. They demonstrate the surprising ability to do *in-context learning*, where an LM "learns" to do a task simply by conditioning on a prompt containing input-output pairs, achieving SOTA results on LAMBADA (Paperno et al., 2016) and TriviaQA (Joshi et al., 2017) tasks (18% and 3% over previous SOTA (Brown et al., 2020)). For example, consider the task of predicting nationalities from names. A prompt (Figure 1) is constructed by concatenating independent "training" examples (e.g., "Albert Einstein was German") followed by a "test example" ("Marie Curie was"). Conditioning on this prompt, GPT-3 places the largest probability on the correct output

$$p(\text{“Polish”} \mid \text{“Albert Einstein was German \textbackslash n Mahatma Gandhi was Indian \textbackslash n Marie Curie was”})$$

by inferring the task from examples. Intruigingly, GPT-3 was not explicitly pretrained to learn from examples, and the distribution of prompts (which concatenate independent examples) is quite different from natural language. Our understanding of in-context learning is limited since (i) real pretraining data is messy and (ii) in-context learning has so far required large-scale datasets and models.

In this paper, we introduce a simple pretraining distribution where in-context learning emerges. To generate a document, we first draw a latent concept θ, which parameterizes the transitions of a Hidden Markov Model (HMM) (Baum & Petrie, 1966), then sample a sequence of tokens from the HMM (Figure 9). This latent variable structure is common in topic models such as LDA (Blei et al., 2003; Gruber et al., 2007). During pretraining, the LM must infer the latent concept across multiple sentences to generate coherent continuations. When conditioning on a prompt, in-context learning occurs when the LM also infers a shared *prompt concept* across examples to make a prediction. We assume the LM fits the pretraining distribution p exactly with enough data and expressivity, so that the question of in-context learning becomes characterizing the conditional distribution of completions given prompts $p(\text{output}|\text{prompt})$ under the pretraining distribution, where the prompt is generated

[1]The code, data, and experiments are located on GitHub and CodaLab.

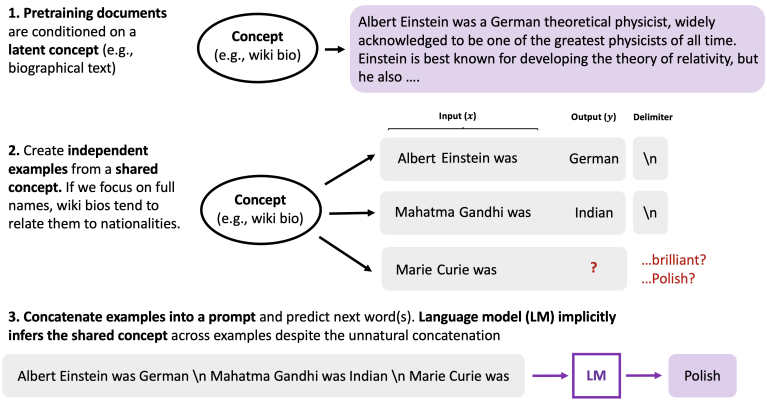

Figure 1: In-context learning can emerge from modeling long-range coherence in the pretraining data. During pretraining, the language model (LM) implicitly learns to infer a latent concept (e.g., wiki bios, which typically transition between name (Albert Einstein) $\rightarrow$ nationality (German) $\rightarrow$ occupation (physicist) $\rightarrow$ ...) shared across sentences in a document. Although prompts are unnatural sequences that concatenate independent examples, in-context learning occurs if the LM can still infer the shared concept across examples to do the task (name $\rightarrow$ nationality, which is part of wiki bios).

from a different distribution p_{prompt}. This conditional distribution, which is the *posterior predictive distribution*, marginalizes out the latent concepts:

$$p(\text{output}|\text{prompt}) = \int_{\text{concept}} p(\text{output}|\text{concept},\text{prompt})p(\text{concept}|\text{prompt})d(\text{concept}). \tag{1}$$

If $p(\text{concept}|\text{prompt})$ concentrates on the prompt concept with more examples, then the LM learns via marginalization by "selecting" the prompt concept. Thus, in-context learning can be viewed as the LM implicitly performing Bayesian inference.

The main challenge is that prompts are sampled from a different distribution than the pretraining distribution. The canonical Bayesian asymptotic tool is the Bernstein-von Mises theorem (van der Vaart, 1998; Kleijn & van der Vaart, 2012; Gunst & Shcherbakova, 2008), which asserts (under regularity conditions) that the posterior distribution of a latent variable concentrates on the maximum likelihood estimate. However, Bernstein-von Mises typically assumes observations are independent and/or drawn from the same distribution as the model, both of which are not satisfied. We prove that despite the distribution mismatch, the asymptotic prediction error of in-context learning is optimal when the signal about the latent concept in each prompt example is larger than the error due to the distribution mismatch. Additionally, we prove that the in-context learning error decreases with the length of each example—thus, information in the inputs, not just the input-output mapping, can be useful for in-context learning.

As a companion to this theory, we created the **G**enerative **IN**-**C**ontext learning dataset (GINC), which is a small-scale synthetic dataset for studying in-context learning. We find that both Transformers (Vaswani et al., 2017) and LSTMs (Hochreiter & Schmidhuber, 1997) trained on GINC exhibit in-context learning. We verify intuitions from the theory, showing that the accuracy of in-context learning improves with the number of examples and example length. Ablations of the GINC dataset show that the latent concept structure in the pretraining distribution is crucial to the emergence of in-context learning.

The experiments also bring up open questions which go beyond our theory, which only studies the pretraining distribution. We find that scaling up the number of model parameters steadily improves the in-context accuracy despite achieving the same pretraining loss, showing that larger models may improve in-context learning beyond increasing the capacity for memorizing the training data better. Previously observed in-context learning phenomena such as sensitivity to example ordering (Zhao et al., 2021) and the existence of settings where zero-shot is better than one/few-shot learning (Brown et al., 2020) are also mirrored in GINC.

2 In-context learning setting

Pretraining distribution. In our framework, a latent concept θ from a family of concepts Θ defines a distribution over observed tokens o from a vocabulary $\mathcal{O}$. To generate a document, we first sample a

concept from a prior $p(\theta)$ and then sample the document given the concept. Each pretraining document is a length T sequence:

$$p(o_1,...,o_T) = \int_{\theta \in \Theta} p(o_1,...,o_T|\theta)p(\theta)d\theta. \tag{2}$$

We assume $p(o_1,...,o_T|\theta)$ is defined by a Hidden Markov Model (HMM). The concept θ determines the transition probability matrix of the HMM hidden states $h_1,...,h_T$ from a hidden state set $\mathcal{H}$.

Prompt distribution. The prompt distribution p_{prompt} generates prompts for in-context learning. The prompt is a concatenation of n independent training examples and 1 test input x_{test}, which are all conditioned on a shared prompt concept θ^*. The goal is to predict the test output y_{test} by predicting the next token.

A prompt example is composed of an input token sequence x (e.g., Albert Einstein was) followed by an output token y (e.g., German). In particular, the i-th training example O_i consists of an input $x_i = O_i[1:\ k-1]$ (the first $k-1$ tokens) followed by an output token $y_i = O_i[k]$ at the end[2]. The i-th training example is independently generated as follows:

1. Generate a start hidden state h_i^{start} from a *prompt start distribution* π_{prompt}.
2. Given h_i^{start}, generate the example sequence $O_i = [x_i, y_i]$ from $p(O_i|h_i^{\text{start}}, \theta^*)$, the *pretraining distribution* conditioned on a prompt concept θ^*.

The test input $x_{\text{test}} = x_{n+1}$ is sampled similarly. Between each example, there is a special delimiter token o^{delim}. The prompt consists of a sequence of training examples (S_n) followed by the test example x_{test}:

$$[S_n, x_{\text{test}}] = [x_1, y_1, o^{\text{delim}}, x_2, y_2, o^{\text{delim}}, ..., x_n, y_n, o^{\text{delim}}, x_{\text{test}}] \sim p_{\text{prompt}}. \tag{3}$$

Mismatch between prompt and pretraining distributions. Since transitions between independent examples can be unnatural, the prompts are low probability sequences under the pretraining distribution. We provide a simple illustration using the names to nationalities example. Suppose that wiki bio documents in the pretraining data typically transition between name $\rightarrow$ nationality $\rightarrow$ occupation $\rightarrow$ .... In the prompt, the examples transition between name $\rightarrow$ nationality $\rightarrow$ name $\rightarrow$ nationality $\rightarrow$ ..., which contains low-probability transitions such as "German" $\rightarrow$ "Mahatma Gandhi". The prompt formatting (e.g., choice of delimiter) can also be a source of mismatch. We aim to show that despite this mismatch, large LMs can infer the prompt concept from examples.

In-context predictor and task. For in-context learning, the output target y for each example x is sampled according to $p_{\text{prompt}}(y|x)$:

$$y_{\text{test}} \sim p_{\text{prompt}}(y|x_{\text{test}}) = \mathbb{E}_{h_{\text{test}}^{\text{start}} \sim \pi_{\text{prompt}}(h_{\text{test}}^{\text{start}}|x_{\text{test}})} \left[p(y|x_{\text{test}}, h_{\text{test}}^{\text{start}}, \theta^*) \right]. \tag{4}$$

where $h_{\text{test}}^{\text{start}}$ denotes the hidden state corresponding to the first token of x_{test} and $\pi_{\text{prompt}}(h_{\text{test}}^{\text{start}}|x_{\text{test}})$ is defined as $p_{\text{prompt}}(x_{\text{test}}|h_{\text{test}}^{\text{start}})\pi_{\text{prompt}}(h_{\text{test}}^{\text{start}})/p_{\text{prompt}}(x_{\text{test}})$.

We analyze the in-context predictor $f_n(x_{\text{test}}) = \text{argmax}_y\, p(y|S_n, x_{\text{test}})$, which outputs the most likely prediction over the *pretraining* distribution conditioned on the prompt from the *prompt* distribution[3]. We study the in-context predictor and its expected 0-1 error with n examples $L_{\text{0-1}}(f_n) = \mathbb{E}_{x_{\text{test}}, y_{\text{test}} \sim p_{\text{prompt}}}[\mathbf{1}[f_n(x_{\text{test}}) \neq y_{\text{test}}]]$.

2.1 Assumptions

We detail the assumptions in our framework, including the structure of delimiters and regularity assumptions. We first assume that there exists a subset of *delimiter hidden states* $\mathcal{D}$ which generates the special delimiter token o^{delim} deterministically.

Assumption 1 (Delimiter hidden states)**.** *Let the delimiter hidden states $\mathcal{D}$ be a subset of $\mathcal{H}$. For any $h \in \mathcal{D}$ and $\theta \in \Theta$, $p(o^{delim}|h,\theta) = 1$ and for any $h' \notin \mathcal{D}$, $p(o^{delim}|h,\theta) = 0$.*

Thus, observing the delimiter o^{delim} reveals that the corresponding hidden state is in $\mathcal{D}$, but does not reveal which element of $\mathcal{D}$ it is. The delimiter is usually a token that can appear in a broad range of contexts (e.g., newline). The delimiter ideally does not distract from the examples — for example, an adversarial delimiter could look like part of the input x. To mitigate these scenarios, we assume that no delimiter (e.g., newline) is significantly more likely under one concept rather than another.

[2]The example length k is fixed for simplicity — we leave extending our analysis to variable k as future work.

[3]In practice, greedy decoding or nucleus sampling (Holtzman et al., 2020) are used for likely completions.

Assumption 2 (Bound on delimiter transitions)**.** *For any delimiter state $h^{delim} \in \mathcal{D}$ and any hidden state $h \in \mathcal{H}$, the probability of transitioning to a delimiter hidden state under θ is upper bounded $p(h^{delim}|h,\theta) < c_1$ for any $\theta \in \Theta \setminus \{\theta^*\}$, and is lower bounded $p(h^{delim}|h,\theta^*) > c_2 > 0$ for θ^*. Additionally, the start hidden state distribution over delimiters is uniform for all concepts θ: $p(h^{delim}|\theta) = \frac{1}{|\mathcal{D}|}$ for all $\theta \in \Theta$ and $h^{delim} \in \mathcal{D}$.*

We also assume the prompt concept θ^* is in the family Θ, which is a broad set of concepts.

Assumption 3 (Well-specification)**.** *The prompt concept θ^* is in Θ.*

Even though the pretraining distribution is broad, the prompt is still low probability under the pretraining distribution since it concatenates independent examples.

Finally, if the prompt has zero probability under the prompt concept θ^*, then Bayesian inference will not be able to infer the prompt concept as in Section 3.1. The following are regularity assumptions which mainly ensure that the prompt is not zero probability under θ^*.

Assumption 4 (Regularity)**.** *The pretraining distribution p satisfies: 1) Lower bound on transition probability for the prompt concept θ^*: for any pair of hidden states $h,h' \in \mathcal{H}$, $p(h|h',\theta^*) > c_3 > 0$. 2) All tokens can be emitted: for every symbol o, there is some hidden state $h \in \mathcal{H}$ such that $p(o|h,\theta^*) > c_4 > 0$. 3) The hidden state Markov chain for concept θ has stationary distribution π^θ over $\mathcal{H}$. 4) The prior $p(\theta)$ has support over the entire concept family Θ and is bounded above everywhere.*

We denote the stationary distribution conditioned on x to be $\pi^\theta(h|x) = (p(x|h,\theta)\pi^\theta(h))/p(x|\theta)$.

3 Theoretical analysis

We prove that in the limit of infinite examples, the error of the in-context predictor is optimal if a *distinguishability* condition holds — the prompt concept θ^* is distinct enough from the other concepts in Θ (e.g., when Θ is a discrete set). When distinguishability does not hold (e.g, Θ is continuous-valued), we show that the expected error still decreases with the length of each example, showing that information in both the inputs and the input-output mapping contribute to in-context learning.

3.1 High-level approach

Our goal is to show that $\mathrm{argmax}_y p(y|S_n,x_{\text{test}}) \to \mathrm{argmax}_y p_{\text{prompt}}(y|x_{\text{test}})$ as the number of examples n grows. In the following, assume that the prompt has non-zero probability under the pretraining distribution p given θ^*, meaning that $p(S_n,x_{\text{test}}|\theta^*) > 0$. We expand $p(y|S_n,x_{\text{test}})$ to analyze its limit:

$$
\begin{aligned}
p(y|S_n,x_{\text{test}}) &= \int_\theta p(y|S_n,x_{\text{test}},\theta)p(\theta|S_n,x_{\text{test}})d\theta \\
&\propto \int_\theta p(y|S_n,x_{\text{test}},\theta)p(S_n,x_{\text{test}}|\theta)p(\theta)d\theta \quad \text{(Bayes' rule, drop the constant } \frac{1}{p(S_n,x_{\text{test}})}\text{)} \\
&= \int_\theta \sum_{h^{\text{start}}_{\text{test}} \in \mathcal{H}} p(y|x_{\text{test}},h^{\text{start}}_{\text{test}},\theta)p(h^{\text{start}}_{\text{test}}|S_n,x_{\text{test}},\theta)\frac{p(S_n,x_{\text{test}}|\theta)}{p(S_n,x_{\text{test}}|\theta^*)}p(\theta)d\theta \qquad (5) \\
&\qquad \text{(Law of total prob, Markov property, divide by } p(S_n,x_{\text{test}}|\theta^*) \text{ (a constant))} \\
&= \int_\theta \sum_{h^{\text{start}}_{\text{test}} \in \mathcal{H}} p(y|x_{\text{test}},h^{\text{start}}_{\text{test}},\theta)p(h^{\text{start}}_{\text{test}}|S_n,x_{\text{test}},\theta)\exp(n\cdot r_n(\theta))p(\theta)d\theta \qquad (6)
\end{aligned}
$$

where $r_n(\theta) = \frac{1}{n}\log\frac{p(S_n,x_{\text{test}}|\theta)}{p(S_n,x_{\text{test}}|\theta^*)}$. In Theorem 1, we prove that under a distinguishability condition, $\exp(n\cdot r_n(\theta)) \to 0$ for all concepts θ except the prompt concept θ^*, where $\exp(n\cdot r_n(\theta^*)) = 1$. The only nonzero term in the integral is when $\theta = \theta^*$, and thus the prompt concept is "selected" as a consequence of Bayesian inference[4]. In Lemma 3, we show that the distribution of the hidden start state for the test example $p(h^{\text{start}}_{\text{test}}|S_n,x_{\text{test}},\theta^*)$ converges to the stationary distribution $\pi^{\theta^*}(h^{\text{start}}_{\text{test}}|x_{\text{test}})$ conditioned on x_{test}. Putting these together with Equation 6, the in-context predictor infers the prompt concept θ^*:

$$
\mathop{\mathrm{argmax}}_y p(y|S_n,x_{\text{test}}) \to \mathop{\mathrm{argmax}}_y \mathbb{E}_{h^{\text{start}}_{\text{test}} \sim \pi^{\theta^*}(h^{\text{start}}_{\text{test}}|x_{\text{test}})}[p(y|x_{\text{test}},h^{\text{start}}_{\text{test}},\theta^*)] \qquad (7)
$$

[4]We can exchange limits and integrals since the probabilities are bounded (dominated convergence).

Recall that the test label is sampled from $p_{\text{prompt}}(y|x_{\text{test}}) = \mathbb{E}_{h_{\text{test}}^{\text{start}} \sim \pi_{\text{prompt}}(h_{\text{test}}^{\text{start}}|x_{\text{test}})}[p(y|x_{\text{test}}, h_{\text{test}}^{\text{start}}, \theta^*)]$. If the prompt start distribution π_{prompt} equals the stationary distribution π^{θ^*}, then the conditional state distributions $\pi_{\text{prompt}}(h_{\text{test}}^{\text{start}}|x_{\text{test}}) = \pi^{\theta^*}(h_{\text{test}}^{\text{start}}|x_{\text{test}})$ are the same between pretraining and prompting. Thus, the in-context predictor makes the optimal prediction when $\pi_{\text{prompt}} = \pi^{\theta^*}$.

3.2 Heuristic derivation

Recall from Section 3.1 that if $\exp(n \cdot r_n(\theta)) \to 0$ for all $\theta \neq \theta^*$, then Bayesian inference "selects" the prompt concept through marginalization. To do this, we focus on showing that $r_n(\theta)$, the average log-likelihood ratio between θ and θ^*, converges to a negative constant, and thus nr_n goes to $-\infty$.

The main technical challenge is to handle the sequence-of-examples structure of the prompt, which makes all the examples dependent with respect to the pretraining distribution. Our approach uses properties of delimiter tokens to approximately factorize the examples, with constant error per example. We let $O_i^{\text{ex}} = [o_{i-1}^{\text{delim}}, O_i]$ be the i-th input-output pair and the previous delimiter together for $i > 1$ and define $O_1^{\text{ex}} = O_1$. Expanding the likelihood term inside $r_n(\theta)$, our goal is to show

$$p(S_n, x_{\text{test}}|\theta) = p(x_{\text{test}}|S_n, \theta)p(S_n|\theta) \approx \prod_{i=1}^{n} O(1)p(O_i|\theta) \tag{8}$$

To show this, we expand $p(S_n|\theta)$ with the chain rule, and with Assumption 4 (to bound $p(x_{\text{test}}|S_n, \theta)$ by $O(1)$) it can be shown that

$$p(x_{\text{test}}|S_n, \theta)p(S_n|\theta) \approx \prod_{i=1}^{n} O(1)p(O_i^{\text{ex}}|O_{1:i-1}^{\text{ex}}, \theta). \tag{9}$$

We then marginalize $p(O_i^{\text{ex}}|O_{1:i-1}^{\text{ex}}, \theta)$ over the hidden state h_{i-1}^{delim} corresponding to the delimiter in $O_i^{\text{ex}} = [o_{i-1}^{\text{delim}}, O_i]$:

$$\prod_{i=1}^{n} O(1)p(O_i^{\text{ex}}|O_{1:i-1}^{\text{ex}}, \theta) = \prod_{i=1}^{n} O(1) \sum_{h_{i-1}^{\text{delim}} \in \mathcal{D}} p(O_i|h_{i-1}^{\text{delim}}, \theta)p(h_{i-1}^{\text{delim}}|O_{1:i-1}^{\text{ex}}, \theta) \approx \prod_{i=1}^{n} O(1)p(O_i|\theta) \tag{10}$$

While summing over $\mathcal{H}$ above would be a trivial equality, we can replace $\mathcal{H}$ with the set of delimiter hidden states $\mathcal{D}$ since $p(h_{i-1}^{\text{delim}}|O_{1:i-1}^{\text{ex}}, \theta) = 0$ for non-delimiter hidden states (Assumption 1). We used in the first equality that $O_{1:i-1}^{\text{ex}} \to h_{i-1}^{\text{delim}} \to O_i^{\text{ex}}$ forms a Markov chain and $p(o_{i-1}^{\text{delim}}|h_{i-1}^{\text{delim}}) = 1$ (Assumption 1) to change O_i^{ex} to O_i. Finally, we can show using properties of delimiter hidden states (Assumption 2) that $p(h_{i-1}^{\text{delim}}|O_{1:i-1}^{\text{ex}}, \theta) \approx O(1)p(h_{i-1}^{\text{delim}}|\theta)$, which allows us to marginalize out h_{i-1}^{delim} in the second step. Therefore, we can upper bound $r_n(\theta)$ as

$$r_n(\theta) \leq \frac{1}{n}\left(O(n) + \sum_{i=1}^{n} \log \frac{p(O_i|\theta)}{p(O_i|\theta^*)}\right) \to O(1) + \mathbb{E}_{O \sim p_{\text{prompt}}}\left[\log \frac{p(O|\theta)}{p(O|\theta^*)}\right]. \tag{11}$$

Since all tokens after the start token in an example O are drawn from the pretraining distribution p conditioned on the prompt concept θ^*, the expectation in Equation 11 becomes a negative KL divergence for tokens 2 through k. There are $O(k)$ KL terms and only $O(1)$ other error terms, which come from the distribution mismatch between the prompt and pretraining distributions. If the KL terms are larger than the error terms, then $r_n(\theta)$ has a negative limit. If this holds for all $\theta \neq \theta^*$, then we have $\exp(n \cdot r_n(\theta)) \to 0$ for all $\theta \neq \theta^*$, enabling in-context learning.

3.3 Formal results

3.3.1 In-context learning under distinguishability

We define a distinguishability condition which formalizes when in-context learning occurs. Letting p_θ^j be the output distribution of the j-th token given the previous tokens, the condition depends on the KL divergence between $p_{\theta^*}^j$ and p_θ^j as well as error terms $\epsilon_{\text{start}}^\theta$ and $\epsilon_{\text{delim}}^\theta$ coming from the distribution mismatch between the prompt and pretraining distributions at the start and delimiter token for each example:

$$p_\theta^j(o) = p(O[j] = o \,|\, O[1:j-1], \theta), \quad KL_j(\theta^* \| \theta) = \mathbb{E}_{O[1:j-1] \sim p_{\text{prompt}}}[KL(p_{\theta^*}^j \| p_\theta^j)] \tag{12}$$

$$\epsilon_{\text{start}}^\theta = \mathbb{E}_{O[1] \sim p_{\text{prompt}}}\left[\log \frac{p(O[1]|\theta)}{p(O[1]|\theta^*)}\right], \quad \epsilon_{\text{delim}}^\theta = (\log(c_1) - \log(c_2)). \tag{13}$$

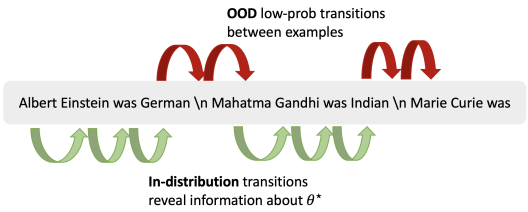

Figure 2: When the signal about the prompt concept within each in-context example (green) is greater than the error from low-probability transitions between examples, in-context learning succeeds in our latent concept setting (Theorem 1). Increasing the example length k increases the number of green transitions. The signal for in-context learning comes from token transitions in both the inputs and the input-output mapping.

Condition 1 (Distinguishability)**.** *We define θ^* to be distinguishable if for all $\theta \in \Theta, \theta \neq \theta^*$,*

$$\sum_{j=2}^{k} KL_j(\theta^* \| \theta) > \epsilon^{\theta}_{start} + \epsilon^{\theta}_{delim}. \tag{14}$$

When the signal from KL divergence (LHS) is larger than the error terms, Equation 14 is satisfied (Figure 2). For larger example lengths k, the LHS increases, improving distinguishability. Intuitively, larger example lengths increase the proportion of the prompt sampled from the pretraining distribution. Under Condition 1, the in-context predictor asymptotically achieves the optimal expected error.

Theorem 1. *Assume the assumptions in Section 2.1 hold. If Condition 1 holds, then as $n \to \infty$ the prediction according to the pretraining distribution is*

$$\operatorname*{argmax}_y p(y|S_n, x_{test}) \to \operatorname*{argmax}_y \mathbb{E}_{h^{start}_{test} \sim \pi^{\theta^*}(h^{start}_{test}|x_{test})}[p(y|x_{test}, h^{start}_{test}, \theta^*)]. \tag{15}$$

As a corollary, if the prompt start distribution π_{prompt} is the stationary distribution π^{θ^*} of the hidden state chain for θ^*, then the in-context predictor f_n achieves the optimal expected 0-1 error: $\lim_{n\to\infty} L_{0\text{-}1}(f_n) = \inf_f L_{0\text{-}1}(f)$.

When the prompt start distribution π_{prompt} matches the stationary distribution π^{θ^*}, the in-context predictor makes the optimal prediction (Equation 4).

3.3.2 Non-distinguishable case

The distinguishability condition (Condition 1) fails when there is some $\theta \neq \theta^*$ for which the KL divergence between θ and θ^* is less than the error terms. However, this also means that the output distributions of θ and θ^* are close in KL. We leverage this to prove that the expected 0-1 error decreases with the example length k under two different settings where distinguishability does not hold.

Continuity. Our first result relies on a continuity assumption between the concept parameter and its corresponding output distribution. Our assumption is based on prior works (Kleijn & van der Vaart, 2012), where the KL divergence is assumed to have a 2nd-order Taylor expansion.

Theorem 2. *Let the set of θ which does not satisfy Equation 14 in Condition 1 to be $\mathcal{B}$ and assume $\pi_{prompt} = \pi^{\theta^*}$. Assume that KL divergences have a 2nd-order Taylor expansion around θ^*:*

$$\forall j,\ KL_j(\theta^* \| \theta) = \frac{1}{2}(\theta - \theta^*)^\top I_{j,\theta^*}(\theta - \theta^*) + O(\|\theta - \theta^*\|^3) \tag{16}$$

where I_{j,θ^*} is the Fisher information matrix of the j-th token distribution with respect to θ^*. Let $\gamma_{\theta^*} = \frac{\max_j \lambda_{max}(I_{j,\theta^*})}{\min j \lambda_{min}(I_{j,\theta^*})}$ where $\lambda_{max}, \lambda_{min}$ return the largest and smallest eigenvalues. Then as $n \to \infty$, the 0-1 risk of the in-context learning predictor f_n is bounded as

$$\lim_{n\to\infty} L_{0\text{-}1}(f_n) \leq \inf_f L_{0\text{-}1}(f) + g^{-1}\left(O\left(\frac{\gamma_{\theta^*} \sup_{\theta \in \mathcal{B}}(\epsilon^{\theta}_{start} + \epsilon^{\theta}_{delim})}{k-1}\right)\right) \tag{17}$$

where $g(\delta) = \frac{1}{2}((1-\delta)\log(1-\delta) + (1+\delta)\log(1+\delta))$ is a calibration function (Steinwart, 2007; Ávila Pires & Szepesvári, 2016) for the multiclass logistic loss for $\delta \in [0,1)$.

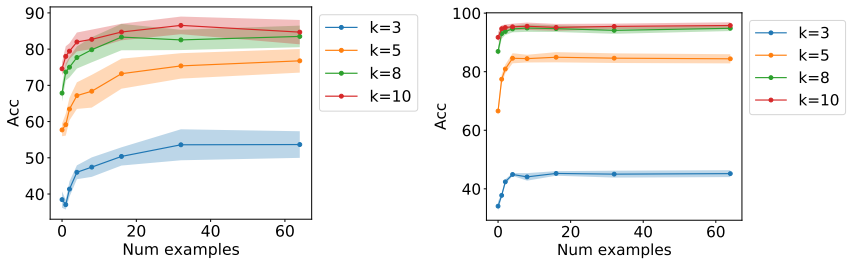

Figure 3: In-context accuracy (95% intervals) of Transformers (left) and LSTMs (right) on the GINC dataset. Accuracy increases with number of examples n and length of each example k.

Since the inverse calibration function g^{-1} is roughly linear in ϵ for $\epsilon \leq 0.7$, the excess risk roughly decreases as $O(1/k)$. When the "worst-case condition number" γ_{θ^*} of the Fisher information matrices is smaller (well-conditioned), the error decreases. Intuitively, this means that there is no direction to vary θ^* in which the output distribution will sharply change. As a consequence, the concepts θ that are not distinguishable from the prompt concept θ^* parameterize distributions that produce similar outputs to the prompt concept and thus achieve a small error.

Varying-length test examples. In the setting where the length of x_{test} is random (uniformly from 2 to k), we can give a similar error guarantee without continuity.

Theorem 3. *Let the set of θ which does not satisfy Equation 14 in Condition 1 to be $\mathcal{B}$ and assume $\pi_{prompt} = \pi^{\theta^*}$. Let the length of the test example x_{test} be uniformly distributed between 2 and k, for $k \geq 2$. As $n \to \infty$, the 0-1 risk of the in-context learning predictor f_n is bounded as*

$$\lim_{n\to\infty} L_{0\text{-}1}(f_n) \leq \inf_f L_{0\text{-}1}(f) + g^{-1}\left(\frac{\sup_{\theta\in\mathcal{B}}(\epsilon^{\theta}_{start} + \epsilon^{\theta}_{delim})}{k-1}\right). \tag{18}$$

Instead of measuring only the error at the k-th token, we average the prediction error on the 2nd to k-th tokens. However, we leave bridging the mismatch between training examples, which are consistently length k, and test examples, which have random length, to future work.

4 Simulations

We generate the GINC dataset and show that Transformers (Vaswani et al., 2017) and LSTMs (Hochreiter & Schmidhuber, 1997) trained on GINC exhibit in-context learning. In the theory, we assumed that the pretrained LM fits the pretraining distribution exactly. Here, we pretrain LMs to approximate the pretraining distribution, showing that the in-context learning properties of the pretraining distribution transfer to the LM.

We construct the GINC dataset to satisfy our theoretical assumptions (see Appendix E.1). For pretraining, we define a uniform mixture of HMMs over a family Θ of 5 concepts to generate 1000 pretraining documents with $\sim$10 million tokens total. For prompting, we generate prompts with 0 to 64 training examples and example lengths $k \in \{3,5,8,10\}$ (2500 prompts for each setting). The target token y_{test} is taken to be the most likely output $\text{argmax}_y p_{\text{prompt}}(y|x_{\text{test}})$ instead of sampling so that the intrinsic error is 0.

We train GPT-2-based Transformers (Radford et al., 2019) and LSTMs on three versions of the GINC dataset with vocabulary sizes 50, 100, and 150, then evaluate the in-context accuracy (see Appendix E.2, E.3). We average all results over 5 pretraining runs. Figure 3 shows that for both Transformer and LSTMs, in-context accuracy improves as the number of prompt examples n and the example length k increase, verifying our theory.

Ablations on the latent concept structure. We ablate the role of the mixture-of-concepts structure in GINC. In Figure 4 (left), we pretrain a 4 layer Transformer on data with only one concept (removing the prior) from Θ, resulting in flat in-context learning curves. Figure 4 (middle) shows that pretraining on random pretraining data, which contains all possible token transitions, in-context learning also fails. Therefore, the mixture-of-concepts structure is important and simply seeing diverse token transitions does not enable in-context learning.

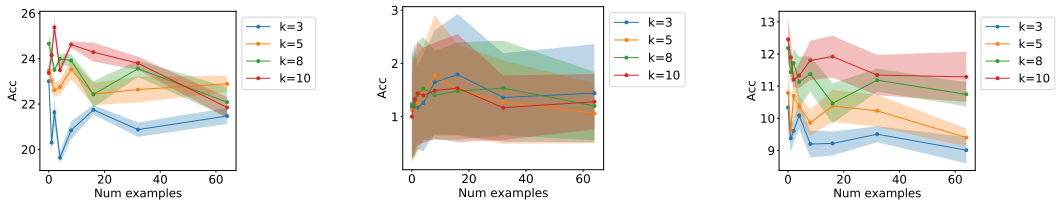

Figure 4: Ablation studies for 4 layer Transformers on the GINC dataset with vocab size 50. **(Left)** When pretrained with only one concept, in-context learning fails. **(Middle)** When the pretraining data has random transitions, the model sees all token transitions but in-context learning fails. **(Right)** When prompts are from random unseen concepts, in-context learning fails to extrapolate.

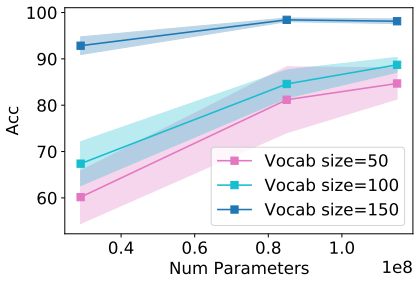

Figure 5: In-context accuracy (95% intervals) of Transformers improves as model size increases on the GINC dataset for vocabulary sizes 50, 100, and 150.

Model	# Params	Train loss (pretraining)	Val loss (pretraining)	In-context Acc
Vocab size 50, $k=10, n=64$				
Transformer (4 layer)	29M	1.49	1.50	60.2 ± 5.7
Transformer (12 layer)	85M	1.31	1.33	81.2 ± 7.1
Transformer (16 layer)	115M	1.31	1.33	84.7 ± 3.4
LSTM	28M	1.31	1.35	95.8 ± 1.11
Vocab size 100, $k=10, n=64$				
Transformer (4 layer)	29M	1.58	1.59	67.4 ± 4.7
Transformer (12 layer)	85M	1.40	1.42	84.6 ± 3.0
Transformer (16 layer)	115M	1.41	1.43	88.7 ± 1.6
LSTM	28M	1.43	1.44	95.8 ± 1.54
Vocab size 150, $k=10, n=64$				
Transformer (4 layer)	29M	1.44	1.45	92.8 ± 1.9
Transformer (12 layer)	85M	1.27	1.28	98.4 ± 0.4
Transformer (16 layer)	115M	1.27	1.28	98.1 ± 0.5
LSTM	28M	1.26	1.31	99.2 ± 1.06

Figure 6: In-context accuracies (95% intervals) on GINC with vocab sizes (50, 100, 150) for Transformers and LSTMs. Accuracy improves with scale even though the pretraining loss may be the same.

Extrapolation to unseen concepts. Full generative control of GINC allows for experimentation with latent variables in the pretraining distribution. For example, in large-scale datasets, it is difficult to test whether a concept or task is in the pretraining data. We test this in GINC by testing the in-context accuracy of a 4 layer Transformer on prompts generated from 5 random concepts that are not in the pretraining family of concepts. Figure 4 (right) shows that in-context learning also fails for these novel concepts.

Effect of model size and architecture. Figure 5 shows that increasing the size of the Transformer (4, 12, 16 layers) steadily increases the in-context accuracy, corroborating the results of Brown et al. (2020). Table 6 shows that even though larger Transformers may have the same pretraining loss (e.g., 12 and 16 layer Transformers both get 1.33 validation loss for vocab size 50), the in-context accuracy still improves (81% to 85% from 12 to 16 layers), suggesting that larger models can improve in-context learning beyond improving pretraining perplexity. This may be related to phenomena from overparameterization and overtraining (Zhang et al., 2017; Power et al., 2021). Finally, the model architecture also plays a role — LSTMs consistently outperform Transformers on GINC despite having fewer parameters, perhaps due to the similarity between HMMs and LSTMs. We leave analysis of the effect of model scaling and model architecture as open questions.

Sensitivity to example ordering. In Figure 7 (left), we test the sensitivity of in-context accuracy on GINC to the ordering of the prompt examples, following Zhao et al. (2021). For this experiment, we consider prompts generated from a single concept and prompt start distribution. We sample 10 different sets (leading to 10 training set IDs) of 4 examples and generate all 24 possible permutations for each example set. We consider the in-context accuracy of the 4 layer Transformer trained on GINC with vocabulary size 50. Similarly to the behavior of GPT-3 (Zhao et al., 2021), there is a significant variation (10–40% difference) between permutations of the same set of examples.

Zero-shot is sometimes better than few-shot. In some settings in GINC, we find that zero-shot performance can be better than few-shot performance. This mirrors GPT-3 on some datasets (e.g., LAMBADA, HellaSwag, PhysicalQA, RACE-m, CoQA/SAT analogies for smaller models (Brown et al., 2020)). This occurs especially when the transition probabilities in GINC are lower entropy (controlled via a temperature parameter). For this experiment, we consider GINC with transition matrix temperature parameter 0.01 (instead of 0.1), 12 concepts, and vocabulary size 100. Figure 7 (right) shows that

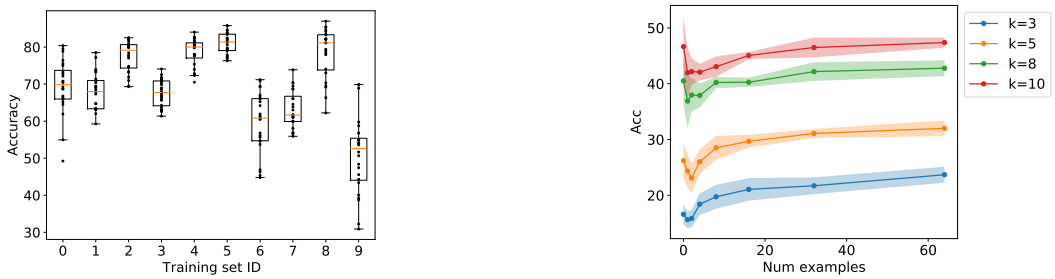

Figure 7: **(Left)** In-context accuracy varies widely with example ordering. Each training ID refers to a set of training examples. Each dot refers to the in-context learning accuracy of one permutation of the training examples for that particular training ID. **(Right)** Zero-shot performance can be higher than one/few-shot performance in some settings in GINC, mirroring the behavior of GPT-3 on some datasets such as LAMBADA (Brown et al., 2020). The few-shot setting introduces the distracting prompt structure, which can initially lower accuracy.

here, few-shot accuracy is initially worse than zero-shot accuracy, but can recover with more examples. We hypothesize that the distracting prompt structure initially decreases the accuracy in this setting.

5 Discussion and Related Work

Learning via Bayesian inference and extrapolation. The canonical Bernstein-von Mises theorem (van der Vaart, 1998) does not apply for in-context learning since the prompt examples are not independent under the pretraining distribution. Gunst & Shcherbakova (2008) show a Bernstein-von Mises-type result for observations from an HMM, but do not handle observations from a different distribution. Future directions include more precise asymptotic results about the posterior distribution and results under misspecification/extrapolation (Kleijn & van der Vaart, 2012).

Topic models and HMMs. Topic models such as LDA (Blei et al., 2003) also have document-level latent variables, but learning is typically relies on algorithms such as EM (Dempster et al., 1977), variational inference (Jordan et al., 1999), or MCMC (Metropolis et al., 1953; Hastings, 1970). We focus on learning as a natural result of Bayesian inference without an explicit inference algorithm. Wei et al. (2021a) also use an HMM model in their pretraining analysis. However, they analyze how pre-trained representations learned with masked LMs (Devlin et al., 2019; Liu et al., 2019; Lewis et al., 2020; Clark et al., 2020) can improve optimization-based downstream learning (Li & Liang, 2021; Lester et al., 2021) rather than in-context learning.

Bridging the mismatch between pretraining and prompting. Prior works support our theoretical intuitions that reducing the prompt distribution mismatch would improve in-context learning. Finetuning LMs on text with a prompting format improves its zero-shot performance (Wei et al., 2021b; Sanh et al., 2021) and optimizing prompt templates improves few-shot finetuning (Jiang et al., 2020; Schick & Schütze, 2021; Shin et al., 2020; Gao et al., 2021). Zhao et al. (2021); Holtzman et al. (2021) improve in-context accuracy via calibration or renormalization, a form of adaptation to the prompt distribution.

Meta-learning. Meta-learning methods can also train a sequence model to learn from examples (Ravi & Larochelle, 2017). However, meta-learning models are trained to learn, while in-context learning emerges from LM pretraining.

Studying large-scale phenomena and making predictions. We can study in-context learning, a large scale phenomenon, at a small scale in GINC because the complexity of the pretraining distribution (HMM hidden state size, number of latent concepts) is small, such that the data and models are relatively larger. Since GINC is synthetic, we can also control the latent data properties (e.g., unseen concepts) to make predictions about large LMs while working at a small scale.

6 Conclusion

We cast in-context learning as implicit Bayesian inference, where the pretrained LM implicitly infers a concept when making a prediction. We show that in-context learning occurs when the pre-training distribution is a mixture of HMMs. Our work provides a first step towards understanding in-context learning, which we hope will provide insight for improving pretraining and prompting.

ACKNOWLEDGEMENTS

We thank Tianyi Zhang, Frieda Rong, Lisa Li, Colin Wei, Shibani Santurkar, Tri Dao, and Ananya Kumar for helpful discussions and feedback. SMX is supported by an NDSEG Fellowship. The work is partially supported by an Open Philanthropy Project Award, SDSI, and SAIL at Stanford University. TM acknowledges support of Google Faculty Award, NSF IIS 2045685, the Sloan Fellowship, and JD.com. Toyota Research Institute provided funds to support this work.

REFERENCES

Leonard E Baum and Ted Petrie. Statistical inference for probabilistic functions of finite state markov chains. *The annals of mathematical statistics*, 37(6):1554–1563, 1966.

D. Blei, Andrew Ng, and M. I. Jordan. Latent Dirichlet allocation. *Journal of Machine Learning Research (JMLR)*, 3:993–1022, 2003.

Tom B. Brown, Benjamin Mann, Nick Ryder, Melanie Subbiah, Jared Kaplan, Prafulla Dhariwal, Arvind Neelakantan, Pranav Shyam, Girish Sastry, Amanda Askell, Sandhini Agarwal, Ariel Herbert-Voss, Gretchen Krueger, Tom Henighan, Rewon Child, Aditya Ramesh, Daniel M. Ziegler, Jeffrey Wu, Clemens Winter, Christopher Hesse, Mark Chen, Eric Sigler, Mateusz Litwin, Scott Gray, Benjamin Chess, Jack Clark, Christopher Berner, Sam McCandlish, Alec Radford, Ilya Sutskever, and Dario Amodei. Language models are few-shot learners. *arXiv preprint arXiv:2005.14165*, 2020.

Kevin Clark, Minh-Thang Luong, Quoc V. Le, and Christopher D. Manning. Electra: Pre-training text encoders as discriminators rather than generators. In *International Conference on Learning Representations (ICLR)*, 2020.

A. P. Dempster, Laird N. M., and Rubin D. B. Maximum likelihood from incomplete data via the EM algorithm. *Journal of the Royal Statistical Society: Series B*, 39(1):1–38, 1977.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. BERT: Pre-training of deep bidirectional transformers for language understanding. In *Association for Computational Linguistics (ACL)*, pp. 4171–4186, 2019.

Tianyu Gao, Adam Fisch, and Danqi Chen. Making pre-trained language models better few-shot learners. *arXiv*, 2021.

Zoubin Ghahramani and Michael Jordan. Factorial hidden Markov models. *Machine Learning*, 29: 245–273, 1997.

Amit Gruber, Yair Weiss, and Michal Rosen-Zvi. Hidden topic Markov models. In *Artificial Intelligence and Statistics (AISTATS)*, 2007.

M. Gunst and O. Shcherbakova. Asymptotic behavior of Bayes estimators for hidden Markov models with application to ion channels. *Mathematical Methods of Statistics*, 17, 2008.

Keith W. Hastings. Monte Carlo sampling methods using Markov chains and their applications. *Biometrika*, 57(1):97–109, 1970.

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

Ari Holtzman, Jan Buys, Li Du, Maxwell Forbes, and Yejin Choi. The curious case of neural text degeneration. In *International Conference on Learning Representations (ICLR)*, 2020.

Ari Holtzman, Peter West, Vered Shwartz, Yejin Choi, and Luke Zettlemoyer. Surface form competition: Why the highest probability answer isn't always right, 2021.

Zhengbao Jiang, Frank F Xu, Jun Araki, and Graham Neubig. How can we know what language models know? In *Association for Computational Linguistics (ACL)*, 2020.

Michael I. Jordan, Zoubin Ghahramani, Tommi S. Jaakkola, and Lawrence K. Saul. An introduction to variational methods for graphical models. *Machine Learning*, 37:183–233, 1999.

Mandar Joshi, Eunsol Choi, Daniel Weld, and Luke Zettlemoyer. TriviaQA: A large scale distantly supervised challenge dataset for reading comprehension. In *Association for Computational Linguistics (ACL)*, 2017.

Diederik Kingma and Jimmy Ba. Adam: A method for stochastic optimization. In *International Conference on Learning Representations (ICLR)*, 2015.

B.J.K. Kleijn and A.W. van der Vaart. The Bernstein-von mises theorem under misspecification. *Electronic Journal of Statistics*, 6, 2012.

Brian Lester, Rami Al-Rfou, and Noah Constant. The power of scale for parameter-efficient prompt tuning. *arXiv preprint arXiv:2104.08691*, 2021.

Mike Lewis, Yinhan Liu, Naman Goyal, Marjan Ghazvininejad, Abdelrahman Mohamed, Omer Levy, Ves Stoyanov, and Luke Zettlemoyer. Bart: Denoising sequence-to-sequence pre-training for natural language generation, translation, and comprehension. In *Association for Computational Linguistics (ACL)*, 2020.

Xiang Lisa Li and Percy Liang. Prefix-tuning: Optimizing continuous prompts for generation. In *Association for Computational Linguistics (ACL)*, 2021.

Opher Lieber, Or Sharir, Barak Lenz, and Yoav Shoham. Jurassic-1: Technical details and evaluation. Technical report, AI21 Labs, August 2021.

Yinhan Liu, Myle Ott, Naman Goyal, Jingfei Du, Mandar Joshi, Danqi Chen, Omer Levy, Mike Lewis, Luke Zettlemoyer, and Veselin Stoyanov. RoBERTa: A robustly optimized BERT pretraining approach. *arXiv preprint arXiv:1907.11692*, 2019.

Ilya Loshchilov and Frank Hutter. Decoupled weight decay regularization. In *International Conference on Learning Representations (ICLR)*, 2019.

Nicholas Metropolis, Arianna W. Rosenbluth, Marshall N. Rosenbluth, Augusta H. Teller, and Edward Teller. Equation of state calculations by fast computing machines. *The journal of chemical physics*, 21(6):1087–1092, 1953.

Denis Paperno, German Kruszewski, Angeliki Lazaridou, Quan Ngoc Pham, Raffaella Bernardi, Sandro Pezzelle, Marco Baroni, Gemma Boleda, and Raquel Fernandez. The LAMBADA dataset: Word prediction requiring a broad discourse context. In *Association for Computational Linguistics (ACL)*, 2016.

Alethea Power, Yuri Burda, Harri Edwards, Igor Babuschkin, and Vedant Misra. Grokking: Generalization beyond overfitting on small algorithmic datasets. In *ICLR MATH AI Workshop*, 2021.

Alec Radford, Jeffrey Wu, Rewon Child, David Luan, Dario Amodei, and Ilya Sutskever. Language models are unsupervised multitask learners. *OpenAI Blog*, 1(8), 2019.

Sachin Ravi and Hugo Larochelle. Optimization as a model for few-shot learning. In *International Conference on Learning Representations (ICLR)*, 2017.

Victor Sanh, Albert Webson, Colin Raffel, Stephen H. Bach, Lintang Sutawika, Zaid Alyafeai, Antoine Chaffin, Arnaud Stiegler, Teven Le Scao, Arun Raja, Manan Dey, M Saiful Bari, Canwen Xu, Urmish Thakker, Shanya Sharma Sharma, Eliza Szczechla, Taewoon Kim, Gunjan Chhablani, Nihal Nayak, Debajyoti Datta, Jonathan Chang, Mike Tian-Jian Jiang, Han Wang, Matteo Manica, Sheng Shen, Zheng Xin Yong, Harshit Pandey, Rachel Bawden, Thomas Wang, Trishala Neeraj, Jos Rozen, Abheesht Sharma, Andrea Santilli, Thibault Fevry, Jason Alan Fries, Ryan Teehan, Stella Biderman, Leo Gao, Tali Bers, Thomas Wolf, and Alexander M. Rush. Multitask prompted training enables zero-shot task generalization, 2021.

Timo Schick and Hinrich Schütze. Exploiting cloze questions for few shot text classification and natural language inference. In *European Association for Computational Linguistics (EACL)*, 2021.

Taylor Shin, Yasaman Razeghi, Robert L Logan IV, Eric Wallace, and Sameer Singh. Eliciting knowledge from language models using automatically generated prompts. In *Empirical Methods in Natural Language Processing (EMNLP)*, 2020.

Ingo Steinwart. How to compare different loss functions and their risks. *Constructive Approximation*, 26, 2007.

A. W. van der Vaart. *Asymptotic statistics*. Cambridge University Press, 1998.

Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Lukasz Kaiser, and Illia Polosukhin. Attention is all you need. *arXiv preprint arXiv:1706.03762*, 2017.

Ben Wang and Aran Komatsuzaki. GPT-J-6B: A 6 Billion Parameter Autoregressive Language Model. `https://github.com/kingoflolz/mesh-transformer-jax`, May 2021.

Colin Wei, Sang Michael Xie, and Tengyu Ma. Why do pretrained language models help in downstream tasks? an analysis of head and prompt tuning. *arXiv*, 2021a.

Jason Wei, Maarten Bosma, Vincent Y. Zhao, Kelvin Guu, Adams Wei Yu, Brian Lester, Nan Du, Andrew M. Dai, and Quoc V. Le. Finetuned language models are zero-shot learners. *arXiv*, 2021b.

Thomas Wolf, Lysandre Debut, Victor Sanh, Julien Chaumond, Clement Delangue, Anthony Moi, Pierric Cistac, Tim Rault, R'emi Louf, Morgan Funtowicz, and Jamie Brew. HuggingFace's transformers: State-of-the-art natural language processing. *arXiv preprint arXiv:1910.03771*, 2019.

Chiyuan Zhang, Samy Bengio, Moritz Hardt, Benjamin Recht, and Oriol Vinyals. Understanding deep learning requires rethinking generalization. In *International Conference on Learning Representations (ICLR)*, 2017.

Tony Z. Zhao, Eric Wallace, Shi Feng, Dan Klein, and Sameer Singh. Calibrate before use: Improving few-shot performance of language models. In *International Conference on Machine Learning (ICML)*, 2021.

Bernardo Ávila Pires and Csaba Szepesvári. Multiclass classification calibration functions. *arXiv*, 2016.