Investigating the Zone of Proximal Development of Language Models for In-Context Learning

Peng Cui   **Mrinmaya Sachan**
Department of Computer Science, ETH Zürich
peng.cui@inf.ethz.ch

Abstract

In this paper, we introduce a learning analytics framework to analyze the in-context learning (ICL) behavior of large language models (LLMs) through the lens of the Zone of Proximal Development (ZPD), an established theory in educational psychology. ZPD delineates the the space between what a learner is capable of doing unsupported and what the learner cannot do even with support. We adapt this concept to ICL, measuring the ZPD of LLMs based on model performance on individual examples before and after ICL. Furthermore, we propose an item response theory (IRT) model to predict the distribution of zones for LLMs. Our findings reveal a series of intricate and multifaceted behaviors of ICL, providing new insights into understanding and leveraging this technique. Finally, we demonstrate how our framework can enhance LLM in both inference and fine-tuning scenarios: (1) By predicting a model's zone of proximal development, we selectively apply ICL to queries that are most likely to benefit from demonstrations, achieving a better balance between inference cost and performance; (2) We propose a human-like curriculum for fine-tuning, which prioritizes examples within the model's ZPD. The curriculum results in improved performance, and we explain its effectiveness through an analysis of the training dynamics of LLMs.[1]

1 Introduction

Human learning is a dynamic and progressive process where learners integrate new information into their knowledge base through interactions with the environment (Piaget, 1977). Research in education and learning sciences has extensively explored what makes learning most effective and efficient. Among them, the Zone of Proximal Development (ZPD) emphasizes the alignment between the learner's capability and the problem's difficulty (Vygotsky, 1978). Specifically, ZPD refers to the range of problems that a learner can solve with appropriate scaffolding but cannot tackle independently. This concept is essential in education as it identifies knowledge that is valuable for learning, feasible to acquire, and not yet mastered. Therefore, learning within ZPD is believed to foster more effective cognitive development (Chaiklin et al., 2003; Tharp and Gallimore, 1991).

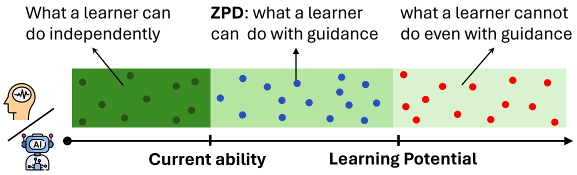
What a learner can do independently
ZPD: what a learner can do with guidance
what a learner cannot do even with guidance
Current ability
Learning Potential

Figure 1: We conceptualize an LLM's Zone of Proximal Development (ZPD) for ICL as the set of queries on which the model's performance can be improved with demonstrations. We introduce a framework to measure and predict this zone and explore its applications.

In this paper, we propose a learning analytics framework to study the *learning behavior* of language models through the lens of ZPD. In particular, we focus on in-context learning (ICL), an emerging ability of LLMs that allows them to learn from a few demonstrations (Brown et al., 2020; Wei et al., 2022). Previous studies have primarily focused on strategies for demonstration optimization (Liu et al., 2021; Qin et al., 2023; Rubin et al., 2021; Ye et al., 2023). However, even with high-quality demonstrations, the performance of ICL still varies significantly across tasks and data (Srivastava et al., 2024). This variability calls for a more comprehensive examination of the *inherent* in-context learnability of LLMs on individual queries.

We first formalize the concept of ZPD in ICL. Drawing on the parallel between ICL and human learning from worked examples, we view LLMs as learners and in-context demonstrations as a form of scaffolding. Then, based on the model's prior

[1] Code is available at https://github.com/nlpcui/llm-zpd

Findings of the Association for Computational Linguistics: NAACL 2025, pages 6485–6498
April 29 - May 4, 2025 ©2025 Association for Computational Linguistics

knowledge and its response to ICL, a query set can be divided into three **zones** ($\mathcal{Z}$): (1) The first zone, denoted as ●$\mathcal{Z}_{\checkmark}$, consists of queries that can be solved by the model via direct prompting, representing the model's prior knowledge; (2) The second zone, denoted as ●$\mathcal{Z}_{\text{✗}\rightarrow\checkmark}$, includes queries that can be solved by the model only with ICL, representing the model's ZPD; and (3) The third zone, denoted as ●$\mathcal{Z}_{\text{✗}\rightarrow\text{✗}}$, contains queries that the model cannot solve even with ICL, representing the knowledge beyond the model's reach. Figure 1 illustrates this conceptualization. This categorization provides a granular look at the model's capability, limitations, and interaction with specific interventions.

We begin by measuring the task-specific zones of various models (§ 3). Since the ICL performance is sensitive to the choice of demonstrations and the ground-truth demonstrations are not available, it is non-trivial to determine whether a problem can potentially benefit from ICL. To address this, we employ a greedy algorithm to construct *Oracle* demonstrations for each query and use them to approximate the zone distribution empirically. Then, we propose to predict the zones of unseen queries using the item Response theory (IRT; Santor and Ramsay (1998)), which jointly captures the latent traits of the model and query (e.g., ability, difficulty). In particular, we introduce a variant of IRT that further takes into account the model's in-context learnability to capture the performance changes with or without ICL (§ 4). We find that the ICL behavior of LLMs is generally predictable even without demonstration information, although the degree of predictability varies across different datasets and tasks.

Finally, we showcase how our framework enhances LLMs in both inference and fine-tuning scenarios (§ 5.3). For inference, we propose a selective ICL strategy, which first predicts the zone of input queries and then applies ICL only to queries that are most likely to benefit from ICL (i.e., within the model's ZPD ●$\mathcal{Z}_{\text{✗}\rightarrow\checkmark}$). Experimental results show this approach achieves competitive or even better performance with reduced inference cost. For fine-tuning, we propose a ZPD-based curriculum that prioritizes challenging yet learnable training examples. We find such a curriculum improves fine-tuning outcomes. Upon further analysis of training dynamics, we find LLMs exhibit *consistent learnability* under both ICL and fine-tuning settings. This consistency explains the effectiveness of our ZPD-based curriculum and suggests potential connections between these two learning paradigms.

In summary, our contributions are threefold:

- We conceptualize the ZPD framework for LLMs, which provides a new perspective on analyzing their ICL behavior.
- We introduce a novel IRT variant that captures LLMs' in-context learnability and predicts their performance with or without ICL.
- We showcase two applications of our framework: a selective ICL strategy and a ZPD-based curriculum, demonstrating its potential to enhance both LLM training and inference.

2 Related Work

In-Context Learning (Brown et al., 2020) has become a popular paradigm for enhancing the capabilities of LLMs across a wide range of tasks. Previous work has extensively focused on optimizing demonstrations, particularly through the selection (Liu et al., 2022; Rubin et al., 2022; Li et al., 2023) and ranking (Zhao et al., 2021a; Lu et al., 2022) of in-context examples. In this paper, we shift the focus from demonstration optimization to the LLM and the target query themselves, highlighting the inherent in-context learnability of LLMs on individual queries. Our study complements these works, contributing to a holistic understanding of what makes ICL (un)successful. Another line of research explores how the ICL capability emerges and functions, with various hypotheses proposed, such as task recognition (Xie et al., 2022; Wang et al., 2024), composition (Li et al., 2024), meta-gradient learning (Garg et al., 2022; Akyürek et al., 2023). This paper also aims to understand ICL but from an empirical perspective by collecting, analyzing, and predicting ICL behaviors.

Adoption of IRT in NLP. IRT is a set of statistical models used in educational assessments to measure the latent abilities of individuals through standardized testing (Lord and Novick, 2008; Santor and Ramsay, 1998). In recent years, it has become increasingly popular in NLP. Byrd and Srivastava (2022) uses IRT to estimate question difficulty and model skills. Gor et al. (2024) proposes a content-aware and identifiable IRT to analyze human-AI complementarity. Polo et al. (2024) argues for the adoption of IRT to build benchmarks for efficient evaluation. In this work, we use IRT to predict LLM in-context learnability on individual queries (conceptualized as ZPD) by capturing the behavior of LLMs before and after (in-context) learning.

Curriculum Learning (Bengio et al., 2009) is the approach that organizes the training examples such that the model converges faster and better, which has been successfully applied in various NLP tasks (Tay et al., 2019; Platanios et al., 2019; Sachan and Xing, 2016). Typically, curriculum learning algorithms organize training examples in increasing order of difficulty. Conversely, there is another line of research that works in the opposite way to start with hard examples, namely Hard Example Mining (Shrivastava et al., 2016; Jin et al., 2018). In this paper, we propose a ZPD-based curriculum that strikes a middle point between the two techniques: prioritizing training examples that are challenging and yet learnable (i.e., within the model's ZPD). Similar strategies have been proven effective in various scenarios (Mindermann et al., 2022). However, this paper proposes a new framework for discovering such desired examples, which can be incorporated into existing approaches.

3 Measuring ZPD of LLMs

3.1 Preliminaries

Let $\mathcal{D} = \{(x_1, y_1), ..., (x_n, y_n)\}$ be a dataset where x_i is a query and y_i is the ground-truth answer. We define the ZPD (🔵$\mathcal{Z}_{\text{✗}\rightarrow\text{✓}}$) of a model $\mathcal{M}$ on D as a subset of examples on which the model's performance can be improved through a learning trial. In this study, we focus on the ICL setting and measure learning outcomes by comparing the model's performance with and without ICL. Specifically, let $c = \{(x_1, y_1)...(x_k, y_k) | x_j \in \mathcal{D}\}$ be a set of demonstrations for x ($x \notin c$), we define 🔵$\mathcal{Z}_{\text{✗}\rightarrow\text{✓}}$ as:

$$🔵\mathcal{Z}_{\text{✗}\rightarrow\text{✓}} \triangleq \{x | \mathcal{F}(y^{\varnothing}) < \tau, \mathcal{F}(y^{c}) > \tau\}, \quad (1)$$

where $\mathcal{F}$ is a scoring function and τ is a threshold deciding whether the predicted answer is acceptable. $y^{\varnothing}$ and y^{c} represent the model's output with direct prompting and with in-context demonstrations, respectively:

$$y^{\varnothing} = \mathcal{M}(\mathcal{T}(x)), y^{c} = \mathcal{M}(\mathcal{T}(c_1) \oplus ... \oplus \mathcal{T}(x)). \quad (2)$$

where $\mathcal{T}$ is a template function and $\oplus$ denotes string concatenation. Due to the potential interference between instruction and demonstration (Srivastava et al., 2024), we adopted a simple prompt template with minimal instruction to focus on the effect of demonstration (See Appendix Table 4).

Similarly, we can define the other two subsets as follows:

$$🟢\mathcal{Z}_{\text{✓}} \triangleq \{x | \mathcal{F}(y^{\varnothing}) > \tau\}, \quad (3)$$

$$🔴\mathcal{Z}_{\text{✗}\rightarrow\text{✗}} \triangleq \{x | \mathcal{F}(y^{\varnothing}) < \tau, \mathcal{F}(y^{c}) < \tau\}, \quad (4)$$

representing queries that can be solved by $\mathcal{M}$ with direct prompting, and queries that cannot be solved even with ICL.

This formalization is flexible and can be applied to other settings. For example, future work could replace ICL with other prompting strategies or analyze fine-tuning behaviors by examining the performance across different epochs.

3.2 Approximating 🔵$\mathcal{Z}_{\text{✗}\rightarrow\text{✓}}$ and 🔴$\mathcal{Z}_{\text{✗}\rightarrow\text{✗}}$

While 🟢$\mathcal{Z}_{\text{✓}}$ is deterministic from the model's base performance $\{y_1^{\varnothing}, y_2^{\varnothing}, ...\}$, 🔵$\mathcal{Z}_{\text{✗}\rightarrow\text{✓}}$ and 🔴$\mathcal{Z}_{\text{✗}\rightarrow\text{✗}}$ depend on the choice of demonstrations c. In this paper, we aim to investigate the ideal ICL behavior of LLMs with *optimal* demonstrations. This is because our goal is to understand the model's *inherent* in-context learnability on individual queries rather than the behavior of a specific ICL strategy. Since optimal demonstrations for each query are unavailable, precise measurements of 🔵$\mathcal{Z}_{\text{✗}\rightarrow\text{✓}}$ and 🔴$\mathcal{Z}_{\text{✗}\rightarrow\text{✗}}$ are infeasible. To address this, we first create *Oracle* demonstrations—the best demonstrations achievable in a practical setting (with a limited demonstration pool and restricted computation resources). Then, we use them to approximate 🔵$\mathcal{Z}_{\text{✗}\rightarrow\text{✓}}$ and 🔴$\mathcal{Z}_{\text{✗}\rightarrow\text{✗}}$.

In concrete, we adopt a retrieve and rank method to construct Oracle demonstrations. Firstly, we retrieve a candidate set $\mathcal{C}$ for each query. The common belief is that demonstrations that are similar to the query are most likely to enhance performance (Liu et al., 2022). Following previous work (Rubin et al., 2022), we employ BM25 (Robertson et al., 2009), a sparse retriever based on surface features, and SBERT (Reimers, 2019), which is based on dense sentence encoding. For the two retrievers, we calculate similarities based on both the (x, y) pair and the ground-truth answer y only, resulting in $2 \times 2 \times K$ candidates. However, similarity may not be the only criterion for demonstration selection. To further enrich the candidate set and recall effective but dissimilar demonstrations, we randomly sample K candidates from the bottom 50 percentile of the retrieving results, doubling the candidate size.

Next, we select Oracle demonstrations c using a greedy scoring approach:

$$c_i = \underset{\mathcal{C}\setminus\{c_1,..,c_{i-1}\}}{\text{argmax}} \ \text{Prob}_{\mathcal{M}}(y|c_1 \oplus ...c_i \oplus x), \quad (5)$$

where c_i is the i^{th} selected demonstration and $\text{Prob}_{\mathcal{M}}(\cdot)$ is the probability from the model $\mathcal{M}$. In other words, we greedily choose demonstrations that can maximize the likelihood of the ground-truth answer. With these demonstrations, the resulting ●$\mathcal{Z}_{\text{✗}\rightarrow\text{✓}}$ is a subset of the actual ZPD while ●$\mathcal{Z}_{\text{✗}\rightarrow\text{✗}}$ is a superset of the actual one. In the rest of the paper, we use ●$\mathcal{Z}_{\text{✗}\rightarrow\text{✓}}$ and ●$\mathcal{Z}_{\text{✗}\rightarrow\text{✗}}$ to denote for the approximated zones unless otherwise specified.

4 Zone Prediction

In this section, we attempt to build a model to predict an LLM's zone distribution on unseen queries. Essentially, the goal is to predict the model's performance, i.e., whether it can solve a query directly (●$\mathcal{Z}_{\text{✓}}$) or with ICL (●$\mathcal{Z}_{\text{✗}\rightarrow\text{✓}}$), or not at all (●$\mathcal{Z}_{\text{✗}\rightarrow\text{✗}}$). We propose a novel variant of item response theory (IRT) to capture the latent traits of the LLM and the queries. A graphic view of our model is shown in Figure 2.

4.1 Background of IRT

IRT is a statistical model that predicts the probability of individual respondents correctly answering a set of queries (or items). In this work, we take a collection of LLMs $\{\mathcal{M}_1, \mathcal{M}_2, ..., \mathcal{M}_m\}$ as respondents. The basic 1 Parameter Logistic (1PL) IRT is defined as:

$$P(r_{i,j} = 1|\mathcal{M}_i, x_j) = \sigma(\theta_i - d_j), \quad (6)$$

where $r_{i,j}$ is the binary correctness label of $\mathcal{M}$'s prediction on x_i. σ is the sigmoid function. θ_i and d_j are latent variables (scalars) to be estimated, representing the ability of the ith model $\mathcal{M}_i$ and the difficulty of the jth query x_j. Simply put, IRT predicts the correctness label based on the gap between model ability and query difficulty.

The 1PL IRT assumes the monotonic relationship between item difficulty and respondent ability.

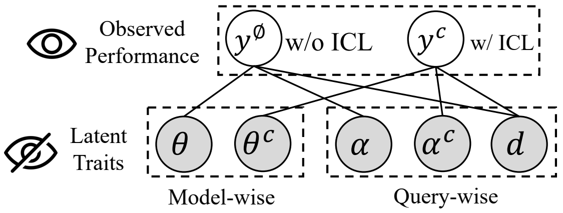

Figure 2: We assume that a model's performance on a given query, y^c (with ICL) or $y^{\varnothing}$ (without ICL), is determined by latent traits (shadowed nodes, bottom) of both the model and the query, including the model's skill $\boldsymbol{\theta}$, ICL skill $\boldsymbol{\theta}^c$, the query's discrimination $\boldsymbol{\alpha}$, ICL discrimination $\boldsymbol{\alpha}^c$, and overall difficulty d.

To relax this, we employ the multi-dimensional IRT (MIRT, Reckase (2006)), which is defined as:

$$P(r_{i,j} = 1|\mathcal{M}_i, x_j) = \sigma(\boldsymbol{\theta}_i^{\mathrm{T}}\boldsymbol{\alpha}_j - d_j), \quad (7)$$

where the model's ability is represented as a *skill vector* $\boldsymbol{\theta}_j \in \mathbb{R}^{\mathrm{H}}$. Correspondingly, an item-wise *discrimination vector* $\boldsymbol{\alpha}_i \in \mathbb{R}^{\mathrm{H}}$ is introduced to represent its latent traits. A closer alignment between $\boldsymbol{\theta}_i$ and $\boldsymbol{\alpha}_j$ indicates a higher likelihood of a correct response.

The training objective of IRT is defined as:

$$\mathcal{L}_{\text{IRT}} = \sum_{i=1}^{\mathrm{M}}\sum_{j=1}^{\mathrm{N}} \text{CE}(P(r_{i,j}), y_j), \quad (8)$$

where $\text{CE}(\cdot)$ stands for the cross-entropy loss between predicted probability and the groud-truth label.

4.2 Content-Aware MIRT

A limitation of MIRT is that it relies on the response data to infer item traits $\boldsymbol{\alpha}_i$. Therefore, it cannot generalize to unseen queries during inference. To overcome this limitation, we use a lightweight neural network to parameterize item traits based on their text features. Specifically, for a given query x_j, we first use an embedding model to obtain its representation $\boldsymbol{e}_j$. Then, we compute its traits by:

$$d_j = f(\mathbf{W}_{\mathrm{d}}\boldsymbol{e}_j + \mathbf{b}_{\mathrm{d}}); \boldsymbol{\alpha}_j = f(\mathbf{W}_{\alpha}\boldsymbol{e}_j + \mathbf{b}_{\alpha}) \quad (9)$$

where $\mathbf{W}_{\mathrm{d}}, \mathbf{W}_{\alpha}, \mathbf{b}_{\mathrm{d}}, \mathbf{b}_{\alpha}$ are learnable weights, trained together with the IRT model, and f is the Relu function.

4.3 Adapting MIRT to Learning Dynamics

While the above model can predict the model's performance on an unseen query, it cannot predict one query's correctness label under two settings and thus cannot predict three zones simultaneously. We propose a variant that incorporates the dynamics of ICL. Concretely, we introduce an additional *ICL skill vector* $\boldsymbol{\theta}^c$ for the model and similarly an *ICL discrimination vector* $\boldsymbol{\alpha}^c$ for the item:

$$P(r_{i,j} = 1 | \mathcal{M}_i, x_j) = \sigma(\boldsymbol{\theta}_i^{\mathrm{T}}\boldsymbol{\alpha}_j - d_j + \boldsymbol{\theta}_i^{c\mathrm{T}}\boldsymbol{\alpha}_j^c), \tag{10}$$

where the alignment between $\boldsymbol{\theta}_i^c$ and $\boldsymbol{\alpha}_j^c$ represents the in-context *learnability* of $\mathcal{M}_i$ with respect to x_j. Similar to d and $\boldsymbol{\alpha}$, $\boldsymbol{\alpha}_j^c$ is computed based on the embedding of x:

$$\alpha_j^c = f(\mathbf{W}_\alpha^c e_j + \mathbf{b}_\alpha^c). \tag{11}$$

Combining Eq. 7, and 10, we have:

$$P(r_{i,j}^{\{\varnothing,c\}} = 1) = \sigma(\boldsymbol{\theta_i}\boldsymbol{\alpha_j} - d_j + g^{\{\varnothing,c\}}\boldsymbol{\theta}_i^c\boldsymbol{\alpha}_j^c), \tag{12}$$

where $r^\varnothing$ and r^c are the correctness labels under direct prompting and ICL. $\{g^\varnothing = 0, g^c = 1\}$ is a gating parameter in align with r to ensure that $\theta_i^c\alpha_j^c$ are only enabled in the ICL setting. In doing this, the latent factors are learned such that:

$$\begin{cases} \boldsymbol{\theta}^{\mathrm{T}}\boldsymbol{\alpha} > d, \boldsymbol{\theta}^{\mathrm{T}}\boldsymbol{\alpha} + \boldsymbol{\theta}^{c\mathrm{T}}\boldsymbol{\alpha}^c > d, \text{ if } r^\varnothing = 1, \\ \boldsymbol{\theta}^{\mathrm{T}}\boldsymbol{\alpha} < d, \boldsymbol{\theta}^{\mathrm{T}}\boldsymbol{\alpha} + \boldsymbol{\theta}^{c\mathrm{T}}\boldsymbol{\alpha}^c > d, \text{ if } r^\varnothing = 0, r^c = 1, \\ \boldsymbol{\theta}^{\mathrm{T}}\boldsymbol{\alpha} < d, \boldsymbol{\theta}^{\mathrm{T}}\boldsymbol{\alpha} + \boldsymbol{\theta}^{c}\mathrm{T}\boldsymbol{\alpha}^c < d, \text{ if } r^\varnothing = 0, r^c = 0. \end{cases} \tag{13}$$

The above three situations correspond to ●$\mathcal{Z}_{\checkmark}$, ●$\mathcal{Z}_{\text{✗}\rightarrow\checkmark}$, and ●$\mathcal{Z}_{\text{✗}\rightarrow\text{✗}}$, respectively. We refer to the proposed model as MIRT$_{\mathrm{ICL}}$.

From a multi-task learning perspective, our model can be seen as jointly training two IRT models, each with its own ability (θ, θ^c) and discrimination (α, α^c) parameters, while sharing the overall item difficulty (d). This allows the model to better capture the relationships between LM behaviors across the two settings.

5 Experiments

We experiment with 8 LLaMA models (Touvron et al., 2023; Dubey et al., 2024) of various sizes, including `LLaMA-2-7B`, `LLaMA-2-7B-chat`, `LLaMA-2-13B`, `LLaMA-2-13B-chat`, `LLaMA-3-8B`, `LLaMA-3-8B-Instruct`, `LLaMA-3-70B`, and `LLaMA-3-70B-Instruct`. In particular, we consider both instruction-tuned (IT) (`-chat/Instruct`

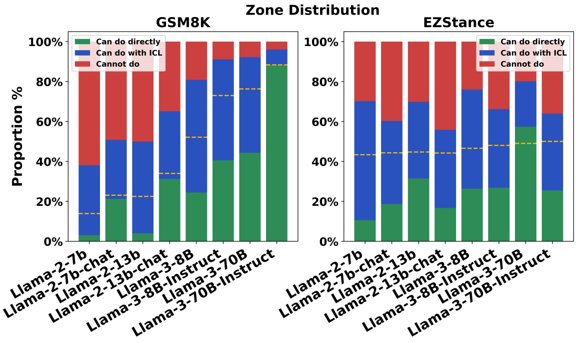

Figure 3: Zone distribution of various LLMs on the two datasets. Yellow lines represent the accuracy of KATE.

models) or non-IT versions to examine the influence of instruction tuning on the model's ZPD. In this study, we focus on the *mathematical reasoning* and *text understanding* abilities of LLMs, using the MathQA dataset **GSM8K** (Cobbe et al., 2021) and the Stance detection (Favor, Neutral, Against) dataset **EZStance** (Zhao and Caragea, 2023) for stance detection. Detailed experiment setup can be found in Appendix A.

We first present and analyze the zone distribution of various LLaMA models (§ 5.1). Then, we evaluate the performance of IRT models on zone prediction (§ 5.2). Finally, we demonstrate two applications of our framework (§ 5.3).

5.1 Zone Distribution Analysis

We measure the three zones of LLaMA models on the test set of GSM8K and the validation set of EZStance. Our observations are as follows.

- ***The potential of ICL remains largely untapped***. In Figure 3, we present the zone distributions of various models. Ideally, the accuracy of ICL should be the combined proportion of ●$\mathcal{Z}_{\checkmark}$ and ●$\mathcal{Z}_{\text{✗}\rightarrow\checkmark}$, which highlights the great potential of ICL. For instance, on the GSM8K dataset, the 8B-Instruct model, with the help of Oracle demonstrations, can achieve competitive performance compared to the two 70B models.

Note that, however, this is only a lower bound of ideal ICL performance, as the Oracle demonstrations are still sub-optimal. Nevertheless, the current method still falls short of fully utilizing even this lower bound. For reference, we highlight the accuracy (yellow line) of KATE (Liu et al., 2021), a similarity-based demonstration selection strategy (with `paraphrase-mpnet-base-v2`). On average, it lags by around 20% on the two datasets.

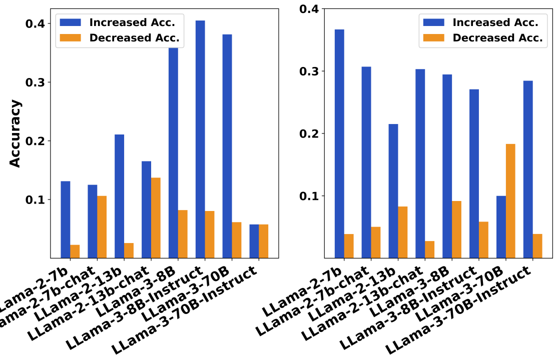

Figure 4: Increased and decreased accuracy by KATE on GSM8K (left) and EZStance (right).

• ***In-context demonstrations can be harmful.*** In Section § 3, we divide a query set into three zones according to the model's performance difference with and without ICL. However, sometimes, ICL can also degrade the performance. We denote the collection of such queries as ●$\mathcal{Z}_{\text{✓}\to\text{✗}}$. We do not frame ●$\mathcal{Z}_{\text{✓}\to\text{✗}}$ into our formalization (§ 3.1) but merge it into ●$\mathcal{Z}_{\text{✓}}$ because we focus on the ideal ICL behavior given Oracle demonstrations. However, this negative effect of ICL is non-negligible in a practical setting.

With KATE as a case study, we compare its increased accuracy (i.e., the proportion of recalled ZPD (●$\mathcal{Z}_{\text{✗}\to\text{✓}}$) examples) and decreased accuracy (i.e., the proportion of ●$\mathcal{Z}_{\text{✓}\to\text{✗}}$ examples) in Figure 4. The sum of the two is the overall performance of KATE. We can see ●$\mathcal{Z}_{\text{✓}\to\text{✗}}$ can reduce up to 14% and 18% accuracy on GSM8K and EZStance. Besides, this negative effect is also model-dependent. For example, `LLaMA-2-7B-chat` and `LLaMA-2-13B-chat` are particularly vulnerable to harmful demonstrations, and this negative effect even overwhelms the benefit for `LLaMA-3-70B`. This granular look at the ICL performance provides a new perspective to improve ICL strategy: recalling examples in ●$\mathcal{Z}_{\text{✓}}$ while minimizing ●$\mathcal{Z}_{\text{✓}\to\text{✗}}$. Previous work mainly focused on the first direction and we will showcase how our IRT model can enhance ICL through the second way in § 5.3.1.

Zones	GSM8K			EZStance		
	Max	Min	Avg	Max	Min	Avg
●$\mathcal{Z}_{\text{✓}}$	0.89	0.74	0.84	0.91	0.46	0.70
●$\mathcal{Z}_{\text{✗}\to\text{✓}}$	0.74	0.21	0.58	0.78	0.34	0.58
●$\mathcal{Z}_{\text{✗}\to\text{✗}}$	0.58	0.20	0.42	0.87	0.32	0.53

Table 1: Pairwise overlap coefficients among zones of different LLMs.

Model	GSM8K			EZStance		
	DP	ICL	Overall	DP	ICL	Overall
IRT$_{\text{1PL}}$	0.808	0.769	0.748	0.736	0.617	0.644
IRT$_{\text{2PL}}$	0.788	0.740	0.728	0.739	0.631	0.651
MIRT	**0.837**	0.770	0.743	0.760	0.608	0.799
MIRT$_{\text{ICL}}$	0.833	**0.821**	**0.862**	**0.770**	**0.662**	**0.799**

Table 2: Performance (AUC) of various IRT models on the two datasets. The best results are in **bold**. Results of Accuracy can be found in Appendix Table 5.

• ***ZPD (●$\mathcal{Z}_{\text{✗}\to\text{✓}}$) of LLMs differ significantly***. We measure the overlap between zones of different models by calculating their averaged pairwise *Overlap Coefficient*, defined as:

$$\text{OVERLAP}(A, B) = \frac{|A \cap B|}{\min(|A|, |B|)}, \tag{14}$$

where A and B are the zones to compare. The results are shown in Table 1, where we can see examples in ●$\mathcal{Z}_{\text{✓}}$ are largely shared across various models, while examples in ●$\mathcal{Z}_{\text{✗}\to\text{✓}}$ and ●$\mathcal{Z}_{\text{✗}\to\text{✗}}$ do not highly overlap, indicating each LLM has its own ZPD. This suggests that ICL strategies should take into account both the data aspect (e.g., similarity) and the model, corroborating the conclusion of Peng et al. (2024).

5.2 Zone Prediction Evaluation

We compare our proposed IRT model MIRT$_{\text{ICL}}$ (Eq. 12) with the following baselines: i) 1PL model (IRT$_{\text{1PL}}$, Eq. 6), ii) 2PL model, which is similar to Eq. 7 but with θ and α as scalars, and iii) Multi-Dimensional IRT MIRT (Eq. 7). We evaluate their ability to predict LLM performance under both direct prompting (DP) and ICL, using AUC as the primary metric. See Appendix A.2 for the implementation details. Note that aside from our MIRT$_{\text{ICL}}$, other baseline models are trained solely on DP data. Nevertheless, we can assess their generalization ability to the ICL setting: since AUC assesses the relative ranking of predicted probabilities, these models should also achieve good AUC if *the LLM's probabilities of correctly answering individual queries are consistent across both settings.*

• ***ICL behavior is, to varying degrees, predictable without demonstrations***. We present the AUC results in Table 2. As a demonstration-agnostic model, MIRT $_{\text{ICL}}$ achieves reasonably decent performance GSM8K but comparatively weaker results on EZStance. We interpret the difference through

	L2-7B		**L2-13B**		**L3-8B**		**L3-70B**	
	base	chat	base	chat	base	instr.	base	instr.
GSM8K	+.10	+.40	$+.13^*$	+.24	+.29	+.07	+.31	+.53
EZStance	−.45	−.60	−.47	−.25	−.35	−.28	−.48	−.36

Table 3: Pearson Correlation between $\boldsymbol{\theta}^{\mathrm{T}}\boldsymbol{\alpha} - d$ (model's ability to solve the query with direct prompting) and $\boldsymbol{\theta}^{c\mathrm{T}}\boldsymbol{\alpha}^{c}$ (the additional gain obtained by ICL). Results with * indicate p-value> 0.05.

the *predictability* and *sensitivity* of ICL: for certain tasks and datasets, ICL performance may hinge more on the model's inherent ICL capacity and the query's difficulty. While for others, it may depend more on the demonstrations or prompts, making the ICL behavior less predictable without the information of demonstrations. Existing work has been focusing on measuring and mitigating sensitivity (Zhao et al., 2021b). We highlight a complementary perspective: measuring and leveraging (See § 5.3 for applications) the predictability of ICL behavior.

• ***(In)consistency between difficulty and in-context learnability.*** In Eq. 12, $\boldsymbol{\theta}\boldsymbol{\alpha} - d$ represents the model's ability to solve the query with DP (or the query's *difficulty*), while $\boldsymbol{\theta}^{c}\boldsymbol{\alpha}^{c}$ captures the additional gain achieved through ICL, reflecting the model's *in-context learnability* of the query. To examine the relationship between the two terms, we compute their Pearson correlation. The results, presented in Table 3, reveal that for the GSM8K dataset, these two terms exhibit weak or moderate positive correlations (from +0.07 to +0.53). Interestingly, the correlation on EZStance is stronger but negative, meaning difficult examples under direct prompting (lower $\boldsymbol{\theta}^{\mathrm{T}}\boldsymbol{\alpha} - d$) seem to benefit more from ICL (higher $\boldsymbol{\theta}^{c\mathrm{T}}\boldsymbol{\alpha}^{c}$) and vice versa. This suggests that a query's difficulty and its in-context learnability are not always aligned. We attribute this phenomenon to the differing abilities required for direct prompting versus ICL. The former primarily relies on the model's prior knowledge of the query, while the latter depends on its ability to leverage contextual information. As a result, this inconsistency could arise in certain tasks and queries where the knowledge is missing but easy to learn in context. A notable example is classification with flipped or semantically unrelated labels (Wei et al., 2023), where an LM struggles to solve the disrupted task in the regular setting but can successfully learn the new mapping through demonstrations.

5.3 Applications

In this section, we demonstrate how our framework can improve in-context learning through a selective ICL strategy (§ 5.3.1) and a ZPD-derived curriculum for fine-tuning LLMs (§ 5.3.2).

5.3.1 Selective ICL

Approach. While ICL has demonstrated effectiveness across a wide range of tasks, it costs k times additional input tokens (k = the number of demonstrations). Moreover, as discussed in § 5.1, ICL sometimes results in worse performance, even with carefully retrieved demonstrations. To address these issues, we propose Selective ICL (SELICL). In specific, given a query x_i, we first predict its correct probability with direct prompting $p_i^{\varnothing}$ and the correct probability with ICL p_i^{c} using Eq. 12 with $g = 0$ and $g = 1$ respectively. Then, we decide the inference prompt for x_i by:

$$\begin{cases} \mathcal{T}(\tilde{c}_1) ... \oplus \mathcal{T}(x) & \text{if } p^{\varnothing} < \tau_1 \text{ and } p^{c} > \tau_2 \\ \mathcal{T}(x) & \text{Otherwise.} \end{cases} \tag{15}$$

where $\{\tilde{c}_1, ..., \tilde{c}_n\}$ are demonstrations retrieved by a certain strategy. τ_1 and τ_2 are predefined thresholds. A lower $p^{\varnothing}$ ($< \tau_1$) and a higher p^{c} ($> \tau_2$) indicate the model is unable to solve this query with direct prompting but is likely to solve it with ICL. In other words, we apply ICL only to queries within the model's ZPD. By doing so, we aim to reduce unnecessary costs by avoiding ICL for either too easy ($p^{\varnothing} < \tau_1$) or too hard ($p^{c} > \tau_2$) queries. Furthermore, this can also potentially improve performance by mitigating the negative effect of ICL observed in Figure 4.

Result and Analysis. We compare our SELICL with the vanilla ICL that applies demonstrations to all queries (denoted as FULICL). Specifically, we use KATE to retrieve demonstrations for FULICL. However, it is worth noting that SELICL is orthogonal to other ICL strategies for two reasons: (1) It focuses on determining when to apply ICL, independent of how demonstrations are selected or organized; (2) The IRT model is trained to predict the model's ICL performance given Oracle demonstrations. Consequently, p^{c} is expected to serve as the predicted upper bound for any ICL strategy.

To select τ_1 and τ_2 for SELICL, we perform a grid search on the IRT validation set by varying their values within the range $[0.01, 0.02, \ldots, 0.99]$. For each combination, we

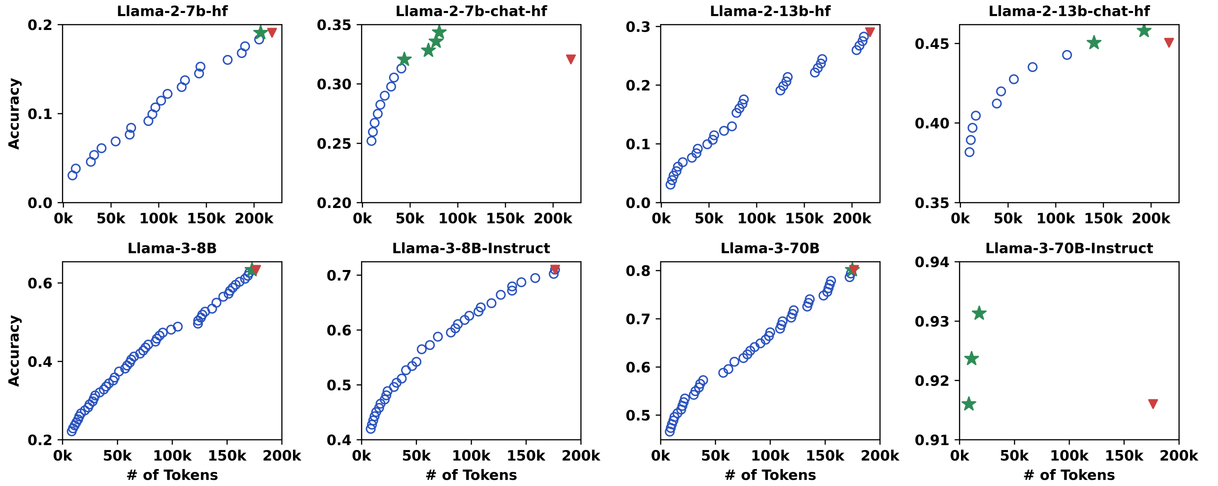

Figure 5: Accuracy and inference cost (number of input tokens) of different ICL strategies on the GSM8K dataset. ▼ is the performance of the baseline FulICL, which applies ICL to all the queries. ○ and ★ are the performance of SelICL under various thresholds τ_1 and τ_2 (not shown), where ★ highlights cases in which SelICL achieves better or equal accuracy with less input tokens compared to the baseline (▼).

decide whether or not to apply ICL to each query according to Eq. 15 and compute the overall accuracy and number of input tokens. Since the prompts and model outputs are already collected when constructing the IRT dataset (Appendix A.2), these results can be obtained without additional model inference.

Then, we plot the Pareto curve (Deb, 2011) of SelICL, approximated with scatter points. In multi-objective optimization, each point on the Pareto curve represents a Pareto-optimal solution that cannot be further improved in one objective without compromising the other (in our case, accuracy and number of input tokens).

Results for GSM8K are shown in Figure 5, and results for EZstance are available in Appendix Figure 8. Solutions that are dominated[2] by others are discarded (apart from the baseline results (▼) for comparison). As can be seen, for 6 out of 8 models, SelICL with proper thresholds (★) can dominate FulICL. Overall, SelICL can serve as a tool to trade off accuracy and cost in resource-limited scenarios. SelICL is paticularly successful for `LLaMA-2-7b-chat` and `LLaMA-70B-Instruct`. Combining with previous findings, both models have relatively narrow ZPD (Figure 3) and are more susceptible to the negative effects of ICL (Figure 4), suggesting that greater caution is needed when applying ICL to them.

Algorithm 1 ZPD-based Curriculum

```
Input: Training set $\mathcal{D}$, model $\mathcal{M}$, correct probability with DP $p^{\varnothing}$ and with ICL $p^{c}$, bucket $k$, epoch $e$
Output: Trained model $\mathcal{M}^*$
$\mathcal{D}^* \leftarrow$ Sort($\mathcal{D}, p_i^c - p_i^{\varnothing}$)
$\{\mathcal{D}_1, ..., \mathcal{D}_n\} \leftarrow$ SplitData($\mathcal{D}^*$) ; $\mathcal{D}_{train} \leftarrow \varnothing$
for $i = 1, i \le k, i$++ do
    $\mathcal{D}_{train} \leftarrow \mathcal{D}_{train} \cup \mathcal{D}_i$        ▷ Update training set
    for $j = 1, j \le e, j$++ do
        Train($\mathcal{M}, \mathcal{D}_{train}$);
    end for
end for
```

5.3.2 ZPD-based Curriculum

It is generally believed that the success of ICL relies on the model's prior knowledge about the query (Xie et al., 2022; Li et al., 2024). Therefore, we assume that queries that can be enhanced by ICL (●$\mathcal{Z}_{\times\rightarrow\checkmark}$) are more learnable than those unsolvable by ICL (●$\mathcal{Z}_{\times\rightarrow\times}$) but also more valuable for learning than those already solvable by DP (●$\mathcal{Z}_{\checkmark}$). Motivated by this, we proposed a ZPD-based curriculum learning algorithm for fine-tuning.

Approach. Typically, curriculum learning consists of a ranking algorithm, which sorts examples according to a certain criterion, and a scheduling algorithm, which sequences examples for training. In our approach, we rank training examples according to $p^c - p^{\varnothing}$ (Eq. 15), which represents the learning gain brought by ICL. For scheduling, we employ the baby-step algorithm (Spitkovsky et al., 2010), which splits examples into buckets and accumula-

[2] In the context of a Pareto curve, a solution dominates another if it is at least as good in all objectives and strictly better in at least one objective.

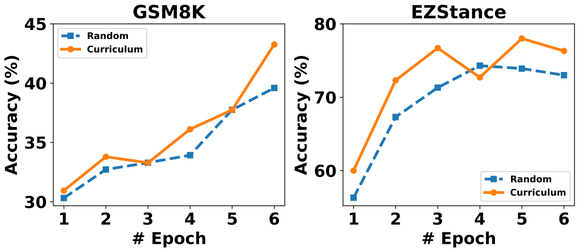

Figure 6: Comparison between random and our ZPD-based curriculum on two datasets.

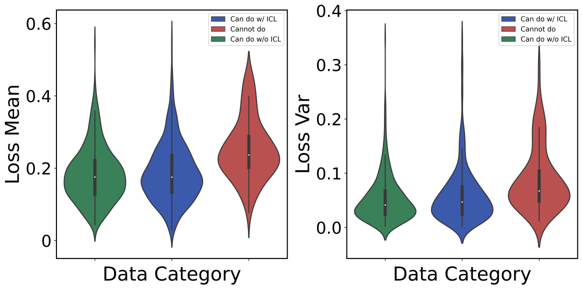

Figure 7: Mean and variance of training loss for queries in different zones. Results are computed over 5 epochs.

tively introduces new buckets. The overall process is outlined in Algorithm 1.

Results and Analysis. We compare our algorithm against a random baseline. Although simple, random is the most widely used baseline in practice and is not necessarily a weak one, as many curriculum strategies fail to outperform it in language modeling (Campos, 2021). We fine-tune the `LLaMA-8B-Instruct` model separately using the two methods with the same scheduler for 6 epochs. See experimental details in Appendix A.3. As shown in Figure 6, our curriculum results in faster convergence and improved performance in most cases. To understand why it works, we analyze the training loss of examples in different zones. Specifically, we compute the **mean** and **variance** of each example's loss across epochs. The two metrics reflect the convergence behavior of individual examples: a higher mean indicates the example is harder to learn, while a higher variance indicates the model is ambiguous about the example (Swayamdipta et al., 2020).

For fair analysis, we fine-tune a new `LLaMA-8B-Instruct` model on the GSM8K dataset for 5 epochs without any curriculum. Figure 7 shows the loss information. We found *consistent learnability* between in-context learning and fine-tuning scenarios: examples in ● $\mathcal{Z}_{\text{✗}\rightarrow\text{✗}}$ are the hardest to learn, followed by ● $\mathcal{Z}_{\text{✗}\rightarrow\text{✓}}$[3], and lastly ● $\mathcal{Z}_{\text{✓}}$. This confirms that our curriculum works as expected: prioritizing examples that are learnable and informative (not yet learned). Such a strategy has been shown effective for various tasks and model architectures (Mindermann et al., 2022; Fan and Jaggi, 2023), and our framework provides a new way to discover these examples.

6 Conclusion

This work presents a novel framework based on the Zone of Proximal Development (ZPD) theory to analyze the ICL behaviors of LLMs. We thoroughly discuss the formalization, measurement, prediction, and application of ZPD in LLMs. Our framework serves as an effective tool for understanding the potential, limitations, and complex dynamics of ICL. Furthermore, we demonstrate its applicability in both inference and training scenarios.

Limitations

We discuss the limitations of this work from the following aspects. First, due to the unavailability of optimal in-context demonstrations, we can only approximate the ZPD of LLMs, which is a lower bound of the model's actual in-context learnability. This challenge is as nuanced and complex as understanding human learning: one can never precisely measure the potential of human learners. Second, we investigate the ZPD of LLMs in a simplified scenario where we only consider demonstrations as guidance and use basic templates without instructions to minimize confounding factors. In practice, ICL is often combined with other prompting strategies, whose influence may warrant further exploration. Finally, the ZPD is a dynamic range that evolves with the learner's knowledge development. Our framework is designed to measure and leverage an LLM's current ZPD, but it is less suited to modeling its developing process (e.g., across different checkpoints during pre-training or fine-tuning). In the future, more advanced learning analytics approaches, such as knowledge tracing, could be adopted to enhance our framework.

[3] (Since we use sub-optimal Oracle demonstrations, some ● $\mathcal{Z}_{\text{✗}\rightarrow\text{✓}}$ examples are not recalled and misclassified into ● $\mathcal{Z}_{\text{✗}\rightarrow\text{✗}}$. As a result, the actual loss value of ● $\mathcal{Z}_{\text{✗}\rightarrow\text{✓}}$ data tends to be slightly closer to ● $\mathcal{Z}_{\text{✗}\rightarrow\text{✗}}$.)

References

Ekin Akyürek, Dale Schuurmans, Jacob Andreas, Tengyu Ma, and Denny Zhou. 2023. What learning algorithm is in-context learning? investigations with linear models. In *The Eleventh International Conference on Learning Representations*.

Yoshua Bengio, Jérôme Louradour, Ronan Collobert, and Jason Weston. 2009. Curriculum learning. In *Proceedings of the 26th annual international conference on machine learning*, pages 41–48.

Tom Brown, Benjamin Mann, Nick Ryder, Melanie Subbiah, Jared D Kaplan, Prafulla Dhariwal, Arvind Neelakantan, Pranav Shyam, Girish Sastry, Amanda Askell, et al. 2020. Language models are few-shot learners. *Advances in neural information processing systems*, 33:1877–1901.

Matthew Byrd and Shashank Srivastava. 2022. Predicting difficulty and discrimination of natural language questions. In *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers)*, pages 119–130.

Daniel Campos. 2021. Curriculum learning for language modeling. *arXiv preprint arXiv:2108.02170*.

Seth Chaiklin et al. 2003. The zone of proximal development in vygotsky's analysis of learning and instruction. *Vygotsky's educational theory in cultural context*, 1(2):39–64.

R Philip Chalmers. 2012. mirt: A multidimensional item response theory package for the r environment. *Journal of statistical Software*, 48:1–29.

Karl Cobbe, Vineet Kosaraju, Mohammad Bavarian, Mark Chen, Heewoo Jun, Lukasz Kaiser, Matthias Plappert, Jerry Tworek, Jacob Hilton, Reiichiro Nakano, Christopher Hesse, and John Schulman. 2021. Training verifiers to solve math word problems. *arXiv preprint arXiv:2110.14168*.

Kalyanmoy Deb. 2011. Multi-objective optimisation using evolutionary algorithms: an introduction. In *Multi-objective evolutionary optimisation for product design and manufacturing*, pages 3–34. Springer.

Abhimanyu Dubey, Abhinav Jauhri, Abhinav Pandey, Abhishek Kadian, Ahmad Al-Dahle, Aiesha Letman, Akhil Mathur, Alan Schelten, Amy Yang, Angela Fan, et al. 2024. The llama 3 herd of models. *arXiv preprint arXiv:2407.21783*.

Simin Fan and Martin Jaggi. 2023. Irreducible curriculum for language model pretraining. *arXiv preprint arXiv:2310.15389*.

Shivam Garg, Dimitris Tsipras, Percy S Liang, and Gregory Valiant. 2022. What can transformers learn in-context? a case study of simple function classes. *Advances in Neural Information Processing Systems*, 35:30583–30598.

Maharshi Gor, Hal Daumé III, Tianyi Zhou, and Jordan Boyd-Graber. 2024. Do great minds think alike? investigating human-ai complementarity in question answering with caimira. *arXiv preprint arXiv:2410.06524*.

SouYoung Jin, Aruni RoyChowdhury, Huaizu Jiang, Ashish Singh, Aditya Prasad, Deep Chakraborty, and Erik Learned-Miller. 2018. Unsupervised hard example mining from videos for improved object detection. In *Proceedings of the European Conference on Computer Vision (ECCV)*, pages 307–324.

Diederik P Kingma. 2014. Adam: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980*.

John Patrick Lalor and Pedro Rodriguez. 2023. py-irt: A scalable item response theory library for python. *INFORMS Journal on Computing*, 35(1):5–13.

Jiaoda Li, Yifan Hou, Mrinmaya Sachan, and Ryan Cotterell. 2024. What do language models learn in context? the structured task hypothesis. In *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 12365–12379, Bangkok, Thailand. Association for Computational Linguistics.

Xiaonan Li, Kai Lv, Hang Yan, Tianyang Lin, Wei Zhu, Yuan Ni, Guotong Xie, Xiaoling Wang, and Xipeng Qiu. 2023. Unified demonstration retriever for in-context learning. *arXiv preprint arXiv:2305.04320*.

Jiachang Liu, Dinghan Shen, Yizhe Zhang, Bill Dolan, Lawrence Carin, and Weizhu Chen. 2021. What makes good in-context examples for gpt-3? *arXiv preprint arXiv:2101.06804*.

Jiachang Liu, Dinghan Shen, Yizhe Zhang, Bill Dolan, Lawrence Carin, and Weizhu Chen. 2022. What makes good in-context examples for GPT-3? In *Proceedings of Deep Learning Inside Out (DeeLIO 2022): The 3rd Workshop on Knowledge Extraction and Integration for Deep Learning Architectures*, pages 100–114, Dublin, Ireland and Online. Association for Computational Linguistics.

Frederic M Lord and Melvin R Novick. 2008. *Statistical theories of mental test scores*. IAP.

Yao Lu, Max Bartolo, Alastair Moore, Sebastian Riedel, and Pontus Stenetorp. 2022. Fantastically ordered prompts and where to find them: Overcoming few-shot prompt order sensitivity. In *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 8086–8098, Dublin, Ireland. Association for Computational Linguistics.

Sören Mindermann, Jan M Brauner, Muhammed T Razzak, Mrinank Sharma, Andreas Kirsch, Winnie Xu, Benedikt Höltgen, Aidan N Gomez, Adrien Morisot, Sebastian Farquhar, et al. 2022. Prioritized training on points that are learnable, worth learning, and not yet learnt. In *International Conference on Machine Learning*, pages 15630–15649. PMLR.

Keqin Peng, Liang Ding, Yancheng Yuan, Xuebo Liu, Min Zhang, Yuanxin Ouyang, and Dacheng Tao. 2024. Revisiting demonstration selection strategies in in-context learning. In *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 9090–9101, Bangkok, Thailand. Association for Computational Linguistics.

Jean Piaget. 1977. *The development of thought: Equilibration of cognitive structures.(Trans A. Rosin)*. Viking.

Emmanouil Antonios Platanios, Otilia Stretcu, Graham Neubig, Barnabas Poczos, and Tom M Mitchell. 2019. Competence-based curriculum learning for neural machine translation. *arXiv preprint arXiv:1903.09848*.

Felipe Maia Polo, Lucas Weber, Leshem Choshen, Yuekai Sun, Gongjun Xu, and Mikhail Yurochkin. 2024. tinybenchmarks: evaluating llms with fewer examples. *arXiv preprint arXiv:2402.14992*.

Chengwei Qin, Aston Zhang, Anirudh Dagar, and Wenming Ye. 2023. In-context learning with iterative demonstration selection. *arXiv preprint arXiv:2310.09881*.

Mark D Reckase. 2006. 18 multidimensional item response theory. *Handbook of statistics*, 26:607–642.

N Reimers. 2019. Sentence-bert: Sentence embeddings using siamese bert-networks. *arXiv preprint arXiv:1908.10084*.

Stephen Robertson, Hugo Zaragoza, et al. 2009. The probabilistic relevance framework: Bm25 and beyond. *Foundations and Trends® in Information Retrieval*, 3(4):333–389.

Ohad Rubin, Jonathan Herzig, and Jonathan Berant. 2021. Learning to retrieve prompts for in-context learning. *arXiv preprint arXiv:2112.08633*.

Ohad Rubin, Jonathan Herzig, and Jonathan Berant. 2022. Learning to retrieve prompts for in-context learning. In *Proceedings of the 2022 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 2655–2671, Seattle, United States. Association for Computational Linguistics.

Mrinmaya Sachan and Eric Xing. 2016. Easy questions first? a case study on curriculum learning for question answering. In *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 453–463.

Darcy A Santor and James O Ramsay. 1998. Progress in the technology of measurement: Applications of item response models. *Psychological assessment*, 10(4):345.

Abhinav Shrivastava, Abhinav Gupta, and Ross Girshick. 2016. Training region-based object detectors with online hard example mining. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 761–769.

Valentin I Spitkovsky, Hiyan Alshawi, and Dan Jurafsky. 2010. From baby steps to leapfrog: How "less is more" in unsupervised dependency parsing. In *Human Language Technologies: The 2010 Annual Conference of the North American Chapter of the Association for Computational Linguistics*, pages 751–759.

Pragya Srivastava, Satvik Golechha, Amit Deshpande, and Amit Sharma. 2024. NICE: To optimize in-context examples or not? In *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 5494–5510, Bangkok, Thailand. Association for Computational Linguistics.

Swabha Swayamdipta, Roy Schwartz, Nicholas Lourie, Yizhong Wang, Hannaneh Hajishirzi, Noah A. Smith, and Yejin Choi. 2020. Dataset cartography: Mapping and diagnosing datasets with training dynamics. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 9275–9293, Online. Association for Computational Linguistics.

Yi Tay, Shuohang Wang, Luu Anh Tuan, Jie Fu, Minh C Phan, Xingdi Yuan, Jinfeng Rao, Siu Cheung Hui, and Aston Zhang. 2019. Simple and effective curriculum pointer-generator networks for reading comprehension over long narratives. *arXiv preprint arXiv:1905.10847*.

Roland G Tharp and Ronald Gallimore. 1991. *Rousing minds to life: Teaching, learning, and schooling in social context*. Cambridge University Press.

Hugo Touvron, Louis Martin, Kevin Stone, Peter Albert, Amjad Almahairi, Yasmine Babaei, Nikolay Bashlykov, Soumya Batra, Prajjwal Bhargava, Shruti Bhosale, et al. 2023. Llama 2: Open foundation and fine-tuned chat models. *arXiv preprint arXiv:2307.09288*.

Lev Semenovich Vygotsky. 1978. *Mind in society: The development of higher psychological processes*, volume 86. Harvard university press.

Xinyi Wang, Wanrong Zhu, Michael Saxon, Mark Steyvers, and William Yang Wang. 2024. Large language models are latent variable models: explaining and finding good demonstrations for in-context learning. Red Hook, NY, USA. Curran Associates Inc.

Jason Wei, Yi Tay, Rishi Bommasani, Colin Raffel, Barret Zoph, Sebastian Borgeaud, Dani Yogatama, Maarten Bosma, Denny Zhou, Donald Metzler, et al. 2022. Emergent abilities of large language models. *arXiv preprint arXiv:2206.07682*.

Jerry Wei, Jason Wei, Yi Tay, Dustin Tran, Albert Webson, Yifeng Lu, Xinyun Chen, Hanxiao Liu, Da Huang, Denny Zhou, et al. 2023. Larger language models do in-context learning differently, 2023. *URL https://arxiv. org/abs/2303.03846*.

Thomas Wolf, Lysandre Debut, Victor Sanh, Julien Chaumond, Clement Delangue, Anthony Moi, Pierric Cistac, Tim Rault, Rémi Louf, Morgan Funtowicz, et al. 2020. Transformers: State-of-the-art natural language processing. In *Proceedings of the 2020 conference on empirical methods in natural language processing: system demonstrations*, pages 38–45.

Sang Michael Xie, Aditi Raghunathan, Percy Liang, and Tengyu Ma. 2022. An explanation of in-context learning as implicit bayesian inference. In *International Conference on Learning Representations (ICLR)*.

Jiacheng Ye, Zhiyong Wu, Jiangtao Feng, Tao Yu, and Lingpeng Kong. 2023. Compositional exemplars for in-context learning. In *International Conference on Machine Learning*, pages 39818–39833. PMLR.

Hugh Zhang, Jeff Da, Dean Lee, Vaughn Robinson, Catherine Wu, Will Song, Tiffany Zhao, Pranav Raja, Dylan Slack, Qin Lyu, et al. 2024. A careful examination of large language model performance on grade school arithmetic. *arXiv preprint arXiv:2405.00332*.

Chenye Zhao and Cornelia Caragea. 2023. Ez-stance: A large dataset for zero-shot stance detection. In *Findings of the Association for Computational Linguistics: EMNLP 2023*, pages 897–911.

Zihao Zhao, Eric Wallace, Shi Feng, Dan Klein, and Sameer Singh. 2021a. Calibrate before use: Improving few-shot performance of language models. In *Proceedings of the 38th International Conference on Machine Learning*, volume 139 of *Proceedings of Machine Learning Research*, pages 12697–12706. PMLR.

Zihao Zhao, Eric Wallace, Shi Feng, Dan Klein, and Sameer Singh. 2021b. Calibrate before use: Improving few-shot performance of language models. In *International conference on machine learning*, pages 12697–12706. PMLR.

A Experimental Setups

A.1 Inference Setting

The datasets are used under the MIT License and with their intended use. For models, we use `LLaMA` checkpoints from Hugging Face Transformers (Wolf et al., 2020). We run experiments with up to 8× RTX 4090 24G GPUs. e. Due to memory constraints, we use Float16 precision for inference, with each run taking around 1~4 hours, depending on the model and data size. The prompt template for GSM8K and EZstance are in Appendix Table 4. For ICL, we set the number of demonstrations to 8 following (Li et al., 2023; Rubin et al., 2021).

Dataset	**Prompt Template**

GSM8K	`Question: {math_problem}`
	`Answer: {step_by_step_answer}.`

EZStance	`Text: {sentence}`
	`Question: Which stance-"favor," "against," or "neutral"-does the above text express toward {target}?`
	`Answer: {stance}.`

Table 4: Prompt templates for the two datasets. highlighted parts are inputs.

A.2 Implementation Details of IRT

Dataset Construction. The dataset for the IRT model is built upon LLM outputs. First, we construct Oracle demonstrations using the approach described in § 3. Then, we run LLMs using prompts in Appendix Table 4 in different settings (DP or ICL). The outputs are represented as tuples consisting of `<model_id, example_id, input, output, setting, label>`. This results in total 2 (Direct prompting or ICL setting) × M (Number of LLMs) × N (Number of queries) instances, where $M = 8$, $N_{\text{GSM8K}} = 1319, N_{\text{EZStance}} = 6703$. We further split them into 80% training set, 10% validation set, and 10% test set.

Training Setup. We set the dimension of latent traits $\boldsymbol{\theta}, \boldsymbol{\alpha}, \boldsymbol{\theta}^c$, $\boldsymbol{\alpha}^c$ to 32. Queries are encoded with SBERT (`paraphrase-mpnet-base-v2`) with an embedding size of 768. We train all the models for 10 epochs with a learning rate of $2e-4$ and batch size of 16. Traditionally, IRT is optimized by marginalized maximum likelihood estimation (Chalmers, 2012). However, this does not scale well to large datasets (Lalor and Rodriguez, 2023). We follow Gor et al. (2024) to use Adam (Kingma, 2014) to optimize our model. The best model is selected based on the performance on the validation set.

A.3 Details of Fine-tuning

We fine-tune `LLaMA-3-8B-Instruct` to evaluate our curriculum learning algorithm (§ 5.3.2). Since `LLaMA` models might already be fine-tuned on the training set of GSM8K (Zhang et al., 2024), we randomly sample 1,000 instances from the test set for fine-tuning and use the remaining 319 instances for evaluation. The EZStance dataset is curated after the release of `LLaMA-3` and, therefore, has no such concern. We sample 5,000 examples from the training set for fine-tuning and directly evaluate the model on the test set. With the scheduler in Algorithm 1, we split the dataset into 3 buckets and fine-tune the model on each bucket for 2 epochs with a learning rate of $1e-5$ and batch size of 4.

B Additional Results

B.1 Additional Results of IRT

The accuracy of IRT models is in Table 5. Note that baseline models are not trained on ICL data and therefore their accuracy is not indicative. We report it only for the completeness of the results.

Model	**GSM8K**			**EZStance**		
	DP	ICL	Overall	DP	ICL	Overall
IRT_{1PL}	69.1	39.6	56.7	76.1	45.6	63.1
IRT_{2PL}	70.4	40.2	58.7	75.3	46.4	63.0
MIRT	68.9	47.2	59.8	76.6	45.9	63.5
MIRT_{ICL}	**77.4**	**78.4**	**77.9**	**77.0**	**68.5**	**72.8**

Table 5: Performance (Accuracy %) of various IRT models. The best results are in **bold**.

B.2 Additional Results of SELICL

The result of SELICL on the EZStance dataset is in Appendix Figure 8.

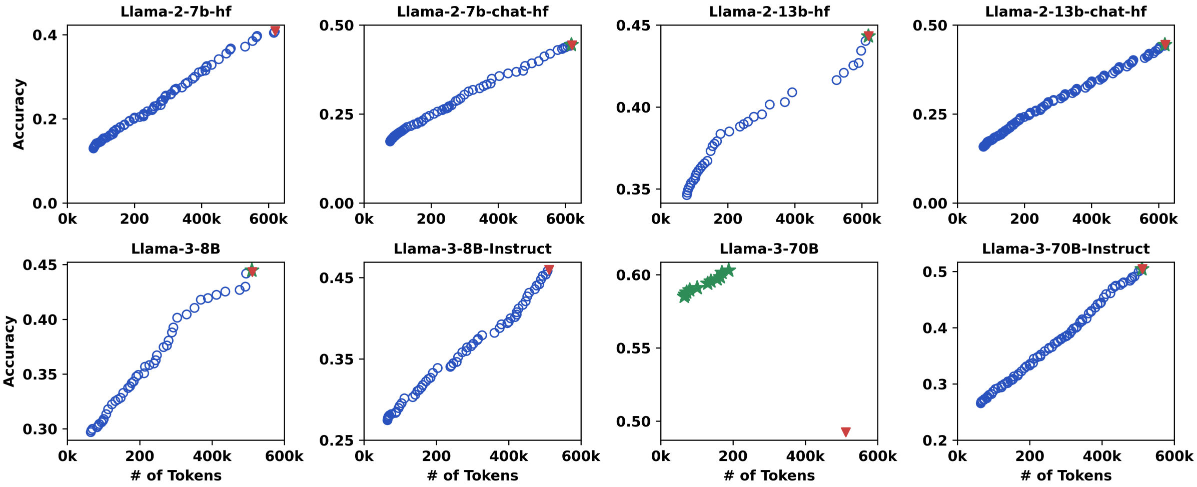

Figure 8: Results of SELICL on EZStance. See detailed explanations in Figure 5.