Explore the In-context Learning Capability of Large Language Models

by

Tianle Li

A thesis
presented to the University of Waterloo
in fulfillment of the
thesis requirement for the degree of
Master of Mathematics
in
Computer Science

Waterloo, Ontario, Canada, 2024

Author's Declaration

This thesis consists of material all of which I authored or co-authored: see Statement of Contributions included in the thesis. This is a true copy of the thesis, including any required final revisions, as accepted by my examiners.

I understand that my thesis may be made electronically available to the public.

Statement of Contributions

This thesis includes the material which is presented at ACL 2023 [60] and a preprint in Arxiv [61]. I conducted most of the experiments and wrote the first drafts of these papers. My supervisor Professor Wenhu Chen contributed ideas, feedback and editing. Other co-authors either help to run experiments or polish the paper.

I understand that my thesis may be made electronically available to the public.

Abstract

The rapid evolution of Large Language Models (LLMs) has marked the beginning of a new age in AI capabilities, particularly in the domain of natural language understanding and processing. Among the forefront of these advancements is the exploration of in-context learning, a paradigm that enables models to adapt to new tasks without explicit retraining. This thesis embarks on a comprehensive investigation into the in-context learning capabilities of LLMs, guided by two pivotal studies: KB-BINDER's deployment in Question Answering over Knowledge Bases (KBQA) and the evaluation of LLMs' performance on LongICLBench, a self-curated benchmark for long-context understanding.

The first facet of this investigation, embodied by KB-BINDER, addresses the challenge of generalizing LLMs to diverse KBQA tasks without task-specific training. KB-BINDER pioneers a novel few-shot in-context learning approach, utilizing Codex to generate logical forms and employing BM25 for draft binding, demonstrating remarkable efficacy across heterogeneous KBQA datasets. We believe KB-BINDER can serve as an important baseline for future research in utilizing the few-shot capability of LLMs to resolve the problem of KBQA.

Complementing this, the second study introduces LongICLBench, a specialized benchmark designed to test long-context LLMs in processing long, context-rich sequences across extreme-label classification tasks with in-context learning. Through evaluation with tasks of increasing difficulty level, an obvious performance threshold is identified, highlighting the current limitations of LLMs in handling extensive context windows and revealing a bias towards labels positioned towards the input's end after grouping the instances with the same labels in demonstration. This underscores a crucial gap in the current long-context LLMs' ability to reason over long sequences, paving the way for further enhancements in long-context comprehension.

Together, these studies form the cornerstone of this thesis, encapsulating the dynamic landscape of in-context learning within LLMs. Through a detailed examination of KB-BINDER and LongICLBench, this work not only charts the current capabilities and boundaries of LLMs but also lays the groundwork for future advancements in making LLMs more adaptable and proficient in handling a wide array of complex tasks.

Acknowledgements

I would like to express my sincere gratitude to my supervisor Prof. Wenhu Chen, who always provides instant and helpful suggestions all the way along my research path during my study pursuing master degree. I also want to thank all my lab-mates and co-authors who create a very good atmosphere for me to do research, which makes it enjoyable to collaborate with them.

Table of Contents

List of Figures

List of Tables

Chapter 1

Introduction

Leveraging the transformative power of large language models (LLMs) for in-context learning heralds a significant advancement in artificial intelligence, extending the frontier of tasks that can be tackled without further fine-tuning [105, 16, 30]. This thesis delves into the broad spectrum of capabilities enabled by in-context learning in LLMs, focusing on two distinct yet interconnected domains: Knowledge Base Question Answering (KBQA) and the evaluation of long-context LLMs through extreme-label classification tasks. By harnessing the in-context learning prowess of LLMs, we aim to address complex challenges or explore potential limitations across these domains, showcasing the models' versatility and their latent deficiency when dealing with natural language processing tasks.

The realm of KBQA has long been a focal point of AI research [7, 115], driven by the quest to make vast repositories of structured knowledge accessible through natural language queries [108, 52, 39]. Despite the progress, the sheer volume and variety of knowledge bases present formidable challenges, notably in terms of data intensiveness and dataset specificity as detailed as follows: 1) Data intensiveness: larger knowledge bases require ever larger quantities of annotated data to allow fine-tuned models to generalize well over them. [116, 93, 38]. 2) Dataset specificity: For relatively small-scale KBQA datasets, the fully-trained models tend to overfit to a specific schema, and can hardly generalize to knowledge base questions in unseen domains [88, 126, 89]. Traditional approaches often depend on the need for extensive annotated data and struggle to generalize across diverse knowledge domains. These challenges are more likely to be overcome with the potential of LLMs, such as GPT-3 and Codex, which have demonstrated remarkable adaptability through few-shot in-context learning [10, 18]. These models, capable of generating draft logical forms from a handful of examples [103, 107, 128, 23, 127, 92], inspire us to devise

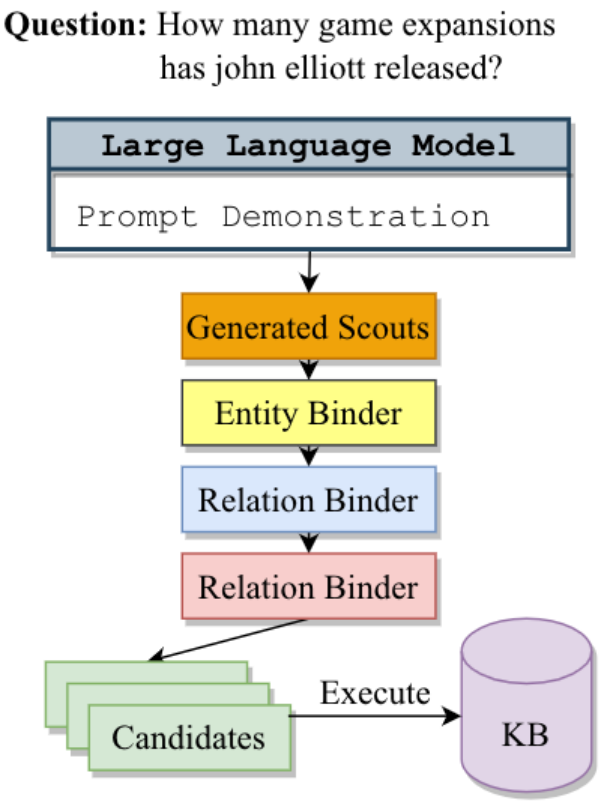
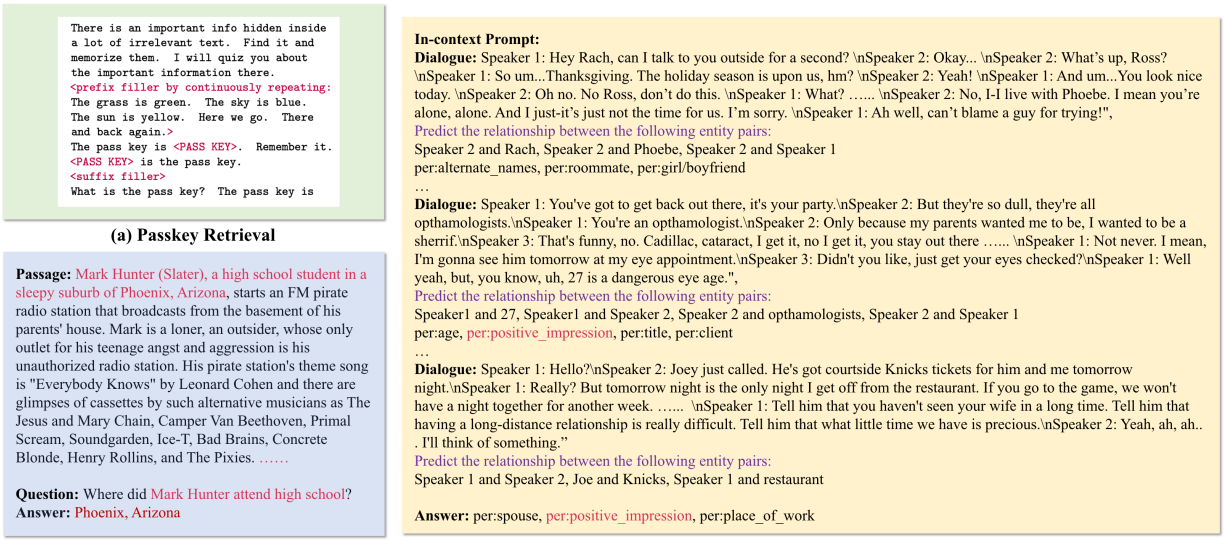


Figure 1.1: Overview of KB-BINDER pipeline. There are two primary stages in our method: 1) Generate the drafts as preliminary logical forms; 2) Bind the drafts to the executable ones with entity and relation binders grounded on the knowledge base. The final answer can be obtained after the execution of the final candidates.

a new framework KB-BINDER(i.e. generate-then-bind as demonstrated in Figure 1.1) to KBQA, which is training-free and resource-efficient, yet powerful in its generalizability.

Simultaneously, the landscape of LLMs has evolved to embrace long-context processing [42, 19, 76, 79, 110], a development critical for applications that require understanding extensive sequences of text, such as long-document question-answering and multi-document summarization. Despite this advancement, there is a notable absence of benchmarks capable of rigorously evaluating the models' comprehension over lengthy inputs. Traditional evaluation metrics fall short, as they often do not reflect the models' true capacity to process and reason over long, complex sequences as presented in Figure 1.2.

Historically, evaluations and benchmarks for long sequences have predominantly con-



(a) Passkey Retrieval

(b) Long-document Question-answer

(c) Extreme-label In-context Learning

Figure 1.2: Comparison extreme-label ICL with the existing evaluation tasks. Passkey Retrieval is a synthetic task. Long-document Question-answering does not require reading the entire document to find the answer. In extreme-label ICL, the model needs to scan through the entire demonstration to understand the whole label space to make the correct prediction.

centrated on three types of assessments:

1. Perplexity measurements on lengthy documents, which is a common metric utilized across numerous studies.

2. Tasks like passkey retrieval or needle-in-a-haystack [69, 19, 56, 95, 33], which involve identifying specific pieces of information inserted randomly into extensive sequences. This task has seen several LLMs surpassing a 99% success rate, indicating its synthetic nature.

3. Answering questions or summarizing content from extensive documents, such as those found in the Qasper dataset [24].

To address these limitations, our work introduces in-context learning with extreme-label classification tasks as a pioneering benchmark. This new benchmark challenges models to fully comprehend extensive inputs and navigate complex label spaces. Such tasks not only push the boundaries of what current LLMs can achieve with lengthy contexts but also can serve as a crucial platform for improving their capabilities in processing long-range information.

Generally, the core contribution of this thesis is the exploration of in-context learning as a unifying framework that can effectively tackle both KBQA challenges and serve as a rigorous evaluation tool for long-context comprehension in LLMs. In KBQA, we leverage in-context learning to navigate the complexities of diverse knowledge bases without the need for extensive training data, thereby mitigating the challenges of data intensiveness and dataset specificity. Through the development of methods like KB-BINDER, we demonstrate the feasibility of training-free, few-shot in-context learning in generating semantically precise logical forms for KBQA, overcoming the limitations of existing approaches.

In parallel, the introduction of extreme-label classification tasks as a benchmark for evaluating long-context LLMs underscores the necessity of comprehensive input understanding. By requiring models to process and reason over entire demonstrations to accurately identify labels from a vast space, we push the boundaries of what is expected from long-context processing capabilities. This dual focus not only highlights the versatility of in-context learning across different applications but also catalyzes advancements in LLMs' ability to comprehend and interact with lengthy sequences of text.

The fusion of in-context learning with LLMs opens new avenues for addressing sophisticated natural language processing tasks. By articulating how in-context learning can revolutionize KBQA and provide a robust framework for evaluating long-context capabilities, this thesis lays the groundwork for future innovations in the field. Through the lens of these two applications, we not only showcase the potential of LLMs to transcend traditional limitations but also chart a course for their evolution, towards models that are increasingly adaptable, efficient, and capable of understanding the nuances of human language.

Chapter 2

Background

This section delves into the pivotal studies and frameworks that have shaped our understanding of in-context learning with large language models (LLMs), particularly focusing on knowledge base question answering (KBQA), reasoning with LLMs, and the nuanced domain of long in-context learning alongside the evaluation of LLMs for handling long-context inputs. Each of these areas contributes to the foundation upon which this thesis is built, highlighting the innovative strides made in the field and identifying the gaps that our research aims to fill.

2.1 In-context Learning with Large Languag Models

In-context learning has revolutionized the application of LLMs across various NLP tasks [10], showcasing remarkable few-shot performance on question answering [23], information extraction [32], and numerical reasoning [55], etc. This capability allows LLMs to perform tasks based on a small set of examples provided directly in their input context, eliminating the need for explicit retraining.

2.1.1 Exploratory Study on In-context Learning with LLMs

Previous works have deepen the understanding of the underlying principles of this phenomenon. For instance, a study has highlighted the utility of designing prompts by pairing inputs with their corresponding labels [68]. Another line of works have analysed the performance of in-context learning with respect to the number of examples provided, along with

retrieving relevant examples to enhance prompt construction [62]. Other recent work takes investigation on how distinct types of explanations, instructions, and controls can affect zero-shot and few-shot performance [50]. These insights form the backbone of our methodology in applying in-context learning to both KBQA and the evaluation of long-context LLM capabilities.

2.1.2 Reasoning with LLMs

Recently, a variety of techniques have been developed to enhance the reasoning abilities of large language models (LLMs) [10, 48]. Among these, Chain of Thought Prompting (CoT) has proven effective by encouraging models to generate intermediate steps in their output, thereby improving reasoning accuracy [106]. Building on this concept, several approaches have introduced the direct creation of formal programs to tackle specific tasks, marking significant advancements in this area [21, 71, 34, 23]. Particularly notable in the context of question answering is the work on Binder, which involves prompting LLMs to perform text-to-SQL conversions to fetch answers from SQL databases [23]. However, simply utilizing SQL table headers to prompt LLMs falls short when confronted with the vast complexity of knowledge bases, which contain thousands of relationships and millions of entities. The proposed framework KB-BINDER addresses this gap through a novel approach that includes the generation of preliminary drafts and a subsequent schema binding process, effectively navigating the extensive search space of knowledge base. Therefore, our approach aligns with this trajectory by exploring generate-then-bind strategies in KBQA, pushing the boundaries of what LLMs can achieve in complex reasoning tasks.

2.1.3 Long In-Context Learning on LLMs

As the scale of pre-trained language models expands, in-context learning (ICL) has become increasingly popular for tackling diverse tasks without significant fine-tuning [30]. Research indicates that augmenting the number of example demonstrations can improve ICL outcomes [63, 109]. However, findings also suggest that overly long input prompts might reduce performance, as the capabilities of earlier large language models (LLMs) are limited by their training on sequences of maximum length[65]. Additionally, it has been observed that LLMs equipped with ICL may struggle with tasks that require detailed specifications, attributed to their limited proficiency in processing extensive texts [77]. In response to these challenges, recent studies have explored memory augmentation

and extrapolation methods, aiming to bolster ICL with a broader array of demonstrations, thereby enhancing model performance in handling complex tasks [59, 102].

2.2 Knowledge Base Question Answering

The majority of cutting-edge KBQA models rely on semantic parsing techniques, transforming natural language questions into corresponding logical forms within a knowledge base (KB) [52, 39]. This process confronts the challenge of navigating through an immense search space, exemplified by databases like Freebase, which is home to 45 million entities and 3 billion facts [9]. Recent advancements leverage the robust generalization capabilities of language models (LMs) to efficiently explore this vast, previously untapped space [20, 40, 114, 84]. These approaches offer greater data efficiency and improved handling of the search space compared to traditional methods based on the independent and identically distributed (i.i.d) assumption [115, 29]. Nonetheless, they typically necessitate thousands of labeled examples for fine-tuning, leaving the potential of few-shot KBQA relatively unexplored due to its perceived complexity and the formidable challenge of mastering the expansive search space with minimal training data. Among the few explorations into this domain, one notable attempt involves training a meta-model to adapt swiftly to new queries using a limited set of examples, though it still requires a substantial initial dataset, thus falling short of a genuine few-shot scenario [44]. In contrast, another concurrent study adopts a unique approach by harnessing the discriminative, rather than generative, capabilities of LLMs for few-shot KBQA [37]. Our work introduces a pioneering effort to facilitate true few-shot learning in KBQA through a generate-then-bind strategy with LLMs, potentially unlocking new avenues for efficient and practical KBQA in settings constrained by data availability.

2.3 Long Context LLMs

This subsection ventures into the domain of long-context LLMs, highlighting research focused on overcoming the challenges and developing techniques for processing extensive textual sequences. We examine the latest methodologies employed by contemporary long-context LLMs, aimed at improving their capacity for managing, reasoning with, and generating lengthy text passages. Furthermore, we assess the evaluation mechanisms and benchmarks designed to measure these models' efficacy and constraints when dealing with long texts. A pivotal element of our discussion is the task of extreme-label classification,

a naturally long-context scenario necessitated by its extensive label array. This task critically evaluates LLMs' understanding abilities and acts as a proving ground for refining long-context processing techniques. Our goal is to shed light on the progress and persisting obstacles in utilizing LLMs for sophisticated, long-range text comprehension and production tasks.

2.3.1 Long Context Techniques on LLMs

The performance of Transformer-based models faces limitations due to the quadratic rise in computational demands as sequence lengths increase, especially when processing inputs with extensive contexts. To overcome this obstacle, recent research has introduced a variety of strategies. Some efforts focus on further fine-tuning LLMs with extended context lengths, thereby conditioning these models to better handle elongated sequences [81, 99]. Meanwhile, other initiatives employ methods like position extrapolation and interpolation, utilizing relative rotary positional embeddings, to increase the range of input lengths that models can process post-training [87, 78, 19]. To address computational challenges, techniques including the use of sliding memory windows and segmenting input into chunks have been suggested [42, 79, 129]. In addition, novel model architectures that deviate from the traditional Transformer framework, such as selective-state-space models — a variant of recurrent neural networks — have been investigated for their innate suitability for long-input processing [75, 36]. Collectively, these varied techniques aim to significantly improve the efficiency of LLMs in managing long-context information.

2.3.2 Long Context Evaluation

Responding to the critical need for evaluating long-range capabilities in large language models, a suite of benchmarks dedicated to long-context assessments has been developed. The Long-Range Arena benchmark features tasks with sequences spanning from 1,000 to 16,000 tokens, designed to test the efficiency of fast Transformers in processing long texts[94]. LongBench includes 21 bilingual datasets across six task categories, with an average text length of 6,000 words, formatted for streamlined [6]. The L-Eval Benchmark offers support for 20 different sub-tasks, featuring input lengths that range from 3,000 to 200,000 tokens [3]. LooGLE targets summarization and various long dependency question-answering tasks, with instances surpassing 100,000 words [57]. The most recent addition, ∞Bench, presents 12 tasks derived from realistic, auto-generated, and manually annotated datasets, with an average text length of 200,000 tokens [124]. While these benchmarks

assess a broad spectrum of capabilities, none specifically address the unique challenges posed by long in-context learning within vast label spaces, a stark contrast to traditional long-document comprehension or synthetic tasks like needle in a haystack. To bridge this gap, we introduce our benchmark, LongICLBench, aimed at providing a thorough long-context evaluation framework for LLMs, particularly focusing on their performance in scenarios requiring extensive in-context learning with a wide range of labels.

2.3.3 Extreme-label Classification

Extreme-label classification entails assigning data to one among a significantly large set of labels, which is a process integral to numerous practical fields such as text-based emotion detection, named entity extraction, and the prediction of biological functionalities [125, 85, 25, 27]. These applications demand accurate classification within expansive label spaces. Approaches to addressing extreme-label classification are varied, spanning from embedding-oriented methods to refined retrieval techniques [8, 101], all aimed at effectively navigating and utilizing broad label landscapes. However, the incorporation of these tasks with long-context large language models introduces unique challenges. The extensive nature of label spaces in extreme-label classification poses a substantial challenge to the in-context learning abilities of LLMs, which must accurately distinguish between closely related labels over lengthy textual inputs [66]. Given these intricacies, our proposed benchmark, LongICLBench, with its increasing levels of difficulty, can offer an ideal platform for assessing the proficiency of LLMs in long-context comprehension.

Chapter 3

Few-show KBQA with In-context Learning

In this section, we introduce KB-BINDER, a novel approach facilitating training-free, few-shot knowledge base question answering(KBQA) with the utilization of in-context learning, which markes a significant advancement in the field. Our method unfolds in two stages, beginning with the generation of a preliminary draft from a small set of KBQA question-and-answer pairs with in-context learning. This draft, while initially imprecise in its entities and relations, serves as a foundational step. Subsequently, KB-BINDER refines this draft through a lexicon-based similarity search across the KB, aligning preliminary entities and relations with their accurate counterparts to produce refined logical forms. These are then executed against the KB to derive answers. In addition, we also develop an enhanced version, KB-BINDER-R, to incorporate additional exemplars for improved accuracy.

Unlike prior approaches that depend on specific heuristics tailored to the KB schema, KB-BINDER leverages the broad applicability of LLMs, eliminating the need for such heuristics. After introducing the framework design in details, we will present the performance of KB-BINDER across four public datasets. These results indicate that in-context learning can resolve complicated reasoning problems after integrating with binder mechanism.

3.1 Methodology

In this first exploration for the reasoning capability of in-context leaning, our KB-BINDER utilizes a large language model (LLM) to generate an initial logical form or draft in response

to a new question. These drafts, while not immediately executable due to their preliminary nature and potential deviation from the specific vocabulary and structure of the knowledge graph, provide valuable insights into the semantic relationships between entities. This insight effectively narrows down the search for accurate entities and schema terms, which are essential for refining the draft into an executable logical form.

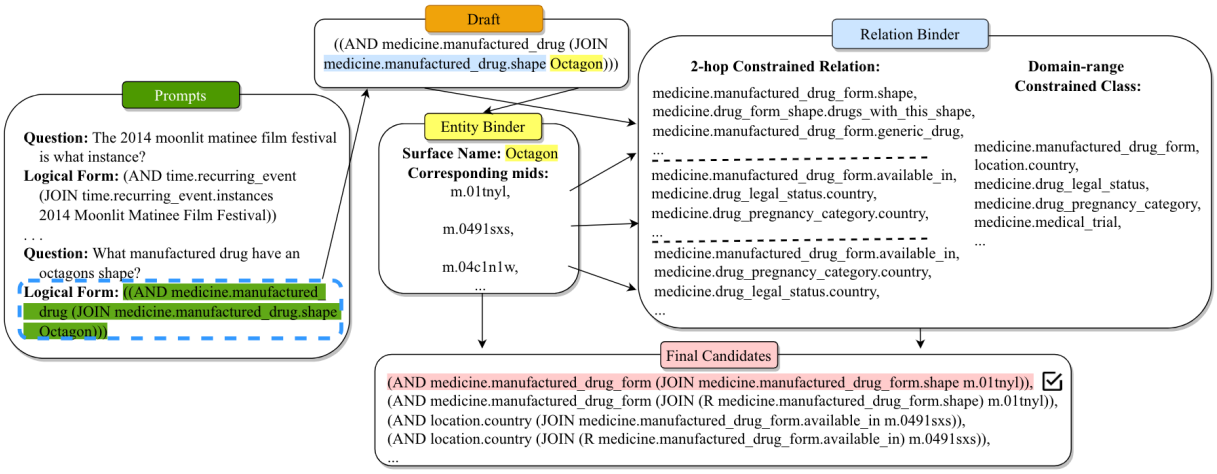


Figure 3.1: KB-BINDER framework: Given a question, the LLM will first generate its corresponding preliminary logical forms as the drafts, imitating the exemplary demonstration. Then the entity and relation binders will operate on the drafts to ground the entities and relations on KB respectively, which produces the final candidates.

3.1.1 Drafts Generator

Utilizing Codex's in-context learning ability, we create drafts by presenting the LLM with random samples from the training set, formatted as question and logical form pairs as shown in Prompts block in Figure 3.1. It's important to highlight the challenge posed by machine identifiers (MIDs) in these forms, which lack interpretability. By replacing MIDs with their descriptive names in our prompts, we facilitate a more intuitive understanding for the LLM, enhancing its ability to produce semantically rich and structurally coherent drafts.

3.1.2 Knowledge Base Binder

The next step involves binding the draft with specific and truly-existed details from the knowledge base (KB), so that it can be actually executed against the target knowledge base. This process is as the follows:

Entity Binder: To accurately determine the machine identifiers (MIDs) for entities referenced in questions, our process begins with extracting the entities' surface names essentially from the preliminary drafts generated by the LLM. Initially, we look for a direct match between the extracted surface names and the friendly names of entities within the knowledge base. If there's a match, we gather all corresponding MIDs for these friendly names. From this collection, we select the most frequently occurring MIDs, based on a frequency count (FACC1), to ensure we're considering the most relevant entities. In cases where a surface name doesn't directly match any friendly name in the knowledge base, we turn to BM25, which is a text retrieval technique to find the closest resembling entity within the KB. This identified entity serves as a pivot or anchor to facilitate the extraction of potential MID candidates, essentially guiding us toward the most plausible matches based on the similarity. When the draft mentions several entities, we address each surface name separately to identify potential MIDs for each one. This step ensures that we accurately represent each entity in the question. After identifying MIDs for individual entities, we explore all possible combinations of these MIDs to cover different ways the entities might relate to each other in the context of the question.

Relation Binder: The preliminary relations that emerge in the drafts we generate are often not direct matches for relations within the knowledge base. However, we expect these initial relations to share a similar logical structure path and semantic essence with those that do exist within the knowledge base, inspired by the examples shown in the prompts. We take each mentioned relation from the draft and pair it with the original question to form a search query. This query is then used with the BM25 algorithm to find the most analogous relations within the entire collection of the knowledge base's relations. To increase the chances that our logical form will execute successfully, we focus on the most promising relations. Specifically, we look only at two-hop relation items that can connect entities through a sequence of two relations and select the top m relations based on their relevance scores from the BM25 search. This selection is made from the pool of relations connected to the current set of entity MIDs we select in Entity Binder. Relations that don't fit within this two-hop framework are set aside, keeping only those most likely to form a executable logical form. For every possible combination of entities, we explore all m top-retrieved relation candidates. This exhaustive approach ensures that we consider all viable ways the entities could be interconnected according to the knowledge base's structure,

significantly improving our chances of constructing an accurate and executable logical form. By applying this generate-then-bind method, we bind vague, preliminary relations into precise, knowledge base-compatible relations. This transformation is essential for moving from a draft that only approximates the answer to a fully formed query that can retrieve accurate information from the knowledge base. It also provides a general framework for in-context learning to be applied to KB-dependent or other analogous applications.

3.1.3 Majority Vote

Following our generate-then-bind method, each draft we produce is associated with numerous possible logical forms. Each of these forms can be translated into a SPARQL query, which is then run against the knowledge base (KB) to fetch answers. We meticulously collect all logical forms that yield answers, alongside the answers themselves. Given that maintaining self-consistency in the model's predictions can significantly enhance their reliability [104], we iterate this process K times for the top K candidates obtained from in-context learning. Through this repetition, we employ a majority voting mechanism to determine the most consistently obtained answer and its corresponding logical form. This technique ensures that our final selection is not just a random pick but is supported by a pattern of recurrence across multiple attempts. We refer to this version of the model, which emphasizes self-consistency across the top K drafts, as KB-BINDER(K).

3.1.4 Retrieved Exemplars

. To enhance our approach without additional training, we develop a variant called KB-BINDER(K)-R. Unlike the original version where the examples in demonstration prompt are chosen at random, KB-BINDER(K)-R uses the BM25 algorithm to find the N most similar questions to the one being asked. This ensures that, in the demonstrations, the logical forms of these N questions are closely related to or exactly match the schema of the target question. By doing so, we're more effectively targeting the specific elements needed to answer the question, particularly benefiting questions that are independent and identically distributed (I.I.D.). This tailored selection process is designed to significantly improve our method's performance by focusing on examples that are directly relevant to the question at hand.

Dataset	Train	Dev	Test
GrialQA	44,337	6,763	13,231
WebQSP	3,098	–	1,639
GraphQA	2,381	–	2,395
MetaQA-1hop	96,106	9,992	9,947
MetaQA-2hop	118,980	14,872	14,872
MetaQA-3hop	114,196	14,274	14,274

Table 3.1: Dataset statistics.

3.2 Experiments

In this section, we provide an overview of the benchmarks utilized to assess the effectiveness of our framework. We detail the specific configuration of KB-BINDER and present its performance on various datasets, comparing it to that of fully-trained baseline models. Finally, we analyze how different design decisions impact performance, exploring the reasons behind these effects.

3.2.1 Datasets

We evaluate KB-BINDER on four public KBQA datasets as follows:

GrailQA [38] is a diverse KBQA dataset built on Freebase. It encompasses a vast array of knowledge, with 32,585 entities and 3,720 relations across 86 domains. GrailQA is carefully designed to evaluate KBQA models across three levels of generalization: identical instance distribution (I.I.D.), compositional, and zero-shot, offering a comprehensive test of model adaptability.

GraphQA [88] Like GrailQA, GraphQA spans a broad spectrum of domains, generated through sentence-level paraphrasing from graph queries. It specifically aims to test a model's ability to understand and generalize across compositional queries, challenging the model's capacity to handle complex, multi-step reasoning.

WebQSP [116] Derived from the WebQuestions dataset and answerable through Freebase, WebQSP focuses on i.i.d. generalization but with simpler questions. This dataset provides a benchmark for evaluating how well a model can handle straightforward queries.

MetaQA [126] Centered around a movie ontology taken from the WikiMovies Dataset, MetaQA features question-answer pairs across three tiers of complexity, assessing a model's performance within a specialized domain.

Table 3.1 offers a detailed breakdown of the training, development, and test splits for these datasets. Our evaluation is conducted on all the available test sets, and we further explore the parameter setting of KB-BINDER through ablation studies on a subset of GrailQA's development set, consisting of 500 examples selected at random.

3.2.2 Baselines

We benchmark our method against models listed on the official leaderboards of each dataset, taking their results directly from their respective publications and using the same evaluation criteria for a fair comparison. It's important to note that all the baseline methods we compared against have used the full training dataset for supervision.

3.2.3 Implementation Details

During the draft generation phase, we utilize the `code-davinci-002`[1] model from the OpenAI API to create the top K drafts for each question. We explore scenarios where $K = 1$ and $K = 6$, which we denote as KB-BINDER(1) and KB-BINDER(6), respectively. Specifically, we randomly select $N = 100$ example questions from the training datasets of both WebQSP and GraphQA. For GrailQA, due to its extensive test dataset and longer inference times, we limit our sample to $N = 40$ exemplars. While MetaQA necessitates only 5 sample questions for demonstrations due to its smaller knowledge base. Each experiment is repeated three times, with the average performance being reported. In the binding phase, we set $n = 15$ for the entity binder across all questions. To match originally unmatched friendly names and identify the top relation items, we employ a combination of BM25 and Contriever[45] provided by Pyserini[2] as a hybrid search approach. Once we've globally ranked the relations, our focus shifts to those that can be connected through 2-hop relations from the identified entities. We examine the top 10 relation candidates ($m = 10$) within this 2-hop limit for GrailQA, WebQSP, and GraphQA, and only the top candidate ($m = 1$) for MetaQA. Following this process, the generated drafts, which are linked with potential candidates, are converted into SPARQL queries for execution on the Virtuoso server following the instructions[3].

[1] https://openai.com/blog/openai-codex/
[2] https://github.com/castorini/pyserini
[3] https://github.com/dki-lab/Freebase-Setup

Method	Overall	
	EM	**F1**
GloVe + Transduction [38]	17.6	18.4
QGG [53]	-	36.7
BERT + Transduction [38]	33.3	36.8
GloVe + Ranking [38]	39.5	45.1
BERT + Ranking [38]	50.6	58.0
ReTraCk [20]	58.1	65.3
S^2QL [119]	57.5	66.2
ArcaneQA [40]	63.8	73.7
RnG-KBQA [114]	68.8	74.4
DecAF [118]	68.4	78.7
TIARA [84]	**73.0**	**78.5**
Few-shot in-context		
KB-BINDER(1)	47.0	51.6
KB-BINDER(6)	50.6	56.0
KB-BINDER(6)-R	53.2	58.5

Table 3.2: 40-shot Results of KB-BINDER/KB-BINDER-R and baselines on GrailQA.

3.2.4 Main Result

In our evaluation, we show how KB-BINDER performs across the test sets of four distinguished public KBQA datasets, as detailed in Tables 3.2, 3.3, 3.4, and 3.5 for GrailQA, WebQSP, GraphQA, and MetaQA, respectively. The model variant KB-BINDER(1) operates in a default setting, generating the top draft for further binding step, while KB-BINDER(6) applies a mass voting strategy across the top six drafts for enhanced self-consistency. An advanced version, KB-BINDER(6)-R, incorporates retrieved exemplars for further optimization, as discussed in Section 3.1.4.

For all datasets, each version of KB-BINDER demonstrates robust performance. Specifically, we observe that KB-BINDER(6) typically surpasses KB-BINDER(1), confirming our initial hypotheses about the efficacy of leveraging mass vote mechanism. Moreover, KB-BINDER(6)-R often boosts performance even more significantly. Notably, our few-shot methodology competes with, and in some cases even exceeds, the fully-supervised state-of-the-art (SOTA) performances on WebQSP, GraphQA, and MetaQA. It also delivers competitive results against a BERT-ranking baseline on GrailQA [38].

Method	F1
ReTraCk [20]	71.0
QGG [53]	74.0
ArcaneQA [40]	75.6
PullNet [89]	62.8
RnG-KBQA [114]	75.6
TIARA [84]	76.7
DecAF [118]	**78.8**
Few-shot in-context	
KB-BINDER(1)	52.5
KB-BINDER(6)	53.2
KB-BINDER(6)-R	74.4

Table 3.3: 100-shot Results of KB-BINDER/KB-BINDER-R and baselines on WebQSP.

Focusing on GrailQA, as shown in Table 3.2, KB-BINDER(6) achieves a 50.6 EM score with just 40 examples, equating to the BERT + Ranking setting, which was fine-tuned with approximately 45k annotations. This parity in performance showcases our model's superior generalization on compositional and zero-shot queries as highlighted in Table 3.6. For GraphQA and MetaQA, KB-BINDER(1) and KB-BINDER(6) outperform previous SOTAs by significant margins, as detailed in Tables 3.4 and 3.5, underscoring KB-BINDER's adeptness in specific scenarios. In the instance of GraphQA, the dataset is characterized by a limited number of training examples, totaling 2,381. However, the questions within the test set are predominantly compositional. This compositionality poses a challenge for models that have been fine-tuned on the available data, making it difficult for them to adapt to new combinations of schema items. On the contrary, large language models (LLMs) find it comparatively easier to generalize in such scenarios, thanks to their broad pre-training corpus [11, 49]. Regarding MetaQA, which draws from the relatively concise WikiMovies knowledge base featuring only a handful of unique relations within a single domain, the alignment between the demonstration context and the target questions is direct and precise. Consequently, just five demonstrations are sufficient for an LLM to produce highly accurate initial relation candidates, illustrating the effectiveness of LLMs in scenarios with tightly defined knowledge domains.

However, improvements with KB-BINDER(K)-R are not uniform across all datasets. For example, on GrailQA, the increase is relatively small, and on GraphQA, performance slightly declines. Yet, on WebQSP, KB-BINDER(K)-R significantly elevates the F1 score. This variability underscores the distinct characteristics of each dataset and the complex

Method	F1
AUDEPLAMBDA [80]	17.7
SPARQA [91]	21.5
BERT + Ranking [38]	25.0
ArcaneQA [40]	31.8
Few-shot in-context	
KB-BINDER(1)	39.3
KB-BINDER(6)	**39.5**
KB-BINDER(6)-R	38.7

Table 3.4: 100-shot Results of KB-BINDER/KB-BINDER-R and baselines on GraphQA.

Method	1-hop	2-hop	3-hop
KV-Mem [67]	96.2	82.7	48.9
VRN [126]	**97.5**	89.9	62.5
GraftNet [90]	97.0	94.8	77.7
PullNet [89]	97.0	**99.9**	91.4
Emb [82]	**97.5**	98.8	94.8
NSM [43]	97.1	**99.9**	98.9
Few-shot in-context			
KB-BINDER(1)	93.5	99.6	96.4
KB-BINDER(1)-R	92.9	**99.9**	**99.5**

Table 3.5: 5-shot Results of KB-BINDER/KB-BINDER-R and baselines on MetaQA.

interplay between dataset specificity and the effectiveness of retrieved exemplars.

In essence, the detailed performance analysis across these datasets reveals that in-context learning approaches like KB-BINDER(K) can match or even surpass fully-trained SOTAs in KBQA tasks, particularly when faced with limited training data or when tasks require nuanced multi-hop reasoning. This performance highlights both the strengths and areas for improvement in applying in-context learning to diverse KBQA or similar challenges.

Method	IID		Compositional		Zero-shot	
	EM	**F1**	**EM**	**F1**	**EM**	**F1**
GloVe + Transduction [38]	50.5	51.6	16.4	18.5	3.0	3.1
BERT + Ranking [38]	59.9	67.0	45.5	53.9	48.6	55.7
RnG-KBQA [114]	86.2	89.0	63.8	71.2	63.0	69.2
TIARA [84]	**87.8**	**90.6**	69.2	76.5	**68.0**	**73.9**
Few-shot in-context						
KB-BINDER(6)	51.9	57.4	50.6	56.6	49.9	55.1
KB-BINDER(6)-R	72.5	77.4	51.8	58.3	45.0	49.9

Table 3.6: Results of KB-BINDER/KB-BINDER-R and baselines on different question types of GrailQA.

3.2.5 Ablation Study

To analyze how the number of examples shown during the draft generation affects the final performance, we carried out ablation studies. Limited by the long inference time required to process all test questions, we assessed our model's performance on a subset of 500 questions randomly selected from the GrailQA development set. We varied the number of few-shot exemplars from 1 to 100, analyzing both the coverage and the EM score for each configuration. Coverage is defined as the proportion of questions that can be associated with at least one executable logical form out of the total in the subset. The results, illustrated in Figure 3.2, clearly indicate a positive correlation between the number of examples and improvements in both coverage and EM score.

Additionally, we examined how the performance of KB-BINDER(K) changes with different numbers of top drafts generated by Codex for majority voting. With 40 examples for reference, the outcomes depicted in Figure 3.3 suggest that increasing drafts from 1 to 6 boosts coverage by 19% and the EM score by 5.6%. It implies that more drafts allow for a wider variety of logical form structures and schema item formats to be initially considered.

However, higher numbers of examples and drafts lead to longer inference times and higher costs for operating KB-BINDER. Consequently, we only report results based on 40 exemplars and the top 6 drafts for GrailQA, highlighting the inherent trade-off between accuracy and computational resource. This suggests the potential for further enhancing KB-BINDER's performance by adjusting these parameters.

Further analysis in Table 3.6 reveals a significant disparity in EM scores between I.I.D. questions and other types among fully supervised models, with scores dropping between 10

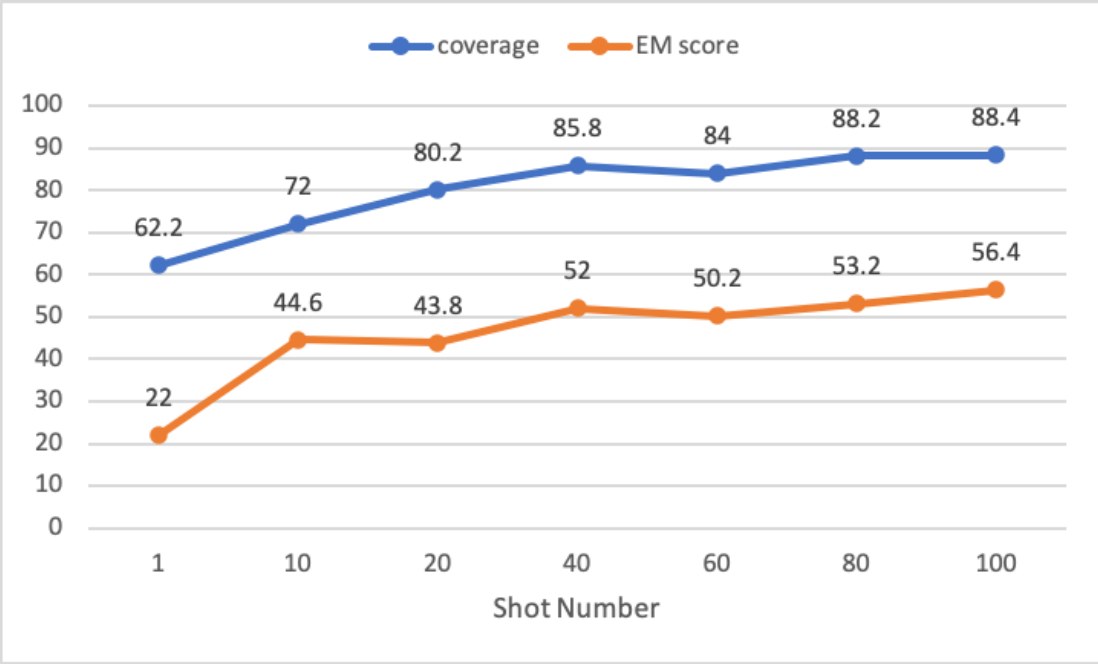


Figure 3.2: KB-BINDER coverage and EM scores trend with shot number.

to 47.5 points. In contrast, KB-BINDER maintains consistent performance across different question types. This is due to the fact that only part of the target questions of I.I.D have been seen in the demonstration for few-shot setting, so there is rarely bias among the three types.

3.2.6 Case Study

In Figure 3.4, we present examples of both successful and unsuccessful outcomes from the KB-BINDER output. For instance, Question P1 showcases a scenario where the logical form produced aligns perfectly with the expected target. Moreover, Question P2 illustrates a situation where, despite correct logical structuring, the draft includes fabricated entity names and relations, necessitating further steps to pinpoint the correct executable logical form. Conversely, Question N1 represents a case where the draft's logic is incorrect. Meanwhile, Question N2 demonstrates accurate draft logic yet unfortunately associates with incorrect entities or relations.

We evaluated the efficacy of individual components within our pipeline, specifically

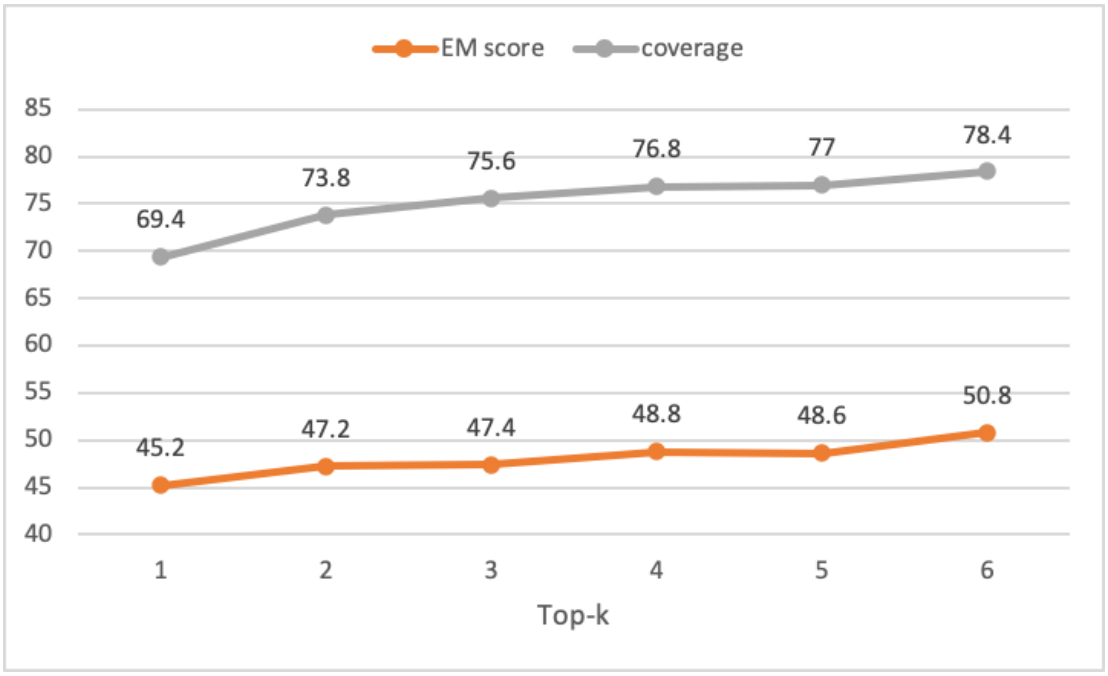


Figure 3.3: KB-BINDER coverage and EM scores trend with top K self-consistency.

focusing on the accuracy of identifying correct MIDs and relations (recall) before and after applying the Entity and Relation Binders. Additionally, we examined the accuracy of the logical framework established in the draft. On a subset of 500 questions from the GrailQA development set, using 40 examples (shots), KB-BINDER(1) achieved recall of 0.9 for entity binding and 0.78 for relation binding, with the logical framework's recall at 0.66 for the top draft, which primarily account for the errors observed. Comparing these results to those obtained prior to the application of the binders under the same conditions, we noted initial recall rates of 0.78 for MIDs and 0.0 for relations. The introduction of our Entity and Relation Binders led to improvements of 12% and 78% in recall rates for MIDs and relations, respectively, revealing the significant impact of these components on enhancing model performance.

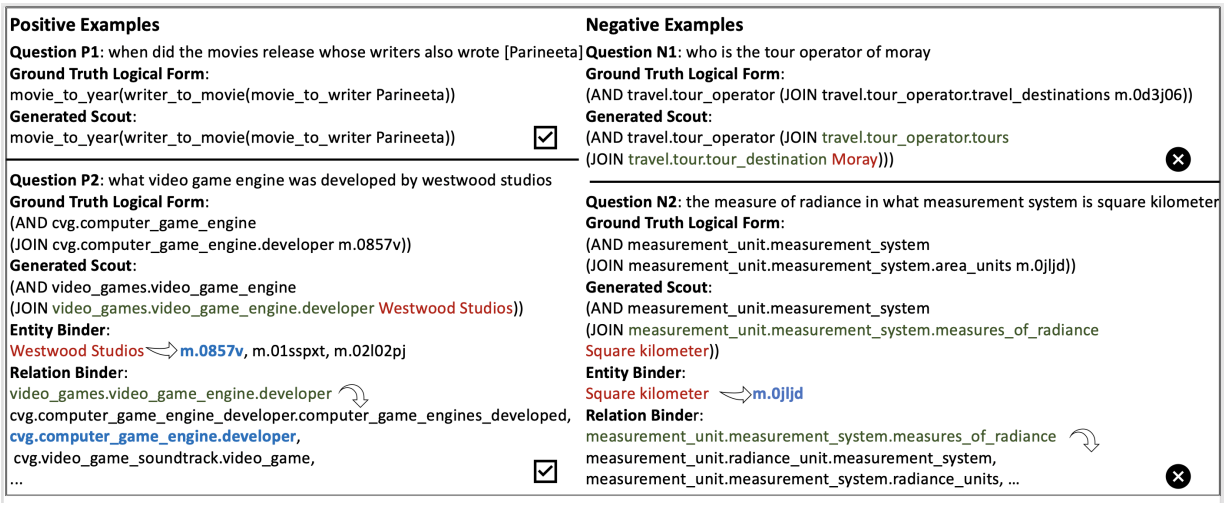


Figure 3.4: Positive and negative examples generated by KB-BINDER.

Chapter 4

Long LLMs evaluation with In-context Learning

After enabling in-context learning for complex reasoning task like KBQA, we further explore its possibility on long-context evaluation as large language models have already entered the era of long texts. We propose to utilize in-context learning for extreme-label classification tasks as a means to evaluate the capabilities of long-context LLMs. This approach differs from previous tasks by requiring LLMs to thorough the entire input to grasp the extensive label space, thus demanding a comprehensive understanding from the models for accurate predictions. The large size of the label space often naturally results in long task demonstrations. For instance, the Discovery dataset includes 174 classes, with each example averaging 61 tokens, leading to demonstrations exceeding 10,000 tokens for just one example per class. Typically, LLMs require more than a single example per class to accurately distinguish among the subtle differences of such a broad label spectrum, making this task an ideal benchmark for evaluating long-context comprehension.

To methodically explore how these capabilities impact model performance in extreme-label text classification with in-context learning, we curate a benchmark as LongICLBench, which comprises six tasks of different difficulty, categorized by context length and label space complexity. Our evaluation of 13 long-context LLMs reveals a general trend where model performance decreases as tasks become more challenging, notably with the need for longer demonstrations, as illustrated in Figure 4.1. Some models, such as Qwen and Mistral, show a linear decline in performance relative to the increase in input length. However, most models demonstrate potential benefits from detailed demonstrations within a certain threshold. Beyond this point, extended inputs may detract from or destabilize performance, as depicted in Figure 4.2. Further analysis into the distribution of label

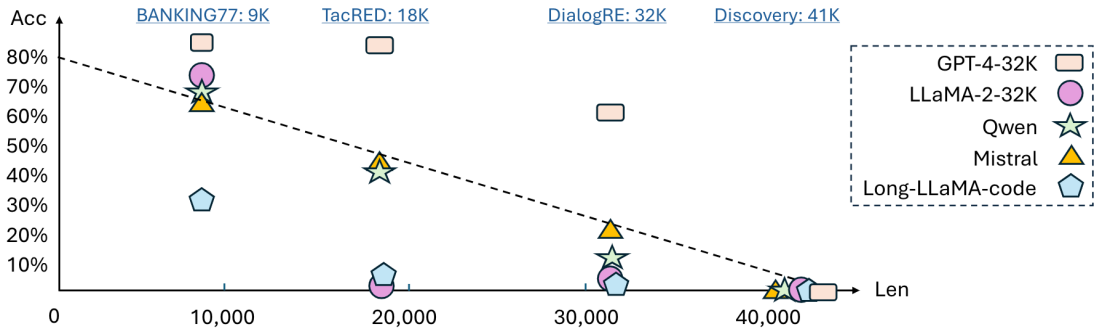


Figure 4.1: Results for representative models across different evaluation datasets. The performance greatly decreases as the task becomes more challenging. Some models even decay linearly w.r.t the demonstration length.

Dataset	Task Type	# Classes	# Tokens/Shot	# Total Tokens
GoEmotion	Emotion Classification	28	28	[1K, 4K]
BANKING77	Intent Classification	77	28	[2K, 11K]
TacRED	Relation Extraction	41	80	[4K, 18K]
Few-NERD	Entity Recognition	66	61	[5K, 23K]
DialogRE	Relation Extraction	36	226	[8K, 32K]
Discovery	Discourse Marker Classification	174	61	[10K, 50K]

Table 4.1: Statistics of the collected sub-dataset in LongICLBench. We evaluate from 1-shot/label to 5-shot/label, which results in the shown #total token range.

positions sheds light on how this aspect significantly impacts the long in-context learning capabilities of the models, including those like GPT-4 turbo, indicating that the position of instances in the prompt crucially influences performances for some models.

In this chapter, we will introduce the benchmark, evaluated models, evaluation results and the exploratory study regarding to the arrangement of instances in demonstrations.

4.1 Long In-context Benchmark

To facilitate the assessment of long in-context learning across extreme-label classification tasks spanning various domains and levels of difficulty, we compile six datasets with context lengths ranging from short to long. To achieve a balance between sequence token length

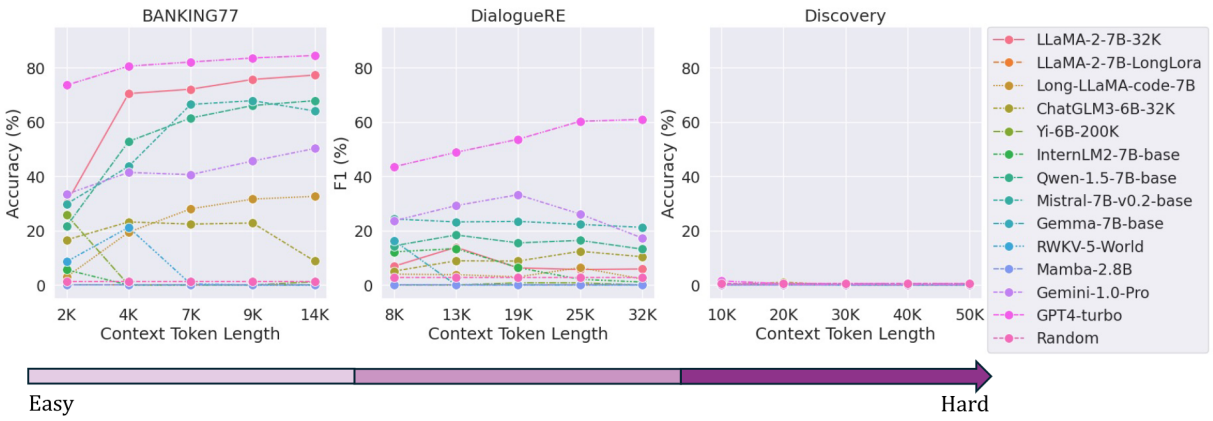


Figure 4.2: LLM performance on long in-context benchmark across different lengths. We curate datasets with different difficulty levels. As we increase the difficulty of the dataset, LLMs struggle to understand the task definition and suffer from significant performance degradation. On the most difficult Discovery dataset, none of the LLMs is able to understand the long demonstration, leading to zero accuracy.

and the objective of evaluating long in-context learning, we evenly sample the examples from each class to create evaluation sets designed for 1 to 5 rounds, with each round comprising a comprehensive set of examples that include all the unique labels. We aim to minimize bias related to label distribution by evenly sampling instances across these classes. Detailed statistics for these datasets are provided in Table 4.1.

GoEmotions [25] is the largest manually annotated dataset of 58k English comments from Reddit, which is labeled into 27 emotion categories or Neutral. There are 27 types of emotion types and drop the rare ones with few examples. Each selected example contains 28 tokens on average.

BANKING77 [13] is focused on intent detection within the banking sector, containing 13,083 annotated examples spanning 77 intents. We retain all intent types for this dataset, with individual instances averaging about 28 tokens.

Few-NERD [27] is a large-scale human-annotated name entity recognition dataset with a hierarchy of 8 coarse-grained and 66 fine-grained entity types. Each of the instances is a paragraph with approximately 61 tokens on average and contains one or multiple entity names as the ground truth answer. There are 66 types of entities in the collection.

TacRED [125] is a comprehensive relation extraction dataset featuring 106,264 examples derived from news and web texts, where each sentence is labeled with a single relation

out of 41 possible types. The average length per example is 80 tokens, making it a good resource for relation-type classification.

DialogRE [117] is an annotated dataset based on dialogue, specifically 1,788 dialogues from the popular American TV show Friends. It identifies 36 potential relation types between pairs of arguments within the dialogues, with an average token count of 226 per example. It offers a unique challenge in understanding relational dynamics in conversational contexts.

Discovery [85] stands out for its emphasis on discourse markers, which are discovered automatically across sentence pairs to form a substantial collection containing 174 discourse markers. Each discourse marker has at least 10,000 examples. The average token count per example is 61, making it the most challenging dataset due to its fine-grained label distinctions.

4.2 Model and Experimental Setup

Model	Size	Initialization	Strategy	Train	Support
Gemma-7B-base [96]	7B	Gemma	RoPE + LF	8K	8K
LLaMA-2-7B-32K [65]	7B	LLaMA-2	Position Interpolation	32K	32K
ChatGLM3-6B-32K [121]	6B	ChatGLM	Position Encoding Scheme	32K	32K
Qwen-1.5-7B-base [5]	7B	Qwen	NTK-Aware Interpolation	32K	32K
Mistral-7B-v0.2-base [46]	7B	Mistral	LF	32K	32K
LLaMA-2-7B-LongLora [22]	7B	LLaMA-2	Shifted Short Attention	100K	100K
Yi-6B-200K [2]	6B	Yi	Position Interpolation +LF	200K	200K
InternLM2-7B-base [12]	7B	InternLM	Dynamic NTK	32K	200K
Long-LLaMA-code-7B [99]	7B	LLaMA-2	Focused Transformer	8K	256K
RWKV-5-World [75]	3B	RWKV	Attention-free Model	4K	∞
Mamba-2.8B [36]	2.8B	Mamba	State Space Model	2K	∞
Gemini-1.0-Pro [95]	-	Gemini	Ring Attention	32K	32K
GPT4-turbo [1]	-	GPT-4	-	-	128K

Table 4.2: The overview of the evaluated models. We utilize base models before instruction-tuning except Gemini and GPT4-turbo. LF means fine-tuning the model on longer-context corpus after pre-training.

In our investigation of in-context learning within the realm of extreme-label classification, we carry out an in-depth evaluation of 13 current long-context language models, each

of which has approximately 7 billion parameters. This evaluation includes cutting-edge models like Gemini and GPT-4-turbo. Table 4.2 outlines the models under review, emphasizing the architectural advancements they introduce to accommodate the long contexts. It's evident from this review that various approaches have been adopted to increase the context window capacity of these models, with some designed to support training within larger context windows and others capable of length extrapolation. Notably, RWKV [75] and Mamba [36], which adopt RNN-like architectures, aim to simplify the computational demands of attention mechanisms, thereby facilitating the processing of much longer inputs with improved efficiency in terms of time and memory.

For each dataset, we design a prompt based on a standardized template, detailed in Table 4.3. This allows us to conduct a fair comparison across both open-source models and those available via API, using input sequences of varying lengths. We ensure a balanced representation of examples for all models by selecting an even distribution of labels for the in-context demonstrations. For instance, a single round of input will cover a complete set of examples across all label types, while five rounds will revisit each label type five times. Our test sample consists of 500 examples from each dataset's test set, with careful attention to maintaining an equitable distribution of label types. The open-source models are accessed through HuggingFace[1], whereas the API-based models are utilized as per instructions in their official documentation[2].

4.3 Evaluation Results

The core findings of our evaluation are presented across Tables 4.4, 4.5, 4.6, and 4.7, 4.8, 4.9. We apply the F1 score as a measure for datasets focusing on entity recognition and relationship extraction, whereas accuracy serves as the metric for other types of datasets. The results generally indicate that Transformer-based models outperform their RNN counterparts across all datasets tested. Nevertheless, none of these models can match the performance levels of sophisticated API-based models, such as GPT-4-turbo.

For simpler tasks like BANKING77, which has context lengths ranging from 2K to 14K tokens across 1 to 5 rounds, most models see improvements with longer contexts and more demonstration examples. As depicted in Figure 4.2 and Table 4.4, accuracy either significantly increases or drastically drops for most open-source models when moving from

[1] https://huggingface.co

[2] https://platform.openai.com/docs/guides/text-generation/chat-completions-api, https://cloud.google.com/vertex-ai/generative-ai/docs/multimodal/overview

2K to 4K tokens. Beyond three rounds, the addition of more examples yields either little or diminishing returns.

In contrast, for more complex tasks such as TacRED and DialogRE, detailed in Tables 4.5 and 4.6, which demand a deeper understanding of long contexts, the performance of all few-shot models declines compared to BANKING77. As shown in Figure 4.2, GPT-4-turbo is the exception, continually benefiting from additional demonstrations, while other models peak with context lengths around 20K tokens.

The Discovery dataset presents an extreme challenge with its vast label space of 174 classes, which illustrates the limits of current models. A single round covering all label possibilities already accumulates a context length of 10K tokens, where all models, including GPT-4-turbo, struggle to differentiate among the nuanced categories, resulting in a score of 0. This outcome implies that the capabilities of models to process different task vary depend on the complexities, suggesting a threshold of complexity that even the most advanced LLMs, such as GPT-4-turbo, cannot surpass, positioned somewhere between the complexities of DialogRE and Discovery.

Another notable insight is the predictable performance pattern of some LLMs on extreme-label in-context learning tasks. As highlighted in Figure 4.1, the performance of models like Qwen and Mistral aligns almost linearly with the length of the demonstration. This pattern hints at a potential mathematical relationship between performance and task complexity in in-context learning scenarios.

4.4 Exploratory Study

Motivated by the Lost in the Middle phenomenon [65], we conduct analytical experiments to assess if the placement of instances within the prompt impacts performance in long in-context learning tasks, especially for extreme-label classification.

4.4.1 Scattered Distribution

Our exploratory work includes pilot studies using the TacRED dataset, characterized by medium complexity. We demonstrate each label type three times, resulting 123 unique instances. In the default scattered distribution, instances with identical labels are randomly dispersed, creating a scattered layout. We monitored the relative position of each instance within the prompt and its label, then calculated the accuracy for every label class. As

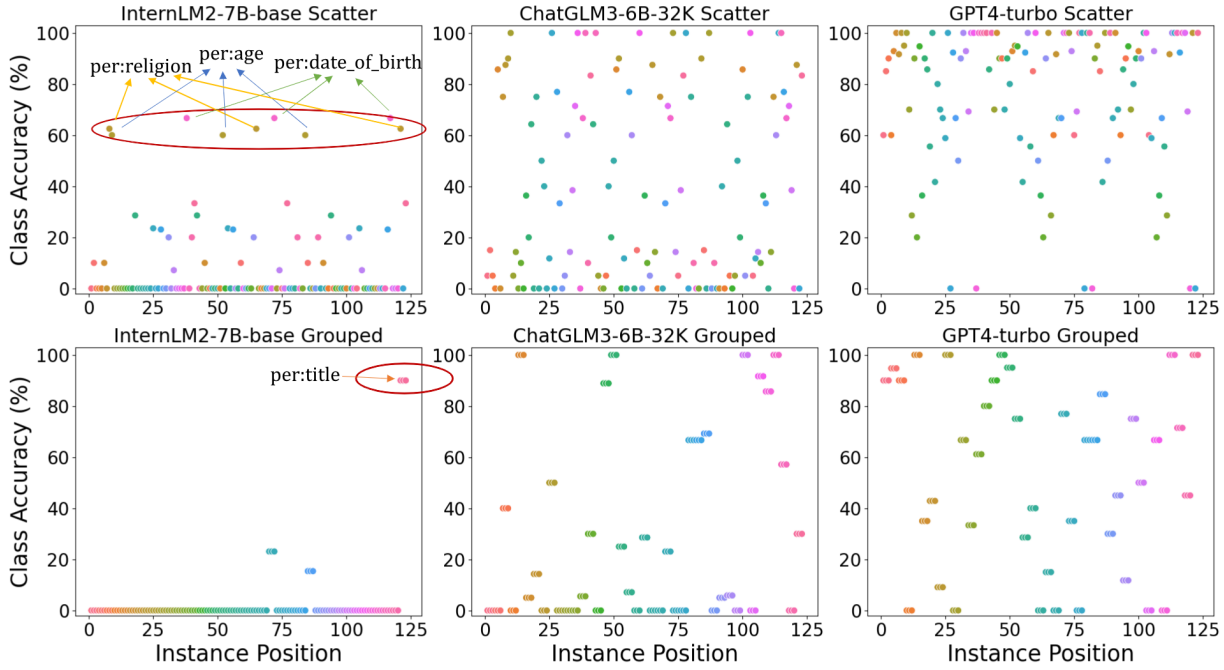


Figure 4.3: Visualization of accuracy for every class when instances from the same class are scattered V.S. grouped in the demonstration prompt.

depicted in the upper row of Figure 4.3, the accuracy visualization for each label, plotted against its position in the prompt, reveals performance variance across different label types. In conditions where instances are scattered, some models, like InternLM2-7B-base, achieve about 60% accuracy only on certain labels, as indicated by a red circle in Figure 4.3, regardless of where the instances are positioned. In contrast, models such as ChatGLM3-6B-32K exhibit strong performance across many labels. Notably, the GPT4-turbo model consistently achieves over 80% accuracy across most labels, with only a few exceptions.

4.4.2 Grouped Distribution

For a direct comparison between scattered and grouped distributions, we arranged instances of the same label to be near by each other in the demonstration prompts. Table 4.10 compares model performances before and after this reorganization, revealing a widespread decline in accuracy when instances are grouped by label. Models like Mistral-7B-v0.2-base and InternLM2-7B-base show significant drops in accuracy, indicating a particular sensitivity to how instances are arranged. Further investigation through visualizing accuracy

for grouped labels, as shown in Figure 4.3, illustrates that same-class instances, marked by color-coded dots, are positioned close to each other. It's clear that some models, such as InternLM2-7B-base, are highly sensitive to instance distribution, performing well only when labels are near the end of the prompt. Meanwhile, other models like ChatGLM3-6B-32K, despite a modest accuracy decline of 3.3%, demonstrate resilience against changes in instance distribution, maintaining similar performance. Interestingly, even the GPT4-turbo model is not immune to the impact of grouped distributions, experiencing a significant performance reduction of 20.3%. This trend of decreased performance does not correlate with the specific label positions within the prompt.

Dataset	Prompt
GoEmotion	Given a comment, please predict the emotion category of this comment. The prediction answer must come from the demonstration examples with the exact format. The examples are as follows: {comment: [comment] emotion category: [emotion]} × *repeat n times*
BANKING77	Given a customer service query, please predict the intent of the query. The predicted answer must come from the demonstration examples with the exact format. The examples are as follows: {service query: [service] intent category: [intent]} × *repeat n times*
TacRED	Given a sentence and a pair of subject and object entities within the sentence, please predict the relation between the given entities. The examples are as follows: {sentence: [sentence], the subject is [subject], the object is [object] the relation between the two entities is: [relation] } × *repeat n times*
Few-NERD	Given the sentence, please find the name entities in the sentence and their corresponding entity types in the strict format of the given examples as following (Entity: EntityType): {[entity]: [entity type]} × *repeat n times*
DialogRE	Given the dialogue, please find the name pair entities in the dialogue and their corresponding relation types in the strict format of given examples as following (note that the number of entities has to strictly have the same value as the number of respective relation): {Dialogue: [dialogue] The list of entity pairs are (subject1, object1), (subject2, object2), etc The [number of pairs] respective relations between each entity pair are: [relation, relation2, etc} × *repeat n times*
Discovery	Given two sentence1 and sentence2, please predict the conjunction word between the two sentences. The predicted answer must come from the demonstration examples with the exact format. The examples are as follows: {[sentence1] ( ) [sentence2] the conjunction word in ( ) is [conjunction]} × *repeat n times*

Table 4.3: The data prompt format of each dataset. Each dataset has a unique prompt format to effectively utilize the context and format of its respective data to get the best output response.

Model	Param	Support	1R	2R	3R	4R	5R
Context Tokens			**2K**	**4K**	**7K**	**9K**	**14K**
Gemma-7B-base	7B	8K	0	0	0	0	0
LLaMA-2-7B-32K	7B	32K	**30.2**	**70.4**	**72.0**	**75.6**	**77.2**
ChatGLM3-6B-32K	6B	32K	16.6	23.2	22.4	22.8	8.8
Qwen-1.5-7B-base	7B	32K	21.6	52.8	61.4	66.0	67.8
Mistral-7B-v0.2-base	7B	32K	29.8	43.6	66.4	67.8	64.0
LLaMA-2-7B-LongLora	7B	100K	0	0	0	0	0
Yi-6B-200K	6B	200K	25.8	0	0	0	1.2
InternLM2-7B-base	7B	200K	5.6	0	0	0	0
Long-LLaMA-code-7B	7B	256K	3.0	19.4	28.0	31.6	32.6
RWKV-5-World	7B	4K	8.6	21.2	0.4	0	0
Mamba-2.8B	2.8B	2K	0	0	0	0	0
Gemini-1.0-Pro	N/A	32K	33.4	41.4	40.6	45.6	50.2
GPT4-turbo	N/A	128K	**73.5**	**80.5**	**82.0**	**83.5**	**84.4**
SoTA (RoBERTA + ICDA)	N/A	-			**94.4**		

Table 4.4: BANKING77 result with respect to increasing context length. **1R** represents one round of traversing all the instances with unique label.

Model	Param	Support	1R	2R	3R	4R	5R
Context Tokens			**4K**	**7K**	**10K**	**14K**	**18K**
Gemma-7B-base	7B	8K	0.4	0.4	0	0	0
LLaMA-2-7B-32K	7B	32K	0	0.4	0.4	0.8	0.4
ChatGLM3-6B-32K	6B	32K	29.7	36.1	38.9	40.1	25.2
Qwen-1.5-7B-base	7B	32K	38.7	47.3	45.2	43.6	40.6
Mistral-7B-v0.2-base	7B	32K	**53.3**	**53.1**	**51.6**	**48.0**	**42.3**
LLaMA-2-7B-LongLora	7B	100K	0	0	0	0	0
Yi-6B-200K	6B	200K	5.6	1.9	8.0	9.5	2.0
InternLM2-7B-base	7B	200K	29.6	27.2	15.5	10.7	8.0
Long-LLaMA-code-7B	7B	256K	3.8	7.1	4.1	6.6	4.9
RWKV-5-World	7B	1K	2.3	2.6	1.0	0	1.2
Mamba-2.8B	2.8B	2K	0	0	0	0	0
Gemini-1.0-Pro	N/A	32K	71.4	77.8	78.2	77.4	76.8
GPT4-turbo	N/A	128K	**74.4**	**76.5**	**79.5**	**80.4**	**84.2**
SoTA (DeepStruct)	N/A	-			76.8		

Table 4.5: TacRED result with respect to increasing context length.

Model	Param	Support	1R	2R	3R	4R	5R
Context Tokens			**8K**	**13K**	**19K**	**25K**	**32K**
Gemma-7B-base	7B	8K	16.3	0	0	0	0
LLaMA-2-7B-32K	7B	32K	6.9	13.9	6.3	5.7	5.9
ChatGLM3-6B-32K	6B	32K	5.1	8.9	8.8	12.4	10.4
Qwen-1.5-7B-base	7B	32K	14.4	18.4	15.5	16.4	13.2
Mistral-7B-v0.2-base	7B	32K	**24.3**	**23.2**	**23.4**	**22.3**	**21.2**
LLaMA-2-7B-LongLora	7B	100K	0	0	0	0	0
Yi-6B-200K	6B	200K	0	0	0.8	0.8	0
InternLM2-7B-base	7B	200K	12.2	13.4	6.4	2.1	1.1
Long-LLaMA-code-7B	7B	256K	4.0	3.8	3.0	6.4	2.2
RWKV-5-World	7B	4K	0	0	0	0	0
Mamba-2.8B	2.8B	2K	0	0	0	0	0
Gemini-1.0-Pro	N/A	32K	23.6	29.2	33.2	26.1	17.3
GPT4-turbo	N/A	128K	**43.5**	**48.8**	**53.6**	**60.2**	**60.9**
SoTA (HiDialog)	N/A	-			**77.1**		

Table 4.6: DialogRE result with respect to increasing context length.

Model	Param	Support	1R	2R	3R	4R	5R
Context Tokens			**10K**	**20K**	**30K**	**40K**	**50K**
Gemma-7B-base	7B	8K	0	0	0	0	0
LLaMA-2-7B-32K	7B	32K	0	0	0	0	✗
ChatGLM3-6B-32K	6B	32k	0	1.0	0	✗	✗
Qwen-1.5-7B-base	7B	32K	0	0	0	0	0
Mistral-7B-v0.2-base	7B	32K	0	0	0	0	0
LLaMA-2-7B-LongLora	7B	100K	0	0	0	0	0
Yi-6B-200K	6B	200k	0	0	0	0	0
InternLM2-7B-base	7B	200K	0	0	0	0	0
Long-LLaMA-code-7B	7B	256K	0	0	0	0	0
RWKV-5-World	7B	4K	0	0.2	0	0	0
Mamba-2.8B	2.8B	2K	0	0	0	0	0
Gemini-1.0-Pro	N/A	32K	0	0	0	✗	✗
GPT4-turbo	N/A	128K	**1.5**	**0.5**	**0.5**	**0.5**	**0.5**
SoTA (MTL)	N/A	-			**87.4**		

Table 4.7: Discovery result with respect to increasing context length.

Model	Param	Support	1R	2R	3R	4R	5R
Context Tokens			**0.8K**	**1.6K**	**2.4K**	**3.2K**	**4K**
Gemma-7B-base	7B	8K	0	0	0	0	0
LLaMA-2-7B-32K	7B	32K	0	0	0	0.2	0.2
ChatGLM3-6B-32K	6B	32K	**22.0**	17.0	15.0	12.6	10.6
Qwen-1.5-7B-base	7B	32K	14.8	**18.2**	**18.6**	**19.0**	**14.2**
Mistral-7B-v0.2-base	7B	32K	2.6	11.4	7.4	11.6	12.4
LLaMA-2-7B-LongLora	7B	100K	0	0	0	0	0
Yi-6B-200K	6B	200K	0	0	0.8	4.0	4.0
InternLM2-7B-base	7B	200K	0	0	0	0	0
Long-LLaMA-code-7B	7B	256K	0	0	0	0.2	0.4
RWKV-5-World	7B	4K	8.8	7.4	4.6	5.2	4.0
Mamba-2.8B	2.8B	2K	0	0	0	0	0
Gemini-1.0-Pro	N/A	32K	20.3	21.4	22.4	24.4	24.0
GPT4-turbo	N/A	128K	**36.5**	**34.4**	**35.0**	**33.3**	**32.0**
SoTA (BERT)	N/A	-			**58.9**		

Table 4.8: GoEmotion result with respect to increasing context length.

Model	Param	Support	1R	2R	3R	4R	5R
Context Tokens			**5K**	**9K**	**14K**	**19K**	**24K**
Gemma-7B-base	7B	8k	**44.0**	44.2	0	0	0
LLaMA-2-7B-32K	7B	32k	36.9	40.8	41.1	41.6	41.3
ChatGLM3-6B-32K	6B	32k	24.1	9.3	23.6	10.4	1.1
Qwen-1.5-7B-base	7B	32k	40.0	46.4	47.6	47.3	47.8
Mistral-7B-v0.2-base	7B	32K	42.2	**47.4**	**48.9**	**50.0**	**50.0**
LLaMA-2-7B-LongLora	7B	100K	0	0	0	0	0
Yi-6B-200K	6B	200k	34.3	40.2	44.8	42.3	43.2
InternLM2-7B-base	7B	200k	43.6	46.2	46.5	47.8	48.3
Long-LLaMA-code-7B	7B	256K	22.3	25.5	26.5	29.4	27.0
RWKV-5-World	7B	1k	13.9	0	0	0.7	9.9
Mamba-2.8B	2.8B	2k	0	0	0	0	0
Gemini-1.0-Pro	N/A	32k	36.8	26.1	28.5	27.4	28.4
GPT4-turbo	N/A	128k	**53.4**	**55.3**	**56.2**	**55.6**	**56.8**
SoTA (PL-Marker)	N/A	-			**70.9**		

Table 4.9: Few-NERD result with respect to increasing context length.

Model	Param	Support	Scatter	Grouped	Δ
Context Tokens				**10K**	
Gemma-7B-base	7B	8K	0	0	0
LLaMA-2-7B-32K	7B	32K	0.4	3.0	+2.6
ChatGLM3-6B-32K	6B	32K	38.9	35.6	-3.3
Qwen-1.5-7B-base	7B	32K	45.2	33.0	-12.2
Mistral-7B-v0.2-base	7B	32K	51.6	5.1	-46.5
LLaMA-2-7B-LongLora	7B	100K	0	0	0
Yi-6B-200K	6B	200K	8.0	0	-8
InternLM2-7B-base	7B	200K	15.5	4.8	-9.7
Long-LLaMA-code-7B	7B	256K	4.1	0	-4.1
RWKV-5-World	7B	4K	1.0	3.6	+2.6
Mamba-2.8B	2.8B	2K	0	0	0
GPT4-turbo	N/A	128K	79.5	59.2	-20.3

Table 4.10: Exploratory Result on TacRED 3 Round. **Grouped** means forcing the same-typed demonstration examples near by each other instead of randomly distributing in the prompt.

Chapter 5

Conclusion

In this thesis, we have conducted a comprehensive exploration of the in-context learning capabilities of Large Language Models (LLMs), focusing on two distinct areas: Knowledge Base Question Answering and the evaluation of long-context LLMs through extreme-label classification tasks. Through the lens of KB-BINDER and LongICLBench, we discover the potential and limitations of LLMs, showcasing their ability to adapt to and excel in complex natural language processing tasks without the need for extensive retraining.

KB-BINDER, as our novel framework for KBQA, leverages the in-context learning prowess of LLMs to generate logical forms from few examples, demonstrating remarkable efficiency and generalizability across various KBQA tasks. This approach not only alleviates the data intensiveness and dataset specificity challenges inherent in traditional KBQA methodologies but also underscores the feasibility of employing in-context learning techniques to foster advancements in this field. Complementarily, LongICLBench served as a rigorous benchmark for evaluating the long-context comprehension capabilities of LLMs, challenging them to process and reason over extensive sequences of text within extreme-label classification scenarios. Our evaluation reveals a performance threshold, highlighting the models' struggles with increasing context complexity and illuminating a path toward enhancing long-context processing abilities.

I hope the insights obtained from these studies can shed light on the current landscape of in-context learning within LLMs, identifying both the strengths and areas for improvement. KB-BINDER's success in KBQA exemplifies the potential of few-shot learning in overcoming domain-specific and grounding-required challenges, while LongICLBench's findings point to the necessity for further advancements in LLMs' ability to navigate and reason over lengthy texts.

In a nutshell, we find that in-context learning can offer a promising avenue for advancing the field of natural language processing, pushing the boundaries of what LLMs can achieve. This thesis strengthens the foundation for future explorations, aiming to enhance LLMs' adaptability and efficiency across a broader spectrum of complex tasks. Through continuous innovation and research, we anticipate the emergence of more sophisticated models capable of surpassing the current limitations, further revolutionizing our interaction with and understanding the performance of large language models.

References

[1] Josh Achiam, Steven Adler, Sandhini Agarwal, Lama Ahmad, Ilge Akkaya, Florencia Leoni Aleman, Diogo Almeida, Janko Altenschmidt, Sam Altman, Shyamal Anadkat, et al. Gpt-4 technical report. *arXiv preprint arXiv:2303.08774*, 2023.

[2] 01. AI, :, Alex Young, Bei Chen, Chao Li, Chengen Huang, Ge Zhang, Guanwei Zhang, Heng Li, Jiangcheng Zhu, Jianqun Chen, Jing Chang, Kaidong Yu, Peng Liu, Qiang Liu, Shawn Yue, Senbin Yang, Shiming Yang, Tao Yu, Wen Xie, Wenhao Huang, Xiaohui Hu, Xiaoyi Ren, Xinyao Niu, Pengcheng Nie, Yuchi Xu, Yudong Liu, Yue Wang, Yuxuan Cai, Zhenyu Gu, Zhiyuan Liu, and Zonghong Dai. Yi: Open foundation models by 01.ai, 2024.

[3] Chenxin An, Shansan Gong, Ming Zhong, Xingjian Zhao, Mukai Li, Jun Zhang, Lingpeng Kong, and Xipeng Qiu. L-eval: Instituting standardized evaluation for long context language models, 2023.

[4] Cem Anil, Yuhuai Wu, Anders Johan Andreassen, Aitor Lewkowycz, Vedant Misra, Vinay Venkatesh Ramasesh, Ambrose Slone, Guy Gur-Ari, Ethan Dyer, and Behnam Neyshabur. Exploring length generalization in large language models. In Alice H. Oh, Alekh Agarwal, Danielle Belgrave, and Kyunghyun Cho, editors, *Advances in Neural Information Processing Systems*, 2022.

[5] Jinze Bai, Shuai Bai, Yunfei Chu, Zeyu Cui, Kai Dang, Xiaodong Deng, Yang Fan, Wenbin Ge, Yu Han, Fei Huang, Binyuan Hui, Luo Ji, Mei Li, Junyang Lin, Runji Lin, Dayiheng Liu, Gao Liu, Chengqiang Lu, Keming Lu, Jianxin Ma, Rui Men, Xingzhang Ren, Xuancheng Ren, Chuanqi Tan, Sinan Tan, Jianhong Tu, Peng Wang, Shijie Wang, Wei Wang, Shengguang Wu, Benfeng Xu, Jin Xu, An Yang, Hao Yang, Jian Yang, Shusheng Yang, Yang Yao, Bowen Yu, Hongyi Yuan, Zheng Yuan, Jianwei Zhang, Xingxuan Zhang, Yichang Zhang, Zhenru Zhang, Chang Zhou, Jingren Zhou, Xiaohuan Zhou, and Tianhang Zhu. Qwen technical report, 2023.

[6] Yushi Bai, Xin Lv, Jiajie Zhang, Hongchang Lyu, Jiankai Tang, Zhidian Huang, Zhengxiao Du, Xiao Liu, Aohan Zeng, Lei Hou, Yuxiao Dong, Jie Tang, and Juanzi Li. Longbench: A bilingual, multitask benchmark for long context understanding, 2023.

[7] Jonathan Berant, Andrew Chou, Roy Frostig, and Percy Liang. Semantic parsing on freebase from question-answer pairs. In *Proceedings of the 2013 conference on empirical methods in natural language processing*, pages 1533–1544, 2013.

[8] Kush Bhatia, Himanshu Jain, Purushottam Kar, Manik Varma, and Prateek Jain. Sparse local embeddings for extreme multi-label classification. In *Neural Information Processing Systems*, 2015.

[9] Kurt D. Bollacker, Colin Evans, Praveen K. Paritosh, Tim Sturge, and Jamie Taylor. Freebase: a collaboratively created graph database for structuring human knowledge. In Jason Tsong-Li Wang, editor, *Proceedings of the ACM SIGMOD International Conference on Management of Data, SIGMOD 2008, Vancouver, BC, Canada, June 10-12, 2008*, pages 1247–1250. ACM, 2008.

[10] 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, T. J. Henighan, Rewon Child, Aditya Ramesh, Daniel M. Ziegler, Jeff 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*, abs/2005.14165, 2020.

[11] 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, 2020.

[12] Zheng Cai, Maosong Cao, Haojiong Chen, ..., Yu Qiao, and Dahua Lin. Internlm2 technical report. *arXiv preprint arXiv:2403.17297*, 2024.

[13] Iñigo Casanueva, Tadas Temčinas, Daniela Gerz, Matthew Henderson, and Ivan Vulić. Efficient intent detection with dual sentence encoders. In Tsung-Hsien Wen,

Asli Celikyilmaz, Zhou Yu, Alexandros Papangelis, Mihail Eric, Anuj Kumar, Iñigo Casanueva, and Rushin Shah, editors, *Proceedings of the 2nd Workshop on Natural Language Processing for Conversational AI*, pages 38–45, Online, July 2020. Association for Computational Linguistics.

[14] Rakesh Chada and Pradeep Natarajan. Fewshotqa: A simple framework for few-shot learning of question answering tasks using pre-trained text-to-text models. *ArXiv*, abs/2109.01951, 2021.

[15] Nilesh Chakraborty, Denis Lukovnikov, Gaurav Maheshwari, Priyansh Trivedi, Jens Lehmann, and Asja Fischer. Introduction to neural network based approaches for question answering over knowledge graphs. *ArXiv*, abs/1907.09361, 2019.

[16] Stephanie C. Y. Chan, Adam Santoro, Andrew K. Lampinen, Jane X. Wang, Aaditya Singh, Pierre H. Richemond, Jay McClelland, and Felix Hill. Data distributional properties drive emergent in-context learning in transformers, 2022.

[17] Danqi Chen, Adam Fisch, Jason Weston, and Antoine Bordes. Reading wikipedia to answer open-domain questions. In *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 1870–1879, 2017.

[18] Mark Chen, Jerry Tworek, Heewoo Jun, Qiming Yuan, Henrique Ponde, Jared Kaplan, Harrison Edwards, Yura Burda, Nicholas Joseph, Greg Brockman, Alex Ray, Raul Puri, Gretchen Krueger, Michael Petrov, Heidy Khlaaf, Girish Sastry, Pamela Mishkin, Brooke Chan, Scott Gray, Nick Ryder, Mikhail Pavlov, Alethea Power, Lukasz Kaiser, Mohammad Bavarian, Clemens Winter, Philippe Tillet, Felipe Petroski Such, David W. Cummings, Matthias Plappert, Fotios Chantzis, Elizabeth Barnes, Ariel Herbert-Voss, William H. Guss, Alex Nichol, Igor Babuschkin, S. Arun Balaji, Shantanu Jain, Andrew Carr, Jan Leike, Joshua Achiam, Vedant Misra, Evan Morikawa, Alec Radford, Matthew M. Knight, Miles Brundage, Mira Murati, Katie Mayer, Peter Welinder, Bob McGrew, Dario Amodei, Sam McCandlish, Ilya Sutskever, and Wojciech Zaremba. Evaluating large language models trained on code. *ArXiv*, abs/2107.03374, 2021.

[19] Shouyuan Chen, Sherman Wong, Liangjian Chen, and Yuandong Tian. Extending context window of large language models via positional interpolation. *ArXiv*, abs/2306.15595, 2023.

[20] Shuang Chen, Qian Liu, Zhiwei Yu, Chin-Yew Lin, Jian-Guang Lou, and Feng Jiang. Retrack: A flexible and efficient framework for knowledge base question answering. In *Annual Meeting of the Association for Computational Linguistics*, 2021.

[21] Wenhu Chen, Xueguang Ma, Xinyi Wang, and William W. Cohen. Program of thoughts prompting: Disentangling computation from reasoning for numerical reasoning tasks. *ArXiv*, abs/2211.12588, 2022.

[22] Yukang Chen, Shengju Qian, Haotian Tang, Xin Lai, Zhijian Liu, Song Han, and Jiaya Jia. Longlora: Efficient fine-tuning of long-context large language models. In *The Twelfth International Conference on Learning Representations*, 2023.

[23] Zhoujun Cheng, Tianbao Xie, Peng Shi, Chengzu Li, R.K. Nadkarni, Yushi Hu, Caiming Xiong, Dragomir R. Radev, Marilyn Ostendorf, Luke Zettlemoyer, Noah A. Smith, and Tao Yu. Binding language models in symbolic languages. *ArXiv*, abs/2210.02875, 2022.

[24] Pradeep Dasigi, Kyle Lo, Iz Beltagy, Arman Cohan, Noah A Smith, and Matt Gardner. A dataset of information-seeking questions and answers anchored in research papers. In *Proceedings of the 2021 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 4599–4610, 2021.

[25] Dorottya Demszky, Dana Movshovitz-Attias, Jeongwoo Ko, Alan Cowen, Gaurav Nemade, and Sujith Ravi. GoEmotions: A dataset of fine-grained emotions. In Dan Jurafsky, Joyce Chai, Natalie Schluter, and Joel Tetreault, editors, *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 4040–4054, Online, July 2020. Association for Computational Linguistics.

[26] Dennis Diefenbach, Vanessa Lopez, Kamal Singh, and Pierre Maret. Core techniques of question answering systems over knowledge bases: a survey. *Knowledge and Information systems*, 55(3):529–569, 2018.

[27] Ning Ding, Guangwei Xu, Yulin Chen, Xiaobin Wang, Xu Han, Pengjun Xie, Haitao Zheng, and Zhiyuan Liu. Few-NERD: A few-shot named entity recognition dataset. In Chengqing Zong, Fei Xia, Wenjie Li, and Roberto Navigli, editors, *Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing (Volume 1: Long Papers)*, pages 3198–3213, Online, August 2021. Association for Computational Linguistics.

[28] Yiran Ding, Li Lyna Zhang, Chengruidong Zhang, Yuanyuan Xu, Ning Shang, Jiahang Xu, Fan Yang, and Mao Yang. Longrope: Extending llm context window beyond 2 million tokens. *arXiv preprint arXiv:2402.13753*, 2024.

[29] Li Dong and Mirella Lapata. Language to logical form with neural attention. In *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 33–43, 2016.

[30] Qingxiu Dong, Lei Li, Damai Dai, Ce Zheng, Zhiyong Wu, Baobao Chang, Xu Sun, Jingjing Xu, Lei Li, and Zhifang Sui. A survey on in-context learning, 2023.

[31] Karel D'Oosterlinck, Omar Khattab, François Remy, Thomas Demeester, Chris Develder, and Christopher Potts. In-context learning for extreme multi-label classification. *arXiv preprint arXiv:2401.12178*, 2024.

[32] Alexander Dunn, John Dagdelen, Nicholas Walker, Sanghoon Lee, Andrew S. Rosen, Gerbrand Ceder, Kristin Persson, and Anubhav Jain. Structured information extraction from complex scientific text with fine-tuned large language models, 2022.

[33] Yao Fu, Rameswar Panda, Xinyao Niu, Xiang Yue, Hannaneh Hajishirzi, Yoon Kim, and Hao Peng. Data engineering for scaling language models to 128k context. *arXiv preprint arXiv:2402.10171*, 2024.

[34] Luyu Gao, Aman Madaan, Shuyan Zhou, Uri Alon, Pengfei Liu, Yiming Yang, Jamie Callan, and Graham Neubig. Pal: Program-aided language models. *ArXiv*, abs/2211.10435, 2022.

[35] Michel Goossens, Frank Mittelbach, and Alexander Samarin. *The LaTeX Companion*. Addison-Wesley, Reading, Massachusetts, 1994.

[36] Albert Gu and Tri Dao. Mamba: Linear-time sequence modeling with selective state spaces. *arXiv preprint arXiv:2312.00752*, 2023.

[37] Yu Gu, Xiang Deng, and Yu Su. Don't generate, discriminate: A proposal for grounding language models to real-world environments, 2023.

[38] Yu Gu, Sue E. Kase, Michelle T. Vanni, Brian M. Sadler, Percy Liang, Xifeng Yan, and Yu Su. Beyond i.i.d.: Three levels of generalization for question answering on knowledge bases. *Proceedings of the Web Conference 2021*, 2020.

[39] Yu Gu, Vardaan Pahuja, Gong Cheng, and Yu Su. Knowledge base question answering: A semantic parsing perspective. In *4th Conference on Automated Knowledge Base Construction*, 2022.

[40] Yu Gu and Yu Su. Arcaneqa: Dynamic program induction and contextualized encoding for knowledge base question answering. In *International Conference on Computational Linguistics*, 2022.

[41] Michael Hahn. Theoretical limitations of self-attention in neural sequence models. *Transactions of the Association for Computational Linguistics*, 8:156–171, 2019.

[42] Yaru Hao, Yutao Sun, Li Dong, Zhixiong Han, Yuxian Gu, and Furu Wei. Structured prompting: Scaling in-context learning to 1, 000 examples. *ArXiv*, abs/2212.06713, 2022.

[43] Gaole He, Yunshi Lan, Jing Jiang, Wayne Xin Zhao, and Ji rong Wen. Improving multi-hop knowledge base question answering by learning intermediate supervision signals. *Proceedings of the 14th ACM International Conference on Web Search and Data Mining*, 2021.

[44] Yuncheng Hua, Yuan-Fang Li, Gholamreza Haffari, Guilin Qi, and Tongtong Wu. Few-shot complex knowledge base question answering via meta reinforcement learning. *ArXiv*, abs/2010.15877, 2020.

[45] Gautier Izacard, Mathilde Caron, Lucas Hosseini, Sebastian Riedel, Piotr Bojanowski, Armand Joulin, and Edouard Grave. Unsupervised dense information retrieval with contrastive learning. 2021.

[46] Albert Q. Jiang, Alexandre Sablayrolles, Arthur Mensch, Chris Bamford, Devendra Singh Chaplot, Diego de las Casas, Florian Bressand, Gianna Lengyel, Guillaume Lample, Lucile Saulnier, Lélio Renard Lavaud, Marie-Anne Lachaux, Pierre Stock, Teven Le Scao, Thibaut Lavril, Thomas Wang, Timothée Lacroix, and William El Sayed. Mistral 7b, 2023.

[47] Donald Knuth. *The TEXbook*. Addison-Wesley, Reading, Massachusetts, 1986.

[48] Takeshi Kojima, Shixiang Shane Gu, Machel Reid, Yutaka Matsuo, and Yusuke Iwasawa. Large language models are zero-shot reasoners, 2022.

[49] Ananya Kumar, Aditi Raghunathan, Robbie Jones, Tengyu Ma, and Percy Liang. Fine-tuning can distort pretrained features and underperform out-of-distribution. *ArXiv*, abs/2202.10054, 2022.

[50] Andrew K. Lampinen, Ishita Dasgupta, Stephanie C. Y. Chan, Kory Matthewson, Michael Henry Tessler, Antonia Creswell, James L. McClelland, Jane X. Wang, and Felix Hill. Can language models learn from explanations in context?, 2022.

[51] Leslie Lamport. *LaTeX — A Document Preparation System*. Addison-Wesley, Reading, Massachusetts, second edition, 1994.

[52] Yunshi Lan, Gaole He, Jinhao Jiang, Jing Jiang, Wayne Xin Zhao, and Ji rong Wen. A survey on complex knowledge base question answering: Methods, challenges and solutions. In *International Joint Conference on Artificial Intelligence*, 2021.

[53] Yunshi Lan and Jing Jiang. Query graph generation for answering multi-hop complex questions from knowledge bases. In *Annual Meeting of the Association for Computational Linguistics*, 2020.

[54] Mosh Levy, Alon Jacoby, and Yoav Goldberg. Same task, more tokens: the impact of input length on the reasoning performance of large language models, 2024.

[55] Aitor Lewkowycz, Anders Andreassen, David Dohan, Ethan Dyer, Henryk Michalewski, Vinay Ramasesh, Ambrose Slone, Cem Anil, Imanol Schlag, Theo Gutman-Solo, Yuhuai Wu, Behnam Neyshabur, Guy Gur-Ari, and Vedant Misra. Solving quantitative reasoning problems with language models, 2022.

[56] Dacheng Li, Rulin Shao, Anze Xie, Ying Sheng, Lianmin Zheng, Joseph Gonzalez, Ion Stoica, Xuezhe Ma, and Hao Zhang. How long can context length of open-source LLMs truly promise? In *NeurIPS 2023 Workshop on Instruction Tuning and Instruction Following*, 2023.

[57] Jiaqi Li, Mengmeng Wang, Zilong Zheng, and Muhan Zhang. Loogle: Can long-context language models understand long contexts?, 2023.

[58] Jiaqi Li, Mengmeng Wang, Zilong Zheng, and Muhan Zhang. Loogle: Can long-context language models understand long contexts? *ArXiv*, abs/2311.04939, 2023.

[59] Mukai Li, Shansan Gong, Jiangtao Feng, Yiheng Xu, Jun Zhang, Zhiyong Wu, and Lingpeng Kong. In-context learning with many demonstration examples, 2023.

[60] Tianle Li, Xueguang Ma, Alex Zhuang, Yu Gu, Yu Su, and Wenhu Chen. Few-shot in-context learning on knowledge base question answering. In Anna Rogers, Jordan Boyd-Graber, and Naoaki Okazaki, editors, *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 6966–6980, Toronto, Canada, July 2023. Association for Computational Linguistics.

[61] Tianle Li, Ge Zhang, Quy Duc Do, Xiang Yue, and Wenhu Chen. Long-context llms struggle with long in-context learning, 2024.

[62] Jiachang Liu, Dinghan Shen, Yizhe Zhang, Bill Dolan, Lawrence Carin, and Weizhu Chen. What makes good in-context examples for gpt-3? *CoRR*, abs/2101.06804, 2021.

[63] Jiachang Liu, Dinghan Shen, Yizhe Zhang, Bill Dolan, Lawrence Carin, and Weizhu Chen. What makes good in-context examples for GPT-3? In Eneko Agirre, Marianna Apidianaki, and Ivan Vulić, editors, *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, May 2022. Association for Computational Linguistics.

[64] Jiaheng Liu, Zhiqi Bai, Yuanxing Zhang, Chenchen Zhang, Yu Zhang, Ge Zhang, Jiakai Wang, Haoran Que, Yukang Chen, Wenbo Su, et al. Eˆ 2-llm: Efficient and extreme length extension of large language models. *arXiv preprint arXiv:2401.06951*, 2024.

[65] Nelson F. Liu, Kevin Lin, John Hewitt, Ashwin Paranjape, Michele Bevilacqua, Fabio Petroni, and Percy Liang. Lost in the middle: How language models use long contexts. *Transactions of the Association for Computational Linguistics*, 12:157–173, 2023.

[66] Aristides Milios, Siva Reddy, and Dzmitry Bahdanau. In-context learning for text classification with many labels, 2023.

[67] Alexander H. Miller, Adam Fisch, Jesse Dodge, Amir-Hossein Karimi, Antoine Bordes, and Jason Weston. Key-value memory networks for directly reading documents. *ArXiv*, abs/1606.03126, 2016.

[68] Sewon Min, Xinxi Lyu, Ari Holtzman, Mikel Artetxe, Mike Lewis, Hannaneh Hajishirzi, and Luke Zettlemoyer. Rethinking the role of demonstrations: What makes in-context learning work? In *EMNLP*, 2022.

[69] Amirkeivan Mohtashami and Martin Jaggi. Landmark attention: Random-access infinite context length for transformers. In *Workshop on Efficient Systems for Foundation Models@ ICML2023*, 2023.

[70] Ramesh Nallapati, Feifei Zhai, and Bowen Zhou. Summarunner: A recurrent neural network based sequence model for extractive summarization of documents. In *Proceedings of the AAAI conference on artificial intelligence*, volume 31, 2017.

[71] Maxwell Nye, Anders Johan Andreassen, Guy Gur-Ari, Henryk Michalewski, Jacob Austin, David Bieber, David Dohan, Aitor Lewkowycz, Maarten Bosma, David Luan, Charles Sutton, and Augustus Odena. Show your work: Scratchpads for intermediate computation with language models, 2021.

[72] Catherine Olsson, Nelson Elhage, Neel Nanda, Nicholas Joseph, Nova DasSarma, Tom Henighan, Ben Mann, Amanda Askell, Yuntao Bai, Anna Chen, Tom Conerly, Dawn Drain, Deep Ganguli, Zac Hatfield-Dodds, Danny Hernandez, Scott Johnston, Andy Jones, Jackson Kernion, Liane Lovitt, Kamal Ndousse, Dario Amodei, Tom Brown, Jack Clark, Jared Kaplan, Sam McCandlish, and Chris Olah. In-context learning and induction heads. *Transformer Circuits Thread*, 2022. https://transformer-circuits.pub/2022/in-context-learning-and-induction-heads/index.html.

[73] Antonio Orvieto, Samuel L. Smith, Albert Gu, Anushan Fernando, Caglar Gulcehre, Razvan Pascanu, and Soham De. Resurrecting recurrent neural networks for long sequences. *ArXiv*, abs/2303.06349, 2023.

[74] Richard Yuanzhe Pang, Alicia Parrish, Nitish Joshi, Nikita Nangia, Jason Phang, Angelica Chen, Vishakh Padmakumar, Johnny Ma, Jana Thompson, He He, et al. Quality: Question answering with long input texts, yes! In *Proceedings of the 2022 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 5336–5358, 2022.

[75] Bo Peng, Eric Alcaide, Quentin Anthony, Alon Albalak, Samuel Arcadinho, Stella Biderman, Huanqi Cao, Xin Cheng, Michael Chung, Leon Derczynski, et al. Rwkv: Reinventing rnns for the transformer era. In *Findings of the Association for Computational Linguistics: EMNLP 2023*, pages 14048–14077, 2023.

[76] Bowen Peng, Jeffrey Quesnelle, Honglu Fan, and Enrico Shippole. Yarn: Efficient context window extension of large language models, 2023.

[77] Hao Peng, Xiaozhi Wang, Jianhui Chen, Weikai Li, Yunjia Qi, Zimu Wang, Zhili Wu, Kaisheng Zeng, Bin Xu, Lei Hou, and Juanzi Li. When does in-context learning fall short and why? a study on specification-heavy tasks, 2023.

[78] Ofir Press, Noah Smith, and Mike Lewis. Train short, test long: Attention with linear biases enables input length extrapolation. In *International Conference on Learning Representations*, 2022.

[79] Nir Ratner, Yoav Levine, Yonatan Belinkov, Ori Ram, Inbal Magar, Omri Abend, Ehud Karpas, Amnon Shashua, Kevin Leyton-Brown, and Yoav Shoham. Parallel context windows for large language models. In Anna Rogers, Jordan Boyd-Graber, and Naoaki Okazaki, editors, *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 6383–6402, Toronto, Canada, July 2023. Association for Computational Linguistics.

[80] Siva Reddy, Oscar Täckström, Slav Petrov, Mark Steedman, and Mirella Lapata. Universal semantic parsing. In *Conference on Empirical Methods in Natural Language Processing*, 2017.

[81] Baptiste Rozière, Jonas Gehring, Fabian Gloeckle, Sten Sootla, Itai Gat, Xiaoqing Ellen Tan, Yossi Adi, Jingyu Liu, Romain Sauvestre, Tal Remez, Jérémy Rapin, Artyom Kozhevnikov, Ivan Evtimov, Joanna Bitton, Manish Bhatt, Cristian Canton Ferrer, Aaron Grattafiori, Wenhan Xiong, Alexandre Défossez, Jade Copet, Faisal Azhar, Hugo Touvron, Louis Martin, Nicolas Usunier, Thomas Scialom, and Gabriel Synnaeve. Code llama: Open foundation models for code, 2024.

[82] Apoorv Saxena, Aditay Tripathi, and Partha Pratim Talukdar. Improving multi-hop question answering over knowledge graphs using knowledge base embeddings. In *Annual Meeting of the Association for Computational Linguistics*, 2020.

[83] Uri Shaham, Maor Ivgi, Avia Efrat, Jonathan Berant, and Omer Levy. ZeroSCROLLS: A zero-shot benchmark for long text understanding. In Houda Bouamor, Juan Pino, and Kalika Bali, editors, *Findings of the Association for Computational Linguistics: EMNLP 2023*, pages 7977–7989, Singapore, December 2023. Association for Computational Linguistics.

[84] Yiheng Shu, Zhiwei Yu, Yuhan Li, Börje F. Karlsson, Tingting Ma, Yuzhong Qu, and Chin-Yew Lin. Tiara: Multi-grained retrieval for robust question answering over large knowledge bases. *ArXiv*, abs/2210.12925, 2022.

[85] Damien Sileo, Tim Van De Cruys, Camille Pradel, and Philippe Muller. Mining discourse markers for unsupervised sentence representation learning. In Jill Burstein, Christy Doran, and Thamar Solorio, editors, *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics:*

Human Language Technologies, Volume 1 (Long and Short Papers), pages 3477–3486, Minneapolis, Minnesota, June 2019. Association for Computational Linguistics.

[86] Jianlin Su, Murtadha Ahmed, Yu Lu, Shengfeng Pan, Wen Bo, and Yunfeng Liu. Roformer: Enhanced transformer with rotary position embedding. *Neurocomputing*, 568:127063, 2024.

[87] Jianlin Su, Yu Lu, Shengfeng Pan, Bo Wen, and Yunfeng Liu. Roformer: Enhanced transformer with rotary position embedding. *ArXiv*, abs/2104.09864, 2021.

[88] Yu Su, Huan Sun, Brian M. Sadler, Mudhakar Srivatsa, Izzeddin Gur, Zenghui Yan, and Xifeng Yan. On generating characteristic-rich question sets for qa evaluation. In *Conference on Empirical Methods in Natural Language Processing*, 2016.

[89] Haitian Sun, Tania Bedrax-Weiss, and William W. Cohen. Pullnet: Open domain question answering with iterative retrieval on knowledge bases and text. *ArXiv*, abs/1904.09537, 2019.

[90] Haitian Sun, Bhuwan Dhingra, Manzil Zaheer, Kathryn Mazaitis, Ruslan Salakhutdinov, and William W. Cohen. Open domain question answering using early fusion of knowledge bases and text. In *Conference on Empirical Methods in Natural Language Processing*, 2018.

[91] Yawei Sun, Lingling Zhang, Gong Cheng, and Yuzhong Qu. Sparqa: Skeleton-based semantic parsing for complex questions over knowledge bases. In *AAAI Conference on Artificial Intelligence*, 2020.

[92] Mirac Suzgun, Nathan Scales, Nathanael Scharli, Sebastian Gehrmann, Yi Tay, Hyung Won Chung, Aakanksha Chowdhery, Quoc V. Le, Ed Huai hsin Chi, Denny Zhou, and Jason Wei. Challenging big-bench tasks and whether chain-of-thought can solve them. *ArXiv*, abs/2210.09261, 2022.

[93] Alon Talmor and Jonathan Berant. The web as a knowledge-base for answering complex questions. In *North American Chapter of the Association for Computational Linguistics*, 2018.

[94] Yi Tay, Mostafa Dehghani, Samira Abnar, Yikang Shen, Dara Bahri, Philip Pham, Jinfeng Rao, Liu Yang, Sebastian Ruder, and Donald Metzler. Long range arena : A benchmark for efficient transformers. In *International Conference on Learning Representations*, 2021.

[95] Gemini Team, Rohan Anil, Sebastian Borgeaud, Yonghui Wu, Jean-Baptiste Alayrac, Jiahui Yu, Radu Soricut, Johan Schalkwyk, Andrew M Dai, Anja Hauth, et al. Gemini: a family of highly capable multimodal models. *arXiv preprint arXiv:2312.11805*, 2023.

[96] Gemma Team, Thomas Mesnard, Cassidy Hardin, Robert Dadashi, Surya Bhupatiraju, Shreya Pathak, Laurent Sifre, Morgane Rivière, Mihir Sanjay Kale, Juliette Love, et al. Gemma: Open models based on gemini research and technology. *arXiv preprint arXiv:2403.08295*, 2024.

[97] InternLM Team. Internlm: A multilingual language model with progressively enhanced capabilities. https://github.com/InternLM/InternLM, 2023.

[98] Hugo Touvron, Louis Martin, Kevin R. Stone, Peter Albert, Amjad Almahairi, Yasmine Babaei, Nikolay Bashlykov, Soumya Batra, Prajjwal Bhargava, Shruti Bhosale, Daniel M. Bikel, Lukas Blecher, Cristian Cantón Ferrer, Moya Chen, Guillem Cucurull, David Esiobu, Jude Fernandes, Jeremy Fu, Wenyin Fu, Brian Fuller, Cynthia Gao, Vedanuj Goswami, Naman Goyal, Anthony S. Hartshorn, Saghar Hosseini, Rui Hou, Hakan Inan, Marcin Kardas, Viktor Kerkez, Madian Khabsa, Isabel M. Kloumann, A. V. Korenev, Punit Singh Koura, Marie-Anne Lachaux, Thibaut Lavril, Jenya Lee, Diana Liskovich, Yinghai Lu, Yuning Mao, Xavier Martinet, Todor Mihaylov, Pushkar Mishra, Igor Molybog, Yixin Nie, Andrew Poulton, Jeremy Reizenstein, Rashi Rungta, Kalyan Saladi, Alan Schelten, Ruan Silva, Eric Michael Smith, R. Subramanian, Xia Tan, Binh Tang, Ross Taylor, Adina Williams, Jian Xiang Kuan, Puxin Xu, Zhengxu Yan, Iliyan Zarov, Yuchen Zhang, Angela Fan, Melanie Kambadur, Sharan Narang, Aurelien Rodriguez, Robert Stojnic, Sergey Edunov, and Thomas Scialom. Llama 2: Open foundation and fine-tuned chat models. *ArXiv*, abs/2307.09288, 2023.

[99] Szymon Tworkowski, Konrad Staniszewski, Mikołaj Pacek, Yuhuai Wu, Henryk Michalewski, and Piotr Miłoś. Focused transformer: Contrastive training for context scaling, 2023.

[100] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Ł ukasz Kaiser, and Illia Polosukhin. Attention is all you need. In *Advances in Neural Information Processing Systems*, volume 30. Curran Associates, Inc., 2017.

[101] Ivan Vulić, Pei-Hao Su, Samuel Coope, Daniela Gerz, Paweł Budzianowski, Iñigo Casanueva, Nikola Mrkšić, and Tsung-Hsien Wen. ConvFiT: Conversational fine-tuning of pretrained language models. In Marie-Francine Moens, Xuanjing Huang,

Lucia Specia, and Scott Wen-tau Yih, editors, *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing*, pages 1151–1168, Online and Punta Cana, Dominican Republic, November 2021. Association for Computational Linguistics.

[102] Weizhi Wang, Li Dong, Hao Cheng, Xiaodong Liu, Xifeng Yan, Jianfeng Gao, and Furu Wei. Augmenting language models with long-term memory. In *Thirty-seventh Conference on Neural Information Processing Systems*, 2023.

[103] Xingyao Wang, Sha Li, and Heng Ji. Code4struct: Code generation for few-shot structured prediction from natural language. *ArXiv*, abs/2210.12810, 2022.

[104] Xuezhi Wang, Jason Wei, Dale Schuurmans, Quoc Le, Ed Chi, and Denny Zhou. Self-consistency improves chain of thought reasoning in language models. *ArXiv*, abs/2203.11171, 2022.

[105] Jason Wei, Yi Tay, Rishi Bommasani, Colin Raffel, Barret Zoph, Sebastian Borgeaud, Dani Yogatama, Maarten Bosma, Denny Zhou, Donald Metzler, Ed H. Chi, Tatsunori Hashimoto, Oriol Vinyals, Percy Liang, Jeff Dean, and William Fedus. Emergent abilities of large language models, 2022.

[106] Jason Wei, Xuezhi Wang, Dale Schuurmans, Maarten Bosma, Ed H. Chi, Quoc Le, and Denny Zhou. Chain of thought prompting elicits reasoning in large language models. *CoRR*, abs/2201.11903, 2022.

[107] Jason Wei, Xuezhi Wang, Dale Schuurmans, Maarten Bosma, Ed Huai hsin Chi, Quoc Le, and Denny Zhou. Chain of thought prompting elicits reasoning in large language models. *ArXiv*, abs/2201.11903, 2022.

[108] Peiyun Wu, Xiaowang Zhang, and Zhiyong Feng. A survey of question answering over knowledge base. In *China Conference on Knowledge Graph and Semantic Computing*, 2019.

[109] Zhiyong Wu, Yaoxiang Wang, Jiacheng Ye, and Lingpeng Kong. Self-adaptive in-context learning: An information compression perspective for in-context example selection and ordering, 2023.

[110] Guangxuan Xiao, Yuandong Tian, Beidi Chen, Song Han, and Mike Lewis. Efficient streaming language models with attention sinks. In *The Twelfth International Conference on Learning Representations*, 2024.

[111] Sang Michael Xie, Aditi Raghunathan, Percy Liang, and Tengyu Ma. An explanation of in-context learning as implicit bayesian inference. *arXiv preprint arXiv:2111.02080*, 2021.

[112] Wenhan Xiong, Jingyu Liu, Igor Molybog, Hejia Zhang, Prajjwal Bhargava, Rui Hou, Louis Martin, Rashi Rungta, Karthik Abinav Sankararaman, Barlas Oguz, et al. Effective long-context scaling of foundation models. *arXiv preprint arXiv:2309.16039*, 2023.

[113] Aiyuan Yang, Bin Xiao, Bingning Wang, Borong Zhang, Ce Bian, Chao Yin, Chenxu Lv, Da Pan, Dian Wang, Dong Yan, Fan Yang, Fei Deng, Feng Wang, Feng Liu, Guangwei Ai, Guosheng Dong, Haizhou Zhao, Hang Xu, Haoze Sun, Hongda Zhang, Hui Liu, Jiaming Ji, Jian Xie, JunTao Dai, Kun Fang, Lei Su, Liang Song, Lifeng Liu, Liyun Ru, Luyao Ma, Mang Wang, Mickel Liu, MingAn Lin, Nuolan Nie, Peidong Guo, Ruiyang Sun, Tao Zhang, Tianpeng Li, Tianyu Li, Wei Cheng, Weipeng Chen, Xiangrong Zeng, Xiaochuan Wang, Xiaoxi Chen, Xin Men, Xin Yu, Xuehai Pan, Yanjun Shen, Yiding Wang, Yiyu Li, Youxin Jiang, Yuchen Gao, Yupeng Zhang, Zenan Zhou, and Zhiying Wu. Baichuan 2: Open large-scale language models, 2023.

[114] Xi Ye, Semih Yavuz, Kazuma Hashimoto, Yingbo Zhou, and Caiming Xiong. Rng-kbqa: Generation augmented iterative ranking for knowledge base question answering. *ArXiv*, abs/2109.08678, 2021.

[115] Scott Wen-tau Yih, Ming-Wei Chang, Xiaodong He, and Jianfeng Gao. Semantic parsing via staged query graph generation: Question answering with knowledge base. In *Proceedings of the Joint Conference of the 53rd Annual Meeting of the ACL and the 7th International Joint Conference on Natural Language Processing of the AFNLP*, 2015.

[116] Wen-tau Yih, Matthew Richardson, Christopher Meek, Ming-Wei Chang, and Jina Suh. The value of semantic parse labeling for knowledge base question answering. In *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers)*, pages 201–206, 2016.

[117] Dian Yu, Kai Sun, Claire Cardie, and Dong Yu. Dialogue-based relation extraction. In Dan Jurafsky, Joyce Chai, Natalie Schluter, and Joel Tetreault, editors, *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 4927–4940, Online, July 2020. Association for Computational Linguistics.

[118] Donghan Yu, Shenmin Zhang, Patrick Ng, Henghui Zhu, Alexander Hanbo Li, J. Wang, Yiqun Hu, William Wang, Zhiguo Wang, and Bing Xiang. Decaf: Joint decoding of answers and logical forms for question answering over knowledge bases. *ArXiv*, abs/2210.00063, 2022.

[119] Daoguang Zan, Sirui Wang, Hongzhi Zhang, Yuanmeng Yan, Wei Wu, Bei Guan, and Yongji Wang. S2ql: Retrieval augmented zero-shot question answering over knowledge graph. In *Pacific-Asia Conference on Knowledge Discovery and Data Mining*, 2022.

[120] John M Zelle and Raymond J Mooney. Learning to parse database queries using inductive logic programming. In *Proceedings of the national conference on artificial intelligence*, pages 1050–1055, 1996.

[121] Aohan Zeng, Xiao Liu, Zhengxiao Du, Zihan Wang, Hanyu Lai, Ming Ding, Zhuoyi Yang, Yifan Xu, Wendi Zheng, Xiao Xia, et al. Glm-130b: An open bilingual pre-trained model. In *The Eleventh International Conference on Learning Representations*, 2022.

[122] Aohan Zeng, Xiao Liu, Zhengxiao Du, Zihan Wang, Hanyu Lai, Ming Ding, Zhuoyi Yang, Yifan Xu, Wendi Zheng, Xiao Xia, Weng Lam Tam, Zixuan Ma, Yufei Xue, Jidong Zhai, Wenguang Chen, Peng Zhang, Yuxiao Dong, and Jie Tang. Glm-130b: An open bilingual pre-trained model, 2023.

[123] Luke S Zettlemoyer and Michael Collins. Learning to map sentences to logical form: structured classification with probabilistic categorial grammars. In *Proceedings of the Twenty-First Conference on Uncertainty in Artificial Intelligence*, pages 658–666, 2005.

[124] Xinrong Zhang, Yingfa Chen, Shengding Hu, Zihang Xu, Junhao Chen, Moo Khai Hao, Xu Han, Zhen Leng Thai, Shuo Wang, Zhiyuan Liu, and Maosong Sun. ∞bench: Extending long context evaluation beyond 100k tokens, 2024.

[125] Yuhao Zhang, Victor Zhong, Danqi Chen, Gabor Angeli, and Christopher D. Manning. Position-aware attention and supervised data improve slot filling. In *Proceedings of the 2017 Conference on Empirical Methods in Natural Language Processing (EMNLP 2017)*, pages 35–45, 2017.

[126] Yuyu Zhang, Hanjun Dai, Zornitsa Kozareva, Alex Smola, and Le Song. Variational reasoning for question answering with knowledge graph. In *AAAI Conference on Artificial Intelligence*, 2017.

[127] Denny Zhou, Nathanael Scharli, Le Hou, Jason Wei, Nathan Scales, Xuezhi Wang, Dale Schuurmans, Olivier Bousquet, Quoc Le, and Ed Huai hsin Chi. Least-to-most prompting enables complex reasoning in large language models. *ArXiv*, abs/2205.10625, 2022.

[128] Hattie Zhou, Azade Nova, H. Larochelle, Aaron C. Courville, Behnam Neyshabur, and Hanie Sedghi. Teaching algorithmic reasoning via in-context learning. *ArXiv*, abs/2211.09066, 2022.

[129] Dawei Zhu, Nan Yang, Liang Wang, Yifan Song, Wenhao Wu, Furu Wei, and Sujian Li. PoSE: Efficient context window extension of LLMs via positional skip-wise training. In *The Twelfth International Conference on Learning Representations*, 2024.

[130] Fengbin Zhu, Wenqiang Lei, Chao Wang, Jianming Zheng, Soujanya Poria, and Tat-Seng Chua. Retrieving and reading: A comprehensive survey on open-domain question answering. *ArXiv*, abs/2101.00774, 2021.