

**ENHANCING NLP CAPABILITIES: STRATEGIES FOR LANGUAGE  
MODEL ADAPTATION IN LOW-RESOURCE TEXT  
CLASSIFICATION TASK AND EVALUATIONS**

---

A Dissertation  
Submitted to  
the Temple University Graduate Board

---

In Partial Fulfillment  
of the Requirements for the Degree  
DOCTOR OF PHILOSOPHY

---

by  
Hanzi Xu  
May 2025

Examining Committee Members:

Slobodan Vucetic, Advisory Chair, Computer and Information Sciences  
Eduard Dragut, Computer and Information Sciences  
Hongchang Gao, Computer and Information Sciences  
Huanmei Wu, External Reader, Health Services Administration and Policy

©  
Copyright  
2025

by

Hanzi Xu  

---

All Rights Reserved

## ABSTRACT

Nowadays, there are two approaches solving classification tasks in Natural Language Processing (NLP). The traditional way usually includes the adaptation of smaller Pre-trained Large Language Models (BERT, RoBERTa, etc.) to specific downstream tasks that offer both remarkable opportunities and significant challenges. While these models have been pivotal in achieving state-of-the-art results across numerous NLP tasks, their dependence on extensive annotated datasets for fine-tuning poses a substantial barrier, particularly in resource-scarce scenarios. The other approach is enabled by the emerging talent of massive-scale LLMs in recent years, where the classification tasks are solved by a one-for-all general-purposed auto-regressive model (GPT, Llama, etc). However, the strong performances of these models are overrated due to their inability to exhibit the expected comprehension of the task.

To address the challenges, we propose three innovative methodologies. Firstly, we introduce “OpenStance”, a novel stance detection system that operates effectively in a zero-shot setting. By leveraging a unique masking mechanism for weak supervision and utilizing existing textual entailment datasets for indirect supervision, OpenStance can handle open-domain topics and generalize across multiple domains without the need for extensive annotated data. Secondly, we present “X-shot”, a robust classification system that addresses the challenges of label variability in real-world applications. This system is capable of handling frequent-shot, few-shot, and zero-shot classification problems simultaneously, employing a flexible framework that adapts to the frequency of label occurrences and manages labels across the spectrum of availability. X-shot shows superior performance across diverse domains and label distributions. Thirdly, we propose “KNOW-

NO”, a new benchmark for evaluating the performance of generative LLMs in classification tasks, especially when gold labels are absent. This benchmark, along with a new evaluation metric called “OMNIACCURACY”, reveals the limitations of LLMs when they are forced to select from available label candidates, even when none are correct. This approach provides a more accurate assessment of LLMs’ performance in classification tasks, both when gold labels are present and absent.

This dissertation proposes innovative methodologies that minimize effort in adapting traditional LLMs to various classification tasks and also propose a novel evaluation metric to accurately assess the human-level discrimination intelligence of the newest LLMs in classification tasks. These methodologies aim to enhance the utility and discrimination ability of different generations of LLMs in the NLP domain, setting a foundation for future advancements in text classification tasks.

## ACKNOWLEDGMENTS

I would like to express my deepest gratitude to my PhD advisor, Slobodan Vucetic, for his invaluable guidance, encouragement, care, and support throughout these five years. Your mentorship has been a cornerstone of my academic journey.

I am also immensely grateful to my co-advisor, Wenpeng, for introducing me to the fascinating world of natural language processing (NLP) during our collaboration over the past two years. Your insights have helped me leave a solid footprint in this field and I will always be grateful.

I extend my sincere thanks to the members of my committee for their numerous teachings throughout my PhD studies. Your advice and feedback have been incredibly beneficial to my development. I am also thankful to all the teachers and individuals who sparked my interest in mathematics and AI, inspiring me to pursue this path. You have played a crucial role in shaping who I am today.

A heartfelt thank you goes to all my friends who have supported me along the way, including Borui, Fangshu (Aria), Chong, and many others, especially my best friends from high school—Mengjie (Marilyn), Zikai (KC), and Sicong (Chris). Your love and warmth have been a constant presence in my life.

I am profoundly grateful to my family, Xiaoping Xu and Yonghong Han. You are the strongest people I know, and your courage and determination in facing challenges have left an indelible mark on my life, guiding me like a beacon.

Lastly, to my furry companions—Aya, Maia, and Theia—who have always shown me unwavering support with their steadfast gazes and comforted me with their furry paws. Although some are no longer with me, they will always hold a significant place in my heart.

Finally, I have some words for myself. I have moved to the Bay Area for a while now. Last night, during a power outage, I found myself at my friend Mengjie's place, seeking refuge for the night. Both of us were immersed in revising our dissertations, sighing occasionally, our faces ghostly pale in the blue light of our laptop screens. I mentioned to her that I only had the acknowledgments left to write. Mengjie showed me her acknowledgments, playfully warning, "You better include me since I've included you" then threatened, "If you don't mention me, I'll remove you from mine—it's not final yet anyway." She laughed, adding, "Who else but ourselves will read these acknowledgments anyway?" Upon reflection, she's probably right. So, I'll write this next part just for myself.

As a child, I dreamed of becoming a scientist like Marie Curie. Now, I might dare to call myself a computer scientist—perhaps this is my dream realized, albeit in a different way than I initially imagined. I was never the perfect student, but numbers always captivated me, and the thrill of solving mathematical problems became my addiction—a fortunate predisposition that has shaped my life. Though circumstances led me away from my beloved geometry in college, I serendipitously discovered the beauty of Machine Learning in graduate school, setting me on the path of AI modeling. I feel incredibly fortunate to have found my place in this transformative technological revolution, and I'm grateful to have discovered something I both love and excel at.

Looking back at these four-plus years of doctoral studies, they often appear in shades of gray, composed of countless nights of anxiety, pain, and self-doubt. Yet it's precisely this gray backdrop that makes every small success shine so brilliantly. In my nearly 28 years of life, this journey has witnessed my greatest efforts and yielded my

proudest achievements. Now, as my 22-year academic journey draws to a close, my real life journey is just beginning. Perhaps years from now, reading these acknowledgments will seem naive, but I want to capture this moment exactly as I feel it. Life brings many emotions, but the feelings of our twenties come only once.

Congratulations to myself. May those I love stay healthy and content, may all women in STEM achieve their desired success, and may a promising future await me.

# TABLE OF CONTENTS

	Page
ABSTRACT.....	iii
ACKNOWLEDGMENTS .....	v
LIST OF TABLES.....	xi
LIST OF FIGURES .....	xii
CHAPTER	
1. INTRODUCTION .....	1
2. RELATED WORKS.....	4
2.1 Stance Detection. ....	4
2.2 Data Scarcity and Supervisions .....	5
2.3 Comprehension Ability of Auto-regressive LLMs.....	6
3. OPENSTANCE: REAL-WORLD ZERO-SHOT STANCE DETECTION.....	8
3.1 Introduction.....	8
3.2 Problem Definition.....	10
3.3 Methodology.....	11
3.4 Experiments. ....	15
3.4.1 Datasets.....	15
3.4.2 Baselines. ....	17
3.4.3 Setting. ....	18
3.4.4 Results.....	19
3.4.5 Analysis.....	20

3.5 Conclusion.....	24
4. X-SHOT: A UNIFIED SYSTEM TO HANDLE FREQUENT, FEW-SHOT AND ZERO-SHOT LEARNING SIMULTANEOUSLY IN CLASSIFICATION.....	26
4.1 Introduction.....	26
4.2 Problem Statement.....	28
4.3 Methodology.....	30
4.3.1 BinBin Architecture.....	30
4.3.2 Supervision Acquisition For Low-shot Labels .....	30
4.4 Experiments.. ..	33
4.4.1 Experimental Setting.....	33
4.4.2 Results.....	37
4.4.3 Analyses.....	38
4.5 Conclusion.. ..	43
5. LLMS’ CLASSIFICATION PERFORMANCE IS OVERCLAIMED.....	46
5.1 Introduction.....	46
5.2 Approach.....	49
5.2.1 KNOW-NO Benchmark.....	49
5.2.2 Prompting LLMs.....	52
5.2.3 OMNIACCURACY: A New Evaluation Metric. ....	54
5.3 Experiments.. ..	54
5.3.1 Main Results. ....	56
5.3.2 Analysis.....	57
5.4 Conclusion and Future Work.. ..	62

6. CONCLUSION.....	64
BIBLIOGRAPHY.....	66
APPENDICES	
A ADDITIONAL MODELING DETAILS OF BINBIN .....	75
B ADDITIONAL MODELING DETAILS OF KNOW-NO. ....	79

## LIST OF TABLES

Table	Page
3.1 Dataset statistics .....	16
3.2 Experiment results on SemT6, VAST and Perspectrum.....	17
4.1 Statistics of dataset labels.....	34
4.2 Main results on three benchmark target tasks.....	37
4.3 Ablation study: deleting top-10 similar tasks.....	40
5.1 Statistics of KNOW-NO.....	49
5.2 Prompting LLMs in KNOW-NO.....	50
5.3 OMNIACCURACY of LLMs and humans.....	55
5.4 Ablation study: w/ $G$ vs w/ $G + None$ .....	62

## LIST OF FIGURES

Figure	Page
3.1 Mean F1 vs. size of $D_{\text{weak}}$ .....	21
4.1 Unified classification problems in BinBin.....	29
4.2 Indirect Supervision for BinBin.....	29
4.3 Weak Supervision template for zero-shot labels.....	32
4.4 Ablation study of BinBin.....	38
4.5 Backbone models with different scales and architectures.....	40
4.6 Ablation study: #instances vs. #tasks.....	41
4.7 Classification to binary BinBin.....	44
4.8 Super-Naturalinstructions task example.....	44
4.9 Super-Naturalinstructions to binary BinBin.....	45
5.1 Comparison between human and latest LLMs.....	47
5.2 Example of EQUINFER, the equation labeled with “B” is correct.....	50
5.3 Humans vs. LLMs on MC-TEST.....	59
5.4 LLMs’ output pattern distribution in NO-HINT on MC-TEST.....	60
A.1 GPT Template.....	76
A.2 GPT Template .....	77
B.1 Scaling the length of context around the equation in EQUINFER.....	79
B.2 Example of MC-TEST.....	80
B.3 Example of BANK-77.....	80
B.4 Example of EQUINFER .....	81

B.5 Error patterns.....	83
B.6 Human behavior.....	84

# CHAPTER 1

## INTRODUCTION

The field of Natural Language Processing (NLP) has witnessed remarkable advancements in recent years, particularly with the advent of Large Language Models (LLMs). These models have revolutionized our approach to various NLP tasks, including text classification. However, as we push the boundaries of what these models can achieve, we also uncover new challenges and limitations that demand innovative solutions.

Our journey begins with the fundamental task of stance detection, a crucial component in developing the inference capabilities of NLP models. Stance detection aims to differentiate the attitude (e.g., support, oppose, or neutral) of a text towards a topic (Walker et al., 2012b). The complexity of this task lies in the unpredictable nature of textual expressions and the varying sizes of topics in real-world scenarios. While previous research has made strides in zero-shot stance detection (Mohammad et al., 2016; Allaway & McKeown, 2020), these approaches often fall short of true zero-shot capabilities due to limitations in domain diversity, topic size, and the need for task-specific supervision.

To address these shortcomings, we introduce OpenStance, a novel open-domain zero-shot stance detection system. OpenStance redefines the zero-shot paradigm by eliminating the need for task-specific supervision and generalizing across multiple domains and topic sizes. Our approach combines indirect supervision from textual entailment datasets like MNLI (Williams et al., 2018) with weak supervision generated through a novel masking mechanism using GPT-3 (Brown et al., 2020a). This innovative method not only achieves robust performance across diverse datasets but also outperforms task-specific supervised models in some cases.

As we delve deeper into the challenges of text classification, we encounter the issue of label variability in real-world applications. Traditional classification systems often struggle with the spectrum of label availability, from frequent-shot to few-shot and zero-shot scenarios. To tackle this problem, we develop X-shot, a unified classification system capable of handling all these scenarios simultaneously. X-shot leverages instruction tuning and a novel binary classification architecture to adapt to varying label frequencies without prior constraints.

To evaluate X-shot, we recompile three representative classification datasets: *FewRel* (Han et al., 2018), *MAVEN* (Wang et al., 2020), and *RAMS* (Ebner et al., 2020). These datasets, sourced from diverse domains and featuring vast label counts, pose a formidable challenge to contemporary text classification systems. Our experiments demonstrate X-shot’s resilience across datasets, consistently outperforming leading baselines, including GPT-3.5.

While our research on OpenStance and X-shot pushes the boundaries of the classification capabilities of traditional LLMs, the new generation of general-purpose auto-regressive model LLMs can directly solve classification problems as a generation task. This new approach, although demonstrating strong performance, leads us to question the true performance of these newest models in classification tasks. This brings us to our final work, KNOW-NO, which addresses a critical gap in the evaluation of LLMs. We notice that even advanced LLMs like GPT-4 struggle and tend to pick a wrong answer when the correct answer is not provided among the options, unlike humans who can express uncertainty in such situation. This observation led us to develop KNOW-NO, a new benchmark for evaluating LLMs in classification tasks when gold labels are both present

and absent. We introduce the CLASSIFY-W/O-GOLD task and propose a novel evaluation metric, OMNIACCURACY, designed to assess the human-level discrimination intelligence of LLMs more comprehensively. Our KNOW-NO benchmark encompasses three standard classification tasks: BANK-77 (Casanueva et al., 2020), MC-TEST (Richardson et al., 2013), and EQUINFER. These tasks cover a wide range of input lengths, label sizes, and label scopes, providing a robust framework for evaluating LLMs’ true classification capabilities.

Through this series of works, we address critical challenges in the application and evaluation of LLMs in NLP text classifications. From redefining zero-shot stance detection to creating a unified system for all-shot classification and finally developing a more comprehensive evaluation framework, our research contributes to enhancing the utility and discrimination ability of LLMs from different generations in the NLP domain. These advancements not only push the boundaries of current LLM capabilities but also pave the way for future research that could further reduce the reliance on large annotated datasets and extend the robustness and reliability of LLMs in real-world applications.

## CHAPTER 2

### RELATED WORKS

#### 2.1 Stance Detection

Stance detection, as a recent member of the NLP family, was mainly driven by newly created datasets. In the past studies, datasets have been constructed from diverse domains like online debate forums Walker et al. (2012a); Hasan & Ng (2014); Abbott et al. (2016), news comments Krejzl et al. (2017); Lozhnikov et al. (2018), Twitter Mohammad et al. (2016); Küçük (2017); Tsakalidis et al. (2018)), etc.

Recently, researchers started to work on zero-shot stance detection in order to build a system that can handle unseen topics. Most work split the collected topic-aware annotations into train and test within the same domain. Allaway & McKeown (2020) made use of topic similarity to connect unseen topics with seen topics. Allaway et al. (2021) designed adversarial learning to learn domain-independent information and topic-invariant representations. Similarly, Wang & Wang (2021) applied adversarial learning to extract stance-related but domain-invariant features existed among different domains. Liu et al. (2021) utilized common sense knowledge from ConceptNet Speer et al. (2017) to introduce extra knowledge of the relations between the texts and topics. Most prior systems worked on a single domain and were tested on a small number of unseen topics. Li et al. (2021) tried to test on various unseen datasets by jointly optimizing on multiple training datasets. However, they still assumed that part of the topics or domains has rich annotations. In contrast, our goal is to design a system that can handle stance detection in an open world without requiring any domain constraints or topic-specific annotations.

## 2.2 Data Scarcity and Supervisions

The topic of data-imbalanced NLP Tasks is first discussed in the context of binary classification datasets, where the negative-to-positive ratio ranges from 5 to 200 (Li et al., 2020). Subsequent works have extended this to multi-class classification settings with a long-tail distribution, where a subset of labels occurs in less than 5% of the training data (Cao et al. (2019); Xu et al. (2023b)). Two common solutions to this problem are reweighting the loss function and resampling the data in mini-batches (Li et al. (2020); Cao et al. (2019); Xu et al. (2023b); Buda et al. (2018); Pouyanfar et al. (2018)). Even though the data imbalance/long-tail problem also tackles different label occurrences, this setting differs from the *X*-Shot problem in three dimensions: i) the presence of zero-shot labels in our setting; ii) the inclusion of a “None” class in the test set, representing cases where none of the labels fit; iii) prior work addressed different imbalance/long-tail problems with separate systems (a system for task/domain A could not be applied to another task/domain), whereas we are modeling these problems within a unified system.

There has been a burgeoning interest in Indirect Supervision (Yin et al., 2023) in recent years. Here, easily available signals from relevant tasks (source tasks) are used to aid in learning the target task. Using the entailment task for Indirect Supervision in zero-shot classification was first proposed by (Yin et al., 2019) and has since been adapted for a variety of NLP tasks, including few-shot intent identification (Zhang et al., 2020; Xu et al., 2023a), event argument extraction (Sainz et al., 2022) and relation extraction (Lu et al., 2022). Beyond entailment, knowledge from areas like question answering (Yin et al., 2021), summarization (Lu et al., 2022) and dense retrievers (Xu et al., 2023a) has been incorporated. However, previous Indirect Supervision is collected from a single source

task. In contrast, our work is inspired by recent studies in instruction learning observing the efficacy of NLP models when given task instructions and their ability to generalize knowledge across tasks (Wang et al., 2022; Mishra et al., 2022; Ye et al., 2021).

### **2.3 Comprehension Ability of Auto-regressive LLMs**

It has been a trend to use LLM generation to solve classification problems, either as standalone classification tasks Sun et al. (2023b,a); Zhang et al. (2024) or mixed with other NLP tasks in multi-task learning Longpre et al. (2023); Wang et al. (2022); Mishra et al. (2022). The classification problem is usually constructed in a traditional setup, where several options, including the correct answer, are provided for a given question. Remarkable performance metrics from GPT have been observed, such as Sun et al. (2023a) demonstrating 95-98% accuracy in sentiment analysis (SST-2/IMDB/Yelp), over 93% in semantic role labeling (CoNLL2009), and 92-98% in part-of-speech identification (Penn, WSJTweets) with few-shot demonstrations. In their subsequent study, an average accuracy above 90% on five well-known NLP benchmark text classification datasets (SST-2, AGNews, R8, R52, MR) reported in Sun et al. (2023b) with zero-shot prompting. As the latest LLMs are seen as reliable solutions for NLP classification, their true understanding of the essence of the classification task has not been properly evaluated.

*LLMs' Challenges in Task Comprehension* It is essential to investigate the rationale behind the model's predictions and determine the extent to which we can trust its output Gunning et al. (2019); Rudin (2019). Recent studies have raised concerns about whether LLMs truly understand the tasks they perform despite their good performance.

Many studies have shown that LLMs can achieve promising performance when they are asked to provide step-by-step reasoning in their answers Wei et al. (2022); Zelikman et al. (2022); Li et al. (2022b). However, LLM-generated reasoning has been found to be unfaithful despite its apparent effectiveness Turpin et al. (2023); Lanham et al. (2023). The performance boost might be attributed to the extra computation provided by the explanation tokens Wei et al. (2022); Lanham et al. (2023). An inspection of the reasoning rationales generated by the model reveals that they often fail to make logical sense Zelikman et al. (2022). It is common to see rationales that simply repeat content from the question without providing a reasonable explanation. Many of the rationales fail to effectively support the claims or address the reasoning required, indicating that the model often does not truly understand the content and reasoning behind the question, even if it arrives at the correct answer.

Similarly, recent works have been evaluating the cognition-inspired intelligence of LLMs by testing latest LLMs on NLP generation tasks to evaluate their capabilities across multiple dimensions, including reading comprehension, commonsense reasoning, discourse comprehension, and paragraph/document-level understanding Wang et al. (2024); Mahowald et al. (2023). When the problem-solving process is decomposed into three sub-steps: knowledge recall, knowledge utilization, and answer generation, the results reveal that, despite high scores in answer generation performance, the score for knowledge utilization is significantly lower, by up to 34%. Additionally, they point out that LLMs' proficiency in language processing does not necessarily translate to a similar level of cognitive capability if looking at the correlation between different capabilities, revealing the current shortcomings of LLMs in true understanding.

## CHAPTER 3

### OPENSTANCE: REAL-WORLD ZERO-SHOT STANCE DETECTION

#### 3.1 Introduction

Stance detection differentiates the attitude (e.g., support, oppose, or neutral) of a text towards a topic (Walker et al., 2012b). The topic can be a phrase or a complete sentence. The same text can express the author’s positions on many different topics. For example, a tweet on climate warming may also express attitudes about environmental policies as well as the debate between electric or fuel cars. Such compound expression can be seen on all online platforms, including News outlets, Twitter, blogs, etc. Therefore, stance detection can be a complicated task that is essential for developing the inference capability of NLP models as well as other disciplines such as politics, journalism, etc.

Since the textual expressions and the size of topics in the real world are unpredictable, zero-shot stance detection has become the mainstream research direction: topics in the testset are unseen during training. For example, (Mohammad et al., 2016) created a dataset SemT6 based on tweets with six noun phrases as topics. One of the topics was reserved for testing and the remaining were used for training. (Allaway & McKeown, 2020) extended the topic size on the domain of news comments by covering 4,000/600 topics in training/testing.

However, despite the change in the domain and topic size, there are three major limitations in previous studies *which make the task not a real zero-shot task*: (i) the dataset only contains texts from a single domain, such as news comments in VAST (Allaway & McKeown, 2020) and tweets in SemT6 (Mohammad et al., 2016); (ii) most literature studied only a limited size of topics with a single textual form (either noun phrases or sentential

claims), e.g., (Mohammad et al., 2016; Conforti et al., 2020); (iii) rich annotation for at least part of the topics is always required, which is not possible in real-world applications because data collection can be very time-consuming and costly (Enayati et al., 2021). Those limitations lead to an impractical zero-shot stance detection system that cannot generalize well to unseen domains and open-form topics.

In this work, we re-define what a zero-shot stance detection should be. Specifically, we define OpenStance: an open-domain zero-shot stance detection, aiming to build a system that can work in the real world without any specific attention to the text domains or topic forms. More importantly, no task-specific supervision is needed. To achieve this, we propose to combine two types of supervision: indirect supervision and weak supervision. The indirect supervision comes from textual entailment—we treat the stance detection problem as a textual entailment task since the attitude toward a topic should be inferred from the input text. Therefore, the existing entailment datasets, such as MNLI (Williams et al., 2018), can contribute supervision to the zero-shot setting. To collect supervision that is more specific to the OpenStance task, we design two MASK choices (MASK-topic and MASK-text) to prompt GPT-3 (Brown et al., 2020a) to generate weakly supervised data. Given an input text and a stance label (support, oppose, or neutral), MASK-topic predicts what topic is appropriate based on the content; given a topic and a label, MASK-text seeks the text that most likely holds this stance. The collection of weakly supervised data only needs the unlabeled texts and the set of topics that users want to include. The joint power of indirect supervision and weak supervision will be evaluated on VAST, SemT6 and Perspectrum (Chen et al., 2019), three popular datasets that cover distinct domains, different sizes and diverse textual forms of topics. Experimental results

show that although no task-specific supervision is used, our system can get robust performance on all three datasets, even outperforming the task-specific supervised models (72.6 vs. 69.3 by mean F1 over the three datasets).

Our contributions are threefold: (i) we define OpenStance, an open-domain zero-shot stance detection task, that fulfills real-world requirements while having never been studied before; (ii) we design a novel masking mechanism to let GPT-3 generate weakly supervised data for OpenStance. This mechanism can inspire other NLP tasks that detect relations between two pieces of texts; (iii) our approach, integrating indirect supervision and weak supervision, demonstrates outstanding generalization among three datasets that cover a wide range of text domains, topic sizes and topic forms.

### 3.2 Problem Definition

OpenStance has the following requirements:

- An instance includes three items: text  $s$ , topic  $t$  and a stance label  $l$  ( $l \in \{\text{support, oppose, neutral}\}$ ); the task is to learn the function  $f(s, t) \rightarrow l$ ;
- The text  $s$  can come from any domain; the topic  $t$  can be any textual expressions, such as a noun phrase “gun control” or a sentential claim “climate change is a real concern”;
- All labeled instances  $\{(s, t, l)\}$  only exist in test; no train or dev is provided;
- Previous work used different metrics for the evaluation. For example, VAST (Allaway & McKeown, 2020) used macro-averaged F1 regarding stance labels, while studies on SemT6 (Allaway et al., 2021; Liang et al., 2022) reported the F1 scores per topic. To make systems be comparable, we unify the evaluation and use the label-oriented macro F1 as our main metric. OpenStance vs. prior zero-shot stance detection. Prior studies of zero-

shot stance detection worked on a single dataset  $D^i$  in which all texts  $s$  comes from the same domain. Topics  $t$  in the dataset are split into *train*, *dev* and *test* disjointly. The main issue is that a model that fits  $D^i$  does not work well on a new dataset  $D^j$  that may contain  $s$  of different domains and unseen  $t$ . For example, a model trained on VAST can only get F1 49.0% on Perspectrum, which is around the performance of random guess. OpenStance aims at handling multiple datasets of open domains and open-form topics without looking at their *train* and *dev*.

Stance detection is essentially a textual entailment problem if we treat the text  $s$  as the premise, and the stance towards the topic  $t$  as the hypothesis. This motivates us to use indirect supervision from textual entailment to deal with the stance detection problem. Nevertheless, there are two distinctions between them: (i) even though we can match  $l$  of stance detection with the labels of textual entailment: support  $\rightarrow$  entailment, oppose  $\rightarrow$  contradict and neutral  $\rightarrow$  neutral, whether a topic  $t$  in stance detection can be treated as a hypothesis depends on the text form of  $t$ . If  $t$  is noun phrases such as “gun control”,  $t$  cannot act as a hypothesis alone as there is no stance in it; if  $t$  is a sentential claim such as “climate change is a real concern”, inferring the truth value of this hypothesis is exactly a textual entailment problem. This observation motivates us to test OpenStance on topics of both phrase forms and sentence forms; (ii) Zero-shot textual entailment means the size of the annotated instances for labels is zero, while OpenStance requires the topics have zero labeled examples.

### 3.3 Methodology

This section introduces how we collect and combine indirect supervision and weak supervision to solve OpenStance.

The part one is *Indirect Supervision*. As we discussed in Section 3.2, stance detection is a case of textual entailment since the stance  $l$  towards a topic  $t$  should be inferred from the text  $s$ . To handle the zero-shot challenge in OpenStance, textual entailment is a natural choice for indirect supervision.

Specifically, we first cast stance detection instances into the textual entailment format by combining  $l$  and  $t$  as a sentential hypothesis  $h$ , such as “it supports topic”, and treating the  $s$  as the premise  $p$ ; then a pretrained model on MNLI (Williams et al., 2018), one of the largest entailment dataset, is ready to predict the relationship between the  $p$  and  $h$ . An entailed (resp. contradicted or neutral)  $h$  means the topic  $t$  is supported (resp. opposed or neutral) by the text  $s$ .

Unfortunately, the indirect supervision from textual entailment may not perform well enough in real-world OpenStance considering the widely known brittleness of pretrained entailment models and the open domains and open-form topics in OpenStance. Therefore, in addition to the indirect supervision from textual entailment, we will collect weak supervision that is aligned better with the texts  $\{x\}$  and the topics  $\{t\}$ .

The part one is *Weak Supervision*. For the next step, we would like to create some weakly supervised data using easily available resources to obtain a better understanding of the target task. We used GPT-3 (Brown et al., 2020a), a pre-trained autoregressive language model that can perform text completion at (arguably) a near-human level, to help us create some weakly labeled instances.

We form incomplete sentences using prompts, and let the GPT-3 complete them. Since a stance label  $l$  connects the text  $s$  and the topic  $t$  and such connection is unavailable in a zero-shot setting, the construction of incomplete sentences is driven by two questions:

(i) given an input text  $s$  and a stance, e.g., support, what topics are supported by  $s$ ? (ii) given a topic and a stance, for example, support, what texts support this topic? As a result, there are two kinds of prompts: MASK-Topic and MASK-Text. To implement the two masking mechanisms, we need to prepare three sets: the raw texts  $\{s\}$ , a set of topics  $\{t\}$ , and the known stance labels  $\{\text{support, oppose, neutral}\}$ . It is noteworthy that no topic-specific human annotations are used here.

The first framework is **MASK-Topic**. In this masking framework, we randomly choose a text from “ $\{s\}$ ” and a stance label from  $\{\text{support, oppose, neutral}\}$ , then build the prompt as: “S/he claims text, so s/he label the idea of MASK”.

For example, when the text is “Coldest and wettest summer in memory” and the label is oppose, the prompt would be “S/he claims coldest and wettest summer in memory, so s/he opposes the idea of”. Then, this prompt is fed into GPT-3, and the completion “global warming” would be the predicted topic.

The second framework is **MASK-Text**: In this case, we randomly choose a topic from  $\{t\}$  and a stance label towards it, then build the prompt as: “His/her attitude towards topic is label because s/he thinks MASK”

For example, when the topic is “climate change is a real concern”, the label is “oppose”, the completed sentence filled by GPT-3 could be “His attitude towards climate change is a real concern is opposition because s/he thinks the science behind climate change is not settled”.

For any dataset of stance detection, we first collect the three sets (i.e.,  $\{s\}$ ,  $\{t\}$ , and  $\{l\}$ ) from the label-free training set without peeking at any gold annotations, then use MASK-Topic and MASK-Text prompts to generate equal number of weakly supervised

examples. We will study which masking scheme is more effective in experiments. In addition, to have a fair comparison with supervised methods that learn on the *train* of a task, we make sure our generated weakly supervised data has the same size as the *train* for any target task.

Although noise is common in weakly supervised data, GPT-3 performs badly on neutral completions for both MASK-Topic and MASK-Text tasks. This is not a surprise for the MASK-Topic since the GPT-3 is asked to provide a topic that the given text has a neutral attitude for, while most texts, obtained from unlabeled train and originally extracted from social networks, usually express a strong attitude. Furthermore, in MASK-Text, even though the GPT-3 can output a text given the neutral label towards a topic, the response is very general and does not provide any insights. For example, when the template is "His attitude towards high school writing skills is neutral because he thinks [MASK]", GPT-3 fills out the MASK with "that they are important but not essential." Obviously, it is much easier to generate text with a clear attitude compared to a neutral stance. On the one hand, GPT-3 may not really understand what a neutral stance is. On the other hand, even humans cannot easily write a neutral opinion towards a topic. Since the quality of generated neutral instances is not very promising, we take the same approach as how VAST (Allaway & McKeown, 2020) collected its neutral samples: matching texts with random topics in the dataset.

For *Training strategy*, to keep consistent format and make full use of the entailment reasoning framework, we convert all phrase-form topic in the weak supervision data into a sentence-form hypothesis with the positive stance, i.e., "he is in favor of topic" (note that this does not change the original label). Then, we randomly split the weak supervision data

as train (80%) and dev (20%). Given the entailment dataset MNLI as the indirect supervision data ( $D_{ind}$ ) and weakly supervised data ( $D_{weak}$ ) from GPT-3, we first pretrain a RoBERTa-large (Liu et al., 2019a) on  $D_{ind}$ , then finetune on  $D_{weak}$ . In inference, we test the final model on the test of each task, checking the system’s generalization ability on diverse domains without optimizing on any domain-specific train.

## 3.4 Experiments

### 3.4.1 Datasets

We choose datasets that can cover (i) multiple domains, (ii) different sizes of unseen topics, and (iii) various textual forms of topics (phrase-form and sentence-form). Therefore, we evaluate on three mainstream stance detection datasets: SemT6 (Mohammad et al., 2016), VAST (Allaway & McKeown, 2020) and Perspectrum (Chen et al., 2019). We discard their training sets and dev sets to satisfy the definition of OpenStance.

The first dataset is *SemT6 (Mohammad et al., 2016)* containing texts from the tweet domain regarding 6 topics: Donald Trump, Atheism, Climate Change is a real Concern, Feminist Movement, Hillary Clinton, and Legalization of Abortion. It is a three-way stance detection problem with labels {support, oppose, neutral}. Note that the prior applications of SemT6 for zero-shot stance detection always trained on five topics and tested on the remaining one. To match the motivation of OpenStance, we treat the whole SemT6 data as test, i.e., all six topics are unseen. When we report the data-specific supervised performance, we follow prior work to regard any five topics as seen and test on the sixth topic; each topic will have the chance to be unseen, and the average performance is reported.

Table 3.1. Dataset statistics

	domain	#topic train/test	topic form	#labels
SemT6	tweet	6	phrase	3
VAST	debate	4641/600	phrase	3
Persp.	debate	541/227	sentence	2

The second dataset is *VAST* (Allaway & McKeown, 2020). In contrast to SemT6, VAST contains text from the New York Times “Room for Debate” section, and many more topics (4,003 in *train*, 383 in *dev* and 600 in *test*). Those diverse topics, covering various themes, such as education, politics, and public health, are short phrases that are first automatically extracted and then modified by human annotators. Like SemT6, it also has three stance labels, but the neutral topics were randomly picked from the whole topic set. For our OpenStance task, we only use its *test* to evaluate our system and do not touch the gold labels of its *train* and *dev*.

The third dataset is *Perspectrum* (Chen et al., 2019) is a binary stance detection benchmark (label is support or oppose) with two main distinctions with SemT6 and VAST: (i) both its text and topics were collected from several debating websites, and (ii) the topics are sentences rather than noun phrases. Similar to VAST, we do not train our model on its *train* and *dev*. The performance on *test* will be reported. Since there are no neutral samples in this dataset, when the model is pretrained as a 3-way classifier, we set the probability threshold as 1/3 on the oppose label: any prediction that has the oppose probabilities lower than 1/3 will be considered as support. Otherwise, the label would be oppose.

Table 3.2. Experiment results on SemT6, VAST and Perspectrum

		F1 Score					
		SemT6	VAST	Persp.	mean		
random guess		32.0	33.3	49.8	38.3		
data-specific supervised learning (prior SOTA)		38.9	78.0	91.0	69.3		
cross-domain transfer		SemT6 as <u>train</u>	38.9	28.9	47.7	38.5	
		VAST as <u>train</u>	55.4	78.0	49.0	60.8	
		Pers as <u>train</u>	26.7	27.0	91.0	48.2	
open-domain transfer	baseline	BERT	22.7	36.8	36.5	32.0	
		GPT-3	30.5	34.2	39.9	34.9	
		Cosine	31.5	35.9	62.7	43.4	
	ours	$D_{ind}$ and	SemT6-based $D_{weak}$	63.7	69.8	82.8	72.1
			VAST-based $D_{weak}$	64.3	72.0	80.4	72.2
			Persp-based $D_{weak}$	64.5	68.7	79.5	70.9
			joint $D_{weak}$	63.2	73.5	81.0	<b>72.6</b>
		w/o indirect	49.6	64.6	38.2	50.8	
		w/o weak	45.3	53.7	79.1	59.4	
		w/o MASK-Topic	45.5	65.2	74.2	61.6	
w/o MASK-Text	63.4	70.8	78.2	70.8			

The detailed statistics of the three datasets are listed in Table 5.1.

### 3.4.2 Baselines

There are no prior systems that work on this new OpenStance problem since no training data is available. Here, we consider three baselines that can work on an unsupervised scheme.

The first baseline is *BERT* (Devlin et al., 2019). Given the (text, topic) as input, “BERT-large-uncased” is used as a masked language model to predict the masked token in “text, it [MASK] topic”. BERT will output the probabilities of the three label tokens {support, oppose, neutral} and the label that receives the highest probability would be the predicted stance.

The second baseline is *GPT-3* (Brown et al., 2020a). Given the text and the topic with the instruction telling the model what task we are trying to accomplish, GPT-3 is able to complete the prompt by choosing one of the given labels {support, oppose, neutral}. GPT-3 also has

functions designed for classification, but the text completion scheme does a better job on this stance detection task. Our prompt is as “Given a topic and a text, determine whether the stance of the text is support, against, or neutral to the topic. Topic: Atheism. Text: Everyone is able to believe in whatever they want. Stance:” and let GPT finish the writing.

The third baseline is *Cosine similarity*. We compare the similarities between the text and a hypothesis sentence that combines label and topic, such as “it supports the topic”, “it opposes the topic”, or “it is unrelated to the topic”. We first get the sentential representations by sentence- BERT (Reimers & Gurevych, 2019a), then choose the label whose resulting hypothesis obtains the highest cosine similarity score.

In addition to the unsupervised baselines, we further consider the data-specific supervised training as the upperbound, and the following variants of our system: i) only MASK-Text or MASK-Topic; ii) only indirect supervision or weak supervision.

### **3.4.3 Setting**

The engine we chose for GPT-3 is “curie”, which gives good quality at a reasonable price. There are several parameters that we played with. We set the temperature, which goes from 0 to 1 and controls the randomness of the completion generated, as 0.8 for MASK-Topic and 0.9 for MASK-Text for more diverse results. The randomness for MASK-Text is slightly higher because we noticed that for some datasets the number of topics is extremely limited, such as SemT6, which only has 6 topics in total; therefore, we want to force diverse responses from GPT-3. The max number of tokens GPT-3 can generate is 6 for MASK-Topic and 150 for MASK-Text. It is worth mentioning that GPT-3 will not necessarily generate as much as the upper bound, sometimes not even close. We let the stop word be "zn", so that it stops generating when it reaches a new paragraph. “top\_p” is set as

1, letting all tokens in the vocabulary been used. “frequency penalty” is 0.3 for MASK-Text to avoid the model producing the same line again and again.

All models are optimized using AdamW (Loshchilov & Hutter, 2019). Learning rate  $1e-6$ , batch size 16, maximal (premise, hypothesis) length is 200. The system is trained for 20 epochs on train and the best model on dev is kept.

#### 3.4.4 Result

Table 3.2 lists the main results. We first include “data-specific supervised learning” as the upperbound performance and the “cross-domain transfer” that takes each dataset as the source domain and tests on others respectively. Both settings try to explore the upper limit when we apply human-annotated supervision. Our core task, OpenStance, is evaluated in the last three blocks.

From the baseline block, we can observe that for all domains, baseline methods mostly perform like random guess, except for the slight improvement of the “cosine” approach over Perspectrum. This result indicates the difficulty of the real-world OpenStance task we proposed. Although BERT and GPT-3 are the top-tier pre-trained language models, they still cannot handle OpenStance well.

Then look at our approach that combines indirect supervision data ( $D_{ind}$ ) and weak supervision data ( $D_{weak}$ ). Note that  $D_{weak}$  is collected based on the label-free train of VAST, SemT6 or Perspectrum. We try  $D_{weak}$  for each task domain and also put them jointly (i.e., “joint  $D_{weak}$ ”). We note that all four versions of  $D_{weak}$  result in very consistent performance—mostly around 72% by the “mean”. This clearly supports the robustness of our model: it is less affected by the original domain of the text and topic, and a single system based on each domain or their combination can perform well on all domains.

The last block of Table 3.2 reports the ablation study, where we discard individual source of supervision (indirect or weak) or individual masking scheme (MASK-Text or MASK-Topic). We observe that i) indirect supervision and weak supervision play complementary roles for the task OpenStance; and they both outperform baselines by large margins, and ii) both masking schemes help, and the MASK-Topic contributes more. This is maybe because MASK-Topic requires the GPT-3 to generate shorter texts than MASK-Text so that MASK-Topic can yield higher-quality data. Additionally, deriving supporting sentences for a given topic sometimes requires substantial background knowledge and solid reasoning, which is still a difficult task for GPT-3.

### 3.4.5 Analysis

Next, we conduct a deep analysis for the system robustness towards prompts ( $Q_1$ ), the required size of  $D_{weak}$  ( $Q_2$ ), the noise in generated  $D_{weak}$  ( $Q_3$ ), and the error patterns made by our system ( $Q_4$ ).

Regarding *the Robustness of dealing with prompts*. Prompt design takes place in both GPT-3 completion and the conversion from stance detection to textual entailment. When generating the prompt for GPT-3, how we construct the prompt in MASK-Topic and MASK-Text can make a huge impact on the completion received. In MASK-Topic, we use the prompt “He said text, so he label the idea of [MASK]”. The reason why we add “the idea of” at the end of the prompt is because it helps the model understand that we want a noun phrase. Otherwise, we will see completions like “that”, “it”, etc. Similarly, in MASK-Text, the final prompt we use is “His attitude towards topic is label because he thinks [MASK]”. Considering the freedom of GPT-3 completion, we add “s/he thinks” at the end of the prompt, forcing GPT-3 to generate a reasoning for the given topic/label pair. If we don’t add “he

thinks” at the end, it would be common to see GPT-3 repeating the given sentence in the generated completion. In addition, when the label is neutral, such as the prompt “His attitude towards high school writing skills is neutral because he thinks [MASK]”, GPT-3 would output sentences like “he does not have a strong opinion either way” if we don’t have “he thinks” at the end. After the modification, responses would make more sense, such as “that they are important but not essential.” These tricks in prompt design suggest that it is essential to make the sentence structure as clear as possible and provide content that helps to instruct the model on what we want.

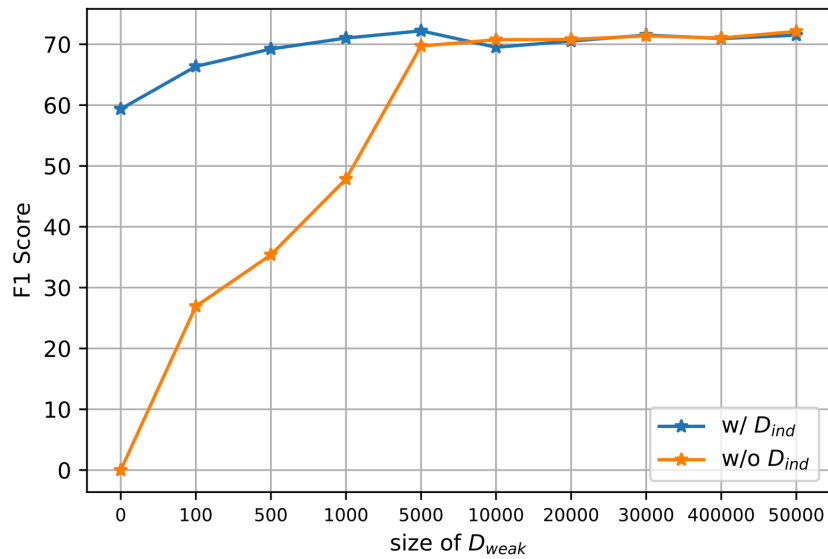


Figure 3.1. Mean F1 vs. size of  $D_{weak}$ .

Considering the freedom of GPT-3 completion, we add “s/he thinks” at the end of the prompt, forcing GPT-3 to generate a reasoning for the given topic/label pair. If we don’t add “he thinks” at the end, it would be common to see GPT-3 repeating the given sentence in the generated completion. In addition, when the label is neutral, such as the prompt “His attitude towards high school writing skills is neutral because he thinks [MASK]”, GPT-3 would output sentences like “he does not have a strong opinion either way” if we don’t have “he

thinks” at the end. After the modification, responses would make more sense, such as “that they are important but not essential.” These tricks in prompt design suggest that it is essential to make the sentence structure as clear as possible and provide content that helps to instruct the model on what we want.

When we convert the topic phrase into a sentential hypothesis, we again get involved in the prompt design. During training, we stick with “he is in favor of topic” template to limit the training size, but in the testing, we found the majority voting of four templates (“he/she is in favor of topic” and “he/she opposes topic”) lead to comparable performance with “he is in favor of topic”. This indicates the pre-trained entailment system is considerably robust in dealing with hypotheses derived from different templates.

Regarding *how much weakly supervised data is needed?* We answer this question by applying  $D_{weak}$  alone or together with  $D_{ind}$ . For each case, we test on sizes varying from 100 to 50,000 and report the average results over 3 random seeds. From the Figure 3.1, we can see that both settings can reach similar performance when we collect over 10k data of  $D_{weak}$ , but the pretraining on  $D_{ind}$  can dramatically reduce the required size of  $D_{weak}$ : from 10k to around 500.

Regarding the *Error patterns of weakly supervised data*. We collect typical error patterns in  $D_{weak}$  derived by MASK-Topic and MASK-Text separately.

For MASK-Topic, there are three typical error types. 1) Incomplete generation. Sometimes GPT-3 fails to give a complete topic phrase and cuts in the middle even though it hasn’t reached the maximum token limit. For example: “He claims 16 year olds are informed enough to cast a vote, so he supports the idea of GIVING 16-YEAR-OLDS” In this example, the topic given by GPT-3 is “giving 16-year-olds”, which is not

a complete phrase as we expected. This kind of errors indicate that GPT-3 sometimes stops generating before providing a complete idea even when the word limit is not exceeded; 2) Failure in understanding the stance. Since we are providing opposite labels (i.e., support and oppose), we hope that GPT-3 would produce distinct topics that hold opposite stances. However, sometimes GPT-3 fails to understand the stances when generating topics. For example: “He claims A higher minimum wage means less crime, so he supports the idea of A MINIMUM WAGE” vs “He claims A higher minimum wage means less crime, so he opposes the idea of A MINIMUM WAGE”. This error type is the most common one in the weakly supervised data (approximately 85% error instances), indicating that GPT-3 is still less effective to interpret negated information. 3) Misunderstanding the text. The GPT-3 does not always understand the meaning of the sentence correctly. For example: “He claims women who are housewives should be paid, so he supports the idea of WOMEN BEING PAID LESS THAN MEN” Here, the predicted topic is related but not the main subject of the sentence. Such a mistake is rare but still exists weak supervision.

For MASK-Text, even though GPT-3 can mostly provide a sentence that is related to the topic and align with the correct stance, more than 50% of the time the content is very short and less informative compared to the texts from the datasets. For example: “His attitude towards middle east oilis oppositionbecause he thinks IT IS A WASTE” vs “His attitude towards miss America is supportbecause he thinks SHE IS TALENTED”. This is not that surprising since GPT-3 was trained to mainly satisfy the language modeling criterion; thus, it would be “lazy” to return with a solid and long response.

These MASK-Text instances are never wrong in the judgment of attitudes, so they can still give the model some help, although limited, in determining the attitudes.

Regarding the *Error analysis of our system*. Due to space limitation, we summarize two common error patterns made by our system. 1) Failed to connect the topic and text. The text often mentions the topic with distinct expressions and contains its stance implicitly. Therefore, it brings more difficulty to the model to successfully locate the topic and identify the stance. For example: “Topic: musician; Text: Spotify and Pandora pay usage rates that are much lower than the radio, records and legal downloads that they are replacing. Low enough to where many potential new artists won’t be able to even earn a living. There must be some alternative other than artists simply being forced to accept the new streaming model that destroys royalties. For example, who set streaming royalty rates? Can artists unionize and negotiate collectively with the streaming services? If we don’t sort this out, we will lose a new generation of artists – which is bad for everyone; Gold label: support; Predicted label: neutral”; 2) Incorrect ground-truth labels. The gold labels are not always correct. Sometimes the model makes a more appropriate judgement than the data provides. For example: “Topic: keep weight; Text: “All the medical evidence points to the fact that it’s nearly impossible to keep off weight once lost. The body just won’t let you.” This is incorrect, and could lead to fatalism that could harm people who are overweight. For example, I lost 70 pounds. That was at least a year ago. It has not come back. It is easy to keep off.”; Gold label: neutral; Predicted label: support”.

### **3.5 Conclusion**

In this work, we define OpenStance, a more realistic and challenging zero-shot stance detection problem in an open world. Under such a setting, multiple domains and

numerous topics can be involved, while no topic-specific annotations are required. To solve this problem, we proposed to combine indirect supervision from textual entailment and weak supervision collected from GPT-3. Our system, without the help of any task-specific supervision, outperforms the supervised method on three benchmark datasets that cover various domains and free-form topics.

## CHAPTER 4

### **X-SHOT: A UNIFIED SYSTEM TO HANDLE FREQUENT, FEW-SHOT AND ZERO-SHOT LEARNING SIMULTANEOUSLY IN CLASSIFICATION**

#### **4.1 Introduction**

For classification problems, the distribution of label occurrences in real-world scenarios often varies widely, with some labels appearing frequently (frequent-shot), others infrequently (few-shot), and some not at all (zero-shot). Given this variability, it becomes imperative to craft learning systems adept at managing labels across the full frequency spectrum. Regrettably, current few-shot systems often fall short when confronted with zero-shot challenges (Zhang et al., 2022; Cui et al., 2022; Zhao et al., 2021). In contrast, zero-shot systems, while adept in their domain, cannot fully benefit from the potential advantages of annotations when available (Zhang et al., 2019; Obamuyide & Vlachos, 2018; Yin et al., 2019; Xu et al., 2022). Thus, developing the skill to manage all possible label occurrences simultaneously is crucial for systems that are intended for practical use.

In this work, we introduce a more challenging and practically useful task:  $X$ -Shot learning. This task mirrors real-world environments where label occurrence spans a continuum, seamlessly incorporating frequent-shot, few-shot, and zero-shot instances, all without a priori constraints. In this paradigm, variable  $X$ , the number of times each label is seen during the training, is unbounded, ranging freely within the interval  $[0, +\infty)$ . At the heart of  $X$ -Shot lies the objective of attaining open-domain generalization and architecting a system resilient across a plethora of label scenarios.

Tackling  $X$ -Shot spawns two core technical conundrums: ( $Q_1$ ) How can one identify suitable sources of Indirect Supervision (Yin et al., 2023) in few-shot and zero-shot settings, given the notable scarcity of annotations. ( $Q_2$ ) Traditional multi-class classifiers struggle with the diversity in label sizes across tasks, frequently requiring customized classification heads for each variation. Here, the challenge is formulating a cohesive system capable of effectively adapting to labels of diverse sizes.

To address  $Q_1$ , we identify the most effective source of Indirect Supervision as being from Instruction Tuning datasets, such as Super-NaturalInstruction (Wang et al., 2022). These datasets primarily contain various NLP tasks enriched with textual instructions. Our method trains the model on these datasets, aiming for robust generalization to the unseen  $X$ -Shot task when supplemented with pertinent instructions, especially for the low-shot (few-shot and zero-shot) labels. For  $Q_2$ , we advocate a triplet-oriented binary classifier. This classifier functions by accepting a triplet of (instruction, input, label), anticipating a binary response (“Yes” or “No”) that confirms the suitability of the label for the specified input under the given instruction. Such a triplet-oriented classifier acts as a cohesive architecture that manages text classification tasks with labels of varied sizes. By combining solutions for both  $Q_1$  and  $Q_2$ , we forge a holistic framework, BinBin (**b**inary **i**nference **b**ased on **i**nstruction following).

There are, however, no existing datasets that explicitly cater to this challenge. To evaluate our system, we turn to three representative classification tasks: relation classification, event detection, and argument role identification. We recompile their associated datasets: *FewRel* (Han et al., 2018), *MAVEN* (Wang et al., 2020), and *RAMS* (Ebner et al., 2020) and simultaneously include frequent-shot, few-shot, and zero-shot

instances. Sourced from diverse domains (Wikipedia, news articles, etc.), and featuring vast label counts (ranging from 30 to 78), these datasets pose a formidable challenge to contemporary text classification systems. Moreover, the *MAVEN* dataset uniquely integrates a “None” label, further amplifying the realistic nature of the task. Experiments on multiple model scales and architectures reveal our system’s resilience across datasets, consistently outperforming leading baselines, including GPT-3.5.

Our contributions can be summarized as follows: (i) We introduce *X*-Shot, a hitherto under-explored, open-domain open-shot text classification problem that mirrors real-world complexities. (ii) We innovate a unique problem setting that reframes any text classification challenge into a binary classification task, adaptable to any number of label sizes and occurrences. (iii) Our BinBin, harnessing the potential of instruction-following datasets, excels past existing approaches, demonstrating versatility across various domains, label magnitudes, and classification paradigms.

## 4.2 Problem Statement

Each *X*-Shot target task has the following components:

- **Input  $t$ :** Versatile text in varied forms, lengths, and domains.
- **Label space  $L$ :**  $L$  contains arbitrary size of labels:  $\{\dots, l_i, \dots\}$  and an optional *None* label (i.e., all labels in  $L$  are incorrect for the input). Within  $L$ , each label can be either zero-shot, few-shot, or more frequent.

The task of *X*-Shot is to figure out label  $L_s \in L$  that is correct for the input  $t$  in the target task, where  $|L_s|$  might be zero (i.e., “None”).

*Research questions of X-Shot:* i) Given that the above formulation encompasses various text classification problems, how can we move away from constructing individual

models for each problem, and instead develop a single classifier adept at handling diverse classification settings? ii) Beyond frequently-encountered labels, low-shot labels necessitate additional supervision for effective reasoning. Where can we find such supervision? In the following

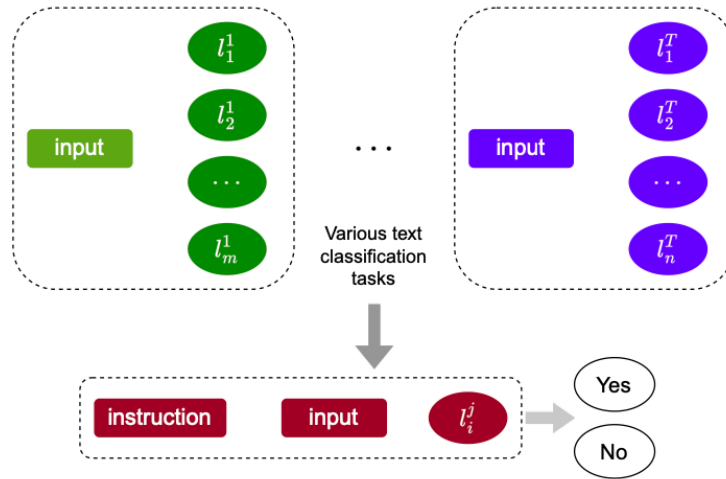


Figure 4.1. Unified classification problems in BinBin.

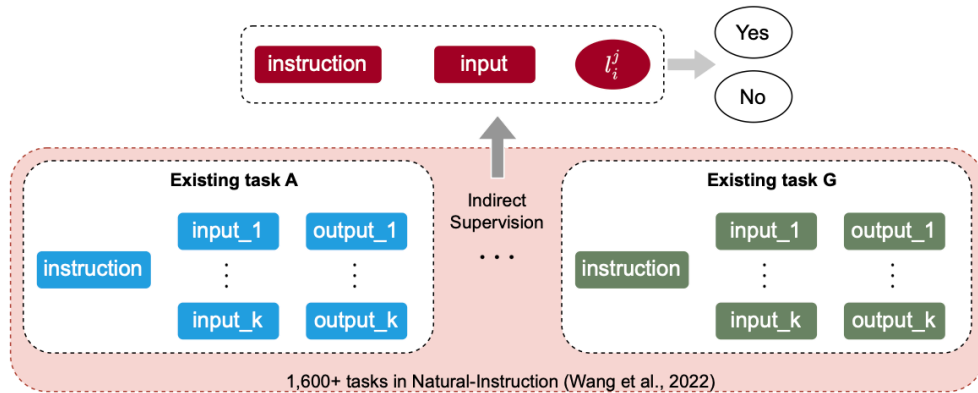


Figure 4.2. Indirect Supervision for BinBin.

section, we delve deeper into our approach concerning the universal system and the provided supervisions.

## 4.3 Methodology

This section first explains how BinBin adapts to different classification problems, then introduces the supervision to train it.

### 4.3.1 *BinBin Architecture*

We have devised a broad approach that converts any classification task into a unified, instruction-driven binary classification formation. As depicted in Figure 4.1, for any text classification task with its set of inputs and labels, we write a short introduction and model it as (instruction, input, label) triplet. The task then becomes determining if the label is appropriate (“Yes”) or not (“No”) given the input under the instruction. This transformation effectively alleviates the frequency gap of the target labels. An example of the conversion can be found in Appendix A.1.2.

BinBin can support classification tasks with any number of class labels. Instead of mapping labels into numerical representations as traditional supervised classifiers do, we retain the actual label names. To pave the way to tackle a variety of low-shot text classification tasks using an instruction-guided approach, two primary challenges arise: i) Ensuring that the model comprehends the instructions, and ii) guiding the model to identify seldom seen or entirely new labels. We will delve deeper into our supervision approaches to address these challenges in the following subsections.

### 4.3.2 *Supervision Acquisition For Low-shot Labels*

*X-Shot* relies on Indirect Supervision and Weak Supervision. We will explain them in this subsection.

*Indirect Supervision.* Previous best-performing systems for low-shot text classification have primarily relied on Indirect Supervision *from a single source task*.

Examples of these source tasks include natural language inference (Yin et al., 2019), summarization (Lu et al., 2022) and passage retrieval (Xu et al., 2023a). This approach presents three main drawbacks: i) the usable supervision from the single source task is limited, and there’s often a domain mismatch between the source task and the target classification tasks; ii) typically, instances of the target problems need to be reformatted into forms of source tasks to enable zero-shot generalization—a process that’s frequently complex; iii) there is not a universally adaptable system to address the  $X$ -Shot learning, where labels might vary in their occurrences.

In this work, we leverage Indirect Supervision from an extensive assortment of NLP tasks. The Super-NaturalInstruction dataset (Wang et al., 2022) encompasses over 1,600 tasks across 76 categories. Each task is accompanied by instructions and numerous input-output instances (an example of tasks is in Appendix A.1.1). This dataset offers an invaluable source of Indirect Supervision for our target  $X$ -Shot. As in Appendix A.1.1, for every task within the Super-NaturalInstruction dataset, we are presented with the associated instruction as well as the input and the ground truth answer. For each instance selected, we will randomly pick one answer that is different from the ground truth answer within the task, whether the task is generation or classification. As a result, we obtain one positive triplet (instruction, input, ground truth) and one negative triplet (instruction, input, random answer) for each instance in our training dataset as in Figure 4.2. Our Indirect Supervision stems from this dataset training. Such training further significantly mitigates the incongruity exist in varying label frequencies.

When evaluated on target classification tasks, we convert every sample into a triplet-oriented binary instance similarly to the transformation for Super-

NaturalInstruction, complemented by a human-written instruction. Given an original instance with text  $t$  and positive label  $l$ , we add an instruction and craft  $|L|$  triplets as [(instruction,  $t$ ,  $l$ ), Yes/No] for each label  $l$  from the label space  $L$ , with the gold label as positive and others as negative.

Through this Indirect Supervision, minor alterations—be it a word or a few words—can change the class completely. By enabling the model to distinguish the positive and negative classes from marginally changed inputs, we hope the model establishes more distinct decision boundaries.

*Weak Supervision for zero-shot labels.* In addition to Indirect Supervision, we aim to specifically enhance our model’s performance on zero-shot labels. Given that we cannot procure annotated instances for these labels, how can we enhance the model’s understanding of zero-shot labels without human intervention or labeling? This is where we leverage the capabilities of GPT-3.5 (Brown et al., 2020b) to produce weakly labeled instances. To generate instances for zero-shot labels, we employ in-context learning by randomly selecting demonstrations from few-shot or frequently labeled data. Here’s a prompt from the *Maven* event detection dataset, aimed at producing text and event triggers for zero-shot event types:

```
event type: Competition
event trigger: tournament
sentence: The final tournament was Played in two stages: the group stage and
the knockout stage.

event type: Motion
event trigger: throwing
sentence: Simultaneously, Sayhood gained a lock on Rodriguez, throwing him
onto the defensive.

event type: Manufacturing
```

Figure 4.3. Weak Supervision template for zero-shot labels.

By exposing GPT-3.5 to event and event statement examples associated with the event type labels “Competition” and “Motion”, we introduce the zero-shot label “Manufacturing.” Subsequently, GPT-3.5 generates an event trigger along with an event statement, serving as a weakly supervised instance for this label.

*Model selection.* In the main results, we adopt the pre-trained RoBERTa-large model (355M parameters) (Liu et al., 2019b) as our backbone model, given its reliability and high efficiency. However, BinBin can also be extended to different model scales and architectures, such as T5 and GPTs. More results can be found in Section 5.3 Analyses.

*Training strategy.* We first train the backbone model (Liu et al., 2019b) on the transformed binary Super-NaturalInstruction dataset, then fine-tune on the converted triplet instances of downstream  $X$ -Shot tasks. The same backbone model will be used in all experiments and baselines.

## 4.4 Experiments

### 4.4.1 Experimental Setting

Regarding the dataset. In this work, we standardize challenging datasets that can cover (i) multiple domains, (ii) various sizes of class labels, and (iii) out-of-domain label scenarios. Therefore, we select: *FewRel* (Han et al., 2018), *MAVEN* (Wang et al., 2020), and *RAMS* (Ebner et al., 2020), referring to relation classification, event detection, and argument role identification problems respectively. We converted each data set into a format appropriate for BinBin. Few/zero/freq-shot labels are evenly distributed in all three datasets to avoid bias on any group when reporting the overall performance. Details of label distribution can be seen in Table 5.1. We rename each resulting dataset as “[ ]<sub>X-Shot</sub>.”

The first one is  $FewRel_{X-Shot}$  :  $FewRel$  is a well-established relation classification dataset where each instance provides a relation statement, two entities from the statement, and their corresponding relation label. Since the test set of  $FewRel$  is not available, we include 78 relations from its train and dev and divide them into 26/26/26 as freq/few/zero-shot labels. We randomly select 500/5/0 instances from each freq/few/zero label in the new *train*, and 200 instances from each label in the new *dev* and *test*.

The second one is  $MAVEN_{X-Shot}$  : As an event detection dataset, the event detection task in  $MAVEN$  includes two steps: detecting the event trigger and predicting the event label from the trigger. In this work, we will focus on the second step, where we assume the event trigger is known and aim to predict the corresponding event label. To make  $MAVEN$  align with our setting, we reorganize its *train* and *dev* sets as follows: since the event label distribution is significantly imbalanced, we select 69 of them who have 400+ instances plus the “None” label as our label set. Labels are divided into 23/23/23+1 as freq/few/zero-shot labels with “None” belonging to the zero-shot group. We select 300/5/0 instances from each freq/few/zero label in the new *train*, and 100 instances from each label in the new *dev* and *test*.

Table 4.1. Statistics of dataset labels

	domain	#freq	#few	#zero
$FewRel_{X-Shot}$	Wikipedia	26	26	26
$MAVEN_{X-Shot}$	Wikipedia	23	23	23+1
$RAMS_{X-Shot}$	News articles	10	10	10

The third one is  $RAMS_{X-Shot}$  :  $RAMS$  tackles the task of identifying semantic role labels given the sentence marked with event triggers and argument terms. There are 30 labels that have more than 100 instances; we split them into 10/10/10 for each label group. Similarly,

we select 300/5/0 instances from each freq/few/zero label in the new *train*, and 50 instances from each label in the new *dev* and *test*.

It’s noteworthy that while these datasets may not be the largest in scale, they introduce complex NLP challenges that are non-trivial for the latest LLMs. This complexity arises from the need for advanced reasoning and dealing with extensive label spaces.

Four typical baselines are included:

- ***Multi-way classification (MWC, (Soares et al., 2019))*** . This methodology is the prior SOTA approach for relation classification which designs a special marker for entity terms. We employ this strategy for all three datasets, given that they all contain term features (entity, event trigger, argument, etc.) similar to relation classification.
- ***In-context learning with GPT-3.5 (GPT-3.5)***. We create a prompt that includes three demonstrations, two positive and one negative, and each comes with the input, label, and a True/False label that indicates whether the prediction is correct. The specific process can be seen in Appendix A.1.3.
- ***Indirect Supervision from Text Entailment (NLI; Li et al. 2022a)*** . NLI is the prior SOTA approach for addressing a zero-shot or few-shot classification with Indirect Supervision from merely the NLI source task. This paradigm uses the input text as the premise and transforms the label into a hypothesis sentence.
- ***Prototypical Prompt learning (PPL; Cui et al. 2022)*** PPL is the prior SOTA system for few- shot classification leveraging prompt learning and Contrastive Learning. For each of the dataset, we select 500 instances per label during

training for prototype learning. For freq and few shot labels, we keep selecting instances from the available instances until we reach the number. For zero-shot labels, we simply put the label itself as the text for the training since we have no instances available.

We elaborate on our implementation details at different stages here.

- ***Indirect Supervision*** . Consistent with the original experimental setup and train/test split (Wang et al., 2022), we select 100 random instances from each task when compiling the Indirect Supervision dataset from Super-NaturalInstruction. Our prefix template follows the previous benchmark strategy, incorporating only the instruction and two positive examples—provided this inclusion doesn’t surpass the word limit. When adjusting target classification tasks to fit BinBin, we draft three distinct instruction prompts and present the average outcomes to demonstrate the system’s stability. All templates are available in Appendix A.1.4.
- ***Weak supervision.*** We use the “text-davinci-003” GPT-3.5 completion model to augment zero-shot instances. For each zero-shot label, we generate 5 instances to serve as Weak Supervision. We attempted to generate 10 or more instances per label but did not observe a notable improvement. We suspect this is due to the limited diversity the GPT-3.5 model can provide, making the benefit of additional samples marginal.
- ***Prediction threshold.*** In the NLI baseline and our method, each instance is converted into  $|L|$  Yes/No instances, one for each label. We compare the probability of the positive class to assign labels. For *FewRel* and *RAMS*, the label with the highest score is chosen. In *MAVEN*, we introduce a threshold parameter,  $t$ . If the label receiving the highest

probability does not exceed this probability threshold, we assign the label as “None”. We experiment with various values of  $t$ , ranging from 0.5 to 1, and select the optimal one based on *dev*.

Table 4.2. Main results on three benchmark target tasks

Models	FewRel <sub><math>X</math>-Shot</sub>				RAMS <sub><math>X</math>-Shot</sub>				MAVEN <sub><math>X</math>-Shot</sub>			
	all	freq	few	zero	all	freq	few	zero	all	freq	few	zero
MWC	49.82	94.23	55.23	0.0	34.47	<b>78.40</b>	25.00	0.0	42.43	85.17	43.96	0.0
NLI	63.46	<b>95.35</b>	48.81	46.22	43.07	71.40	20.40	37.40	56.31	<b>85.65</b>	39.83	44.00
PPL	53.23	95.15	<b>63.54</b>	0.0	27.13	65.00	16.20	0.20	46.84	85.04	<b>55.52</b>	0.0
GPT-3.5	18.24	18.22	25.33	11.17	18.19	21.21	15.15	18.19	21.43	15.15	12.12	37.50
BinBin	<b>68.48</b>	94.06	58.04	<b>53.34</b>	<b>54.70</b>	77.00	<b>29.00</b>	<b>58.07</b>	<b>64.96</b>	84.32	46.64	<b>63.97</b>

#### 4.4.2 Results

Table 4.2 compares BinBin system with baselines. The “freq”, “few”, and “zero” columns refer to the accuracy of freq-shot, few-shot, and zero-shot labels respectively. Our model consistently outperforms all baselines by a large margin in the “all” and “zero” dimensions, while occasionally showing slightly lower but on-par performance with the baselines in “freq” and “few”. Analyzing these baselines, we notice that most are ill-suited for the  $X$ -Shot problem setting, particularly in zero-shot scenarios where annotations are absent. MWC is influenced by the number of label-wise training instances; therefore, its performance, although pretty high for “freq”, drops quickly to be 0.0 for “zero”. Similarly, the few-shot prompting (PPL) baseline does well for “few” but encounters difficulties with unseen class instances, underscoring the limitations of classification models in the  $X$ -Shot context. NLI, representing the SOTA in low-shot learning settings, is the only model adept at managing all three types of labels. Nonetheless, when compared with BinBin, NLI’s accuracy remains lower in few-shot and zero-shot situations. This indicates that, despite its

competency in handling low-shot labels, NLI’s capacity for exploiting limited supervision is inferior to our system.

As one of the most advanced closed-source LLMs, GPT-3.5 shows limited effectiveness in this task, with its performance across three label sets appearing strikingly similar. Although GPT-like models demonstrate robust capabilities in in-context learning, they fall short in utilizing rich annotations when available and often struggle in scenarios with a large label space. This highlights the flexibility of our BinBin in handling classification labels of different sizes and occurrences.

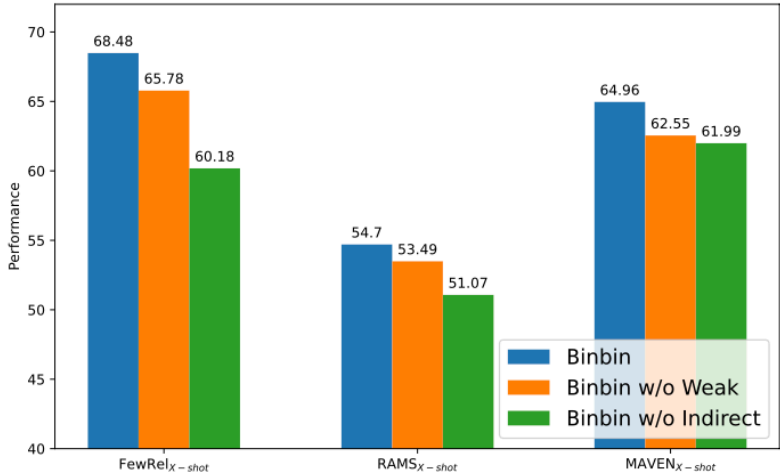


Figure 4.4. Ablation study of BinBin.

### 4.4.3 Analyses

In addition to reporting the main results, we further analyze our system in the following dimensions: ( $Q_1$ ) the individual contribution of Indirect Supervision and Weak Supervision; ( $Q_2$ ) is BinBin adaptive to other model scales and architectures? ( $Q_3$ ) why does “zero” show better performance than “few” in RAMS<sub>x-Shot</sub> and MAVEN<sub>x-Shot</sub>? ( $Q_4$ ) Given that our Indirect Supervision is derived from a diverse range of NLP tasks in Natural-Instruction Wang et al. (2022), is there a possibility of task leakage? ( $Q_5$ ) When selecting

source tasks for Indirect Supervision in instruction-following, which configuration is more effective: having more (diverse) tasks or having more (task-wise) instances? ( $Q_6$ ) The efficiency of our system. ( $Q_7$ ) The mistakes our system makes.

( $Q_1$ ) *Ablation study.* Figure 4.3 depicts the ablation study, where either Indirect Supervision or Weak Supervision is discarded from our system BinBin. Our findings reveal that both supervision sources fulfill complementary roles in the  $X$ -Shot task. Encouragingly, while their combined usage yields the best results, each type of supervision, on its own, still surpasses the baselines. Such a result clearly underscores the high efficiency and effectiveness of our innovative system.

( $Q_2$ ) *How does BinBin adapt to other model scales and architectures.* Even though we use RoBERTa as our backbone model, BinBin can be adapted to any popular pretrained language model architectures. Besides our main results with RoBERTa-large, an encoder-only transformer with 355M parameters, we also integrate our system into T5-3b (Raffel et al., 2020) and GPT-Neo 1.3B (Black et al., 2021), which are representative models for encoder- decoder and decoder-only transformers, respectively. For RoBERTa, we use the [CLS] token for classification. Similarly, for T5, we only adopt the encoder part and feed the first token into the classification head. For GPT-Neo, since it is a decoder-only model designed for generation tasks, we adopt the last token and add a classification head on top, as other casual models do. The results are in Figure 4.4. Given the larger parameter size, it is not surprising to see T5-3B outperform RoBERTa across all three datasets. However, GPT-Neo 1.3B consistently underperforms compared to RoBERTa, despite having a similar large parameter size. Considering that both RoBERTa and T5 provide encoder token

representations for classification heads, we conclude that decoder-only architectures, such as GPT-Neo, are not as effective in sequence classification.

(Q<sub>3</sub>) *Why do zero-shot labels outperform few-shot labels in the MAVEN<sub>X-Shot</sub> and RAMS<sub>X-Shot</sub> benchmarks?* We observe that this phenomenon applies not only to our system, but also to baselines “NLI” and “GPT-3.5”. We suspect two reasons: i) Some zero-shot labels in RAMS<sub>X-Shot</sub> seem easier upon visual inspection; ii) In MAVEN<sub>X-Shot</sub>, “None” is treated as a zero-shot label in the test set, contributing notably due to threshold tuning.

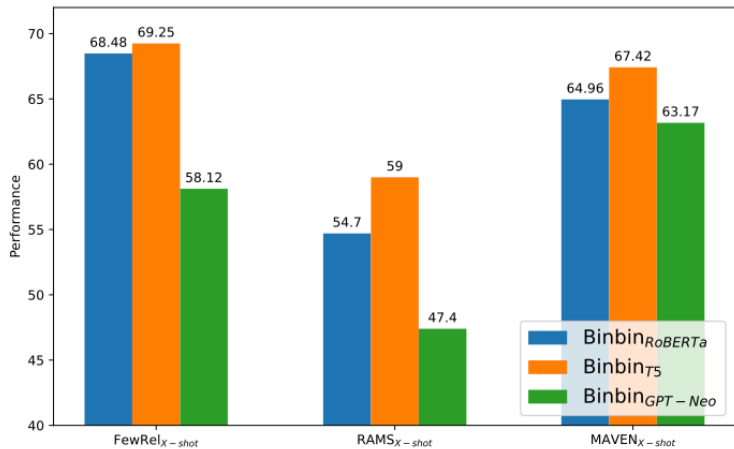


Figure 4.5. Backbone models with different scales and architectures.

Table 4.3. Ablation study: deleting top-10 similar tasks

	all	freq	few	zero
FewRel <sub>X-Shot</sub>	63.34	89.04	<b>60.95</b>	40.04
RAMS <sub>X-Shot</sub>	51.64	<b>78.74</b>	<b>30.13</b>	40.07
MAVEN <sub>X-Shot</sub>	63.83	<b>85.68</b>	<b>47.48</b>	58.57

(Q<sub>4</sub>) *Influence of Task Type Overlap.* Although the Natural-Instruction task repository doesn’t directly contain our target datasets, we remove the top 10 tasks closest to each target dataset to assess the impact of similar tasks. The measurement is based on cosine similarity between Sentence-BERT Reimers & Gurevych (2019b) embeddings of the task definitions in the Natural-Instruction dataset and each X-Shot target dataset’s instruction.

From Table 4.3, we can observe that: i) The main decreases when the top-10 similar tasks are deleted happen to zero-shot labels. Recall that we only provided Weak Supervision for them; this phenomenon indicates that pretraining on similar source tasks can help diminish the impact of noise in the weakly supervised data. ii) Despite slight decreases in “all”, our results still surpass baselines in Table 4.2, underscoring the value of diverse training tasks. This is further supported by subsequent analysis.

(Q<sub>5</sub>) *Number of Tasks vs Number of Instances.* Balancing the number of tasks and the number of instances per task is pivotal in curating instruction-following datasets Lou et al. (2023).

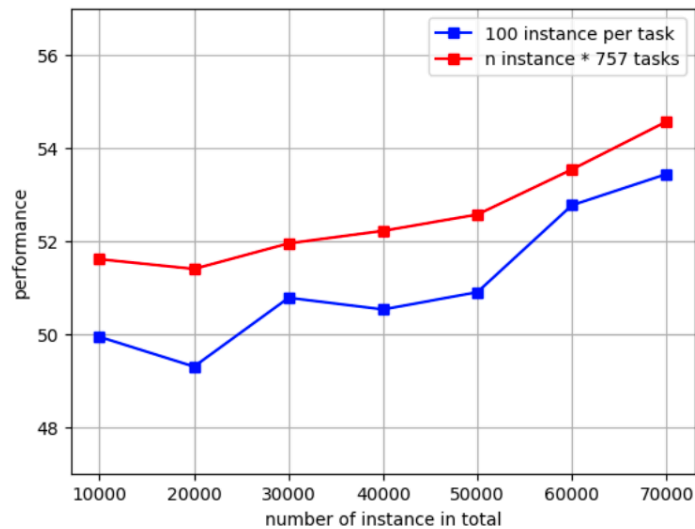


Figure 4.6. Ablation study: #instances vs. #tasks.

We wonder, by keeping the total instance count constant, should we have more tasks or more instances per task? We try [100,200,...,700] for the varying number of tasks, each with 100 instances. In total, we have [10,000, 20,000, ... 70,000] instances. Accordingly, for the varying number of instances per task, we have datasets with [10,000/757, 20,000/757, ... 70,000/757] number of instances. The overall instances remain

the same in each step. From Figure 4.5, it’s evident that both task count and instance count boost performance. While increasing either is beneficial, having more (diverse) tasks has a greater impact than adding more instances to each task. Given these insights, future work should focus on diversifying the types of tasks exposed to the model, considering data constraints.

*(Q<sub>6</sub>) Efficiency Analysis.* Efficiency concerns center around the inference stage, where our system converts varied-label classification problems into a binary inference task. This step of BinBin aligns with the NLI baseline, the previous SOTA method for low-shot learning. The training in our system takes more time due to pretraining on Natural-Instruction, but during testing, both systems are equally efficient as they make binary decisions for each label. More importantly, using a unified system like BinBin, instead of separate systems for different label groups, actually reduces overall training time and computational effort. More details in terms of training time and computational resources are in Appendix A.1.5

*(Q<sub>7</sub>) Error Analysis.* We collect the most typical errors as follows:

- **Multiple labels make sense** In datasets with many labels, multiple labels can fit a context, with the model’s interpretation sometimes more accurate than the original data. Consider the instance from *RAMS* dataset: “Many high-ranking figures in companies tied to Skolkovo have also donated to the Clinton Foundation” While the ground truth label for the argument “Clinton Foundation” is “recipient”, the model strongly suggests “beneficiary”—a label that is equally justifiable.
- **Bias towards more frequent labels** Models often favor frequently encountered labels in cases of semantic overlap among multiple labels. For example,

consider a sentence from the *FewRel* dataset: “The Spanish Andorran border runs 64 km between the south of Andorra and northern Spain ( by the autonomous community of Catalonia ) in the Pyrenees Mountains.”. Here, the entities are “Catalonia” and “autonomous community”. Although the gold relation for the two entities is “instance of”, the model assigns the highest probability to “part of”—a frequent group label. This suggests that not only does the label share semantic similarities with others, but its frequent occurrence also biases the prediction, especially when many labels lead to potential confusion.

- **identifying reciprocal or inverse relationships** This issue arises when the model struggles to differentiate between roles that represent opposite positions in a given context, such as in a “receiver” and “giver” scenario while both roles are part of the same transaction, but the model confuses who is who. For instance, in a sentence from *RAMS*: “She was shouting, ‘I am a terrorist,’ and reportedly threatened to blow herself up he couldn’t believe that the decapitated child ’s head being carried by the woman was real.” Where “she” is a “killer”. However, the model incorrectly labels “she” as a “victim”, demonstrating the difficulty in accurately discerning reciprocal roles.

## 4.5 Conclusion

This work introduces *X-Shot*, a text classification setting characterized by diverse label occurrences: freq-shot, few-shot, and zero-shot. Our approach, BinBin, leverages Indirect Supervision and LLMs’ Weak Supervision to consistently outperform state-of-the-art methods across three benchmark datasets in various domains.

**Original Instance:**

Sentence: "3D Friends ( stylized as 3D FRIENDS ) is an American indie rock band from Austin , Texas  
Entity 1: 3D Friends  
Entity 2: indie rock  
Relation: genre

**Unified Schema:**

Instruction Template

**Input:**  
**Definition:** Given a sentence about two entities, return a relation between the two entities that can be referred from the sentence.  
**Positive Example 1 -**  
 Sentence: *Mount Storer* ( ) is a jagged peak in the *Tula Mountains* , 4 nautical miles ( 7 km ) east - northeast of *Mount Harvey*.  
 Entity 1: *Mount Harvey*  
 Entity 2: *Tula Mountains*  
 Relation: mountain range  
**Positive Example 2 -**  
 Sentence: On the east side of the square stands the impressive mansion of *Dundas House*, built by *Sir William Chambers* for Sir Lawrence Dundas between 1772 and 1774  
 Entity 1: *Sir William Chambers*  
 Entity 2: *Dundas House*  
 Relation: notable work  
**Now complete the following example -**  
 Input: sentence: "3D Friends ( stylized as 3D FRIENDS ) is an American indie rock band from Austin , Texas  
 Entity 1: 3D Friends  
 Entity 2: indie rock  
 Relation: genre / company

Instance for Prediction

label: Yes No

Figure 4.7. Classification to binary BinBin.

**Definition**

In this task, you will be shown a short story with a beginning, two potential middles, and an ending. Your job is to choose the middle statement that makes the story coherent / plausible by writing "1" or "2" in the output. If both sentences are plausible, pick the one that makes most sense.

**Positive Examples**

**Input:** Beginning: John was on the trail running. Middle 1: John accelerated the speed and broke his leg accidentally. Middle 2: John was chased by a bear. Ending: He ran even faster until he got to his car safely.  
**Output:** 2  
**Explanation:** When someone breaks his/her leg, it is difficult to run. Therefore, we choose 2 in this case.

**Negative Examples**

**Input:** Beginning: Jon decided to steal a police car. Middle 1: Jon crashed the police car into a telephone poll. Middle 2: Jon wasn't caught. Ending: Jon went to prison for three years.  
**Output:** Jon crashed the police car into a telephone poll.  
**Explanation:** You should not answer with the chosen sentence. You should only answer with 1 or 2

**Instances**

**Input:** Beginning: Today I was cooking hamburgers inside. Middle 1: I burned my hand. Middle 2: I burned my feet. Ending: Now I have a blister.  
**Output:** 1  
 .....

Figure 4.8. Super-Naturalinstructions task example.

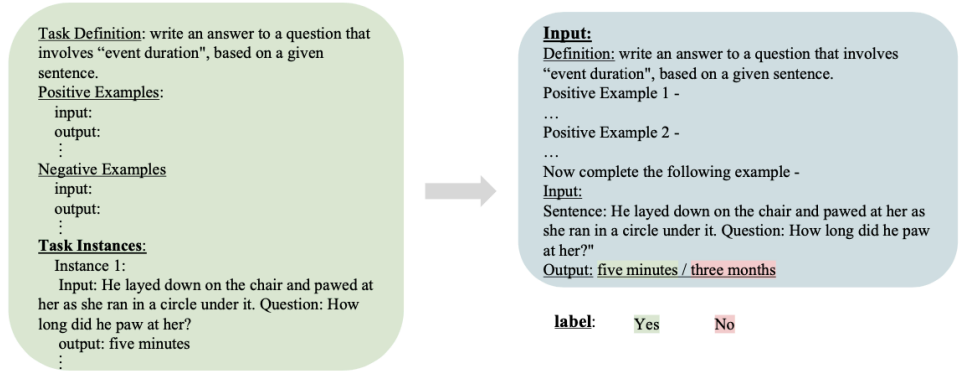


Figure 4.9. Super-Naturalinstructions to binary BinBin.

## CHAPTER 5

### LLMS' CLASSIFICATION PERFORMANCE IS OVERCLAIMED

#### 5.1 Introduction

While large language models (LLMs) often demonstrate strong performance in classification tasks where gold labels are included by default, they also reveal limitations when these gold labels are excluded, as they may still choose from incorrect options. To address this, let's begin with the example in Figure 5.1, which illustrates the use of the latest LLMs for a straightforward classification problem.

In this simple scenario, GPT-4o, the latest large language model (LLM), performs comparably to humans when the correct answer is included among the options. However, interestingly, even this advanced LLM does not show uncertainty by indicating “no correct answer” or “all options seem incorrect”, a behavior consistently demonstrated by humans when the correct answer is not provided.

Why should we be concerned about this particular phenomenon, especially in the era of LLMs? There are two primary reasons: i) Given the versatility of LLMs, they can process inputs with any set of labels by following natural language instructions, even when the correctness of the labels is unknown. The expected behavior from LLMs should mirror that of humans in the previous example: identifying correct labels when present or indicating the absence of correct ones without risking users accepting false responses. In contrast, traditional classifiers, trained on fixed label sets, are limited to predicting within those specific labels and lack the flexibility to handle open sets of labels. ii) LLMs are predominantly designed as generative models, prioritizing the enhancement of their generative capabilities, often at the expense of discriminative capabilities Sun et al. (2023b).

Many researchers argue that classification tasks are perceived as easy for LLMs, as evidenced by their consistently high performance Sun et al. (2023b,a); Zhang et al. (2024). However, the above example raises the question of whether the performance of LLMs in classification tasks has been overstated due to current evaluation benchmarks and metrics only capturing incomplete human behavior.

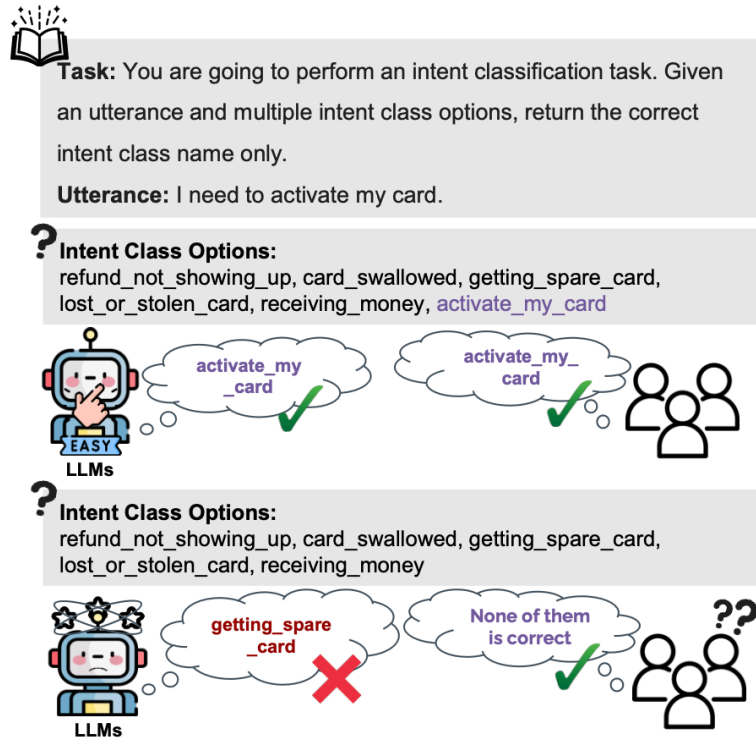


Figure 5.1. Comparison between human and latest LLMs<sup>1</sup>.

To investigate this question, we present three standard classification tasks as benchmarks: BANK-77 (intent classification task, Casanueva et al., 2020) , MC-TEST (multiple-choice question answering task, Richardson et al., 2013) , and EQUINFER, a newly assembled task where the objective is to infer the correct equation from four

<sup>1</sup> Latest LLMs (GPT-4o, claude-3-opus, and gemini-1.5-pro as of July, 2024) vs. Human when the gold label is present or absent. API access with temperature set as 0

candidates given surrounding paragraphs in scientific papers. This benchmark, termed KNOW-NO, encompasses classification tasks with inputs of varying lengths, label sizes, and label scopes, including instance-level and task-level label spaces.

We define a novel evaluation metric, OMNIACCURACY, designed to accurately assess the human-level discrimination intelligence of LLMs in classification tasks. This metric integrates the performance of LLMs across two dimensions within the KNOW-NO framework:

i) ACCURACY-W/-GOLD: representing the conventional accuracy when the correct label is provided.

ii) ii) ACCURACY-W/O-GOLD: indicating the accuracy when the correct label is not provided. We argue that OMNIACCURACY offers a more comprehensive reflection of LLMs' classification performance.

In summary, our contributions can be outlined as follows:

- To the best of our knowledge, this is the first study to uncover the limitations of LLMs in classification tasks when gold labels are absent. We designate this task as CLASSIFY-W/O-GOLD and propose it as a novel evaluation framework for LLMs.
- We introduce a benchmark, KNOW-NO, which encompasses two established classification tasks alongside a newly devised one, aimed at evaluating CLASSIFY-W/O-GOLD.
- This study introduces and proposes a new evaluation metric, OMNIACCURACY, tailored to assess LLMs in classification tasks. This metric combines performance metrics when gold labels are both present and absent, offering a more comprehensive evaluation of LLMs' capabilities.

Table 5.1. Statistics of KNOW-NO.

	#input	input	label format	#label	label scope	challenge
BANK-77	1,000	12	phrase	77	task-level	moderate
MC-TEST	1,000	220	phrase&sent	4	instance-level	low
EQUINFER	1,049	1,925	latex equation	4	instance-level	high

## 5.2 Approach

### 5.2.1 *KNOW-NO Benchmark*

In this work, we collect representative classification datasets to cover (i) multiple task types and difficulty levels, and (ii) various label sizes and label scopes (task-level label space or instance-level label space). Specifically, we build this benchmark “KNOW-NO” with two existing datasets BANK-77 (Casanueva et al., 2020), MC-TEST (Richardson et al., 2013), along with a new dataset proposed by us, named EQUINFER. An overview of the KNOW-NO statistics is provided in Table 5.1. We will first introduce BANK-77 and MC-TEST, followed by a detailed explanation of the construction process for EQUINFER.

**5.2.1.1 BANK-77** Inputs are simple sentences (customer service queries) in the banking and financial domain, all sharing the same label space of 77 intents (“task-level label”). The original dataset comprises 13,083 inputs; however, due to budget limitations, we randomly selected 1,000 instances for this study, ensuring coverage of all 77 labels. We chose this dataset because of its moderate difficulty level and large label size.

**5.2.1.2 MC-TEST** MC-TEST is a pioneering multiple-choice reading comprehension benchmark. It includes 660 elementary-school-level stories, each accompanied by 4 multiple-choice questions with 4 unique answer options for each question (“instance-level” label). We randomly selected 250 stories (1000 questions) as our test set. We chose MC-TEST because of its simplicity (most questions can be answered by

keyword matching rather than deep reasoning), which may help us uncover surprising behaviors of the latest LLMs.

**Context before:**  
 “[.....] Therefore, the positive sample is the corresponding augmented sentence, while the negative samples are the augmented versions of other original source sentences from the same mini-batch.  $e_{x^i}$  and  $e_{z^i}$  are the average representations along the sequence dimension from the encoder outputs. Apart from the contrastive loss, the standard cross-entropy loss is calculated as:”

**Context after:**  
 “We combine both losses as the final loss:

$$\mathcal{L} = \mathcal{L}_{ce} + \lambda \mathcal{L}_{ctr}$$

where  $\lambda$  is an interpolation factor. We incorporate the augmented source inputs  $z$  to ensure that the model can still generate correct translations with noisy input. [.....]”

**Equation options:**  
 A:  $\mathcal{L}_{ce} = -\sum_{i=1}^N \sum_{c=1}^C y_{i,c} \log(p_{i,c})$     B:  $\mathcal{L}_{ce} = -\sum_{i=1}^N (\log P_{\theta}(y^i|x^i) + \log P_{\theta}(y^i|z^i))$   
 C:  $\mathcal{L}_{ce} = -\frac{1}{N} \sum_{i=1}^N \sum_{c=1}^C y_c^i \log(p_c^i)$     D:  $\mathcal{L}_{ce} = -\sum_{i=1}^N \sum_{k=1}^C y_k^i \log \hat{y}_k^i$

Figure 5.2. Example of EQUINFER, the equation labeled as “B” is correct.

Table 5.2. Prompting LLMs in KNOW-NO

w/ $\mathcal{G}$	Instruction: For the following input and options, please return the correct option(s). Input: ... Options: $ID_1$ . [option <sub>1</sub> ]; $ID_2$ . [option <sub>2</sub> ]; ...; $ID_G$ . [option <sub>G</sub> ]; ...
hint as option.	Instruction: For the following input and options, please return the correct option(s). Input: ... Options: $ID_1$ . [option <sub>1</sub> ]; $ID_2$ . [option <sub>2</sub> ]; ... $ID_n$ . none-of-them
w/o $\mathcal{G}$	Instruction: For the following input and options, please return the correct option(s), or return “none-of-them” if you believe none of the options is correct. Input: ... Options: $ID_1$ . [option <sub>1</sub> ]; $ID_2$ . [option <sub>2</sub> ]; ...
no hint	Instruction: For the following input and options, please return the correct option(s). Input: ... Options: $ID_1$ . [option <sub>1</sub> ]; $ID_2$ . [option <sub>2</sub> ]; ...

### 5.2.1.3 EQUINFER We design this task to mimic the paper reviewing process,

where reviewers must determine if an equation is valid based on its context. This task requires intensive domain expertise.

**5.2.1.3.1 Data Crawling.** We crawled a total of 4,951 papers’ LaTeX source packages from ArXiv, focusing on papers accepted by top-tier NLP conferences.<sup>2</sup> We excluded papers that were unsuitable for this task, including 1) papers without any LaTeX equations and 2) papers with overly complicated equations (e.g., equations with nested structures or custom commands). This filtering process resulted in 1,449 papers. From each paper, we randomly sampled up to 3 equations, leading to a final set of 3,877 equations.

**5.2.1.3.2 Task Formulation.** We formulate this task as a multiple-choice classification, where each instance includes N-word context before the equation and N words after, all in the original LaTeX format. To keep an optimal context length for both sides of the equation, we tested ten different context lengths ranging from 100 to 1500 on GPT-4. The results are displayed in Figure A.1, Appendix A.2.1. We found that starting from 1000 words, the model’s performance did not show significant improvement. Therefore, we decided to retain 1000 words for either side. Additionally, when truncating the context, we ensure that complete sentences are presented. This means if the 1000-word limit would cut off a sentence, we extend the truncation to ensure sentence completeness. The model must select the correct LaTeX equation from one positive option (the gold equation from the original paper) and three negative options.

**5.2.1.3.3 Label Space Construction.** To craft high-quality negative options, we mask out the target gold equation in a paper and prompt GPT-4 to generate the masked equation based on the context before and after the equation (100 words on each side).<sup>3</sup>

---

<sup>2</sup> ACL, EMNLP, NAACL, TACL, and EACL, etc.

<sup>3</sup> We also provide GPT-4 with the left part of the “” sign from the gold equation to make the LLM-crafted negative equations more similar to the gold equation, thereby increasing the challenge.

**5.2.1.3.4 Quality Control (Automatic and Manual).** We filtered out negative equations if: a) they were identical to the gold equation; b) GPT-4 could easily recognize the flaws. For the latter, we provided GPT-4-Turbo with the negative equation and asked whether the equation had significant flaws. The remaining negative equations are thus “hard” options that can easily deceive the LLMs.<sup>4</sup> Among the original 3,877 instances, the instance will be abandoned if it cannot gather 3 qualified negative equations. The above process results in a total of 1,449 instances. Since all the above filtering steps are based on LLMs that may still leave some false negative equations, we asked humans to further filter out classification instances with any suspicious false negative equations (i.e., LLM-crafted negative equations that are logically correct). After human filtering, we have a total of 1,049 classification instances.

An example instance of EQUINFER is shown in Figure 5.2.

## 5.2.2 Prompting LLMs

Here we elaborate on our prompts for scenarios where gold labels are present and where gold labels are absent, respectively.

*Prompt when the gold label is present (w/ G).* The prompt used can be seen in the first block of Table 5.2.

*Prompt when the gold label is absent (w/o G).* In this case, the gold option is deleted. The key question is whether we should provide hints for the LLMs on how to handle situations where all options appear incorrect, and how to implement these hints. In

---

<sup>4</sup> In practice, for each classification instance, if any of the three negative equations can fool GPT-4, we will keep all three negative equations in our dataset. This ensures there is at least one “hard” negative equation in each classification instance.

this work, based on real-world scenarios, we design three types of hints (blocks 2-4 in Table 5.2):

- HINT-AS-OPTION: Even though no gold label is available, we provide “none-of-them” as one of the options. This mirrors common human behavior when no valid options are found, leading to the selection of this choice. An answer will only be counted as correct when LLMs choose “none-of-them”.

- HINT-IN-INSTRU: In contrast to the above hint type, here we do not include “none-of-them” as an option. Instead, in the instruction, we explicitly request the LLM to output “none-of-them” if no correct option is found. An answer will only be counted as correct when LLMs return “none-of-them”.

- NO-HINT: No hint at all. The instruction is the same as “w/  $G$ ” except for the absence of the gold label.

The evaluation of NO-HINT is more complicated: based on our observations, some LLMs, especially top-performing ones, tend to generate a new option with an explanation if they believe no correct options are provided (this may also indicate data leakage of our datasets in LLM pretraining, which we will analyze in  $Q_4$  of Section 5.3.2). This type of LLM response creates two challenges: i) it lacks a fixed format, making automatic parsing for system evaluation infeasible; ii) in reality, if an LLM response contains a reasonable label for the input, it requires us to understand the task and conduct some reasoning, which is beyond the scope of automatic processing. Also, we want to avoid using a separate LLM for evaluation, which might introduce even more errors. Therefore, for NO-HINT, we always report human performance by manually reviewing LLM responses.

We believe that any of the hint types mentioned above would work for human users. Using diverse prompts, we aim to comprehensively evaluate the model’s performance without the gold option and avoid behavior specific to a particular prompt.

### 5.2.3 OMNIACCURACY: *A New Evaluation Metric*

Our goal with OMNIACCURACY is to have it reflect model performance both when the gold label is present and absent. Therefore, we define OMNIACCURACY in the following straightforward form:

$$\text{OMNIACCURACY} = \frac{1}{2} \cdot (\mathcal{A}_{w/} + E[\mathcal{A}_{w/o}]) \tag{5.1}$$

where  $\mathcal{A}_{w/}$  represents the accuracy when the gold label is present, and  $E[\mathcal{A}_{w/o}]$  indicates the expectation of accuracy when the gold label is absent. In practice,  $E[\mathcal{A}_{w/o}]$  can be achieved by combining multiple prompting techniques, such as the three hints styles described in Section 5.2.2. We also encourage researchers to explore the most appropriate form for their particular research. In this work, we adopt the following form:

$$E[\mathcal{A}_{w/o}] = \frac{\sum_{i=1}^n \mathcal{A}_{w/o}^i}{n} \tag{5.2}$$

i.e., we take the average performance across all conditions where no correct answer is presented, including “HINT-AS-OPTION”, “HINT-IN-INSTRU” and “NO-HINT”, as the comprehensive and robust assessment of LLMs when the gold label is missing.

## 5.3 Experiments

We evaluate several popular open-source and closed-source LLMs in this study:

- 1) **Closed-source LLMs.** GPT-4 OpenAI (2023) and Claude3 Anthropic (2024).
- 2) **Open-**

**source LLMs.** Llama-3 Meta (2024), Gemma Mesnard et al. (2024), and Mistral Jiang et al. (2023).

We adopt the most recently released instruction-tuned versions of each LLM model as follows: *gpt-4o-2024-05-13*, *claude-3-opus-20240229*, *Meta-Llama-3- 8B-Instruct*, *gemma-7b-it*, *Mistral-7B-Instruct-v0.2*. During inference, the temperature is always set to 0 for reproducibility. The top\_p is set to 0.9, and the max\_length for generation is 1000.

Table 5.3. OMNIACCURACY of LLMs and humans.

		Closed-source		Open-source			Human	
		GPT-4	Claude3	Llama3	Gemma	Mistral		
MC-TEST	w/ $\mathcal{G}$ (i.e., $\mathcal{A}_{w/}$ )	98.67	98.26	94.23	39.0	87.53	100.00	
	w/o $\mathcal{G}$	Hint as option	80.17	49.83	43.10	4.33	39.97	96.00
		Hint in instru.	80.40	62.17	3.83	15.26	30.27	97.00
		No hint	41.30	60.30	50.10	15.90	33.60	93.00
		$E[\mathcal{A}_{w/o}]$	67.29	57.43	32.34	11.83	34.61	95.33
		OMNIACCURACY	<b>82.98</b>	<b>77.85</b>	<b>63.29</b>	<b>25.41</b>	<b>61.07</b>	<b>97.67</b>
BANK-77	w/ $\mathcal{G}$ (i.e., $\mathcal{A}_{w/}$ )	69.40	65.75	42.53	39.03	45.1	–	
	w/o $\mathcal{G}$	Hint as option	1.83	0.9	2.13	1.43	1.13	–
		Hint in instru.	6.17	5.3	2.9	0.87	1.60	–
		no hint	1.60	2.00	8.50	2.20	9.30	–
		$E[\mathcal{A}_{w/o}]$	3	2.73	4.51	1.5	4.01	–
		OMNIACCURACY	<b>36.30</b>	<b>34.24</b>	<b>23.52</b>	<b>20.27</b>	<b>24.56</b>	–
EquationInf.	w/ $\mathcal{G}$ (i.e., $\mathcal{A}_{w/}$ )	44.71	55.39	30.31	20.40	29.90	–	
	w/o $\mathcal{G}$	Hint as option	1.91	8.67	38.90	9.06	5.79	–
		Hint in instru.	2.86	9.06	0.67	0.19	0.29	–
		No hint	0.0	0.0	0.0	0.0	0.0	–
		$E[\mathcal{A}_{w/o}]$	1.59	5.91	13.19	3.08	2.03	–
		OMNIACCURACY	<b>23.15</b>	<b>30.65</b>	<b>21.75</b>	<b>11.74</b>	<b>15.96</b>	–

We run each set of experiments 3 times with options shuffled by different random seeds and report the averaged results. For more details, such as hyper-parameters or costs, please see Appendix. A.2.3.

### 5.3.1 Main Results

The main results are presented in Table 5.3.

*G is present (w/ G).* First, all LLMs, and especially closed-source LLMs, demonstrate exceptionally high accuracy, achieving around 98% on MC-TEST and over 65% on BANK-77. Second, the performance of LLMs decreases progressively from MC-TEST, BANK-77 to EQUINFER, which aligns with our expectations due to the increasing level of difficulty of these tasks.

*G is absent (w/o G).* For all three prompt styles in w/o  $G$ , LLM performance decreases notably as all models still tend to return one of the incorrect options offered. When comparing hint styles, “HINT-AS-OPTION” and “HINT-IN-INSTRU”, we find that LLMs’ preference varies. Such variability indicates the difficulty of consistently evaluating  $A_{w/o}$ , as it is somewhat dependent on prompt design. We encourage researchers to design multiple prompts and use the expected value  $E[A_{w/o}]$  rather than any individual  $A_{w/o}^i$ .

Llama3 exhibits unexpected behavior on the EQUINFER dataset, as “HINT-AS-OPTION” performs even better than “w/  $G$ ”. We suspect this may be due to data bias during model pretraining. Please see Section 5.3.2  $Q_5$  for more analysis.

*Observations about OMNIACCURACY.* In the last column of Table 5.3, we report the human performance on MC-TEST (more details in Section 5.3.2  $Q_2$ ). Although both GPT-4 and Claude3 perform on par with humans (98%+ vs. 100%), OMNIACCURACY clearly shows they are still behind humans—both overall (around 80% by LLMs vs. 97.67% by humans) and in “ $E[A_{w/o}]$ ” category.

We can see that when using the standard evaluation approach (with the gold label as an option), some LLMs appear to reach human-like performance. However, our evaluation

reveals that LLMs still lag considerably behind human performance because they cannot recognize the absence of the true answer as effectively as humans can. Therefore, OMNI-ACCURACY offers a more comprehensive measure to evaluate LLMs’ understanding of the classification task and their ability to perform human-level discrimination.

### 5.3.2 Analysis

*Q<sub>1</sub>: Most effective prompt when gold label is missing: HINT-AS-OPTION, HINT-IN-INSTRU or NO-HINT?* NO-HINT is clearly the worst among the three. Based on Table 5.3, for strong closed-source LLMs, HINT-IN-INSTRU consistently results in higher performance than HINT-AS-OPTION. This can be attributed to their superior ability to follow instructions. Additionally, considering the poorer performance of NO-HINT, it is suggested that when working with widely recognized top-performing LLMs and needing to try only one type of hint (perhaps due to budget constraints), HINT-IN-INSTRU is the better option.

Among open-source LLMs, HINT-AS-OPTION more frequently outperforms HINT-IN-INSTRU, similar to our observations about Llama 3 on EQUINFER. We suspect this is because open-source LLMs are generally less adept at following instructions, and they might be relying on some specific classification patterns seen during pretraining. Therefore, including none-of-them in the instruction is less clear for them compared to setting it as a separate option.

In addition, we notice that these open-source LLMs achieve the highest performance under NO-HINT among the three w/o *G* prompts on both MC-TEST and BANK-77. In fact, open-source LLMs tend to generate a new option whenever the gold option is absent, regardless of whether there are hints about none-of-them. Therefore,

even with HINT-AS-OPTION and HINT-IN-INSTRU, these models often ignore hints and propose self-generated answers without returning none-of-them. This leads to poorer performance HINT-AS-OPTION and HINT-IN-INSTRU.

We also notice interesting differences in the accuracy ranking of LLMs between when the gold label is available (i.e.,  $A_{w\{o\}}$ ) and when it is deleted (i.e.,  $ErA_{w\{o\}}$ ). More details and analysis will be illustrated in Appendix A.2.4.

*Q<sub>2</sub>: Human performance analysis when the gold label is absent* In the last column of Table 5.3, we report the average human performance on MC-TEST. We randomly selected 25 stories, resulting in 100 questions, each with four answer candidates. We randomly divided the 100 questions into four groups of 25 questions each, with each group corresponding to one of the four prompts (w/G, HINT-AS-OPTION, HINT-IN-INSTRU, or NO-HINT). We invited four human participants to work separately, with each person responsible for all four groups. To ensure unbiased results, we did not allow the same human to annotate the same question with different prompts, so their annotations of “w/o  $G$ ” questions would not be influenced by “w/  $G$ ” questions.

Figure 5.3 depicts the statistics of human performance versus the maximum performance of LLMs on 4 prompts. We observe two key dimensions. First, across prompts, human performance is barely affected by the presence of the gold option. In HINT-IN-INSTRU and HINT-AS-OPTION, when the gold option is deleted and the “none-of-them” option is provided either in the options or in the instruction, human performance differed by less than 4% compared to when the gold option was present. Even in NO-HINT, without any gold option or “none-of-them” option hints, humans were only slightly confused, with the difference being up to 8%. This indicates that CLASSIFY-

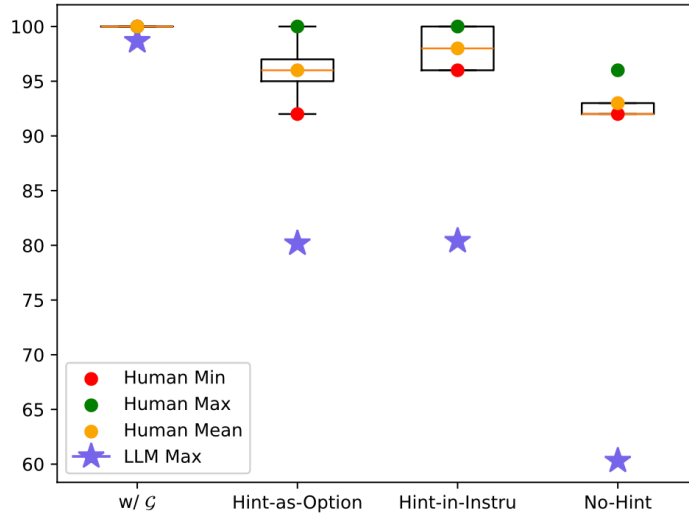


Figure 5.3. Humans vs. LLMs on MC-TEST.

-W/O-GOLD is not a very challenging task for humans, whereas it is for the models. Second, across humans, there is a 4-8% difference between Human Min and Human Max when the gold option is absent. Despite this, it is evident that even the Human Min performance is significantly higher than the LLM Max performance. Particularly under NO-HINT, human performance did not show a significant decline compared to other “w/ G” prompts, while all LLMs experienced dramatic drops.

*Q3: What different behaviors do LLMs exhibit when the gold option is absent in NO-HINT?* In NO-HINT, there are two types of patterns from models’ responses. The first type is to declare that none of the provided options is correct. The second type is to generate a new answer that the model believes to be correct. Each model behaves differently, and their response patterns vary. We identified three behaviors:

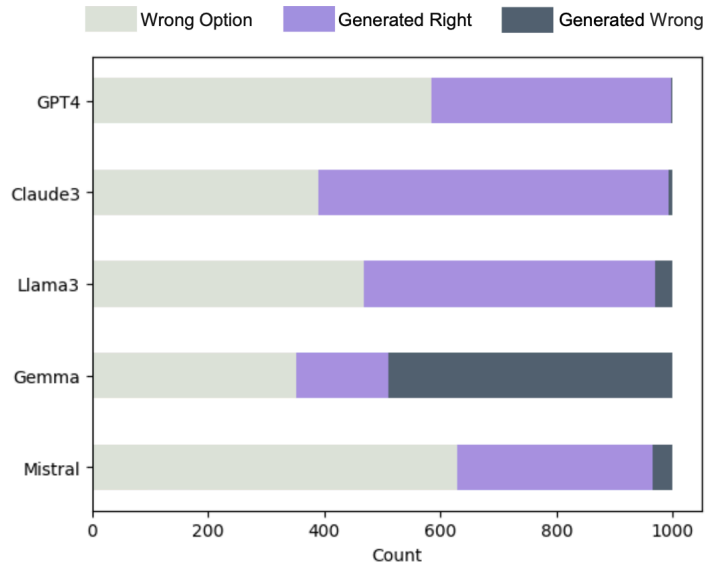


Figure 5.4. LLMs’ output pattern distribution in NO-HINT on MC-TEST.

- GPT-4 usually combines both patterns, while other models tend to generate a new answer directly.

- Figure 5.4 illustrates the distribution of LLM responses in NO-HINT on MC-TEST. GPT-4 and Mistral make the most mistakes on incorrect options, while Claude3 and Llama 3 are most likely to generate correct new labels. Gemma performs the worst, with mostly wrong labels generated.

- When generating new answers, a significant difference lies in the letter assigned by LLMs to this new option. We observed that when only options A-C exist in NO-HINT, GPT-4 consistently labels its self-generated answer as “D” while other LLMs tend to choose letters from A to C.

Sample outputs and further analysis of behavior patterns are available in Appendix A.2.5.

*Q4: If LLMs can generate a correct label in NO-HINT, is it because the LLMs have seen this dataset during pretraining?* When eyeballing the answers suggested by LLMs

in NO-HINT for MC-TEST and BANK-77, we noticed that the model sometimes generates a new option identical to the gold option. This is not surprising for MC-TEST, where options can be direct terms from the story. However, for BANK-77, with its fixed and specific label space, this raises questions about whether LLMs have had this dataset in its parameterized knowledge base.

To answer this question, we employed a small trick within the BANK-77 dataset. Given that the tokens in BANK-77’s options are delimited by “\_”, such as “lost\_or\_stolen\_card” or “activate\_my\_card”, we replaced “\_” with “-” and reran the experiments for the NO-HINT scenario. Among our five LLMs, Llama-3 was the only model that still generated “\_” in the output. Therefore, we highly suspect that Llama-3 was exposed to BANK-77 during its pretraining. Consequently, its performance may be biased, especially in the human evaluation metrics.

Even though all other LLMs consistently follow the new delimiter “-”, this can be attributed to their strong instruction-following capabilities. Therefore, our proposed format-aware trick is unable to conclusively determine whether these models were exposed to this dataset during pre-training, given the massive amount of data these models have seen.

*Q5: Would the model be misled when we add none-of-them in w/ G?* We have observed that the model exhibits different behaviors when encountering none-of-them hints. We hypothesize that this behavior might stem from data bias introduced during model pretraining or instruction tuning. To investigate this, we conducted an ablation study by introducing none-of-them options in w/ G prompts on a subset (250 instances) on MC-TEST and EQUINFER.

Table 5.4. Ablation study: w/  $\mathcal{G}$  vs w/  $\mathcal{G} + \text{None}$ 

		GPT 4	Claude 3	Llama 3	Gemma	Mistral
MC	w/ $\mathcal{G}$	98.67	98.26	94.23	39.00	87.53
	+ None	99.60	97.60	91.70	40.00	87.60
EQ	w/ $\mathcal{G}$	44.71	55.39	30.31	20.40	29.90
	+ None	45.60	58.80	15.30	10.10	22.40

To ensure fairness, we randomly replaced one incorrect answer with none-of-them, ensuring the models always select from options A-D in both scenarios. The results, presented in Table 5.4, show that for the simpler MC-TEST dataset, model performance remains nearly identical between w/  $\mathcal{G}$  and w/  $\mathcal{G} + \text{None}$  settings. For the more challenging EQUINFER dataset, closed-source models maintains their robustness, while open-source models, particularly Llama 3 and Gemma, experienced significant performance declines. This decrease is attributed to the confusion caused by the none-of-them option, which might also explain why Llama 3 performs exceptionally well in HINT-AS-OPTION on EQUINFER.

#### 5.4 Conclusion and Future Work

Our study reveals critical insights into the limitations of LLMs in classification tasks under CLASSIFY-W/O-GOLD where gold labels can be present or absent. The KNOW- NO benchmark and OMNIACCURACY metrics provide a more comprehensive reflection of LLMs’ classification performance compared to traditional accuracy metrics by combining evaluation criteria for both the presence and absence of gold labels. These findings challenge the current perception of LLMs’ performance and highlight the importance of evaluating their comprehension and discrimination abilities in the absence of gold labels.

This work establishes a new testbed for assessing LLMs' human-level discrimination intelligence, opening up avenues for understanding and improving AI comprehension. We encourage future research to utilize our framework for investigating novel techniques and architectures that can enhance LLMs' ability to handle the absence of gold labels. By building upon our work, researchers can contribute to the development of more robust and reliable LLMs, ultimately enabling them to effectively tackle real-world classification challenges across various domains.

## CHAPTER 6

### CONCLUSION

This thesis has presented a series of innovative approaches to address critical challenges in the field of Natural Language Processing, particularly focusing on the adaptation and evaluation of Large Language Models (LLMs) for classification tasks. Through our three main works - OpenStance, X-Shot, and KNOW-NO- we have made significant strides in enhancing the capabilities and assessment of LLMs in real-world scenarios.

OpenStance introduced a novel approach to zero-shot stance detection, redefining the paradigm to operate effectively across open domains without task-specific supervision. By combining indirect supervision from textual entailment datasets with weak supervision generated through an innovative masking mechanism, OpenStance demonstrated robust performance across diverse datasets, outperforming even task-specific supervised models in some cases. This work not only advances the state of the art in stance detection but also provides a framework that can inspire similar approaches in other NLP tasks involving relational text analysis.

Building on the challenges of label variability, X-shot presented a unified classification system capable of handling frequent-shot, few-shot, and zero-shot scenarios simultaneously. By leveraging instruction tuning and a novel binary classification architecture, X-shot demonstrated superior adaptability and performance across diverse domains and label distributions. This flexible framework offers a scalable solution for real-world text classification challenges, potentially revolutionizing how we approach classification tasks in dynamic, data-scarce environments.

Finally, KNOW-NO addressed a critical gap in the evaluation of the newest autoregressive LLMs for classification tasks. By introducing the CLASSIFY-W/O-GOLD task and the OMNIACCURACY metric, we provided a more comprehensive framework for assessing the human-level discriminative capabilities of LLMs in classification scenarios. This work not only unveiled limitations in current LLM performance but also set a new standard for evaluating AI models in classification tasks, encouraging the development of more robust and reliable systems.

Collectively, these works contribute significantly to the broader goal of enhancing the utility and discrimination ability of LLMs in the NLP domain. Our research has paved the way for future advancements in several key areas:

1. **Adaptive Learning:** Future research could explore methods to dynamically adapt LLMs to new domains and tasks with minimal supervision, building on the principles established in OpenStance and X-shot.
2. **Uncertainty Quantification:** Inspired by the findings in KNOW-NO, there is a need for developing LLMs that can better express uncertainty and recognize when they lack sufficient information to make accurate classifications.
3. **Multimodality:** Extending our approaches to incorporate multimodal data could lead to more robust classification system integrating textual and visual information.

In conclusion, this thesis has made substantial contributions to the field of NLP, particularly in advancing the capabilities of LLMs for classification tasks. By addressing the key challenges in adaptation, generalization, and evaluation, our work sets a strong foundation for future research aimed at creating more robust, versatile, and reliable understanding and classification NLP systems

## BIBLIOGRAPHY

- Abbott, R., Ecker, B., Anand, P., & Walker, M. A. (2016), “Internet Argument Corpus 2.0: An SQL schema for Dialogic Social Media and the Corpora to go with it,” in Proceedings of the Tenth International Conference on Language Resources and Evaluation LREC 2016, Portorož, Slovenia, May 23-28, 2016, eds. N. Calzolari, K. Choukri, T. Declerck, S. Goggi, M. Grobelnik, B. Maegaard, J. Mariani, H. Mazo, A. Moreno, J. Odijk, & S. Piperidis, European Language Resources Association (ELRA).
- Allaway, E. & McKeown, K. R. (2020), “Zero-Shot Stance Detection: A Dataset and Model using Generalized Topic Representations,” in Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing, EMNLP 2020, Online, November 16-20, 2020, eds. B. Webber, T. Cohn, Y. He, & Y. Liu, pp. 8913–8931, Association for Computational Linguistics.
- Allaway, E., Srikanth, M., & McKeown, K. R. (2021), “Adversarial Learning for Zero-Shot Stance Detection on Social Media,” in Proceedings of the 2021 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, NAACL-HLT 2021, Online, June 6-11, 2021, eds. K. Toutanova, A. Rumshisky, L. Zettlemoyer, D. Hakkani-Tür, I. Beltagy, S. Bethard, R. Cotterell, T. Chakraborty, & Y. Zhou, pp. 4756–4767, Association for Computational Linguistics.
- Anthropic (2024), “Introducing the next generation of Claude,” .
- Black, S., Gao, L., Wang, P., Leahy, C., & Biderman, S. (2021), “GPT-Neo: Large Scale Autoregressive Language Modeling with Mesh-Tensorflow,” .
- Brown, T. B., Mann, B., Ryder, N., Subbiah, M., Kaplan, J., Dhariwal, P., Neelakantan, A., Shyam, P., Sastry, G., Askell, A., Agarwal, S., Herbert-Voss, A., Krueger, G., Henighan, T., Child, R., Ramesh, A., Ziegler, D. M., Wu, J., Winter, C., Hesse, C., Chen, M., Sigler, E., Litwin, M., Gray, S., Chess, B., Clark, J., Berner, C., McCandlish, S., Radford, A., Sutskever, I., & Amodei, D. (2020a), “Language Models are Few-Shot Learners,” in Advances in Neural Information Processing Systems 33: Annual Conference on Neural Information Processing Systems 2020, NeurIPS 2020, December 6-12, 2020, virtual, eds. H. Larochelle, M. Ranzato, R. Hadsell, M. Balcan, & H. Lin.
- Brown, T. B., Mann, B., Ryder, N., Subbiah, M., Kaplan, J., Dhariwal, P., Neelakantan, A., Shyam, P., Sastry, G., Askell, A., Agarwal, S., Herbert-Voss, A., Krueger, G., Henighan, T., Child, R., Ramesh, A., Ziegler, D. M., Wu, J., Winter, C., Hesse, C., Chen, M., Sigler, E., Litwin, M., Gray, S., Chess, B., Clark, J., Berner, C., McCandlish, S., Radford, A., Sutskever, I., & Amodei, D. (2020b), “Language Models are Few-Shot Learners,” CoRR, abs/2005.14165.
- Buda, M., Maki, A., & Mazurowski, M. A. (2018), “A systematic study of the class imbalance problem in convolutional neural networks,” Neural Networks, 106, 249–259.

- Cao, K., Wei, C., Gaidon, A., Aréchiga, N., & Ma, T. (2019), “Learning Imbalanced Datasets with Label-Distribution-Aware Margin Loss,” in Advances in Neural Information Processing Systems 32: Annual Conference on Neural Information Processing Systems 2019, NeurIPS 2019, December 8-14, 2019, Vancouver, BC, Canada, eds. H. M. Wallach, H. Larochelle, A. Beygelzimer, F. d’Alché-Buc, E. B. Fox, & R. Garnett, pp. 1565–1576.
- Casanueva, I., Temcinas, T., Gerz, D., Henderson, M., & Vulic, I. (2020), “Efficient Intent Detection with Dual Sentence Encoders,” CoRR, abs/2003.04807.
- Chen, S., Khashabi, D., Yin, W., Callison-Burch, C., & Roth, D. (2019), “Seeing Things from a Different Angle: Discovering Diverse Perspectives about Claims,” in Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, NAACL-HLT 2019, Minneapolis, MN, USA, June 2-7, 2019, Volume 1 (Long and Short Papers), eds. J. Burstein, C. Doran, & T. Solorio, pp. 542–557, Association for Computational Linguistics.
- Conforti, C., Berndt, J., Pilehvar, M. T., Giannitsarou, C., Toxvaerd, F., & Collier, N. (2020), “Will-They-Won’t-They: A Very Large Dataset for Stance Detection on Twitter,” in Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics, ACL 2020, Online, July 5-10, 2020, eds. D. Jurafsky, J. Chai, N. Schluter, & J. R. Tetreault, pp. 1715–1724, Association for Computational Linguistics.
- Cui, G., Hu, S., Ding, N., Huang, L., & Liu, Z. (2022), “Prototypical Verbalizer for Prompt-based Few-shot Tuning,” in Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers), ACL 2022, Dublin, Ireland, May 22-27, 2022, eds. S. Muresan, P. Nakov, & A. Villavicencio, pp. 7014–7024, Association for Computational Linguistics.
- Devlin, J., Chang, M., Lee, K., & Toutanova, K. (2019), “BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding,” in Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, NAACL-HLT 2019, Minneapolis, MN, USA, June 2-7, 2019, Volume 1 (Long and Short Papers), eds. J. Burstein, C. Doran, & T. Solorio, pp. 4171–4186, Association for Computational Linguistics.
- Ebner, S., Xia, P., Culkin, R., Rawlins, K., & Durme, B. V. (2020), “Multi-Sentence Argument Linking,” in Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics, ACL 2020, Online, July 5-10, 2020, eds. D. Jurafsky, J. Chai, N. Schluter, & J. R. Tetreault, pp. 8057–8077, Association for Computational Linguistics.
- Enayati, S., Yang, Z., Lu, B., & Vucetic, S. (2021), “A Visualization Approach for Rapid Labeling of Clinical Notes for Smoking Status Extraction,” in Proceedings of the Second Workshop on Data Science with Human in the Loop: Language Advances, pp. 24–30, Online, Association for Computational Linguistics.
- Gunning, D., Stefik, M., Choi, J., Miller, T., Stumpf, S., & Yang, G. (2019), “XAI - Explainable artificial intelligence,” Sci. Robotics, 4.

- Han, X., Zhu, H., Yu, P., Wang, Z., Yao, Y., Liu, Z., & Sun, M. (2018), “FewRel: A Large-Scale Supervised Few-shot Relation Classification Dataset with State-of-the-Art Evaluation,” in Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing, Brussels, Belgium, October 31 - November 4, 2018, eds. E. Riloff, D. Chiang, J. Hockenmaier, & J. Tsujii, pp. 4803–4809, Association for Computational Linguistics.
- Hasan, K. S. & Ng, V. (2014), “Why are You Taking this Stance? Identifying and Classifying Reasons in Ideological Debates,” in Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing, EMNLP 2014, October 25-29, 2014, Doha, Qatar, A meeting of SIGDAT, a Special Interest Group of the ACL, eds. A. Moschitti, B. Pang, & W. Daelemans, pp. 751–762, ACL.
- Jiang, A. Q., Sablayrolles, A., Mensch, A., Bamford, C., Chaplot, D. S., de Las Casas, D., Bressand, F., Lengyel, G., Lample, G., Saulnier, L., Lavaud, L. R., Lachaux, M., Stock, P., Scao, T. L., Lavril, T., Wang, T., Lacroix, T., & Sayed, W. E. (2023), “Mistral 7B,” CoRR, abs/2310.06825.
- Krejzl, P., Hourová, B., & Steinberger, J. (2017), “Stance detection in online discussions,” CoRR, abs/1701.00504.
- Küçük, D. (2017), “Stance Detection in Turkish Tweets,” in Workshops Proceedings and Tutorials of the 28th ACM Conference on Hypertext and Social Media (HT 2017), Prague, Czech Republic, July 4-7, 2017, eds. J. Rubart & Y. Yesilada, vol. 1914 of CEUR Workshop Proceedings, CEUR-WS.org.
- Lanham, T., Chen, A., Radhakrishnan, A., Steiner, B., Denison, C., Hernandez, D., Li, D., Durmus, E., Hubinger, E., Kernion, J., Lukosiute, K., Nguyen, K., Cheng, N., Joseph, N., Schiefer, N., Rausch, O., Larson, R., McCandlish, S., Kundu, S., Kadavath, S., Yang, S., Henighan, T., Maxwell, T., Telleen-Lawton, T., Hume, T., Hatfield-Dodds, Z., Kaplan, J., Brauner, J., Bowman, S. R., & Perez, E. (2023), “Measuring Faithfulness in Chain-of-Thought Reasoning,” CoRR, abs/2307.13702.
- Li, B., Yin, W., & Chen, M. (2022a), “Ultra-fine Entity Typing with Indirect Supervision from Natural Language Inference,” Trans. Assoc. Comput. Linguistics, 10, 607–622.
- Li, S., Chen, J., Shen, Y., Chen, Z., Zhang, X., Li, Z., Wang, H., Qian, J., Peng, B., Mao, Y., Chen, W., & Yan, X. (2022b), “Explanations from Large Language Models Make Small Reasoners Better,” CoRR, abs/2210.06726.
- Li, X., Sun, X., Meng, Y., Liang, J., Wu, F., & Li, J. (2020), “Dice Loss for Data-imbalanced NLP Tasks,” in Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics, ACL 2020, Online, July 5-10, 2020, eds. D. Jurafsky, M. J. Chai, N. Schluter, & J. R. Tetreault, pp. 465–476, Association for Computational Linguistics.

- Li, Y., Zhao, C., & Caragea, C. (2021), “Improving Stance Detection with Multi-Dataset Learning and Knowledge Distillation,” in Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing, EMNLP 2021, Virtual Event / Punta Cana, Dominican Republic, 7-11 November, 2021, eds. M. Moens, X. Huang, L. Specia, & S. W. Yih, pp. 6332–6345, Association for Computational Linguistics.
- Liang, B., Chen, Z., Gui, L., He, Y., Yang, M., & Xu, R. (2022), “Zero-Shot Stance Detection via Contrastive Learning,” in WWW ’22: The ACM Web Conference 2022, Virtual Event, Lyon, France, April 25 - 29, 2022, eds. F. Laforest, R. Troncy, E. Simperl, D. Agarwal, A. Gionis, I. Herman, & L. Médini, pp. 2738–2747, ACM.
- Liu, R., Lin, Z., Tan, Y., & Wang, W. (2021), “Enhancing Zero-shot and Few-shot Stance Detection with Commonsense Knowledge Graph,” in Findings of the Association for Computational Linguistics: ACL/IJCNLP 2021, Online Event, August 1-6, 2021, eds. C. Zong, F. Xia, W. Li, & R. Navigli, vol. ACL/IJCNLP 2021 of Findings of ACL, pp. 3152–3157, Association for Computational Linguistics.
- Liu, Y., Ott, M., Goyal, N., Du, J., Joshi, M., Chen, D., Levy, O., Lewis, M., Zettlemoyer, L., & Stoyanov, V. (2019a), “RoBERTa: A Robustly Optimized BERT Pretraining Approach,” CoRR, abs/1907.11692.
- Liu, Y., Ott, M., Goyal, N., Du, J., Joshi, M., Chen, D., Levy, O., Lewis, M., Zettlemoyer, L., & Stoyanov, V. (2019b), “RoBERTa: A Robustly Optimized BERT Pretraining Approach,” CoRR, abs/1907.11692.
- Longpre, S., Hou, L., Vu, T., Webson, A., Chung, H. W., Tay, Y., Zhou, D., Le, Q. V., Zoph, B., Wei, J., & Roberts, A. (2023), “The Flan Collection: Designing Data and Methods for Effective Instruction Tuning,” in International Conference on Machine Learning, ICML 2023, 23-29 July 2023, Honolulu, Hawaii, USA, eds. A. Krause, E. Brunskill, K. Cho, B. Engelhardt, S. Sabato, & J. Scarlett, vol. 202 of Proceedings of Machine Learning Research, pp. 22631–22648, PMLR.
- Loshchilov, I. & Hutter, F. (2019), “Decoupled Weight Decay Regularization,” in 7th International Conference on Learning Representations, ICLR 2019, New Orleans, LA, USA, May 6-9, 2019, OpenReview.net.
- Lou, R., Zhang, K., Xie, J., Sun, Y., Ahn, J., Xu, H., Su, Y., & Yin, W. (2023), “MUFFIN: Curating Multi-Faceted Instructions for Improving Instruction-Following,” CoRR, abs/2312.02436.
- Lozhnikov, N., Derczynski, L., & Mazzara, M. (2018), “Stance Prediction for Russian: Data and Analysis,” in Proceedings of 6th International Conference in Software Engineering for Defence Applications, SEDA 2018, Rome, Italy, June 7-8, 2018, eds. P. Ciancarini, M. Mazzara, A. Messina, A. Sillitti, & G. Succi, vol. 925 of Advances in Intelligent Systems and Computing, pp. 176–186, Springer.

- Lu, K., Hsu, I., Zhou, W., Ma, M. D., & Chen, M. (2022), “Summarization as Indirect Supervision for Relation Extraction,” in Findings of the Association for Computational Linguistics: EMNLP 2022, Abu Dhabi, United Arab Emirates, December 7-11, 2022, eds. Y. Goldberg, Z. Kozareva, & Y. Zhang, pp. 6575–6594, Association for Computational Linguistics.
- Mahowald, K., Ivanova, A. A., Blank, I. A., Kanwisher, N., Tenenbaum, J. B., & Fedorenko, E. (2023), “Dissociating language and thought in large language models: a cognitive perspective,” CoRR, abs/2301.06627.
- Mesnard, T., Hardin, C., Dadashi, R., Bhupatiraju, S., Pathak, S., Sifre, L., Rivière, M., Kale, M. S., Love, J., Tafti, P., Hussenot, L., Chowdhery, A., Roberts, A., Barua, A., Botev, A., Castro-Ros, A., Slone, A., Héliou, A., Tacchetti, A., Bulanova, A., Paterson, A., Tsai, B., Shahriari, B., Lan, C. L., Choquette-Choo, C. A., Crepy, C., Cer, D., Ippolito, D., Reid, D., Buchatskaya, E., Ni, E., Noland, E., Yan, G., Tucker, G., Muraru, G., Rozhdestvenskiy, G., Michalewski, H., Tenney, I., Grishchenko, I., Austin, J., Keeling, J., Labanowski, J., Lespiau, J., Stanway, J., Brennan, J., Chen, J., Ferret, J., Chiu, J., & et al. (2024), “Gemma: Open Models Based on Gemini Research and Technology,” CoRR, abs/2403.08295.
- Meta (2024), “Introducing Meta Llama 3: The most capable openly available LLM to date,”.
- Mishra, S., Khashabi, D., Baral, C., & Hajishirzi, H. (2022), “Cross-Task Generalization via Natural Language Crowdsourcing Instructions,” in Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers), ACL 2022, Dublin, Ireland, May 22-27, 2022, eds. S. Muresan, P. Nakov, & A. Villavicencio, pp. 3470–3487, Association for Computational Linguistics.
- Mohammad, S. M., Kiritchenko, S., Sobhani, P., Zhu, X., & Cherry, C. (2016), “SemEval-2016 Task 6: Detecting Stance in Tweets,” in Proceedings of the 10th International Workshop on Semantic Evaluation, SemEval@NAACL-HLT 2016, San Diego, CA, USA, June 16-17, 2016, eds. S. Bethard, D. M. Cer, M. Carpuat, D. Jurgens, P. Nakov, & T. Zesch, pp. 31–41, The Association for Computer Linguistics.
- Obamuyide, A. & Vlachos, A. (2018), “Zero-shot relation classification as textual entailment,” in Proceedings of the first workshop on fact extraction and VERification (FEVER), pp. 72–78.
- OpenAI (2023), “GPT-4 Technical Report,” CoRR, abs/2303.08774
- Pouyanfar, S., Chen, S., & Shyu, M. (2018), “Deep Spatio-Temporal Representation Learning for Multi-Class Imbalanced Data Classification,” in 2018 IEEE International Conference on Information Reuse and Integration, IRI 2018, Salt Lake City, UT, USA, July 6-9, 2018, pp. 386–393, IEEE.
- Raffel, C., Shazeer, N., Roberts, A., Lee, K., Narang, S., Matena, M., Zhou, Y., Li, W., & Liu, P. J. (2020), “Exploring the Limits of Transfer Learning with a Unified Text-to-Text Transformer,” J. Mach. Learn. Res., 21, 140:1–140:67.

- Reimers, N. & Gurevych, I. (2019a), “Sentence-BERT: Sentence Embeddings using Siamese BERT-Networks,” in Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing, EMNLP-IJCNLP 2019, Hong Kong, China, November 3-7, 2019, eds. K. Inui, J. Jiang, V. Ng, & X. Wan, pp. 3980–3990, Association for Computational Linguistics.
- Reimers, N. & Gurevych, I. (2019b), “Sentence-BERT: Sentence Embeddings using Siamese BERT-Networks,” in Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing, EMNLP-IJCNLP 2019, Hong Kong, China, November 3-7, 2019, pp. 3980–3990, Association for Computational Linguistics.
- Richardson, M., Burges, C. J. C., & Renshaw, E. (2013), “MCTest: A Challenge Dataset for the Open-Domain Machine Comprehension of Text,” in Proceedings of the 2013 Conference on Empirical Methods in Natural Language Processing, EMNLP 2013, 18-21 October 2013, Grand Hyatt Seattle, Seattle, Washington, USA, A meeting of SIGDAT, a Special Interest Group of the ACL, pp. 193–203, ACL.
- Rudin, C. (2019), “Stop explaining black box machine learning models for high stakes decisions and use interpretable models instead,” Nat. Mach. Intell., 1, 206–215.
- Sainz, O., Gonzalez-Dios, I., de Lacalle, O. L., Min, B., & Agirre, E. (2022), “Textual Entailment for Event Argument Extraction: Zero- and Few-Shot with Multi-Source Learning,” in Findings of the Association for Computational Linguistics: NAACL 2022, Seattle, WA, United States, July 10-15, 2022, eds. M. Carpuat, M. de Marneffe, & I. V. M. Ruíz, pp. 2439–2455, Association for Computational Linguistics.
- Soares, L. B., FitzGerald, N., Ling, J., & Kwiatkowski, T. (2019), “Matching the Blanks: Distributional Similarity for Relation Learning,” in Proceedings of the 57th Conference of the Association for Computational Linguistics, ACL 2019, Florence, Italy, July 28- August 2, 2019, Volume 1: Long Papers, eds. A. Korhonen, D. R. Traum, & L. Màrquez, pp. 2895–2905, Association for Computational Linguistics.
- Speer, R., Chin, J., & Havasi, C. (2017), “ConceptNet 5.5: An Open Multilingual Graph of General Knowledge,” in Proceedings of the Thirty-First AAAI Conference on Artificial Intelligence, February 4-9, 2017, San Francisco, California, USA, eds. S. Singh & S. Markovitch, pp. 4444–4451, AAAI Press.
- Sun, X., Dong, L., Li, X., Wan, Z., Wang, S., Zhang, T., Li, J., Cheng, F., Lyu, L., Wu, F., & Wang, G. (2023a), “Pushing the Limits of ChatGPT on NLP Tasks,” CoRR, abs/2306.09719.
- Sun, X., Li, X., Li, J., Wu, F., Guo, S., Zhang, T., & Wang, G. (2023b), “Text Classification via Large Language Models,” in Findings of the Association for Computational Linguistics: EMNLP 2023, Singapore, December 6-10, 2023, eds. H. Bouamor, J. Pino, & K. Bali, pp. 8990–9005, Association for Computational Linguistics.

- Tsakalidis, A., Aletras, N., Cristea, A. I., & Liakata, M. (2018), “Nowcasting the Stance of Social Media Users in a Sudden Vote: The Case of the Greek Referendum,” in Proceedings of the 27th ACM International Conference on Information and Knowledge Management, CIKM 2018, Torino, Italy, October 22-26, 2018, eds. A. Cuzzocrea, J. Allan, N. W. Paton, D. Srivastava, R. Agrawal, A. Z. Broder, M. J. Zaki, K. S. Candan, A. Labrinidis, A. Schuster, & H. Wang, pp. 367–376, ACM.
- Turpin, M., Michael, J., Perez, E., & Bowman, S. R. (2023), “Language Models Don’t Always Say What They Think: Unfaithful Explanations in Chain-of-Thought Prompting,” in Advances in Neural Information Processing Systems 36: Annual Conference on Neural Information Processing Systems 2023, NeurIPS 2023, New Orleans, LA, USA, December 10 - 16, 2023, eds. A. Oh, T. Naumann, A. Globerson, K. Saenko, M. Hardt, & S. Levine.
- Walker, M. A., Tree, J. E. F., Anand, P., Abbott, R., & King, J. (2012a), “A Corpus for Research on Deliberation and Debate,” in Proceedings of the Eighth International Conference on Language Resources and Evaluation, LREC 2012, Istanbul, Turkey, May 23-25, 2012, eds. N. Calzolari, K. Choukri, T. Declerck, M. U. Dogan, B. Maegaard, J. Mariani, J. Odijk, & S. Piperidis, pp. 812–817, European Language Resources Association (ELRA).
- Walker, M. A., Anand, P., Abbott, R., & Grant, R. (2012b), “Stance Classification using Dialogic Properties of Persuasion,” in Human Language Technologies: Conference of the North American Chapter of the Association of Computational Linguistics, Proceedings, June 3-8, 2012, Montréal, Canada, pp. 592–596, The Association for Computational Linguistics.
- Wang, L. & Wang, D. (2021), “Solving Stance Detection on Tweets as Multi-Domain and Multi-Task Text Classification,” IEEE Access, 9, 157780–157789.
- Wang, X., Wang, Z., Han, X., Jiang, W., Han, R., Liu, Z., Li, J., Li, P., Lin, Y., & Zhou, J. (2020), “MAVEN: A Massive General Domain Event Detection Dataset,” in Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing, EMNLP 2020, Online, November 16-20, 2020, eds. B. Webber, T. Cohn, Y. He, & Y. Liu, pp. 1652–1671, Association for Computational Linguistics.
- Wang, X., Liu, B., & Wu, L. (2024), “FAC<sup>2</sup>E: Better Understanding Large Language Model Capabilities by Dissociating Language and Cognition,” CoRR, abs/2403.00126.
- Wang, Y., Mishra, S., Alipoormolabashi, P., Kordi, Y., Mirzaei, A., Naik, A., Ashok, A., Dhanasekaran, A. S., Arunkumar, A., Stap, D., Pathak, E., Karamanolakis, G., Lai, H. G., Purohit, I., Mondal, I., Anderson, J., Kuznia, K., Doshi, K., Pal, K. K., Patel, M., Moradshahi, M., Parmar, M., Purohit, M., Varshney, N., Kaza, P. R., Verma, P., Puri, R. S., Karia, R., Doshi, S., Sampat, S. K., Mishra, S., A, S. R., Patro, S., Dixit, T., & Shen, X. (2022), “Super-NaturalInstructions: Generalization via Declarative Instructions on 1600+ NLP Tasks,” in Proceedings of the 2022 Conference on Empirical Methods in Natural

- Language Processing, EMNLP 2022, Abu Dhabi, United Arab Emirates, December 7-11, 2022, eds. Y. Goldberg, Z. Kozareva, & Y. Zhang, pp. 5085–5109, Association for Computational Linguistics.
- Wei, J., Wang, X., Schuurmans, D., Bosma, M., Ichter, B., Xia, F., Chi, E. H., Le, Q. V., & Zhou, D. (2022), “Chain-of-Thought Prompting Elicits Reasoning in Large Language Models,” in Advances in Neural Information Processing Systems 35: Annual Conference on Neural Information Processing Systems 2022, NeurIPS 2022, New Orleans, LA, USA, November 28 - December 9, 2022, eds. S. Koyejo, S. Mohamed, A. Agarwal, D. Belgrave, K. Cho, & A. Oh.
- Williams, A., Nangia, N., & Bowman, S. R. (2018), “A Broad-Coverage Challenge Corpus for Sentence Understanding through Inference,” in Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, NAACL-HLT 2018, New Orleans, Louisiana, USA, June 1-6, 2018, Volume 1 (Long Papers), eds. M. A. Walker, H. Ji, & A. Stent, pp. 1112–1122, Association for Computational Linguistics.
- Xu, H., Vucetic, S., & Yin, W. (2022), “OpenStance: Real-world Zero-shot Stance Detection,” CoRR, abs/2210.14299.
- Xu, N., Wang, F., Dong, M., & Chen, M. (2023a), “Dense Retrieval as Indirect Supervision for Large-space Decision Making,” in Findings of the Association for Computational Linguistics: EMNLP 2023, eds. H. Bouamor, J. Pino, & K. Bali, pp. 15021–15033, Singapore, Association for Computational Linguistics.
- Xu, P., Xiao, L., Liu, B., Lu, S., Jing, L., & Yu, J. (2023b), “Label-Specific Feature Augmentation for Long-Tailed Multi-Label Text Classification,” in Thirty-Seventh AAAI Conference on Artificial Intelligence, AAAI 2023, Thirty-Fifth Conference on Innovative Applications of Artificial Intelligence, IAAI 2023, Thirteenth Symposium on Educational Advances in Artificial Intelligence, EAAI 2023, Washington, DC, USA, February 7-14, 2023, eds. B. Williams, Y. Chen, & J. Neville, pp. 10602–10610, AAAI Press.
- Ye, Q., Lin, B. Y., & Ren, X. (2021), “CrossFit: A Few-shot Learning Challenge for Cross-task Generalization in NLP,” in Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing, EMNLP 2021, Virtual Event / Punta Cana, Dominican Republic, 7-11 November, 2021, eds. M. Moens, X. Huang, L. Specia, & S. W. Yih, pp. 7163–7189, Association for Computational Linguistics.
- Yin, W., Hay, J., & Roth, D. (2019), “Benchmarking Zero-shot Text Classification: Datasets, Evaluation and Entailment Approach,” in Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing, EMNLP-IJCNLP 2019, Hong Kong, China, November 3-7, 2019, eds. K. Inui, J. Jiang, V. Ng, & X. Wan, pp. 3912–3921, Association for Computational Linguistics.

- Yin, W., Radev, D. R., & Xiong, C. (2021), “DocNLI: A Large-scale Dataset for Document-level Natural Language Inference,” in Findings of the Association for Computational Linguistics: ACL/IJCNLP 2021, Online Event, August 1-6, 2021, eds. C. Zong, F. Xia, W. Li, & R. Navigli, vol. ACL/IJCNLP 2021 of Findings of ACL, pp. 4913–4922, Association for Computational Linguistics.
- Yin, W., Chen, M., Zhou, B., Ning, Q., Chang, K., & Roth, D. (2023), “Indirectly Supervised Natural Language Processing,” in Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics: Tutorial Abstracts, ACL 2023, Toronto, Canada, July 9-14, 2023, eds. Y. V. Chen, M. Mieskes, & S. Reddy, pp. 32–40, Association for Computational Linguistics.
- Zelikman, E., Wu, Y., Mu, J., & Goodman, N. D. (2022), “STaR: Bootstrapping Reasoning With Reasoning,” in Advances in Neural Information Processing Systems 35: Annual Conference on Neural Information Processing Systems 2022, NeurIPS 2022, New Orleans, LA, USA, November 28 - December 9, 2022, eds. S. Koyejo, S. Mohamed, A. Agarwal, D. Belgrave, K. Cho, & A. Oh.
- Zhang, H., Zhang, X., Huang, H., & Yu, L. (2022), “Prompt-Based Meta-Learning For Few-shot Text Classification,” in Proceedings of the 2022 Conference on Empirical Methods in Natural Language Processing, EMNLP 2022, Abu Dhabi, United Arab Emirates, December 7-11, 2022, eds. Y. Goldberg, Z. Kozareva, & Y. Zhang, pp. 1342–1357, Association for Computational Linguistics.
- Zhang, J., Lertvittayakumjorn, P., & Guo, Y. (2019), “Integrating Semantic Knowledge to Tackle Zero-shot Text Classification,” in Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, NAACL-HLT 2019, Minneapolis, MN, USA, June 2-7, 2019, Volume 1 (Long and Short Papers), eds. J. Burstein, C. Doran, & T. Solorio, pp. 1031–1040, Association for Computational Linguistics.
- Zhang, J., Hashimoto, K., Liu, W., Wu, C., Wan, Y., Yu, P. S., Socher, R., & Xiong, C. (2020), “Discriminative Nearest Neighbor Few-Shot Intent Detection by Transferring Natural Language Inference,” in Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing, EMNLP 2020, Online, November 16-20, 2020, eds. B. Webber, T. Cohn, Y. He, & Y. Liu, pp. 5064–5082, Association for Computational Linguistics.
- Zhang, Y., Wang, M., Ren, C., Li, Q., Tiwari, P., Wang, B., & Qin, J. (2024), “Pushing The Limit of LLM Capacity for Text Classification,” CoRR, abs/2402.07470.
- Zhao, Z., Wallace, E., Feng, S., Klein, D., & Singh, S. (2021), “Calibrate Before Use: Improving Few-shot Performance of Language Models,” in Proceedings of the 38th International Conference on Machine Learning, ICML 2021, 18-24 July 2021, Virtual Event, eds. M. Meila & T. Zhang, vol. 139 of Proceedings of Machine Learning Research, pp. 12697–12706, PMLR.

## APPENDIX A

### ADDITIONAL MODELING DETAILS OF BINBIN

#### A.1 Super-NaturalInstruction to BinBin

We convert Super-NaturalInstruction (Wang et al., 2022) into our binary schema for the Indirect Supervision. Super-NaturalInstruction is a benchmark In-context learning dataset with 757 train tasks and 119 test tasks. Each task includes a definition, several positive/negative demonstrations, and thousands of instances. A task instance from Super-NaturalInstruction is presented in Figure 4.7. We select 100 instances from each task and convert them into BinBin schema for Indirect Supervision training as shown in Figure 4.8.

#### A.2 X-Shotdata to Binbin

As discussed in Section 4.1, each *X-Shot* instance is converted into the unified binary format to align with BinBin. A detailed example from *FewRel* is illustrated in Figure 4.6.

#### A.3 In-context Learning baseline

For the in-context learning baseline, we provide 3 demonstrations, 2 positive ones and 1 negative one, and let GPT-3.5 complete the label of the test instance. A sample template is as follows for *FewRel*:

Sentence: Pan was appointed director of the National Academy (Zhejiang Academy of Fine Arts) by the Kuomintang Minister of Culture, Chen Lifu, in 1945.  
Entity 1: Chen Lifu  
Entity 2: Kuomintang  
Relation: member of political party  
Label: Yes

Sentence: Aldo Protti (July 19 ,1920 - August 10 , 1995 ) was an Italian baritone opera singer  
Entity 1: Aldo Protti  
Entity 2: baritone  
Relation: voice type  
Label: Yes

Sentence: Part of DirectX' Direct3D is used to render three - dimensional graphics in applications  
Entity 1: DirectX  
Entity 2: Direct3D  
Relation: movement  
Label: No

Sentence: The Suzuki GS500 is an entry level motorcycle manufactured and marketed by the Suzuki Motor Corporation.  
Entity 1: Suzuki GS500  
Entity 2: Suzuki Motor Corporation  
Relation: winner  
Label:

Figure A.1. GPT Template.

We use the OpenAI API to extract and exponentiate the log probability of the model predicting "Yes", converting it into a regular probability. We then select the label with the highest probability as the predicted label, similar to our BinBinapproach.

### ***A.3.1 BinBinTask Instructions***

To prove the robustness of our model, we create 3 versions of the task instructions for each of the datasets (*FewRel*, *MAVEN*, *RAMS*) as follows:

*FewRel*

Instruction A: Given a sentence about two entities, return a relation between the two entities that can be inferred from the sentence.

Instruction B: Your task is to identify a relationship between two entities mentioned in a given sentence.

Instruction C: Identify the relationship between two entities in a given sentence that can be inferred from the sentence.

*RAMS*

Instruction A: Your task is to identify the role of a specified argument within a given sentence, in relation to an identified event trigger.

Instruction B: Identify the role of the argument given the event trigger within the sentence.

Instruction C: Identify the role of the argument given the event trigger within the sentence.

*MAVEN*

Instruction A: Given the sentence and the identified trigger word, determine the most appropriate event category for this trigger.

Instruction B: Identify the event type in the sentence associated with the trigger word.

Instruction C: Classify the event represented by the trigger word in the context of the following sentence.

Figure A.2. GPT Template.

### A.3.2 *Efficiency Analysis*

- **Time Cost** Our system is trained on NVIDIA A100 GPUs. On a single GPU, it takes 6/30/30 hours on average using the RoBERTa/T5/GPT-Neo model for each task with bf16 precision acceleration. We incorporate packages mainly from Pytorch for the modeling.

- **Memory Cost** The memory requirements for our proposed system include the model parameters and the dataset, similar to other methods and the latest state-of-the-art baseline. The sizes of parameters for the RoBERTa, T5-3B and GPT-Neo models are

- 355M, 3B, and 1.3B, respectively. For T5, since it is encoder-decoder architecture and we only adopt the encoder, the real memory usage would be 1.5B, half of the original size. \_\_\_\_\_

## APPENDIX B

### ADDITIONAL MODELING DETAILS OF KNOW-NO

#### *B. Scaling EQUINFER*

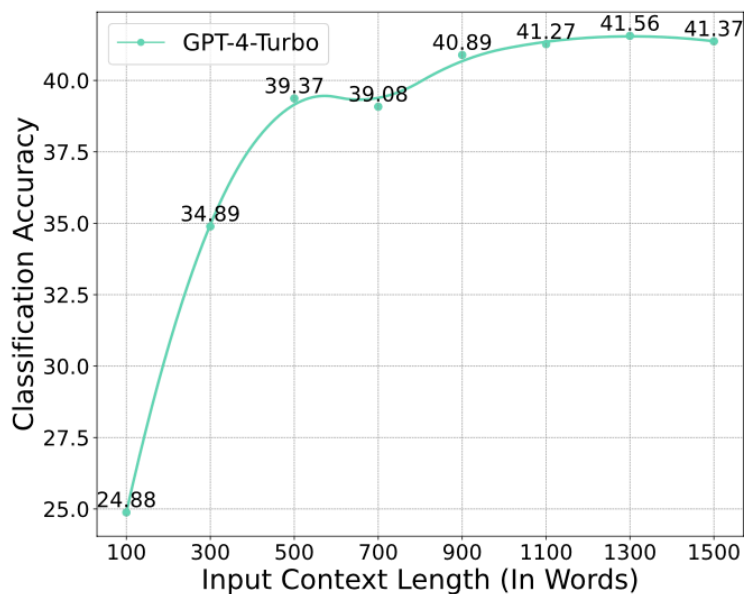


Figure B.1. Scaling the length of context around the equation in EQUINFER.

Figure B.1 shows how the length of context around equations affects the classification performance in EQUINFER.

#### *B.1 PROMPT ILLUSTRATION*

Below, we present sample inputs from the MC-TEST, BANK-77, and EQUINFER datasets. These examples represent w/  $G$  prompts, where the gold option is present (in blue). For w/o  $G$  prompts, the gold option is removed, and the prompt is adjusted accordingly. Specifically, we add hints (none-of-them) in options/instruction for HINT-AS-OPTION/HINT-IN-INSTRU, or no hint at all for NO-HINT.

**Task:**  
Given the story and an associated question, please return the correct option for the question without explanation.

**Story:**  
[.....] "I'm going to play for the Yankees ma!" Tom said. Tom's mom was so excited that she took Tom and the whole family out for dinner. Grandpa, Grandma, Mom and Dad were all there, and bought Tom a big cake! [.....]

**Question:**  
What did Tom's family buy him to celebrate?

**Options:**  
A: A cake  
B: A car  
C: New clothes  
D: A baseball

**Your answer:**

Figure B.2. Example of MC-TEST.

**Task:**  
You are going to perform an intent classification task. Given an utterance and multiple intent class options, return the correct intent class name only.

**Utterance:**  
An unauthorized payment is in my app

**Intent Class Options:**  
refund\_not\_showing\_up, apple\_pay\_or\_google\_pay,  
pending\_card\_payment, [card\\_payment\\_not\\_recognised](#),  
[.....]  
balance\_not\_updated\_after\_cheque\_or\_cash\_deposit

**Your answer:**

Figure B.3. Example of BANK-77.

**Task:**

You are given the latex source code of the context before and after an equation in an NLP paper and multiple options for the equation. Only return the correct option letter as the answer without explanation.

**Context before:**

“[.....]  $e_{x^i}$  and  $e_{z^i}$  are the average representations along the sequence dimension from the encoder outputs. Apart from the contrastive loss, the standard cross-entropy loss is calculated as:”

**Context after:**

“We combine both losses as the final loss:

$$\mathcal{L} = \mathcal{L}_{ce} + \lambda \mathcal{L}_{ctr}$$

where  $\lambda$  is an interpolation factor. [.....]”

**Equation options:**

A:  $\mathcal{L}_{ce} = -\sum_{i=1}^N \sum_{c=1}^C y_{i,c} \log(p_{i,c})$

B:  $\mathcal{L}_{ce} = -\sum_{i=1}^N (\log P_{\theta}(y^i|x^i) + \log P_{\theta}(y^i|z^i))$

C:  $\mathcal{L}_{ce} = -\frac{1}{N} \sum_{i=1}^N \sum_{c=1}^C y_c^i \log(p_c^i)$

D:  $\mathcal{L}_{ce} = -\sum_{i=1}^N \sum_{k=1}^C y_k^i \log \hat{y}_k^i$

**Your answer:**

Figure B.4. Example of EQUINFER.

## B.2 Experimental Setting Details

The cost of using the two closed-source LLMs is detailed below. For BANK-77, MC-TEST, and EQUINFER, it takes 1 million, 0.4 million, and 5 million input tokens, respectively, to run one prompt on all instances. Given that one set of experiments includes 4 prompts, the costs for GPT-4 (including both input and output tokens) are approximately \$15, \$8, and \$100 for BANK-77, MC-TEST, and EQUINFER, respectively. For Claude 3, the costs are roughly \$60, \$24, and \$300 for BANK-77, MC-TEST, and EQUINFER, respectively.

We want to highlight that KNOW-NO and OMNIACCURACY represent a novel evaluation approach under CLASSIFY-W/O-GOLD. They can be applied to any model in any classification setting, extending beyond the three datasets and five models reported in this study.

### ***B.3 Ranking Differences of LLMs between $A_{w\{}$ and $A_{w\{o}$***

Are there any differences in the accuracy ranking of LLMs between when the gold label is available (i.e.,  $A_{w\{}$ ) and when it is deleted (i.e.,  $ErA_{w\{oS}$ )? We notice that closed-source models consistently achieve impressive accuracy when the gold option is available. However, when the gold option is removed, they resist acknowledging the absence of a correct label, especially in more challenging tasks. In MC-TEST, the simplest of the three tasks, GPT-4 and Claude 3 confidently suggest “none” or generate a new answer, outperforming other models by a large margin. Conversely, for BANK-77 and EQUINFER, they tend to select from the available incorrect options, performing as poorly as the open-source models.

### ***B.4 Model Behaviour under NO-HINT***

In this section, we provide a detailed analysis of each model’s responses in NO-HINT. Since MC-TEST is the simplest of the three tasks, it is the most likely scenario for the model to correctly identify the NO-HINT context. In contrast, for EQUINFER and BANK-77, it is very rare for the model to identify the situation. Hence, we will focus on MC-TEST for our primary discussion.

Even with MC-TEST, models fail to recognize the absence of the gold label approximately half of the time. When this occurs, the typical response patterns are as follows:

**Gold option:** Rick  
**Options:** A. Bob, B. James, C. Stephanie  
**Response type 1:** “None of the above”  
**Response type 2:** “None of the options provided are correct. The correct answer is Rick.”  
**Response type 3:** “D. Rick”  
**Response type 4:** “C. Rick”

Figure B.5. Error patterns.

Each model displays these patterns differently, with unique characteristics specific to each model.

*GPT-4* Surprisingly, GPT-4 is among the least likely models to acknowledge the absence of a gold label. However, when it does recognize this situation and adopts an “out-of-the-box” approach, its accuracy is notably the highest. The most common responses for GPT-4 are **Response types 1, 2, and 3**. For the new answers proposed, GPT-4 always labels them as option D.

*Claude 3 / Llama 3 / Mistral* Claude 3, Llama 3, and Mistral exhibit consistent and standardized response patterns in the format “**Letter. Option**”, without giving any additional discussion. The option might be one of the incorrect options provided or a new answer generated by the model. Interestingly, unlike GPT-4, which always assigns its new answer as option D, Claude 3, Mistral, and Llama 3 tend to assign a letter from A to C, as per **Response type 4**.

*Gemma* Gemma’s responses are the most chaotic and illogical. It often forces an explanation on the incorrect option or provides an incorrect option but includes the correct answer in the reasoning. This inconsistency aligns with its poor performance across all datasets and most prompts. One example of such an error is as follows:

**Question:** What color was the animal’s stripe?

**Gold option:** green

**Options:** A. playground, B. sand, C. frosting

**Gemma output:**

*“Answer: C. The animal had a green stripe across its back, therefore the answer is C.”*

As a result, it is very rare for Gemma to generate correct output, and none of the **Response types** fit its behavior.

### ***B.5 Human Behavior under NO-HINT***

**Gold option:** Rick  
**Options:** A. Bob, B. James, C. Stephanie  
**Response type 1:** *“None of the options is correct / something is wrong with the problem setting”*  
**Response type 2:** *“Rick”*

Figure B.6. Human behavior.

When faced with NO-HINT, humans respond more straightforwardly and directly compared to the models. Humans often either point out there might be flaws in the question design or provide the correct answer directly as the two **response types** above. Very interestingly, we notice that humans would not assign a letter to a self-generated answer and treat it as one of the provided options. This behavior seems to be unique to models, likely because they are trained to follow the provided pattern.