X-Shot: A Unified System to Handle Frequent, Few-shot and Zero-shot Learning Simultaneously in Classification

Hanzi Xu* **Muhao Chen†** **Lifu Huang‡** **Slobodan Vucetic*** **Wenpeng Yin§**
*Temple University †University of California, Davis ‡Virginia Tech §Penn State University
{hanzi.xu, slobodan.vucetic}@temple.edu wenpeng@psu.edu

Abstract

In recent years, few-shot and zero-shot learning, which learn to predict labels with limited annotated instances, have garnered significant attention. Traditional approaches often treat frequent-shot (freq-shot; labels with abundant instances), few-shot, and zero-shot learning as distinct challenges, optimizing systems for just one of these scenarios. Yet, in real-world settings, label occurrences vary greatly. Some of them might appear thousands of times, while others might only appear sporadically or not at all. For practical deployment, it is crucial that a system can adapt to any label occurrence. We introduce a novel classification challenge: X-Shot, reflecting a real-world context where freq-shot, few-shot, and zero-shot labels co-occur without predefined limits. Here, X can span from 0 to $+\infty$. The crux of X-Shot centers on open-domain generalization and devising a system versatile enough to manage various label scenarios. To solve X-Shot, we propose BinBin (**bin**ary **in**ference **b**ased on **in**struction following) that leverages the *Indirect Supervision* from a large collection of NLP tasks via instruction following, bolstered by *Weak Supervision* provided by large language models. BinBin surpasses previous state-of-the-art techniques on three benchmark datasets across multiple domains. To our knowledge, this is the first work addressing X-Shot learning, where X remains variable.[1]

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 and 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: ($\mathcal{Q}_1$) 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. ($\mathcal{Q}_2$) 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 $\mathcal{Q}_1$, 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 supple-

[1] Code and data are publicly available at https://github.com/xhz0809/X-shot.

Findings of the Association for Computational Linguistics: ACL 2024, pages 4652–4665

mented with pertinent instructions, especially for the low-shot (few-shot and zero-shot) labels. For $\mathcal{Q}_2$, 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 $\mathcal{Q}_1$ and $\mathcal{Q}_2$, we forge a holistic framework, `BinBin` (**bin**ference **b**ased on **in**struction 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 underexplored, 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.

2 Related Work

Data Imbalance 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., 2023c). 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., 2023c; 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.

Indirect Supervision 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., 2023b), 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., 2023b) 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).

Unified Discriminative Classifier Previous research, such as the work presented in (Xu et al., 2023a), also attempts to transform classification problems into binary tasks. While this system represents a discriminative classifier approach similar to ours, there are several significant differences. The most notable distinction is that it does not cover multiple learning scenarios, whereas our X`-Shot` encompasses the entire range of label occurrences. Additionally, without any supervision,

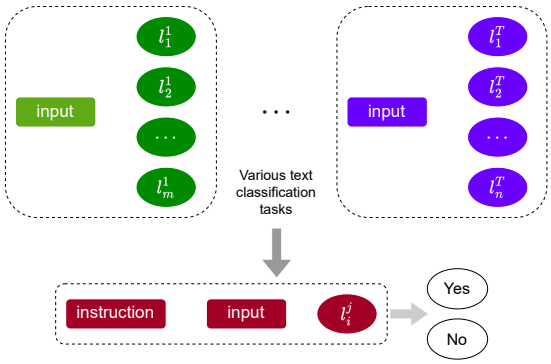

Figure 1: `BinBin` unifies T various text classification tasks as an Instruction Tuning problem. l_i^j: the i-th label in the j-th task. A detailed example is in Appendix A.2.

their approach cannot be adapted to diverse tasks, and therefore, is less flexible than ours. Most importantly, this system benchmarks its performance against generative models, rather than comparing it with state-of-the-art (SOTA) systems specifically designed for target classification tasks.

3 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: $\{\cdots, l_i, \cdots\}$ 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 section, we delve deeper into our approach concerning the universal system and the provided supervisions.

4 Methodology

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

4.1 `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 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.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.

4.2 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., 2023b). 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 cat-

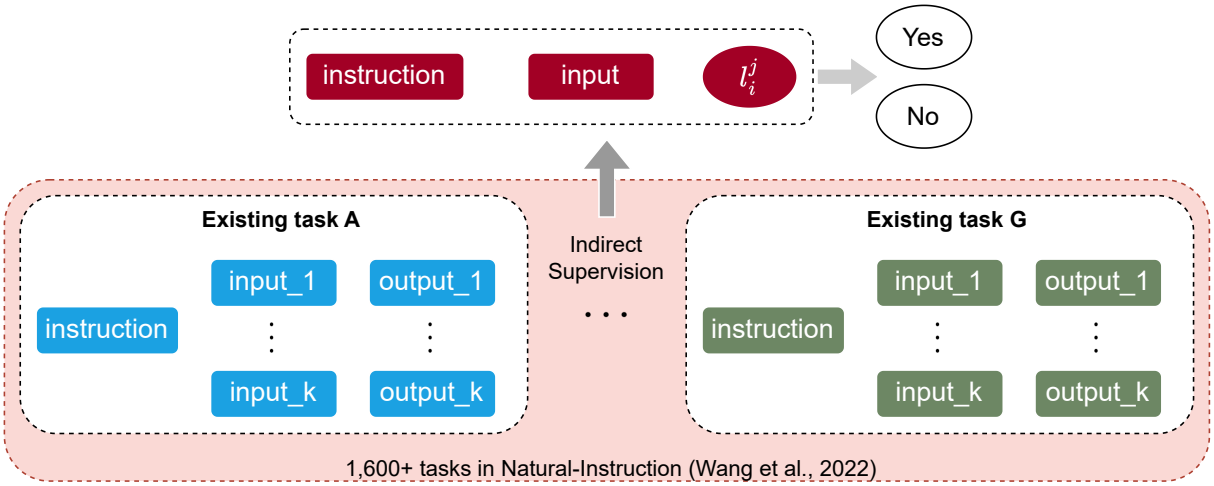

Figure 2: *Indirect Supervision* for `BinBin`. *Indirect Supervision* enables `BinBin` to learn from diverse tasks in Super-NaturalInstruction before applying this knowledge to a target classification task j. A detailed example is in Appendix A.1

egories. Each task is accompanied by instructions and numerous input-output instances (an example of tasks is in Appendix A.1). This dataset offers an invaluable source of *Indirect Supervision* for our target X`-Shot`. As in Appendix A.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 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., 2020) 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
```

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., 2019) 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., 2019) 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.

5 Experiments

5.1 Experimental setting

Datasets. 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 1. We rename each resulting dataset as "[]$_{X\text{-Shot}}$."

• **FewRel**$_{X\text{-Shot}}$: *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 $test$.

• **MAVEN**$_{X\text{-Shot}}$: 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 $test$.

	domain	#freq	#few	#zero
FewRel$_{X\text{-Shot}}$	Wikipedia	26	26	26
MAVEN$_{X\text{-Shot}}$	Wikipedia	23	23	23+1
RAMS$_{X\text{-Shot}}$	News articles	10	10	10

Table 1: Statistics of dataset labels.

• **RAMS**$_{X\text{-Shot}}$: *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*.

Baselines. 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.3.

• ***Indirect Supervision* from Text Entailment (NLI; Li et al. 2022)**. 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.

Implementation details. 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.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.

5.2 Results

Table 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.

5.3 Analyses

In addition to reporting the main results, we further analyze our system in the following dimensions: ($\mathcal{Q}_1$) the individual contribution of *Indirect Supervision* and *Weak Supervision*; ($\mathcal{Q}_2$) is `BinBin` adaptive to other model scales and architectures? ($\mathcal{Q}_3$) why does "zero" show better performance than "few" in RAMS$_{X\texttt{-Shot}}$ and MAVEN$_{X\texttt{-Shot}}$? ($\mathcal{Q}_4$) 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? ($\mathcal{Q}_5$) 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? ($\mathcal{Q}_6$) The efficiency of our system. ($\mathcal{Q}_7$) The mistakes our system makes.

($\mathcal{Q}_1$) Ablation study. Figure 3 depicts the ablation study, where either *Indirect Supervision* of *Weak Supervision* is discarded from our system `BinBin`. Our findings reveal that both supervision sources fulfill complementary roles in the X`-Shot`

Models	FewRel$_{X\text{-Shot}}$				RAMS$_{X\text{-Shot}}$				MAVEN$_{X\text{-Shot}}$			
	all	freq	few	zero	all	freq	few	zero	all	freq	few	zero
MWC (Soares et al., 2019)	49.82	94.23	55.23	0.0	34.47	**78.40**	25.00	0.0	42.43	85.17	43.96	0.0
NLI (Li et al., 2022)	63.46	**95.35**	48.81	46.22	43.07	71.40	20.40	37.40	56.31	**85.65**	39.83	44.00
PPL (Cui et al., 2022)	53.23	95.15	**63.54**	0.0	27.13	65.00	16.20	0.20	46.84	85.04	**55.52**	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	**68.48**	94.06	58.04	**53.34**	**54.70**	77.00	**29.00**	**58.07**	**64.96**	84.32	46.64	**63.97**

Table 2: Main results on three benchmark target tasks

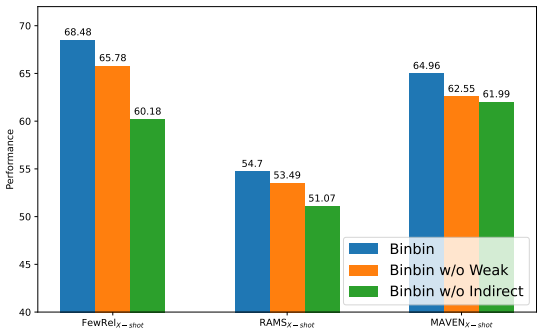

Figure 3: Ablation study of BinBin

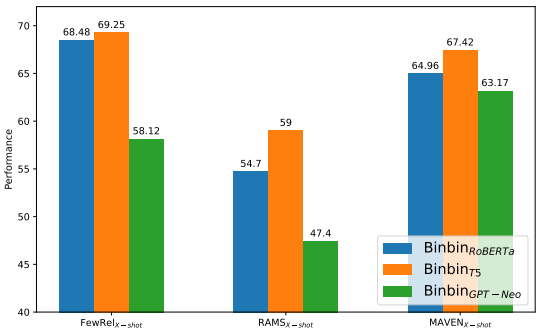

Figure 4: Backbone models across different scales and architectures

task. Encouragingly, while their combined usage yields the best results, each type of supervision, on its own, still surpasses the baselines. Such a result underscores the efficiency of our system.

($\mathcal{Q}_2$) **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. 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.

	all	freq	few	zero
FewRel$_{X\text{-Shot}}$	63.34	89.04	**60.95**	40.04
RAMS$_{X\text{-Shot}}$	51.64	**78.74**	**30.13**	40.07
MAVEN$_{X\text{-Shot}}$	63.83	**85.68**	**47.48**	58.57

Table 3: Results of training BinBin after deleting top-10 similar tasks from Natural-Instruction. Bold: enhanced performance compared to the pre-deletion state.

($\mathcal{Q}_3$) **Why do zero-shot labels outperform few-shot labels in the *MAVEN*$_{X\text{-Shot}}$ and *RAMS*$_{X\text{-Shot}}$ 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*$_{X\text{-Shot}}$ seem easier upon visual inspection; ii) In *MAVEN*$_{X\text{-Shot}}$, "None" is treated as a zero-shot label in the test set, contributing notably due to threshold tuning.

($\mathcal{Q}_4$) **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

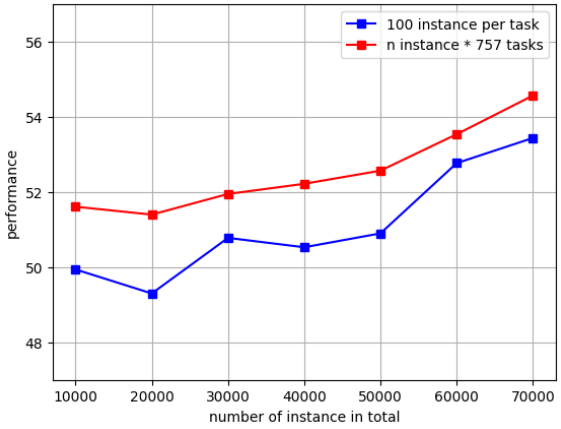

Figure 5: #instances vs. #tasks

the impact of similar tasks. The measurement is based on cosine similarity between Sentence-BERT (Reimers and Gurevych, 2019) embeddings of the task definitions in the Natural-Instruction dataset and each X-Shot target dataset's instruction.

From Table 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 2, underscoring the value of diverse training tasks. This is further supported by subsequent analysis.

($\mathcal{Q}_5$) 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). 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 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.

($\mathcal{Q}_6$) 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 that of 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, as opposed to separate systems for different label groups, actually reduces overall training time and computational effort. A more detailed quantitative report in terms of training time and computational resources can be found in Appendix A.5

($\mathcal{Q}_7$) 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.

6 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.

Limitation

The primary limitation of our model is its efficiency, particularly when handling datasets with a large number of labels when converting the original task into a binary task. This results in extended training times and increased computational efforts. It is important to note that this limitation is not an isolated challenge for our model; it aligns with the experiences reported in previous state-of-the-art models. Future work can focus on optimizing the training process to enhance efficiency without compromising the model's performance.

References

Sid Black, Leo Gao, Phil Wang, Connor Leahy, and Stella Biderman. 2021. GPT-Neo: Large Scale Autoregressive Language Modeling with Mesh-Tensorflow.

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

Mateusz Buda, Atsuto Maki, and Maciej A. Mazurowski. 2018. A systematic study of the class imbalance problem in convolutional neural networks. *Neural Networks*, 106:249–259.

Kaidi Cao, Colin Wei, Adrien Gaidon, Nikos Aréchiga, and Tengyu Ma. 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*, pages 1565–1576.

Ganqu Cui, Shengding Hu, Ning Ding, Longtao Huang, and Zhiyuan Liu. 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*, pages 7014–7024. Association for Computational Linguistics.

Seth Ebner, Patrick Xia, Ryan Culkin, Kyle Rawlins, and Benjamin Van Durme. 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*, pages 8057–8077. Association for Computational Linguistics.

Xu Han, Hao Zhu, Pengfei Yu, Ziyun Wang, Yuan Yao, Zhiyuan Liu, and Maosong Sun. 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*, pages 4803–4809. Association for Computational Linguistics.

Bangzheng Li, Wenpeng Yin, and Muhao Chen. 2022. Ultra-fine entity typing with indirect supervision from natural language inference. *Trans. Assoc. Comput. Linguistics*, 10:607–622.

Xiaoya Li, Xiaofei Sun, Yuxian Meng, Junjun Liang, Fei Wu, and Jiwei Li. 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*, pages 465–476. Association for Computational Linguistics.

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

Renze Lou, Kai Zhang, Jian Xie, Yuxuan Sun, Janice Ahn, Hanzi Xu, Yu Su, and Wenpeng Yin. 2023. MUFFIN: curating multi-faceted instructions for improving instruction-following. *CoRR*, abs/2312.02436.

Keming Lu, I-Hung Hsu, Wenxuan Zhou, Mingyu Derek Ma, and Muhao Chen. 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*, pages 6575–6594. Association for Computational Linguistics.

Swaroop Mishra, Daniel Khashabi, Chitta Baral, and Hannaneh Hajishirzi. 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*, pages 3470–3487. Association for Computational Linguistics.

Abiola Obamuyide and Andreas Vlachos. 2018. Zero-shot relation classification as textual entailment. In *Proceedings of the first workshop on fact extraction and VERification (FEVER)*, pages 72–78.

Samira Pouyanfar, Shu-Ching Chen, and Mei-Ling Shyu. 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*, pages 386–393. IEEE.

Colin Raffel, Noam Shazeer, Adam Roberts, Katherine Lee, Sharan Narang, Michael Matena, Yanqi Zhou, Wei Li, and Peter J. Liu. 2020. Exploring the limits of transfer learning with a unified text-to-text transformer. *J. Mach. Learn. Res.*, 21:140:1–140:67.

Nils Reimers and Iryna Gurevych. 2019. 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*, pages 3980–3990. Association for Computational Linguistics.

Oscar Sainz, Itziar Gonzalez-Dios, Oier Lopez de Lacalle, Bonan Min, and Eneko Agirre. 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*, pages 2439–2455. Association for Computational Linguistics.

Livio Baldini Soares, Nicholas FitzGerald, Jeffrey Ling, and Tom Kwiatkowski. 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*, pages 2895–2905. Association for Computational Linguistics.

Xiaozhi Wang, Ziqi Wang, Xu Han, Wangyi Jiang, Rong Han, Zhiyuan Liu, Juanzi Li, Peng Li, Yankai Lin, and Jie Zhou. 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*, pages 1652–1671. Association for Computational Linguistics.

Yizhong Wang, Swaroop Mishra, Pegah Alipoormolabashi, Yeganeh Kordi, Amirreza Mirzaei, Atharva Naik, Arjun Ashok, Arut Selvan Dhanasekaran, Anjana Arunkumar, David Stap, Eshaan Pathak, Giannis Karamanolakis, Haizhi Gary Lai, Ishan Purohit, Ishani Mondal, Jacob Anderson, Kirby Kuznia, Krima Doshi, Kuntal Kumar Pal, Maitreya Patel, Mehrad Moradshahi, Mihir Parmar, Mirali Purohit, Neeraj Varshney, Phani Rohitha Kaza, Pulkit Verma, Ravsehaj Singh Puri, Rushang Karia, Savan Doshi, Shailaja Keyur Sampat, Siddhartha Mishra, Sujan Reddy A, Sumanta Patro, Tanay Dixit, and Xudong Shen. 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*, pages 5085–5109. Association for Computational Linguistics.

Haike Xu, Zongyu Lin, Jing Zhou, Yanan Zheng, and Zhilin Yang. 2023a. A universal discriminator for zero-shot generalization. In *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers), ACL 2023, Toronto, Canada, July 9-14, 2023*, pages 10559–10575. Association for Computational Linguistics.

Hanzi Xu, Slobodan Vucetic, and Wenpeng Yin. 2022. Openstance: Real-world zero-shot stance detection. *CoRR*, abs/2210.14299.

Nan Xu, Fei Wang, Mingtao Dong, and Muhao Chen. 2023b. Dense retrieval as indirect supervision for large-space decision making. In *Findings of the Association for Computational Linguistics: EMNLP 2023*, pages 15021–15033, Singapore. Association for Computational Linguistics.

Pengyu Xu, Lin Xiao, Bing Liu, Sijin Lu, Liping Jing, and Jian Yu. 2023c. 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*, pages 10602–10610. AAAI Press.

Qinyuan Ye, Bill Yuchen Lin, and Xiang Ren. 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*, pages 7163–7189. Association for Computational Linguistics.

Wenpeng Yin, Muhao Chen, Ben Zhou, Qiang Ning, Kai-Wei Chang, and Dan Roth. 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*, pages 32–40. Association for Computational Linguistics.

Wenpeng Yin, Jamaal Hay, and Dan Roth. 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*, pages 3912–3921. Association for Computational Linguistics.

Wenpeng Yin, Dragomir R. Radev, and Caiming Xiong. 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*, volume ACL/IJCNLP 2021 of *Findings of ACL*, pages 4913–4922. Association for Computational Linguistics.

Haoxing Zhang, Xiaofeng Zhang, Haibo Huang, and Lei Yu. 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*, pages 1342–1357. Association for Computational Linguistics.

Jian-Guo Zhang, Kazuma Hashimoto, Wenhao Liu, Chien-Sheng Wu, Yao Wan, Philip S. Yu, Richard Socher, and Caiming Xiong. 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*, pages 5064–5082. Association for Computational Linguistics.

Jingqing Zhang, Piyawat Lertvittayakumjorn, and Yike Guo. 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)*, pages 1031–1040. Association for Computational Linguistics.

Zihao Zhao, Eric Wallace, Shi Feng, Dan Klein, and Sameer Singh. 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*, volume 139 of *Proceedings of Machine Learning Research*, pages 12697–12706. PMLR.

A Appendix

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 7. We select 100 instances from each task and convert them into `BinBin` schema for *Indirect Supervision* training as shown in Figure 8.

A.2 `X-Shot` data 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 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*:

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 -

Instance for Prediction

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

label: Yes No

Figure 6: 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 7: Super-Naturalinstructions task example

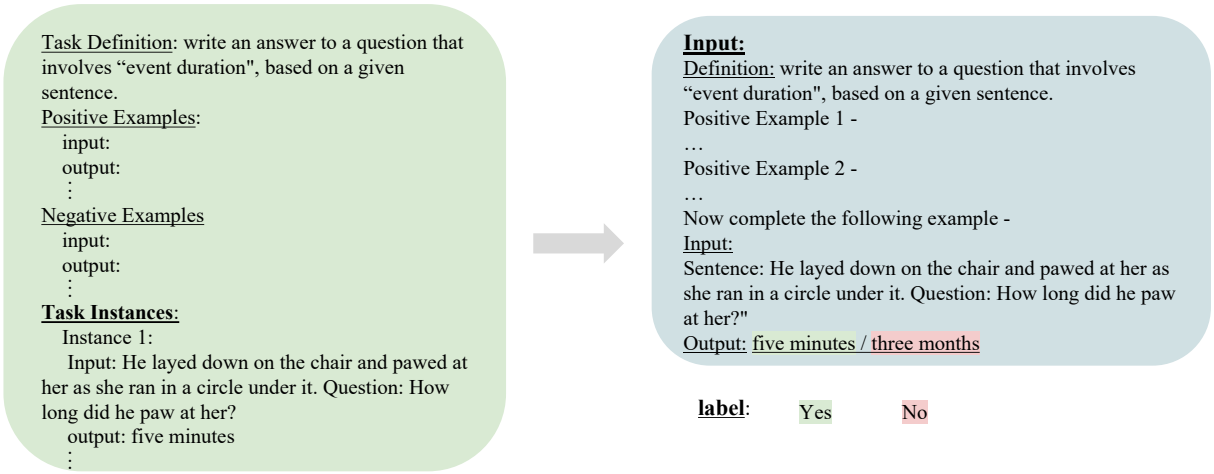

Figure 8: Super-Naturalinstructions to binary BinBin

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:

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 BinBin approach.

A.4 BinBin Task Instuctions

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.`

A.5 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.

A.6 ACL ethics code discussion

• **Scientific artifacts usage** The existing Scientific artifacts included in this work are RoBERTa, T5 and GPT-Neo model (Liu et al., 2019; Raffel et al., 2020; Black et al., 2021) and 3 NLP classification datasets. The model and datasets used in this work are publicly available for research purposes and do not contain any sensitive information. Our use of existing Scientific artifacts is consistent with their intended usage.

The license, copyright information, the asset we proposed, and terms of use information regarding `BinBin`, will be specified once the code is released.