AS-ES Learning: Towards Efficient CoT Learning in Small Models

Nuwa Xi[*], Yuhan Chen[*], Sendong Zhao[†], Haochun Wang, Bing Qin and Ting Liu
Research Center for Social Computing and Information Retrieval,
Harbin Institute of Technology, China
{nwxi,yuhanchen,sdzhao,hcwang,bqin,tliu}@ir.hit.edu.cn

Abstract

Chain-of-Thought (CoT) serves as a critical emerging ability in LLMs, especially when it comes to logical reasoning. Attempts have been made to induce such ability in small models as well by distilling from the data with CoT generated by Large Language Models (LLMs). However, existing methods often simply generate and incorporate more data from LLMs and fail to note the importance of efficiently utilizing existing CoT data. We here propose a new training paradigm AS-ES (Abstractive Segments - Extractive Segments) learning, which exploits the inherent information in CoT for iterative generation. Experiments show that our methods surpass the direct seq2seq training on CoT-extensive tasks like MWP and PET summarization, without data augmentation or altering the model itself. Furthermore, we explore the reason behind the inefficiency of small models in learning CoT and provide an explanation of why AS-ES learning works, giving insights into the underlying mechanism of CoT.

1 Introduction

CoT is one of the most important emerging abilities that distinguishes LLMs from prior models with smaller scales (Wei et al., 2022). The explicit introduction of CoT enables LLMs to tackle complex problems that necessitate critical thinking and intricate logical reasoning, thus enhancing the overall performance of LLMs. (Wei et al., 2022; Zhang et al., 2023; Wang et al., 2022a).

Intuitively, numerous works extended the CoT capacity to smaller-scale models (Shridhar et al., 2023; Hsieh et al., 2023; Fu et al., 2023; Ma et al., 2023; Chen et al., 2023; Wang et al., 2022b; Ho et al., 2022). Yet, these endeavors predominantly concentrate on generating an increased quantity of high-quality CoT data from LLMs and resort to a direct seq2seq training approach, where the query forms the input and the CoT-enriched answer serves as the target. Such methods often overlook the limited capacity of small models in learning the complex reasoning in CoT. Taking this into account, Some works tried to decompose CoT into more fine-grained reasoning steps, and used LLMs to generate rationales for each individual step to augment the original CoT (Hsieh et al., 2023; Ma et al., 2023; Zhang et al., 2023). Shridhar et al. (2023) further improved the direct seq2seq paradigm using the augmented CoT to train separate models for iterative generation. However, the use of augmented CoT is just another way of generating more CoT data, which is costly and still fails to fully exploit the inherent information in existing datasets.

Furthermore, although proven to achieve better performance, using separate small models for CoT learning raises another question. Similar to using different modules for information retrieval and reasoning in multi-hop QA (Deng et al., 2020; Jiang and Bansal, 2019; Feng et al., 2020; Mavi et al., 2022), these methods specialize small models in single-task operations, in contrast with LLMs using a singular framework. Such implementation underlines an assumption that small models, due to their constrained computational capacity compared to LLMs, may require separate, specialized models to perform distinct phases of CoT processing (Weichert et al., 2019; Fu et al., 2023).

In response to the above issues, we introduce a novel paradigm different from earlier approaches for distilling CoT to small models. We classify the statements inside the CoT-format results of LLMs into two categories: Extractive Segments (ES) that remind the model of the context and set the stage for subsequent conclusions, and Abstractive Segments (AS) that infer additional insights not explicitly stated in the context. With the deconstruction of CoT into AS and ES components, we curate a dataset tailored for an iterative learning process,

[*] Equal Contribution
[†] Corresponding author

Findings of the Association for Computational Linguistics ACL 2024, pages 10686–10697
August 11-16, 2024 ©2024 Association for Computational Linguistics

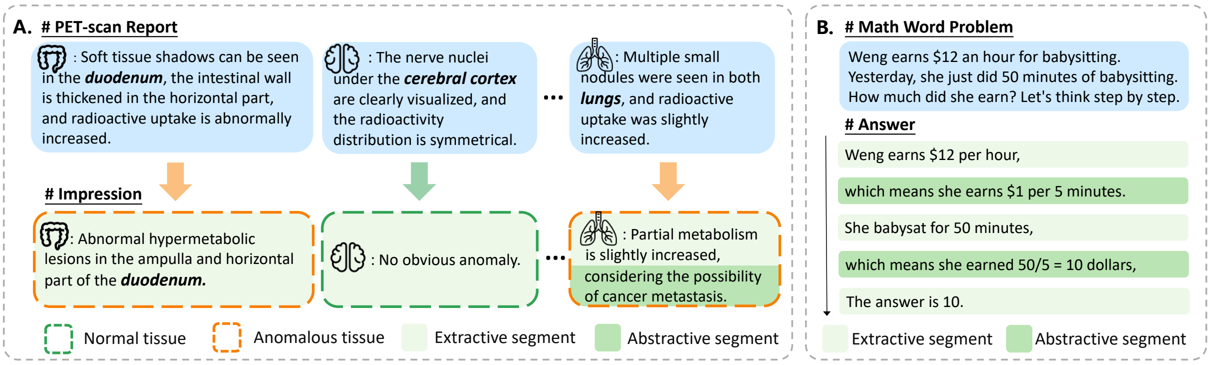

A.
# PET-scan Report
: Soft tissue shadows can be seen in the *duodenum*, the intestinal wall is thickened in the horizontal part, and radioactive uptake is abnormally increased.
: The nerve nuclei under the *cerebral cortex* are clearly visualized, and the radioactivity distribution is symmetrical.
: Multiple small nodules were seen in both *lungs*, and radioactive uptake was slightly increased.
# Impression
: Abnormal hypermetabolic lesions in the ampulla and horizontal part of the *duodenum.*
: No obvious anomaly.
: Partial metabolism is slightly increased, considering the possibility of cancer metastasis.
Normal tissue
Anomalous tissue
Extractive segment
Abstractive segment
B.
# Math Word Problem
Weng earns $12 an hour for babysitting. Yesterday, she just did 50 minutes of babysitting. How much did she earn? Let's think step by step.
# Answer
Weng earns $12 per hour,
which means she earns $1 per 5 minutes.
She babysat for 50 minutes,
which means she earned 50/5 = 10 dollars,
The answer is 10.
Extractive segment
Abstractive segment

Figure 1: **A.** Description of PET-scan summarization task: Each blue section denotes the part of the report involving a particualr organ, with the green representing the related impression. Organ names are ***bolded and italicized***, while light and dark green distinguish between impression segments. **B.** Description of Math Word Problem task: The blue section highlights the question, while light and dark green sections denote distinct segments of the answer.

defined as AS-ES learning to maximize the latent potential of small models for CoT-intensive tasks without the need for additional data.

We further experiment with two training strategies, using two models for generating AS/ES respectively and using one unified model for generating AS/ES together, as an attempt to answer whether a single small model can handle both extraction and abstractive reasoning effectively akin to LLMs, and furthermore, whether the suboptimal performance of small models trained with CoT stems from intrinsic limitations or data utilization inefficiencies.

To cover different scenarios, we take two representative problems – impression generation for PET scan report (PET) and Math Word Problem (MWP). As shown in Figure 1, the CoT of the two tasks expands in different ways: MWP requires a step-by-step solution with CoT following a sequential pattern, while PET involves more parallel processing across multiple body regions.

Here is a paragraph for rebuttal. The first citation is (Huang and Qiao, 2023). And the second one is (Xiong et al., 2023).

In summary, the main contributions of our paper are as follows:

- We introduce AS-ES learning, a novel data-efficient training paradigm that maximizes the intrinsic value of existing CoT data, adaptable across various model sizes and tasks.

- We explore the use of AS-ES dataset and find that the limitations in CoT learning previously attributed to the inherent capabilities of small models can be substantially mitigated through an improved data utilization strategy without additional data.

- We explain the efficacy of AS-ES learning from the loss perspective, offering insights into the underlying dynamics of CoT that may benefit future research in the field.

2 Related Work

Deduction in NLP Tasks Deduction is a logical process of reasoning or inferring specific information from given premises or data. In NLP, deduction is used in various tasks, such as summarization, question answering and information extraction (Mirzaee and Kordjamshidi, 2023; Minervini et al., 2020; Deng et al., 2020; Mavi et al., 2022; Qu et al., 2020; Nye et al., 2021). Early research employs end-to-end models to directly learn reasoning strategies from labeled and structured data (Minervini et al., 2020; Qu et al., 2020), which requires the effort of human annotation. Nye et al. (2021) separates the inference process into a traditional generation part and an extra validation part. They construct a symbolic reasoning module to validate the generated facts using a minimal world model. However, the minimal world model must be hand-engineered. In our work, we employ a fully automatic strategy to disentangle the deduction process, achieving both efficiency and effectiveness.

CoT in Small Models CoT is a significant ability to improve the performance of complex reasoning, which is considered as an emergent ability of LLMs (Wei et al., 2022). Numerous work aims to transfer this ability to small models (Shridhar et al., 2023; Hsieh et al., 2023; Fu et al., 2023; Ma et al., 2023;

Chen et al., 2023). One prevailing method is distilling from LLMs, such as utilizing LLMs to generate rationales or multi-step solutions as training data for smaller models (Shridhar et al., 2023; Hsieh et al., 2023). Chen et al. (2023) also allows LLMs to generate multiple rationales whose consistency is enforced by KL-divergence. Ma et al. (2023) employ a two-stage distillation strategy, in which LLMs not only generate rationale but also provide an answer inference process according to the rationale generated by the student model. However, all these methods require LLMs as a teacher model. In our work, we require no additional data generated by LLMs, which substantially reduces the cost of time and computational resources.

3 Method

This section elucidates the methodology of our study, addressing three pivotal questions: (1) How to segment the complete CoT into AS and ES? (2) How to construct a dataset using the segmented AS and ES? (3) How does the AS-ES learning work for training and generation?

3.1 AS-ES Segmentation

We explore a variety of segmentation techniques based on distinct characteristics of AS and ES. After splitting all sub-sentences S_i in the complete CoT S by punctuation, we calculate the respective score M using different metrics based on different segmentation methods. All segmentation methods except *inter* and *bleu/rouge* follow the criteria below to determine which part falls into AS or ES (β here is a pre-determined hyperparameter).

$$AS = \{S_i | M(S_i) > \beta\overline{M(S)}\} \tag{1}$$

$$ES = \{S_i | M(S_i) <= \beta\overline{M(S)}\} \tag{2}$$

The logic behind the two equations is that, for ES, model generally shows a higher certainty while the certainty about AS is relevantly lower. Therefore, we use the average score of all the sub-sentences as the borderline. Sub-sentences above this borderline will be AS and below will be ES. To give a finer adjustment about the borderline, we multiply the average score by β.

We next introduce the specific segmentation strategies we experiment with. Each strategy is denoted by an abbreviation that will be used subsequently to identify the segmentation applied.

Entropy-oriented Segmentation (*ent/ent)** Entropy measures uncertainty, and is intrinsically linked to the cross-entropy loss commonly employed in seq2seq training. Denote the input query as Q and the generated response as R. We can approximate $P(R|Q)$ by Equation 3, and calculate the entropy for the response as in Equation 4.

$$P(R|Q) \approx softmax(logits(R|Q)) \tag{3}$$

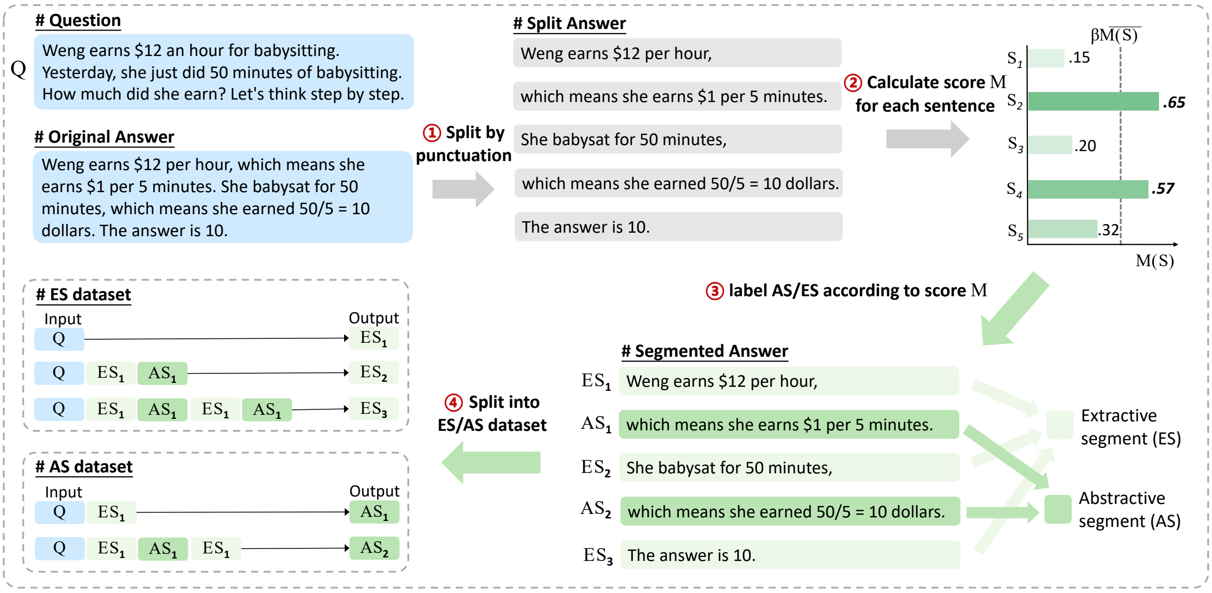
$$H_n = - \sum_{\forall i \in |V|} P(R_i|Q) \log P(R_i|Q) \tag{4}$$

The rationale is that trained models should exhibit greater certainty regarding the ES, which closely mirrors the input query, thereby resulting in lower entropy compared to AS. This is a way that measures the "extractive" and "abstractive" from the perspective of a model rather than a human one. The two can diverge largely from each other, and the latter one is almost impossible due to the large cost of human annotation. For comparison, we also employ this segmentation using pre-trained but not fine-tuned models, designated as ent*.

Location-oriented Segmentation (*inter*) Many CoTs naturally fall into an $ES|AS|ES|AS|...$ pattern, especially where in-depth and step-by-step reasoning is required. Each sentence typically presents a complete reasoning step, beginning with context followed by deduction. We exploit this pattern, designating sub-sentences as ES and AS in an interleaving fashion.

Loss-oriented Segmentation (*loss*) This approach is akin to entropy-oriented segmentation, where the loss of the trained model is used to estimate its certainty about a sub-sentence. We hypothesize that the efficacy of AS-ES learning may be attributed to its ability to lower the loss boundary, as will be shown in Section 5.3. Therefore, segmenting based on loss could potentially enable the model to concentrate more effectively on less well-understood segments.

Similarity-oriented Segmentation (*bleu/rouge*) ES typically derives directly from the original context, sharing greater similarity with the query, whereas AS, often involves new reasoning, making it more different from the query. We utilize BLEU and ROUGE scores to quantify the similarity between the query and the CoT segments. The



# Question
Q
Weng earns $12 an hour for babysitting. Yesterday, she just did 50 minutes of babysitting. How much did she earn? Let's think step by step.
# Original Answer
Weng earns $12 per hour, which means she earns $1 per 5 minutes. She babysat for 50 minutes, which means she earned 50/5 = 10 dollars. The answer is 10.
① Split by punctuation
# Split Answer
Weng earns $12 per hour,
which means she earns $1 per 5 minutes.
She babysat for 50 minutes,
which means she earned 50/5 = 10 dollars.
The answer is 10.
② Calculate score M for each sentence
βM(S)
S1 .15
S2 .65
S3 .20
S4 .57
S5 .32
M(S)
③ label AS/ES according to score M
# Segmented Answer
ES1 Weng earns $12 per hour,
AS1 which means she earns $1 per 5 minutes.
ES2 She babysat for 50 minutes,
AS2 which means she earned 50/5 = 10 dollars.
ES3 The answer is 10.
Extractive segment (ES)
Abstractive segment (AS)
④ Split into ES/AS dataset
# ES dataset
Input
Output
Q → ES1
Q ES1 AS1 → ES2
Q ES1 AS1 ES1 AS1 → ES3
# AS dataset
Input
Output
Q ES1 → AS1
Q ES1 AS1 ES1 → AS2

Figure 2: The workflow for labeling raw data as either ES or AS parts, followed by constructing ES/AS datasets.

classification of AS/ES using *bleu/rouge* segmentation follows the equations below, the opposite of Equation 1 and 2.

$$AS = \{S_i | M(S_i) < \beta\overline{M(S)}\} \qquad (5)$$

$$ES = \{S_i | M(S_i) >= \beta\overline{M(S)}\} \qquad (6)$$

3.2 AS-ES Dataset Construction

Training Data Organization Mere segmentation into AS and ES is insufficient for effective model training, as the formats of input and target of the training data play an important role in the training results. After segmenting the original targets into AS and ES, we construct the AS-ES dataset used for AS-ES learning. As shown in Figure 2, we first merge the adjacent AS or ES as one, and then structure the data as $Q|ES_1AS_1...ES_{i-1}AS_{i-1} \rightarrow ES_i$ and $Q|ES_1AS_1...ES_{i-1}AS_{i-1}ES_i \rightarrow AS_i$. Both the ES dataset and AS dataset are compiled from samples generating ES and AS respectively.

Stop Sign Another question is when to put a stop to iterative generation. We employ a stop sign at the end of sequences generating the final sentence of the original CoT. In this way, the loop can stop once this stop sign is detected during generation. For MWP, the conclusive phrase "the answer is ..." serves as this marker. In PET summarization where no inherent stop sign exists, we introduce a special token $<$ STOP $>$ as the end of each CoT.

Irrelevant Information Processing One problem of the AS-ES dataset is that it is rather lengthy compared to the original dataset, which often leads to the problem of exceeding the maximum input length of the model. PET summarization serves as a good example. Due to the limitation of the input length and consideration and the training cost, we first divide whole-body PET scans into sections according to anatomical regions, with each section treated as an independent CoT instance. Unlike MWP, where the entire context provided in the query is integral to formulating the output, PET scan reports typically contain extensive normality, which are generally not included in the final impression. To address this discrepancy, segments depicting normal findings are annotated with "No obvious anomaly" as the ground truth, and then incorporated proportionally to ES-dataset. This enables the ES model to identify and selectively extract sentences to the final summary. The proportion of normality incorporated is denoted by γ, which is the ratio of normal findings included compared to the total number of PET reports.

3.3 AS-ES Learning

Dual-path Learning Existing work that uses iterative generation for CoT learning in small models uses two separate models, one for heuristic questioning and the other for answering. AS-ES training can also adapt to this paradigm by training two separate models, Extractive Segment generation Model (ESM) for retrieving and Abstractive Segment generation Model (ASM) for reasoning, together designated as Dual-path Segment generation Models (DSM). This training paradigm reflects an underlying assumption that a singular small model may struggle with the complexity of performing both extraction and reasoning tasks within an iterative sequence. The generation process under the DSM framework, as depicted in Algorithm 1, mirrors the procedural logic of the AS-ES dataset construction. The training set of ASM and ESM can be denoted as follows.

$$tr(ASM) = \{D_{AS}\} \tag{7}$$

$$tr(ESM) = \{D_{ES}\} \tag{8}$$

Uni-path Learning While the dual-path learning approach has demonstrated efficacy, it necessitates the training and maintenance of two separate models, each with its independent parameter space, complicating the training process and inflating computational costs. In response to these challenges, we propose the uni-path learning framework, a more streamlined method that consolidates all AS-ES data into a single model. The training set for the Uni-path Segment generation Model (USM) is thus a combined dataset:

$$tr(USM) = \{D_{AS}, D_{ES}\} \tag{9}$$

Algorithm 1 Dual-Path Generation Process

```
Require: ESM, ASM ▷ Two models gained from
    dual-path learning
input ← Start token
while True do
    ESM_output ← ESM(input)
    input ← input + ESM_output
    if stop_sign in ESM_output then
        output ← input return output
    end if
    ASM_output ← ASM(input)
    input ← input + ASM_output
    if stop_sign in ESM_output then
        output ← input return output
    end if
end while
```

As detailed in Algorithm 2, USM undertakes the identical iterative generation procedure as DSM in Algorithm 1, except that the singular model assumes the responsibilities of both ASM and ESM. This unified approach posits that a single model, given an effective training strategy, can successfully navigate both extractive and abstractive tasks, thereby simplifying the learning process and reducing the requisite resources for model training.

Algorithm 2 Uni-Path Generation Process

```
Require: USM          ▷ The model gained from
    uni-path learning
input ← Start token
while True do
    USM_output ← USM(input)
    input ← input + USM_output
    if stop_sign in USM_output then
        output ← input return output
    end if
end while
```

4 Experiment

4.1 Dataset

Math Word Problems (MWP) For the MWP task, we employ the dataset curated by Fu et al. (2023), which consists of chain-of-thought data generated by the code-davinci-002 model from OpenAI. The original questions are sourced from the GSM8K dataset. Fu et al. enhanced the dataset by prepending four in-context examples to each question to serve as prompts for the GPT model. In our study, we extract only the original questions and their corresponding answers augmented with CoT as our dataset.

PET Report Summarization (PET) For the PET task, we introduce the cPET-11K dataset, a novel compilation of 11.6k Chinese PET/CT report-impression pairs. This dataset is a collection of PET/CT report data from patients with pancreatic cancer, originating from real clinical data of a major tertiary hospital. These PET/CT scans were performed using three PET/CT machines. All patient reports have been anonymized, with only the content of the reports retained. These reports focus on determining whether the patient has pancreatic cancer and whether there is distant metastasis of the pancreatic cancer to other abdominal organs.

4.2 Implementation

Base Model We use two variants of the typical seq2seq model T5 (Raffel et al., 2020). For MWP, we follow Fu et al. (2023) and use Flan-T5 (Chung et al., 2022). For PET impression generation, we

Metric	Para	Model				
		direct	DSM(*ent*)	USM(*ent*)	DSM(*inter*)	USM(*inter*)
BLEU	77M	**31.39**	29.18 (↓7.04%)	28.53 (↓9.11%)	28.27 (↓9.94%)	28.93 (↓7.84%)
	250M	**34.65**	30.87 (↓10.91%)	31.12 (↓10.19%)	30.63 (↓11.60%)	29.82 (↓13.94%)
	800M	**33.53**	31.04 (↓7.43%)	31.21 (↓6.92%)	30.08 (↓10.29%)	29.65 (↓11.57%)
Acc	77M	15.25	13.74 (↓9.90%)	16.76 (↑9.90%)	13.19 (↓13.51%)	**17.58** (↑15.28%)
	250M	19.23	20.60 (↑ 7.12%)	20.74 (↑7.85%)	20.60 (↑7.12%)	**21.70** (↑12.84%)
	800M	22.53	24.18 (↑7.32%)	**24.86** (↑10.34%)	21.29 (↓5.50%)	21.57 (↓4.26%)

Table 1: Results of different models for Math Word Problem. The best scores and improvements are in **bold**.

Metric	Model				
	direct	DSM(*ent*)	USM(*ent*)	DSM(*inter*)	USM(*inter*)
BLEU	2.17	3.29 (↑51.61%)	**3.83** (↑76.49%)	0.37 (↓82.94%)	0.67 (↓69.12%)
ROUGE-L	17.15	22.25 (↑29.73%)	**23.96** (↑39.70%)	7.39 (↓56.90%)	8.38 (↓51.13%)
MR	39.34	28.4 (↓27.80%)	**24.78** (↓37.01%)	58.87 (↑49.64%)	33.04 (↓16.01%)

Table 2: Results of different models for PET-scan ummarization. The best scores are in **bold**. Lower MR metric indicates better performance.

use mt5 (Xue et al., 2020) for its multilingual capabilities. We conduct experiments with three different sizes of Flan-T5 (small, base, large) to investigate the impact of model size on the efficacy of AS-ES learning. Unless otherwise specified, the default models used are Flan-T5-base for MWP and mT5-base for PET impression generation.

Training Process As the size of the AS-ES dataset is usually bigger than the original dataset, to ensure a fair comparison among different methods, all uni-Path approaches are trained with the same amount of batch size and learning rate given the same amount of training time. Conversely, dual-path approaches undergo training for around half the duration per model in accordance with their data amount.

Checkpoint Selection Considering the final generated results can not be directly obtained during evaluation for iterative approaches like AS-ES learning, traditional evaluation metrics like BLEU score or validation loss may not directly correlate with actual model performance. Furthermore, even for a direct approach, a higher BLEU score or lower loss does not necessarily lead to a higher accuracy for MWP. As for PET scan impression generation, the BLEU score can be significantly influenced by variations in formatting, necessitating additional post-processing steps that can alter the metric's relevance to actual performance.

To address the above issues, we select three different checkpoints for each approach, and report the best performance among the three. The detailed criteria are as follows. (1) best_train: the model with the lowest loss on the training set. (2) best_loss: the model with the lowest loss on the validation set. (3) best_bleu: the model with the highest BLEU on the validation set. For MWP, accuracy is deemed the primary performance indicator, while for PET, we prioritize BLEU scores.

$$MR = \frac{1}{N}\sum_{i=1}^{N}\left(\frac{|R_{\mathrm{GT}_i} - (R_{\mathrm{GT}_i} \cap R_{\mathrm{GR}_i})|}{|R_{\mathrm{GT}_i}|}\right) \tag{10}$$

Evaluation Metrics For MWP, we use BLEU (Papineni et al., 2002) and accuracy as the evaluation metrics with a focus on accuracy. For PET, we use BLEU, ROUGE (Lin, 2004) and MR (missing ratio) as the evaluation metrics with a focus on BLEU. MR is calculated using Equation 10, assessing the proportion of anomalies omitted in the generated summaries. Here, R denotes the various organs or regions identified via keyword mapping, GT and GR represent the ground truth and generated results respectively, and N is the size of the test set.

4.3 Overall Performance of AS-ES Learning

As AS-ES learning has multiple combinations of segmentation strategy and training strategy, we here report the most effective segmentation for the two tasks respectively, and apply both USM and DSM for the two segmentation strategies. As shown in Table 1 and Table 2, AS-ES learning improves the model performance on both tasks with the appropriate strategy. Entropy-oriented segmentation shows a generalizability across different model sizes and tasks, while interleaving segmentation is more targeted.

We here summarize some key aspects in the experiments as a summary of the characteristics of AS-ES learning, as well as an attempt to answer the question mentioned in Section 1, that is, whether the inferiority of small models in CoT-related tasks stem from its inherent incapability to do extraction and reasoning with a singular model.

4.3.1 Lower BLEU with Higher Accuracy

The first thing to notice in Table 1 is that, although all AS-ES learning leads to lower BLEU score compared to the direct approach, most of them leads to a higher accuracy. This may implicate that instead of memorizing the solution provided in the training set in direct approach, AS-ES learning enables logical reasoning of higher granularity in small models through a literal step-by-step process. Models trained through AS-ES learning could yield different solution and therefore although differs from the ground truth in terms of textual similarity, still leads to the same correct results.

4.3.2 Different Model Size Reacts Differently to Segmentation Strategy

As shown in Table 1, entropy-oriented segmentation works best for Flan-T5-large while interleaving segmentation works best for Flan-T5-small. The reasons behind this could be two-fold. First, entropy calculation is the key for entropy-oriented segmentation. The inherent superiority of larger models after fine-tuning results in a better grasp of data, leading to a better segmentation. Furthermore, when it comes to task of highly sequential CoT solution like MWP, entropy-oriented segmentation does not seem that straightforward and easy for small models to grasp, especially with the suboptimal entropy calculation by small models. Overall, the specific performance of AS-ES learning depends on both the quality of AS-ES dataset and the capacity of models to handle to data.

4.3.3 Segmentation Works Differently for Different Tasks

As shown in Table 1 and Table 2, PET scan summarization and MWP show rather large divergence using entropy segmentation and interleaving segmentation. The former benefits largely from entropy segmentation while suffers a lot from interleaving segmentation. On the contrary, interleaving segmentation works for most cases in MWP, depending on the size of the model. The reason behind this is quite intrinsic. The structure of math word problems typically follows a logical progression where statements provide context or premises (ES) followed by a step in reasoning or calculation (AS), and this pattern tends to repeat as the problem is broken down into solvable parts therefore naturally aligns with the interleaving segmentation approach. Impression generation from PET scans, on the other hand, tend to consist of multiple observations (ES) followed by a collective diagnostic insight (AS), or vice versa without an interleaving pattern, therefore better accommodates to entropy segmentation.

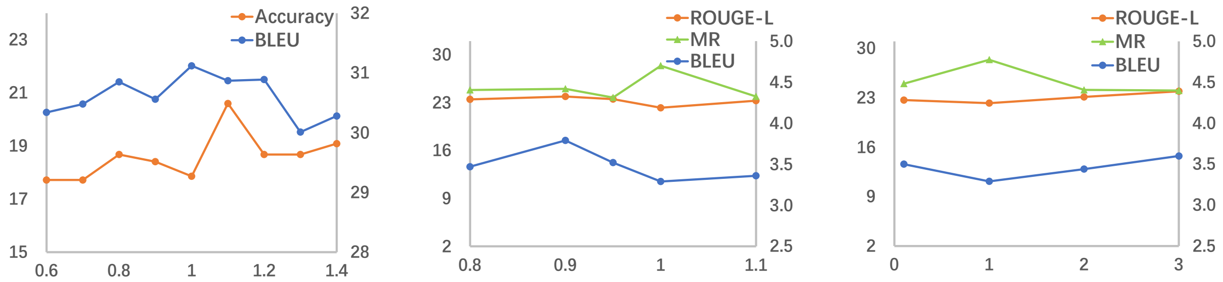
4.3.4 USM v.s. DSM: Is One Model Enough?

Although existing works all use two separate models for iterative generation approach, experiments results for both MWP and PET suggest that one model is enough and lead to even better results compared to using two models, at least for AS-ES learning. Although both suffering from error accumulation, USM consistently outperforms direct approach compared to DSM. The integrated context for training USM might enable the model to better understand the interplay between different types of reasoning, leading to a more nuanced understanding of the data. Furthermore, the separate model in DSM leads to separate optimization, making the checkpoint used for test may not be fully display its capacity, even with multiple checkpoint selection strategy. This phenomenon suggests that it is more about the training strategy but rather the capacity of model itself.

5 Discussion

5.1 Effect of Segmentation Strategy

As shown in Table 3, different segmentation strategies do play a key role in AS-ES learning. segmentation by *inter*, *loss* and *ent* calculated from fine-tuned models all yield better results compared to the direct approach. The decreased performance using *bleu/rouge* segmentation indicates that simply segmenting AS/ES by its textual similarity (as what is intrinsic to humans) is not the same way for models. Furthermore, using entropy calculated by



(a) DSM for MWP with different β

(b) USM for PET with different β

(c) USM for PET with different γ.

Figure 3: results of different hyperparameter settings for AS-ES learning. The BLUE metric values (blue line) correspond to the right Y-axis (secondary axis). γ is set to 1.0 in Figure 3b, while β is set to 1.0 in Figure 3c.

Segment	BLEU	Accuracy
baseline	**34.65**	19.23
inter	30.62 (↓11.63%)	**20.60** (↑7.12%)
ent	30.87 (↓10.90%)	**20.60** (↑7.12%)
*ent**	30.88 (↓10.88%)	18.96 (↓1.40%)
bleu	29.97 (↓13.50%)	17.72 (↓7.85%)
rouge	27.46 (↓20.75%)	16.21 (↓15.70%)
loss	30.34 (↓12.43%)	20.19 (↑4.99%)

Table 3: Results of DSM for MWP using different segmentation method. Baseline refers to direct approach. The best scores are in **bold**.

pre-trained but not fine-tuned models introduces no further improvement, which makes sense since pre-trained models have less grasp about which part is extractive/abstractive.

5.2 Effect of Hyperparameters

Most segmentation methods introduce a hyperparameter β as in Equation 1 and 2, and PET summarization introduces a hyperparameter of γ which evaluates the amount of incorporated normality. Here we evaluate how these hyperparameters affect different training strategies for different tasks.

Effect of Ratio β As shown in Figure 3a and Figure 3b, entropy segmentation for both USM and DSM for different tasks has a β threshold (around 1) where the model achieves the best balance between ES and AS.

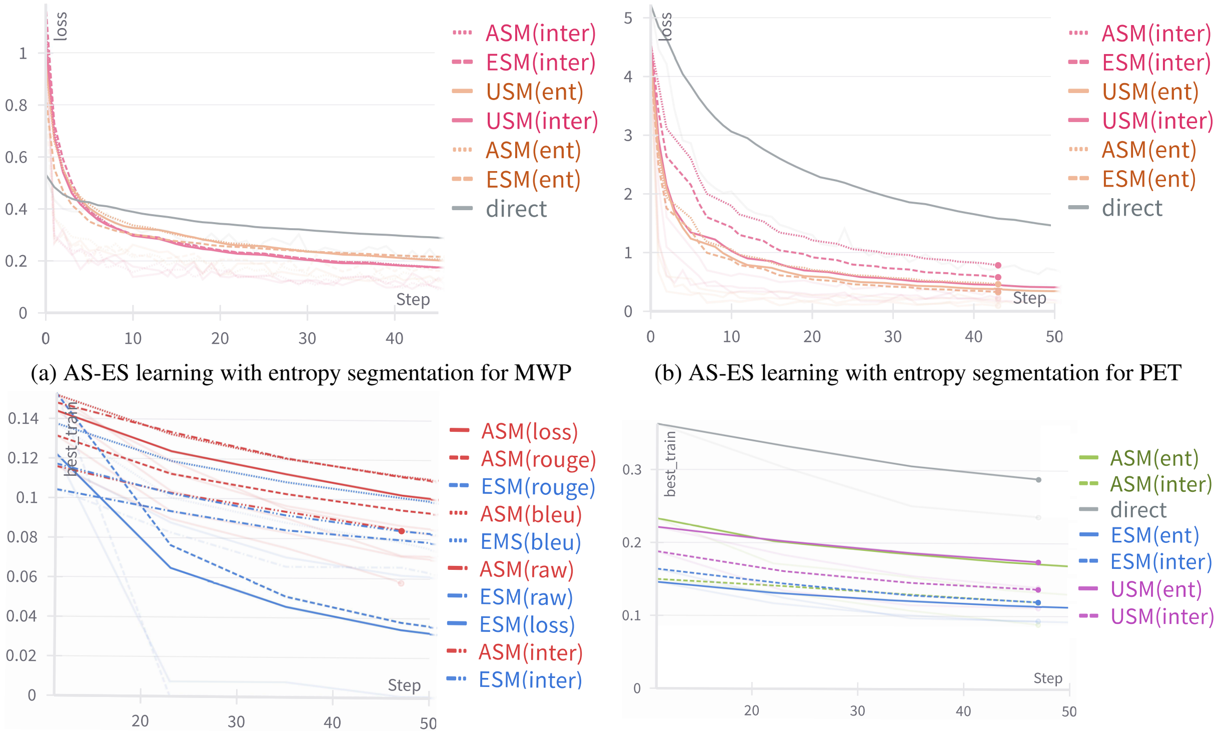
Effect of Ratio γ As shown in Figure 3c, γ at 1.0 reaches the lowest point on the curve, which might be a tipping point where the inclusion of normal findings is enough to dilute the model's focus on anomalies without providing the additional contextual benefits seen at higher γ values. Overall speaking, the model performance is less affected by γ compared to segmentation and training strategy.

5.3 Why AS-ES Learning Works?

We here further explore the underlying mechanism of AS-ES learning to see why it works from a different perspective other than just performance and intuition. As shown in Figure 4c, the best training loss boundary of ESM is generally lower than ASM, which aligns with our hypothesis that extraction comes more easily while logical reasoning is not quite so. Figure 4d shows the best training loss boundary for the direct approach, USM, ASM and ESM respectively. As expected, the lowest loss boundary for the direct approach is significantly larger than AS-ES learning, which partly explains why AS-ES learning works. The divergence of MWP and PET to segmentation strategy could also be explained from this perspective. As shown in Figure 4a and 4b, interleaving segmentation generally reaches a lower boundary than entropy segmentation for MWP therefore resulting in a better performance using interleaving segmentation, and vice versa for PET. The overall discovery suggests that AS-ES learning works by achieving a generally lower loss boundary compared to the direct approach.

6 Conclusion

In this paper, we introduce a data-efficient CoT distillation strategy for small models. By segmenting CoT data into extractive part and abstractive part respectively, we improve the model performance through an iterative generation approach without incorporating additional data. The generalization of AS-ES learning to different model sizes and tasks shows its effectiveness. We further explore whether two models are necessary for this interactive generation approach, and answer the derivative question that the limitation of small models in CoT learning stems from the training paradigm instead



(a) AS-ES learning with entropy segmentation for MWP

(b) AS-ES learning with entropy segmentation for PET

(c) ASM/ESM with different segmentation for MWP

(d) USM/ASM/ESM with entropy segmentation for MWP

Figure 4: Smoothed training curve for AS-ES Learning. In Figure 4a and Figure 4b, grey, pink and yellow denote direct approach, interleaving segmentation and entropy segmentation respectively. Figure 4c and Figure 4d show the curve of the best training loss among time. All curves for ASM are displayed in red with ESM in bleu in Figure 4c. In Figure 4d, curves for ASM, ESM, USM are displayed in green, blue and pink, respectively.

of its inherent capacity, providing insights into the underlying mechanism of CoT.

Limitations

In this paper, we mainly experiment and discuss the use of AS-ES Learning based on the direct training approach. Although experiments show that the use of one single model is better than two separate models, this may not be the case when there is a solution for DSM to be simultaneously trained. Furthermore, although we come up with a general range of the hyperparameter settings for AS-ES learning, the specific optimal settings of AS-ES Learning on different datasets may vary and therefore require a need for specific exploration.

Ethical Considerations

In this work, we introduce a new Chinese PET report-impression dataset. The data collection protocol is approved by an ethics review board. All experimental datasets involved have been de-identified by dataset providers and used for research only.

Acknowledgement

We gratefully acknowledge the support of the National Science and Technology Major Project (2021ZD0113301, 2021ZD0113302), the National Natural Science Foundation of China Youth Fund (62206079), and the Heilongjiang Provincial Natural Science Foundation of China (YQ2022F006).

References

Hongzhan Chen, Siyue Wu, Xiaojun Quan, Rui Wang, Ming Yan, and Ji Zhang. 2023. Mcc-kd: Multi-cot consistent knowledge distillation. *arXiv preprint arXiv:2310.14747*.

Hyung Won Chung, Le Hou, Shayne Longpre, Barret Zoph, Yi Tay, William Fedus, Yunxuan Li, Xuezhi Wang, Mostafa Dehghani, Siddhartha Brahma, et al. 2022. Scaling instruction-finetuned language models. *arXiv preprint arXiv:2210.11416*.

Yang Deng, Wenxuan Zhang, and Wai Lam. 2020. Multi-hop inference for question-driven summarization. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 6734–6744.

Yanlin Feng, Xinyue Chen, Bill Yuchen Lin, Peifeng Wang, Jun Yan, and Xiang Ren. 2020. Scalable multi-hop relational reasoning for knowledge-aware question answering. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 1295–1309, Online. Association for Computational Linguistics.

Yao Fu, Hao Peng, Litu Ou, Ashish Sabharwal, and Tushar Khot. 2023. Specializing smaller language models towards multi-step reasoning. *arXiv preprint arXiv:2301.12726*.

Namgyu Ho, Laura Schmid, and Se-Young Yun. 2022. Large language models are reasoning teachers. *arXiv preprint arXiv:2212.10071*.

Cheng-Yu Hsieh, Chun-Liang Li, Chih-Kuan Yeh, Hootan Nakhost, Yasuhisa Fujii, Alexander Ratner, Ranjay Krishna, Chen-Yu Lee, and Tomas Pfister. 2023. Distilling step-by-step! outperforming larger language models with less training data and smaller model sizes. *arXiv preprint arXiv:2305.02301*.

Mouxiao Huang and Yu Qiao. 2023. Uncertainty-estimation with normalized logits for out-of-distribution detection. In *International Conference on Computer, Artificial Intelligence, and Control Engineering (CAICE 2023)*, volume 12645, pages 524–530. SPIE.

Yichen Jiang and Mohit Bansal. 2019. Self-assembling modular networks for interpretable multi-hop reasoning. *arXiv preprint arXiv:1909.05803*.

Chin-Yew Lin. 2004. Rouge: A package for automatic evaluation of summaries. In *Text summarization branches out*, pages 74–81.

Yuhan Ma, Haiqi Jiang, and Chenyou Fan. 2023. Sci-cot: Leveraging large language models for enhanced knowledge distillation in small models for scientific qa. *arXiv preprint arXiv:2308.04679*.

Vaibhav Mavi, Anubhav Jangra, and Adam Jatowt. 2022. A survey on multi-hop question answering and generation. *arXiv preprint arXiv:2204.09140*.

Pasquale Minervini, Sebastian Riedel, Pontus Stenetorp, Edward Grefenstette, and Tim Rocktäschel. 2020. Learning reasoning strategies in end-to-end differentiable proving. In *International Conference on Machine Learning*, pages 6938–6949. PMLR.

Roshanak Mirzaee and Parisa Kordjamshidi. 2023. Disentangling extraction and reasoning in multi-hop spatial reasoning. *arXiv preprint arXiv:2310.16731*.

Maxwell Nye, Michael Tessler, Josh Tenenbaum, and Brenden M Lake. 2021. Improving coherence and consistency in neural sequence models with dual-system, neuro-symbolic reasoning. *Advances in Neural Information Processing Systems*, 34:25192–25204.

Kishore Papineni, Salim Roukos, Todd Ward, and Wei-Jing Zhu. 2002. Bleu: a method for automatic evaluation of machine translation. In *Proceedings of the 40th annual meeting of the Association for Computational Linguistics*, pages 311–318.

Adam Paszke, Sam Gross, Francisco Massa, Adam Lerer, James Bradbury, Gregory Chanan, Trevor Killeen, Zeming Lin, Natalia Gimelshein, Luca Antiga, et al. 2019. Pytorch: An imperative style, high-performance deep learning library. *Advances in neural information processing systems*, 32.

Meng Qu, Junkun Chen, Louis-Pascal Xhonneux, Yoshua Bengio, and Jian Tang. 2020. Rnnlogic: Learning logic rules for reasoning on knowledge graphs. *arXiv preprint arXiv:2010.04029*.

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. *Journal of Machine Learning Research*, 21(140):1–67.

Kumar Shridhar, Alessandro Stolfo, and Mrinmaya Sachan. 2023. Distilling reasoning capabilities into smaller language models. In *Findings of the Association for Computational Linguistics: ACL 2023*, pages 7059–7073.

Xuezhi Wang, Jason Wei, D. Schuurmans, Quoc Le, E. Chi, and Denny Zhou. 2022a. Self-consistency improves chain of thought reasoning in language models. *ArXiv*, abs/2203.11171.

Yueqian Wang, Chang Liu, Kai Chen, Xi Wang, and Dongyan Zhao. 2022b. SMASH: Improving SMAll language models' few-SHot ability with prompt-based distillation. In *Findings of the Association for Computational Linguistics: EMNLP 2022*, pages 6608–6619, Abu Dhabi, United Arab Emirates. Association for Computational Linguistics.

Jason Wei, Xuezhi Wang, Dale Schuurmans, Maarten Bosma, Fei Xia, Ed Chi, Quoc V Le, Denny Zhou, et al. 2022. Chain-of-thought prompting elicits reasoning in large language models. *Advances in Neural Information Processing Systems*, 35:24824–24837.

D. Weichert, Patrick Link, Anke Stoll, S. Rüping, S. Ihlenfeldt, and S. Wrobel. 2019. A review of machine learning for the optimization of production processes. *The International Journal of Advanced Manufacturing Technology*, 104:1889 – 1902.

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

Miao Xiong, Zhiyuan Hu, Xinyang Lu, Yifei Li, Jie Fu, Junxian He, and Bryan Hooi. 2023. Can llms express their uncertainty? an empirical evaluation of confidence elicitation in llms. *arXiv preprint arXiv:2306.13063*.

Linting Xue, Noah Constant, Adam Roberts, Mihir Kale, Rami Al-Rfou, Aditya Siddhant, Aditya Barua, and Colin Raffel. 2020. mt5: A massively multilingual pre-trained text-to-text transformer. *arXiv preprint arXiv:2010.11934*.

Zhuosheng Zhang, Aston Zhang, Mu Li, Hai Zhao, G. Karypis, and Alexander J. Smola. 2023. Multimodal chain-of-thought reasoning in language models. *ArXiv*, abs/2302.00923.

A Details of Implementation

Our model utilizes the Pytorch-based (Paszke et al., 2019) Huggingface Transformers (Wolf et al., 2020) packages. All experiments are conducted with the same batch size of 64. The learning rate is set to 5e-4 for MWP and 1e-3 for PET using early stopping. Both MWP and PET datasets are split into *train*, *validation*, *test* sets with a ratio of 80%, 10%, and 10% respectively. The results reported are the average of three separate runs. Most experiments were conducted on NVIDIA A100-80GB-PCIe GPUs or A100-SXM4-80GB GPUs, some were conducted on Tesla V100S-PCIE-32GB GPUs. Code and dataset are available at https://github.com/rootnx/AS-ES-learning/.

B Human Evaluation of AS-ES Dataset

AS-ES learning initially gains inspirations from multi-hop QA. Following the idea of separating retrieval and reasoning, ES is considered to be extractive and AS is considered to be deductive. However, do the automatic AS-ES segmentation methods share the same taste as human? To answer this question, we randomly sample 100 different question-answer pairs for human evaluation. We annotate each sub-sentence in the answer with "AS" or "ES" and compare with the results of different segmentation strategy. *ent* shows high average matching rate compared to human evaluation (76.90%), while *inter* does not (51.90%). Although this may contradict with the improvement in small (77M) USM(inter) and base(250M) DSM(inter)/USM(inter), we further find that CoT with high matching rate (>75%) under *inter* segmentation makes up about half of the dataset (49%). These questions tend to have simpler and more structured solution. This indicates that for models with smaller scales, *inter* improves their performance on these relatively "simple" problems, leading to its performance gain; while for large models, these "simple" problems can be already solved using direct seq2seq training, thus remain comparable performance.

C Case Study of Model Outputs

Here we demonstrate a few cases to give a general impression about how the outputs of AS-ES learning differs from the direct approach.

We find that the direct approach tends to miss steps or add additional but erroneous step. In some complicated question, the direct approach could lose its logic during the process. All these issues could be alleviated in AS-ES learning.

Furthermore, the AS-ES learning does not necessarily use the same solution as the ground truth answer, which aligns with our findings in Section 4.3.1, suggesting the more diverse solution generated by AS-ES learning, rather than simply remembering and copying solution following a fixed pattern.

Question	Ground Truth Answer	Direct	AS-ES Learning
Marcus is having a water balloon party. He has 100 balloons. Each balloon holds 3 ounces of water. He can buy 50 ounces of water for $2.5 a bottle. If he walks into the store with 2 $10 bills, how much change will he have after he buys all the water he needs? Let's think step by step	Marcus needs 100 balloons x 3 ounces of water = 300 ounces of water. He can get 50 ounces of water for $2.5 a bottle, so he needs to buy 300 / 50 = 6 bottles of water. Each bottle of water costs $2.5, so he needs to spend 6 x $2.5 = $15 for water. Marcus has $10 x 2 = $20 to spend. He will have $20 - $15 = $5 in change. The answer is 5.	Marcus has 100 balloons, each holding 3 ounces of water. So, he needs 100 x 3 = 300 ounces of water. He can buy 50 ounces of water for $2.5 a bottle. So, he needs 300/50 = 6 bottles of water. He has 2 $10 bills, so he has $10 + $10 = $20. He will have $20 - $6 = $14 in change. The answer is 14.	Marcus has 100 balloons, Each balloon holds 3 ounces of water, So, he needs 100 x 3 = 300 ounces of water, He can buy 50 ounces of water for $2.5 a bottle, So, he needs 300/50 = 6 bottles of water, Each bottle costs $2.5, So, he needs 6 x $2.5 = $15 worth of water, He walks into the store with 2 $10 bills, So, he has $10 + $10 = $20, He will have $20 - $15 = $5 in change, The answer is 5.
Ben's hockey team is 60% boys and the rest are girls. Half the girls are juniors and the other half are seniors. If the team has 50 players, how many junior girls are there? Let's think step by step	Of the 50 players, the team is 60% boys, so there are 50 x 0.6 = 30 boys. This means there are 50 - 30 = 20 girls. Of these 20 girls, half are juniors and half are seniors, so there are 20/2 = 10 junior girls. The answer is 10.	The team is 60% boys and 40% girls. There are 50 players, so there are 50 * 0.6 = 30 boys. There are 50 - 30 = 20 girls. Half of the girls are juniors, so there are 20 * 0.5 = 10 junior girls. The other half are seniors, so there are 20 * 0.5 = 10 seniors. There are 10 + 10 = 20 junior girls. The answer is 20.	The team is 60% boys, so 60% of 50 = 30 boys, The rest of the team is girls, so 50 - 30 = 20 girls, Half the girls are juniors, so half of 20 = 10 junior girls, The other half of the team are seniors, so the other half of the team is 20 - 10 = 10 seniors, The answer is 10.
A portable computer drive has enough kilobytes of storage space to store 2000 photos. Each photo is 1.5 kilobytes in size. How many 200-kilobyte videos can it store if it already has 400 photos on it? Let's think step by step	Since the drive has 2000 photos, it has 2000 * 1.5 = 3000 kilobytes of space. It has 400 photos, so 400 * 1.5 = 600 kilobytes of space are already used. This means that there are 3000 - 600 = 2400 kilobytes of free space. The videos are 200 kilobytes each, so 2400 / 200 = 12 videos can be stored. The answer is 12.	The computer drive has enough storage space to store 2000 photos. Each photo is 1.5 kilobytes in size. So, 2000 photos will take 2000 x 1.5 = 3000 kilobytes. The computer drive already has 400 photos on it. So, it can store 3000 - 400 = 2400 kilobytes more. Each 200-kilobyte video will take 200/1.5 = 160 kilobytes. So, the computer drive can store 2400/160 = 10 200-kilobyte videos. The answer is 10.	The computer drive has enough kilobytes of storage space to store 2000 photos, Each photo is 1.5 kilobytes in size, So, 2000 x 1.5 = 3000 kilobytes of storage space can be stored, The computer drive already has 400 photos on it, So, 3000 - 400 x 1.5 = 3000 - 600 = 2400 kilobytes of storage space can be stored, Each 200-kilobyte video is 200 kilobytes in size, So, 2400 / 200 = 12 200-kilobyte videos can be stored, The answer is 12.

Table 4: Case study of model outputs.