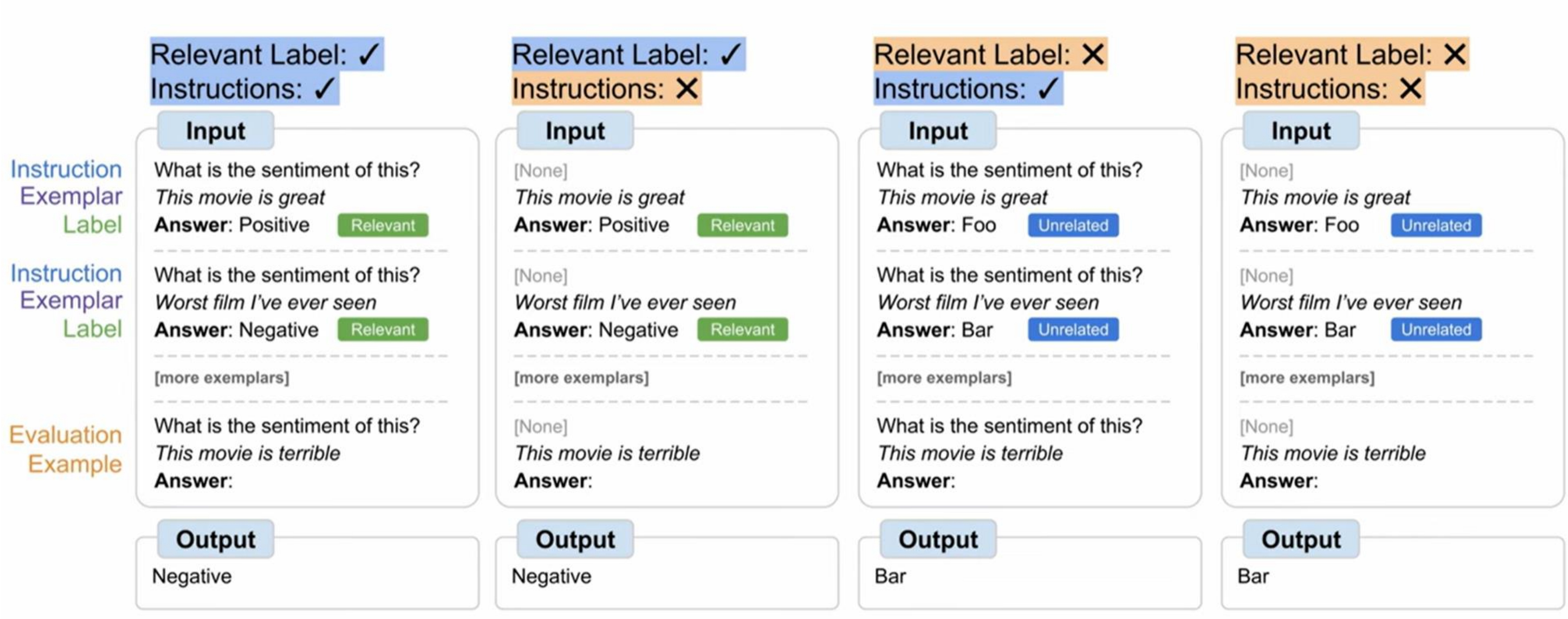
In-Context Learning

Instruction Fine-Tuning & In-Context Learning

- (Instruct FT) & (ICL)

• Instruction	What is the sentiment of this?
• Exemplar	The movie is great
• Label	Positive | 1 | +

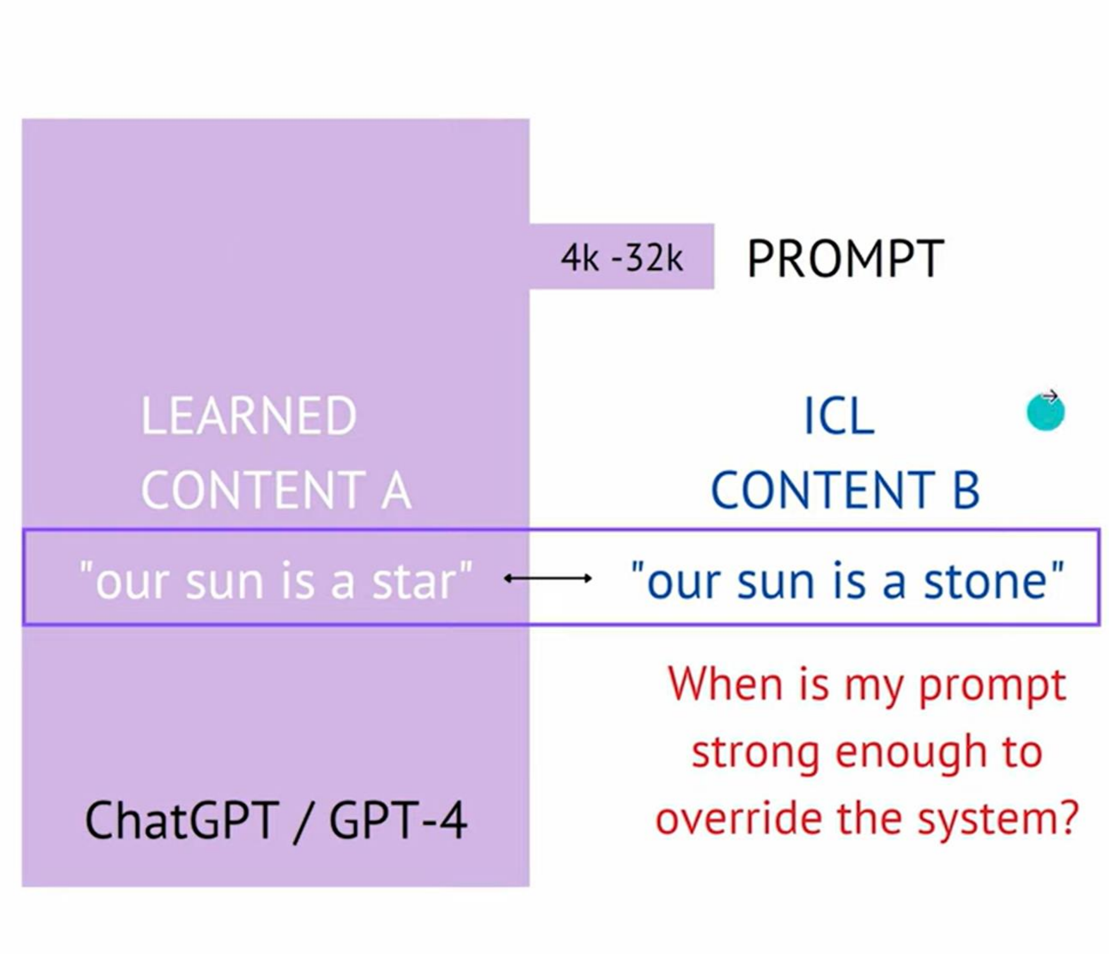
Instruction Fine-Tuning & In-Context Learning



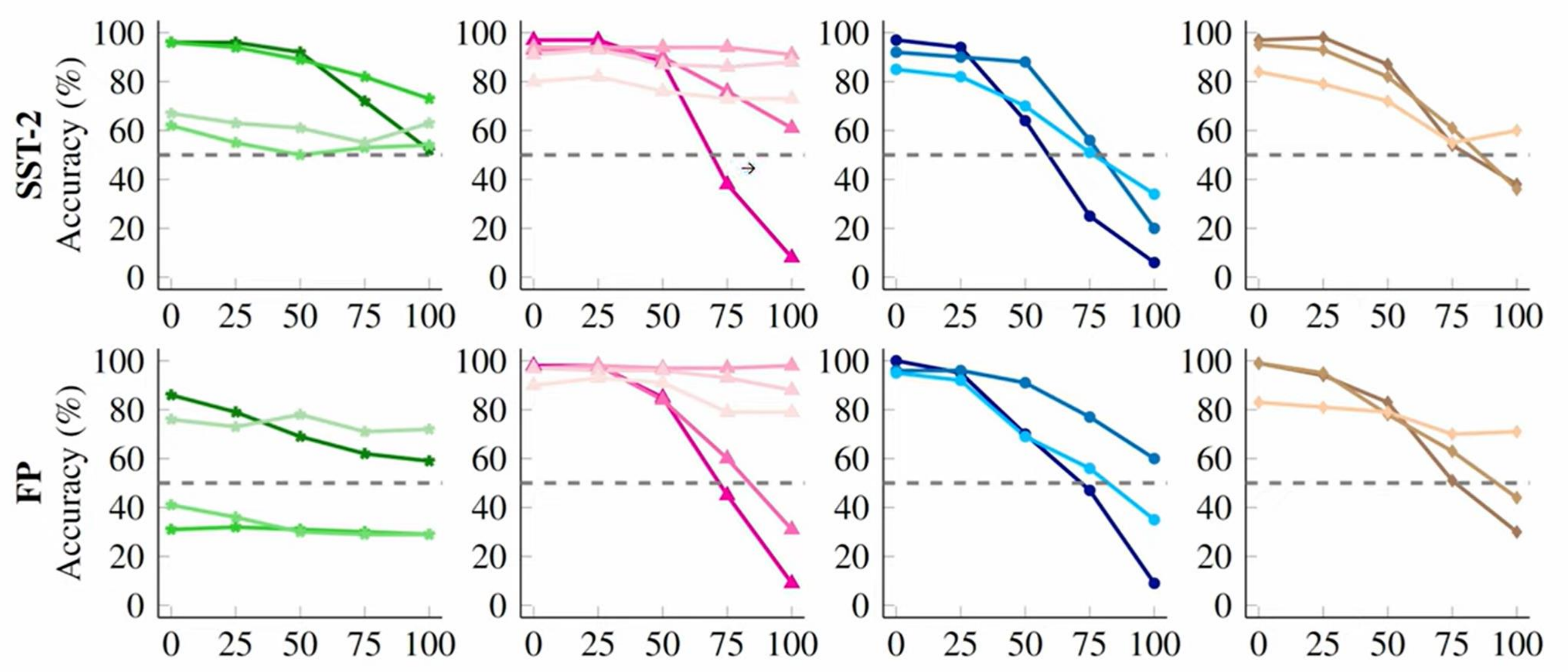
Instruction Fine-Tuning & In-Context Learning

- Flipped-label ICL
- SUL-ICL

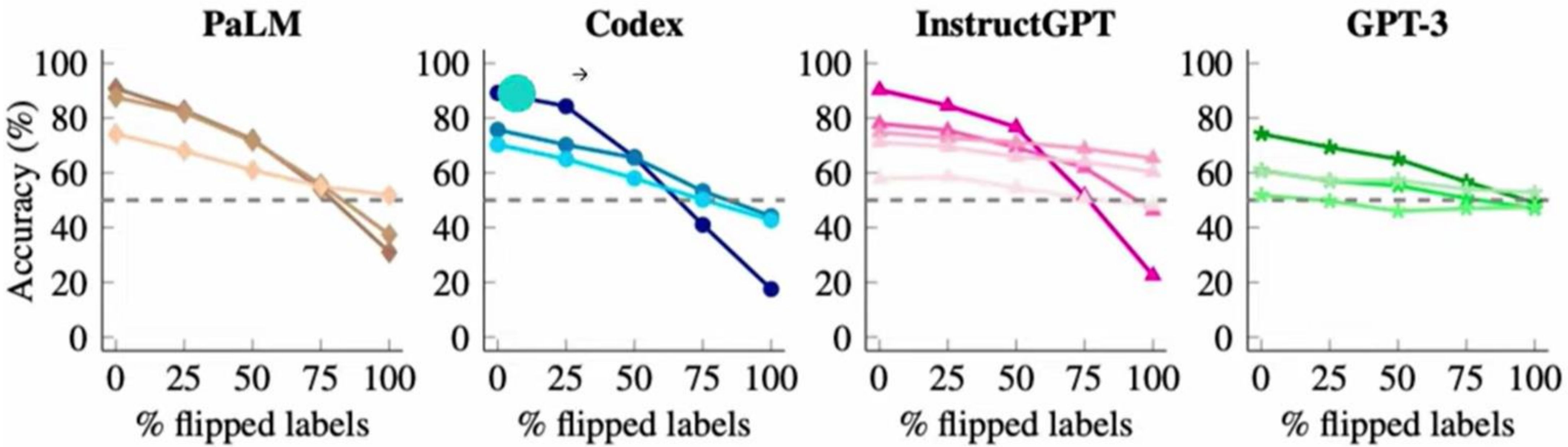
- Larger models > Smaller models



Flipped-label ICL



Flipped-label ICL



In-Context Learning

- What about ICL PROMPT content overrides?

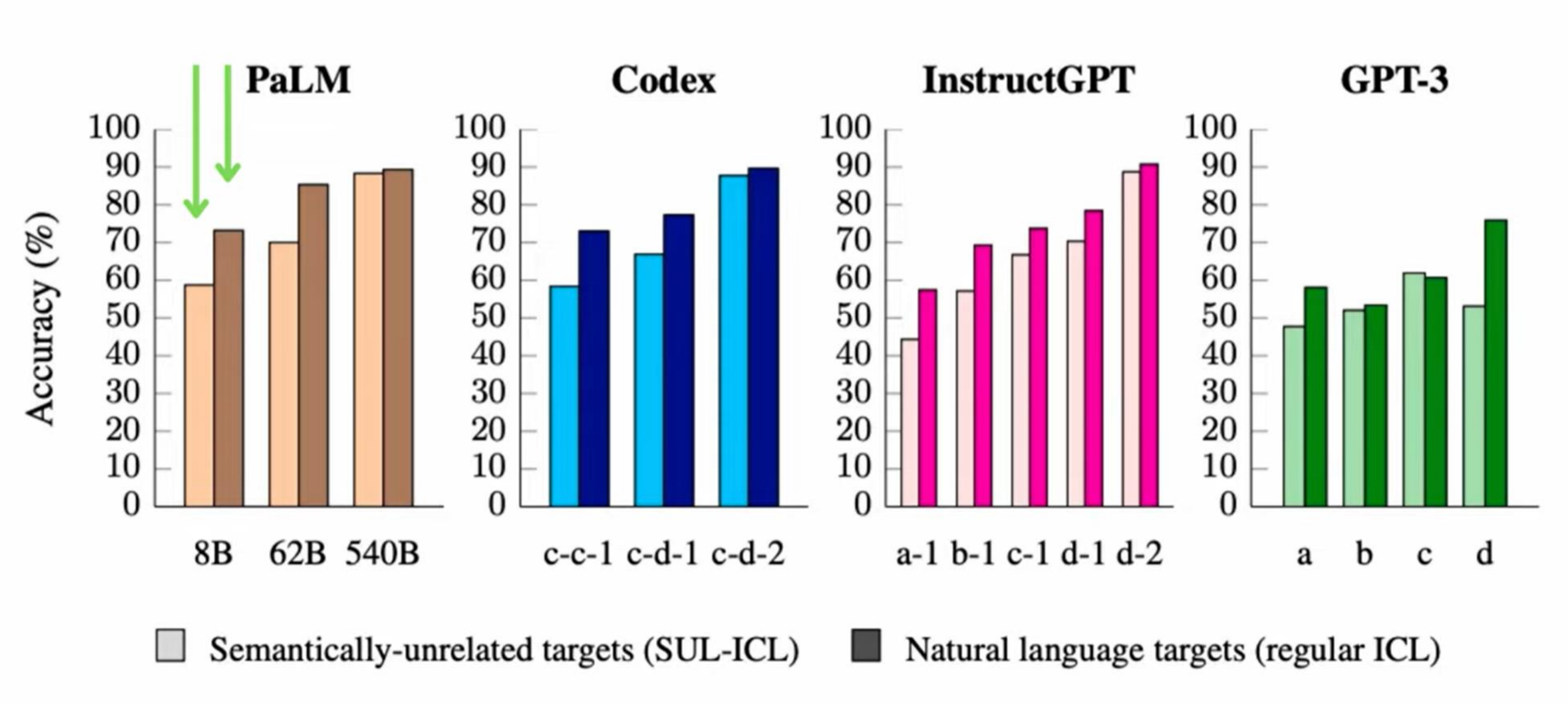
• Weights	vs	Activations
• Fine-tuning	vs	Prompt-tuning

- What about combination of two approaches?

In-Context Learning

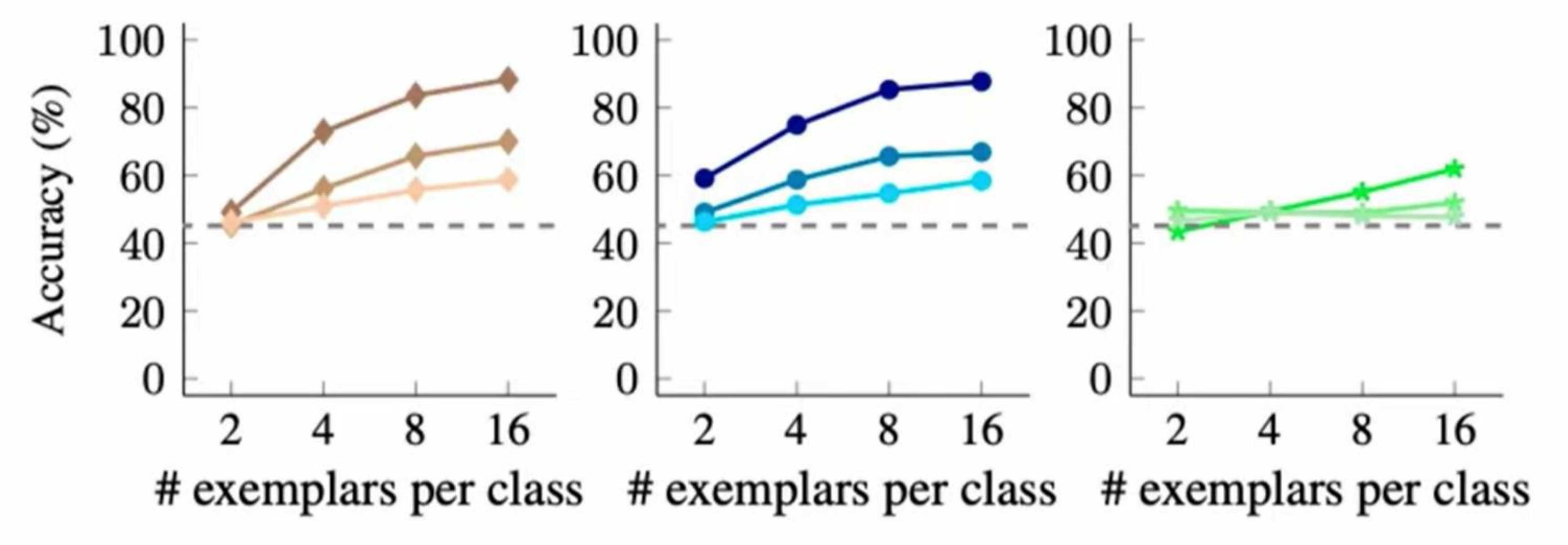
- Papers:
  - In-Context Retrieval-Augmented Language Models
  - What Learning Algorithms is In-Context Learning? Investigating With Linear Models
  - Large Language Models Do In-Context Learning Differently
  - Symbol Tuning Improves In-Context Learning in Language Models

Semantically-unrelated targets ICL



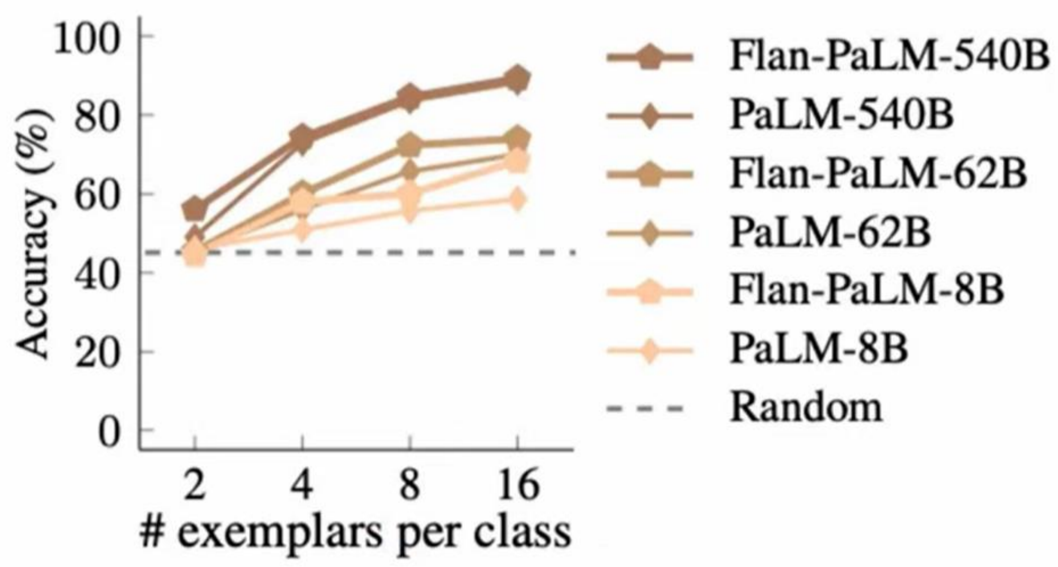
Semantically-unrelated targets ICL

- Small models rely more on semantic priors
- Large models have the ability to learn input-label mapping in-context when the semantic nature of label is removed.
- The effect of number of examples



Instruction tuning

- Learning input-label mappings vs Semantic prior knowledge
  - Which one is stronger?
- More confident on their inherent instruction fine-tuned knowledge



Symbol tuning

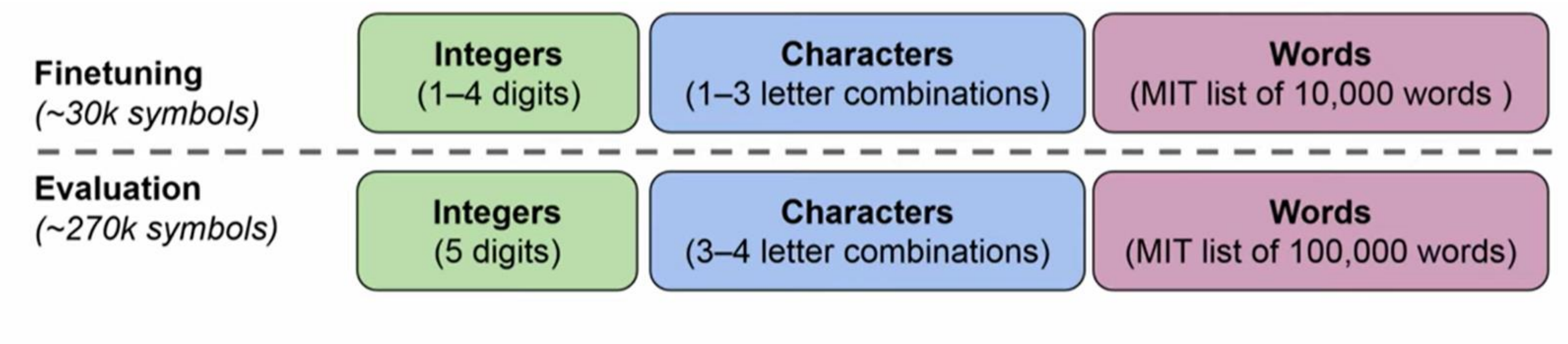
- A form of fine-tuning on input-label pairs where labels are remapped to arbitrary symbols
- Symbol tuning: Remove instructions, change labels to unrelated symbols. Task can only be learned from exemplars.

Symbol tuning

Dataset	Instruction
SUBJ	"Is the following sentence subjective or objective?"
TEH	"Label the following tweet based on whether it contains hate speech."
TEAB	"Read the following tweet and determine its stance on abortion."
TEAT	"Read the following tweet and determine its stance on atheism."
TEFE	"Read the following tweet and determine its stance on feminism."
TEHI	"Read the following tweet and determine its stance on Hillary Clinton."
ADEC	"Label the following sentence based on whether it is related to an adverse drug event."
OR	"Label the following sentence based on whether it is overruling or not."
SOT	"Read the following paper title and institution name and classify the institution as a university, company, or research institute."
TOS	"Label the following sentence from a Terms of Service based on whether it is potentially unfair."
TC	"Label the following tweet text based on whether it contains a complaint."

Symbol tuning

- Which symbol to use?



Symbol tuning

- Which symbol to use?

Model	Algorithmic Reasoning: Turing Concepts	Algorithmic Reasoning: List Functions	In-Context Learning: Flipped Labels	In-Context Learning: Relevant Target + Instruction	In-Context Learning: Relevant Target + No Instruction	In-Context Learning: No Relevant Target + Instruction	In-Context Learning: No Relevant Target + No Instruction
Random Guessing	0	0	50	42.4	42.4	42.4	42.4
Flan-PaLM-8B	17.6	19.2	26.5	63.9	61.6	42.4	44.2
+ Symbol tuning (integers)	34.1	38.1	33.3	**66.9**	**65.5**	54.0	53.5
+ Symbol tuning (characters)	32.9	32.7	34.3	63.5	61.8	56.7	54.7
+ Symbol tuning (words)	**52.9**	**42.5**	**54.8**	60.6	56.6	**56.9**	**54.9**
Flan-PaLM-62B	61.2	56.1	23.8	74.3	70.0	57.0	50.5
+ Symbol tuning (integers)	75.3	64.4	30.7	74.4	70.4	65.4	52.7
+ Symbol tuning (characters)	72.9	64.5	33.5	76.9	70.1	70.8	59.4
+ Symbol tuning (words)	**78.8**	**68.9**	**54.2**	**77.3**	**73.4**	**71.4**	**60.7**

Symbol tuning

- Prompt Formats

- “Input: [input] \n Output: [label]”
- “Input: [input] \n Target: [label]”
- “Input: [input] \n Symbol: [label]”
- “Input: [input] \n Label: [label]”
- “Question: [input] \n Answer: [label]”
- “Student: [input] \n Teacher: [label]”
- “X = [input] \n Y = [label]”
- “Q: [input] \n A: [label]”
- “[input] -> [label]”
- “Sentences: [input] \n Mapped To: [label]”

Symbol tuning prompts

- Prompt containing k = 2 In-context examples per class. The original labels [“entailment”, “not entailment”] have been remapped to [“4348”, “forests”]

Prompt:
Input: In the May 2005 general election Michael Howard failed to unseat the Labour Government, although the Conservatives did gain 33 seats, playing the most significant role in reducing Labour’s majority from 167 to 66.

In the May 2005 general election Conservatives got 33 seats.

Output: forests

Prompt:
X = Which restaurant did Madonna work in New York City?.

In 1978, she dropped out of college and relocated to New York City.

Y = 8529

Symbol tuning results

	Average performance on eleven tasks			
Relevant labels:	✓	✓	✗	✗
Task instructions:	✓	✗	✓	✗
Random Guessing	42.4	42.4	42.4	42.4
Flan-PaLM-8B	63.9	61.6	42.4	44.2
+ Symbol tuning (ours)	57.6 (**-6.3**)	54.3 (**-7.3**)	58.2 (**+15.8**)	52.8 (**+8.6**)
Flan-PaLM-62B	74.3	70.0	57.0	50.5
+ Symbol tuning (ours)	75.5 (**+1.2**)	70.8 (**+0.8**)	71.4 (**+14.4**)	60.3 (**+9.8**)
Flan-cont-PaLM-62B	77.3	70.3	56.3	51.0
+ Symbol tuning (ours)	78.9 (**+1.6**)	74.5 (**+4.2**)	71.8 (**+15.5**)	62.1 (**+11.1**)
Flan-PaLM-540B	82.2	77.4	70.7	58.1
+ Symbol tuning (ours)	84.4 (**+2.2**)	78.8 (**+1.4**)	80.0 (**+9.3**)	63.6 (**+5.5**)

Symbol tuning

- It works best when relevant labels are unavailable
- The symbol tuning can allow much smaller models to perform as well as large models
- Potential of improvements especially when tasks are ***not clear***
- For small models when the task is clear the performance decreases
    - This may suggest that symbol tuning can override its prior knowledge

Symbol tuning

- Symbol tuning is based of the intuition that when models cannot use instructions or relevant labels, it must do so by instead learning from in-context exemplars.
- Much better on algorithmic tasks…

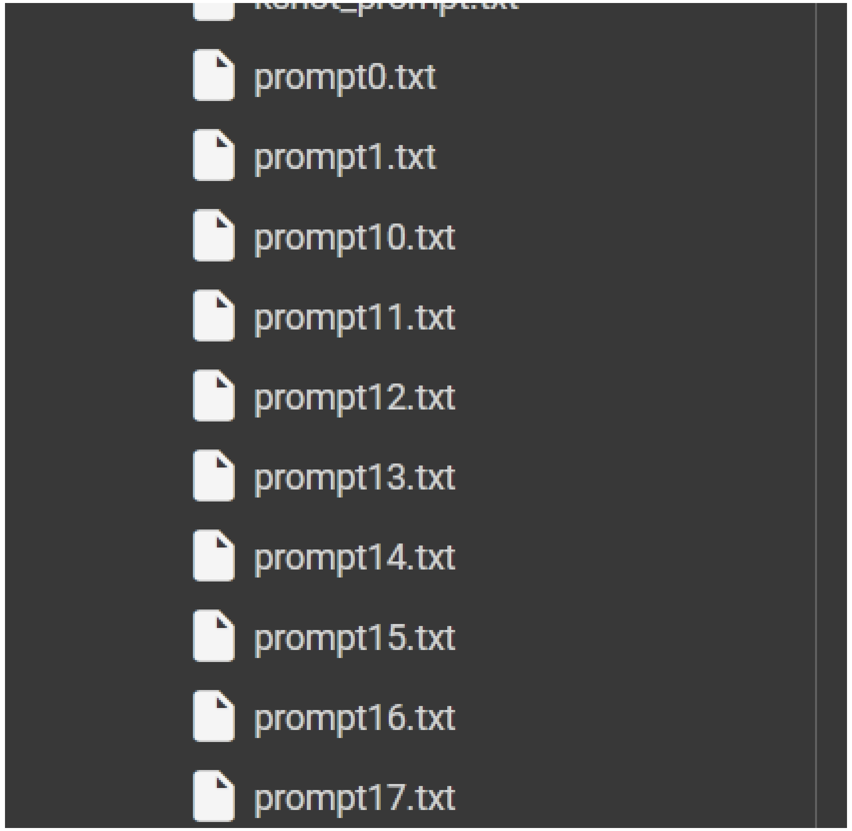
Symbol Tuning Benchmark

Symbol tuning

- Change labels to numbers.
- Trying to not mention anything related to ‘important’ or ‘not important’ tags in the prompt.

Symbol tuning

- Cleaned the code.
- Saving each prompt for more analyses.

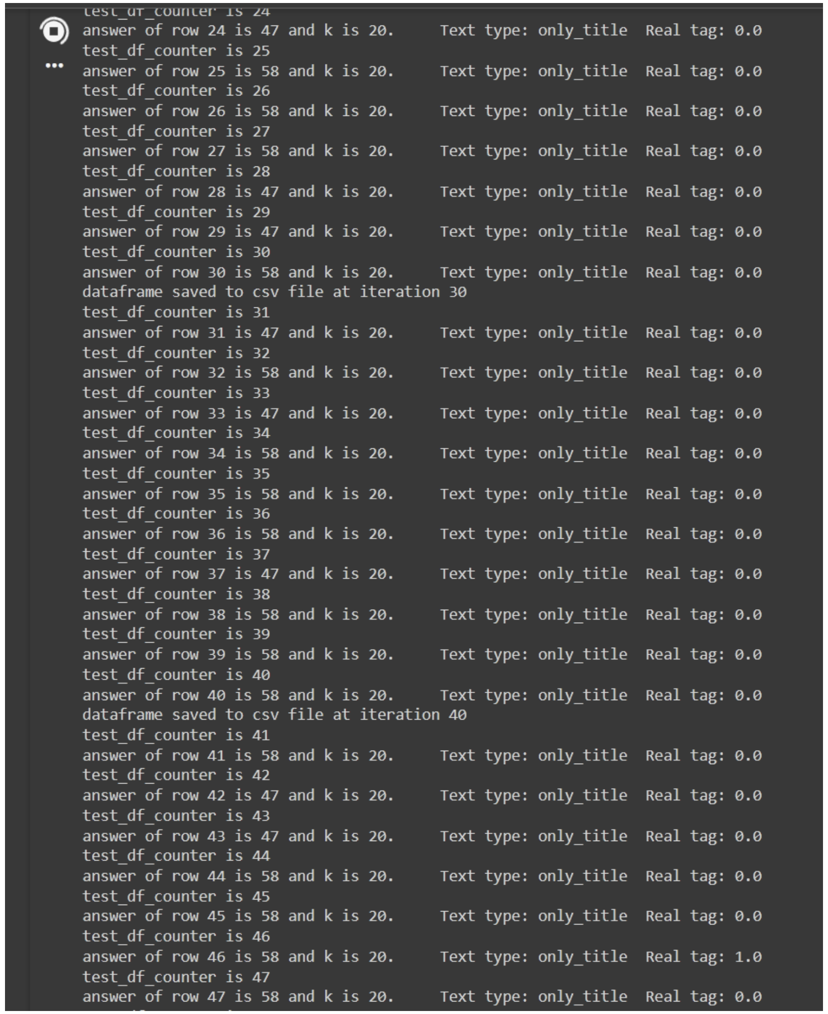


Symbol tuning

- *First step:*
  - Change the labels ‘1’s to ’58’s.
  - Change the labels’0’s to ‘47’s.
- We tried to use labels that the model hasn’t seen.
- So, it doesn’t use its’ predefined knowledge to tag news with ‘important’ or ‘not important’ tags.

Symbol tuning

- Surprisingly the Aya LLM tends to generate ‘58’ more than ‘47’ ones.
- This might because there is more details or definition defined about ’58’ label.
- Or this might be caused by ’58’ being the first label.
- Or the dataset being imbalanced!
- The result shown is in ‘k=20’ mode.



Symbol tuning

- The result shown here is with k=0 shot prompts.
- The model only generates ‘58’ as an answer!
- We can interpret two things from the observation:
  - First the k shot example help the model to obtain knowledge about ‘47’ labels therefore resulting to predict some titles as ‘not important’ or ’47’.
  - Second, we should include in prompt what is ‘not important’ or ‘47’ label, only including information about what is known as ‘important’ result in generating only ‘important’ labels.

```
        print(f"dataframe saved to csv file at iteration {i}")
```

```
test_df_counter is 0
answer of row 0 is 58 and k is 0.     Text type: only_title  Real tag: 0.0
dataframe saved to csv file at iteration 0
test_df_counter is 1
answer of row 1 is 58 and k is 0.     Text type: only_title  Real tag: 0.0
test_df_counter is 2
answer of row 2 is 58 and k is 0.     Text type: only_title  Real tag: 1.0
test_df_counter is 3
answer of row 3 is 58 and k is 0.     Text type: only_title  Real tag: 1.0
test_df_counter is 4
answer of row 4 is 58 and k is 0.     Text type: only_title  Real tag: 0.0
test_df_counter is 5
answer of row 5 is 58 and k is 0.     Text type: only_title  Real tag: 0.0
test_df_counter is 6
answer of row 6 is 58 and k is 0.     Text type: only_title  Real tag: 0.0
test_df_counter is 7
answer of row 7 is 58 and k is 0.     Text type: only_title  Real tag: 0.0
test_df_counter is 8
answer of row 8 is 58 and k is 0.     Text type: only_title  Real tag: 0.0
test_df_counter is 9
answer of row 9 is 58 and k is 0.     Text type: only_title  Real tag: 0.0
test_df_counter is 10
answer of row 10 is 58 and k is 0.     Text type: only_title  Real tag: 1.0
dataframe saved to csv file at iteration 10
test_df_counter is 11
answer of row 11 is 58 and k is 0.     Text type: only_title  Real tag: 0.0
test_df_counter is 12
answer of row 12 is 58 and k is 0.     Text type: only_title  Real tag: 0.0
test_df_counter is 13
answer of row 13 is 58 and k is 0.     Text type: only_title  Real tag: 0.0
test_df_counter is 14
answer of row 14 is 58 and k is 0.     Text type: only_title  Real tag: 0.0
test_df_counter is 15
```

Symbol tuning

- The result shown here is with k=1 shot prompts.
- The model generates ’47’ labels sporadically.
- This means that one example provided in the prompt was not enough to give the model enough information to predict more labels as ’47’.
- But it shows that even providing one example can change the output!

```
answer of row 0 is 58 and k is 1.     Text type: only_title  Real tag: 0.0
dataframe saved to csv file at iteration 0
test_df_counter is 1
answer of row 1 is 58 and k is 1.     Text type: only_title  Real tag: 0.0
test_df_counter is 2
answer of row 2 is 58 and k is 1.     Text type: only_title  Real tag: 1.0
test_df_counter is 3
answer of row 3 is 58 and k is 1.     Text type: only_title  Real tag: 1.0
test_df_counter is 4
answer of row 4 is 58 and k is 1.     Text type: only_title  Real tag: 0.0
test_df_counter is 5
answer of row 5 is 58 and k is 1.     Text type: only_title  Real tag: 0.0
test_df_counter is 6
answer of row 6 is 58 and k is 1.     Text type: only_title  Real tag: 0.0
test_df_counter is 7
answer of row 7 is 58 and k is 1.     Text type: only_title  Real tag: 0.0
test_df_counter is 8
answer of row 8 is 58 and k is 1.     Text type: only_title  Real tag: 0.0
test_df_counter is 9
answer of row 9 is 58 and k is 1.     Text type: only_title  Real tag: 0.0
test_df_counter is 10
answer of row 10 is 47 and k is 1.     Text type: only_title  Real tag: 1.0
dataframe saved to csv file at iteration 10
test_df_counter is 11
answer of row 11 is 58 and k is 1.     Text type: only_title  Real tag: 0.0
test_df_counter is 12
answer of row 12 is 58 and k is 1.     Text type: only_title  Real tag: 0.0
test_df_counter is 13
```

Symbol tuning

- The result shown here is with k=50 shot prompts.
- The model generates more ’47’ labels.
- The results shows that the information and details about the ‘not important’ news is a necessity to override LLM predefined knowledge.

```
test_df_counter is 19
answer of row 19 is 47 and k is 50.     Text type: only_title  Real tag: 0.0
test_df_counter is 20
answer of row 20 is 47 and k is 50.     Text type: only_title  Real tag: 0.0
dataframe saved to csv file at iteration 20
test_df_counter is 21
answer of row 21 is 58 and k is 50.     Text type: only_title  Real tag: 1.0
test_df_counter is 22
answer of row 22 is 47 and k is 50.     Text type: only_title  Real tag: 0.0
test_df_counter is 23
answer of row 23 is 58 and k is 50.     Text type: only_title  Real tag: 0.0
test_df_counter is 24
answer of row 24 is 47 and k is 50.     Text type: only_title  Real tag: 0.0
test_df_counter is 25
answer of row 25 is 58 and k is 50.     Text type: only_title  Real tag: 0.0
test_df_counter is 26
answer of row 26 is 58 and k is 50.     Text type: only_title  Real tag: 0.0
test_df_counter is 27
answer of row 27 is 58 and k is 50.     Text type: only_title  Real tag: 0.0
test_df_counter is 28
answer of row 28 is 58 and k is 50.     Text type: only_title  Real tag: 0.0
test_df_counter is 29
answer of row 29 is 47 and k is 50.     Text type: only_title  Real tag: 0.0
test_df_counter is 30
answer of row 30 is 58 and k is 50.     Text type: only_title  Real tag: 0.0
dataframe saved to csv file at iteration 30
```

Symbol tuning

- The challenge to make predictions more accurate is to include clear definition and details for both ‘important’ and ‘not important’ news.
- This causes the language model to rely more on the information given in the prompt (or, as we know, in-context learning) rather than on its prior knowledge.

Symbol tuning results

- Results for k = 0 shot learning:

K = 0	Accuracy	Precision	Recall	F1-Score	# of ’58’	# of ’47’
Title	17%	14%	93%	24%	96	5

- The shown results is for first 101 entities in test data.

Symbol tuning results

- Results for k = 1 shot learning:

K = 1	Accuracy	Precision	Recall	F1-Score	# of ’58’	# of ’47’
Title	48%	19%	86%	31%	63	38

- The shown results is for first 101 entities in test data.

Symbol tuning results

- Results for k = 5 shot learning:

K = 5	Accuracy	Precision	Recall	F1-Score	# of ’58’	# of ’47’
Title	47%	16%	64%	25%	58	43

- The shown results is for first 101 entities in test data.

Symbol tuning results

- Results for k = 20 shot learning:

K = 20	Accuracy	Precision	Recall	F1-Score	# of ’58’	# of ’47’
Title	49%	15%	57%	24%	54	47

- The shown results is for first 101 entities in test data.

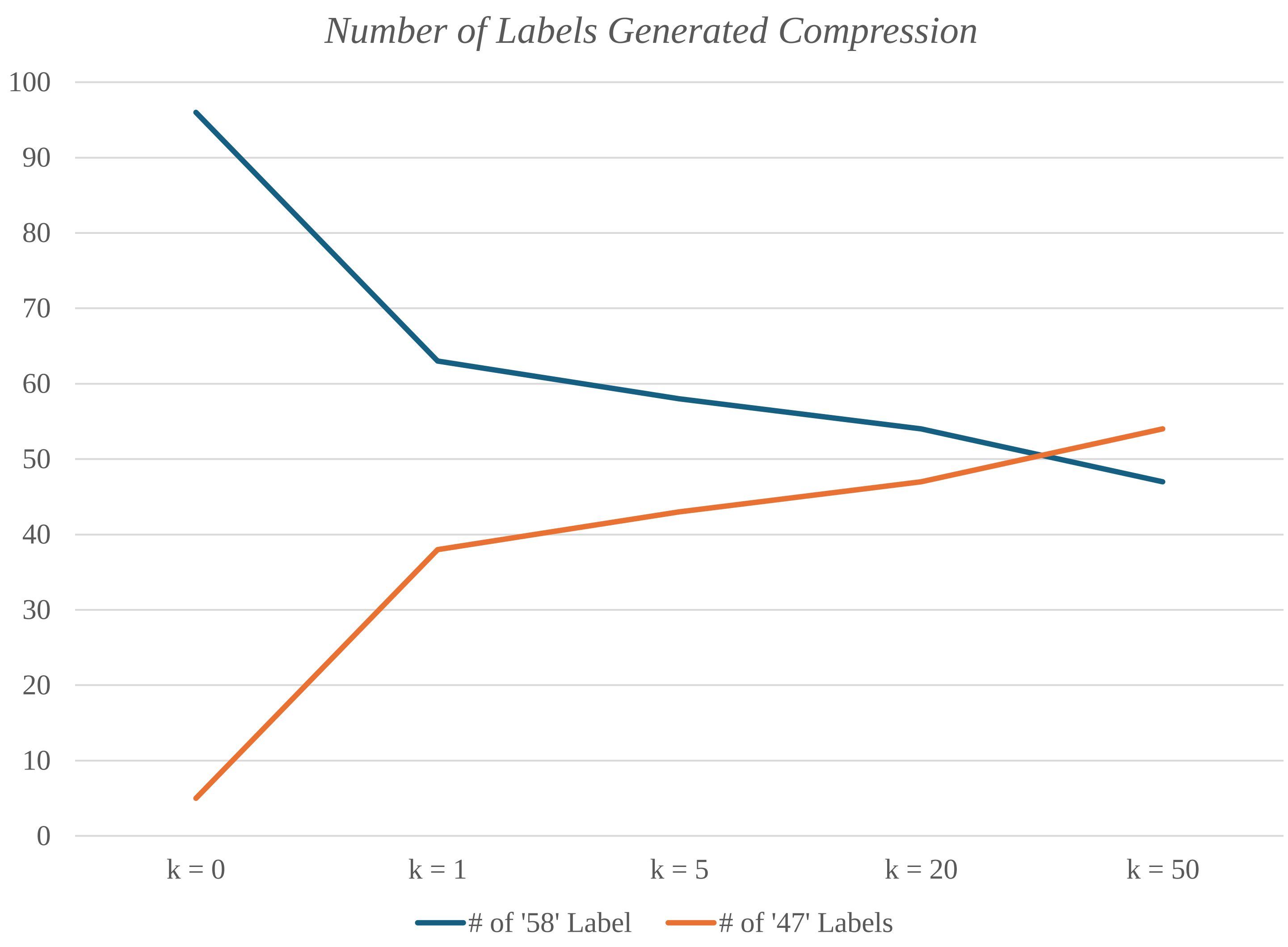
Symbol tuning results

- Results for k = 50 shot learning:

K = 50	Accuracy	Precision	Recall	F1-Score	# of ’58’	# of ’47’
Title	55%	17%	57%	26%	47	54

- The shown results is for first 101 entities in test data.

Symbol tuning results



Symbol tuning feasible improvements

- Possible future improvements:
  - Change the prompt: The problem observed here is that the prompt lacks a definition for ‘important’ news but details and definitions for ‘not important’ ones.
  - Changing the ‘important’ label to something that is harder to generate because our dataset is imbalanced, and we have little ‘important’ news compared to ‘not important’ ones. Therefore, it is logical to make the ‘important’ label harder to generate for the LLM model.
  - Including in the prompt that we have way less ‘important’ news than ‘not important’ ones; therefore, the model should be more sensitive and conservative in generating the ‘important’ label.
  - Including the chain of thoughts context with the examples provided in the prompt to make the decision for the model more logical and with more reasoning information.