ANLP – Assignment 4

Course Coordinator: Manish Srivatsava
Name : Lakshmipathi Balaji
Roll No: 2021114007
Mail : lakshmipathi.balaji@gmail.com

QUESTION - 1

Concept of Soft Prompts: How does the introduction of ”soft prompts” address the limitations of discrete text prompts in large language models? Why might soft prompts be considered a more flexible and efficient approach for task-specific conditioning?

Soft prompts, in contrast to traditional, discrete text prompts, offer a more nuanced approach to interacting with large language models. Here's why they stand out:

1. **Overcoming the Constraints of Fixed Vocabulary**: Regular text prompts are hamstrung by a rigid set of words and require manual crafting, which may miss out on the finer details needed for certain tasks. Soft prompts, however, are more malleable. They learn and evolve through backpropagation, enabling them to grasp and reflect subtle elements essential for specific tasks, something beyond the reach of standard text prompts.

2. **Adapting Dynamically to the Task at Hand**: Unlike their static counterparts, soft prompts are not set in stone. They can modify their representations during the training process to better align with the unique requirements of a particular task.

3. **Enhanced Conditioning for Tasks**: Soft prompts aren't bound by the limitations of length or vocabulary that constrain discrete prompts. This freedom allows them to condition the language model more effectively for specific tasks.

As for why soft prompts are seen as a more flexible and efficient solution:

1. **Task Specific**: Soft prompts excel in their ability to fine-tune to the nuances of a task. Their embeddings are optimized during training, making

them task-specific, as opposed to relying on generic, pre-trained embeddings.

2. **Minimizing Manual Effort**: Finding the right discrete prompts can be a time-consuming process that involves much trial and error. Soft prompts remove this hurdle by self-optimizing, thereby reducing the manual labor involved.

3. **Efficient Use of Parameters**: A key advantage of soft prompts is their ability to achieve superior performance with a leaner use of parameters. This efficiency is particularly beneficial in large-scale models where managing resource usage is critical.

QUESTION -2

Scaling and Efficiency in Prompt Tuning: How does the efficiency of prompt tuning relate to the scale of the language model? Discuss the implications of this relationship for future developments in large-scale language models and their adaptability to specific tasks.

Scaling and Efficiency in Prompt Tuning: Exploring the Connection Between Model Size and Prompt Tuning Effectiveness

- **Enhanced Performance in Larger Models**: As we scale up language models, increasing their parameter count, they become more adept at utilizing soft prompts. This is because larger models have a greater ability to assimilate and make use of the nuanced instructions embedded in these prompts, leading to improved overall performance.

- **Limited Impact in Smaller Models**: On the flip side, smaller language models might not fully exploit the advantages of prompt tuning. Their restricted capacity can result in a less significant boost in performance when compared to their larger counterparts.

Implications for Future Language Model Developments

- **Promising Outlook for Larger Models**: This correlation indicates that as we continue to build bigger language models, the effectiveness of prompt tuning is likely to grow as well. This presents an exciting opportunity for advancing more powerful and efficient models, particularly for tasks that demand a subtle grasp of language or intricate generation capabilities.

- **Task-Specific Adaptability**: Large models, when paired with prompt tuning, show enhanced flexibility in adapting to specific tasks. This flexibility is invaluable for applications that demand tailored responses, like generating personalized content or handling intricate question-and-answer scenarios.

- **Resource Efficiency**: For organizations and researchers, the ability to employ a single expansive model across various tasks, minimizing the need for exhaustive fine-tuning (thanks to effective prompt tuning), translates to a more economical allocation of resources.

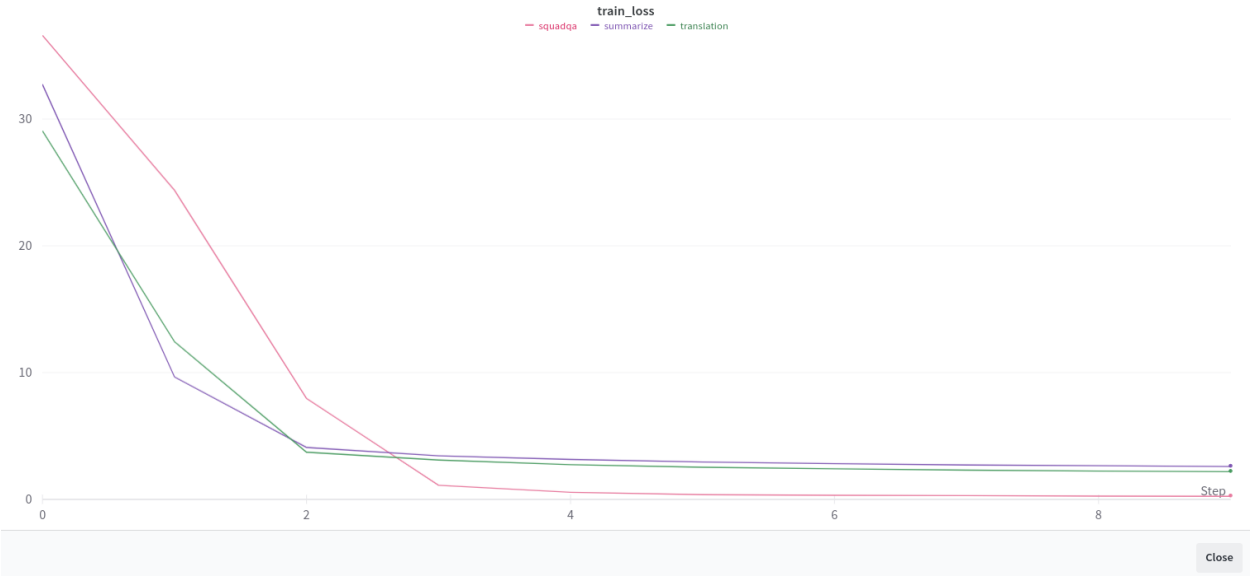
- **Balancing Act in Resource Utilization**: Despite the optimism, this approach also introduces challenges in terms of computational resources. Bigger models necessitate more memory and processing power. Hence, balancing model size, efficiency, and resource demands will be a crucial factor in the ongoing evolution of language models.

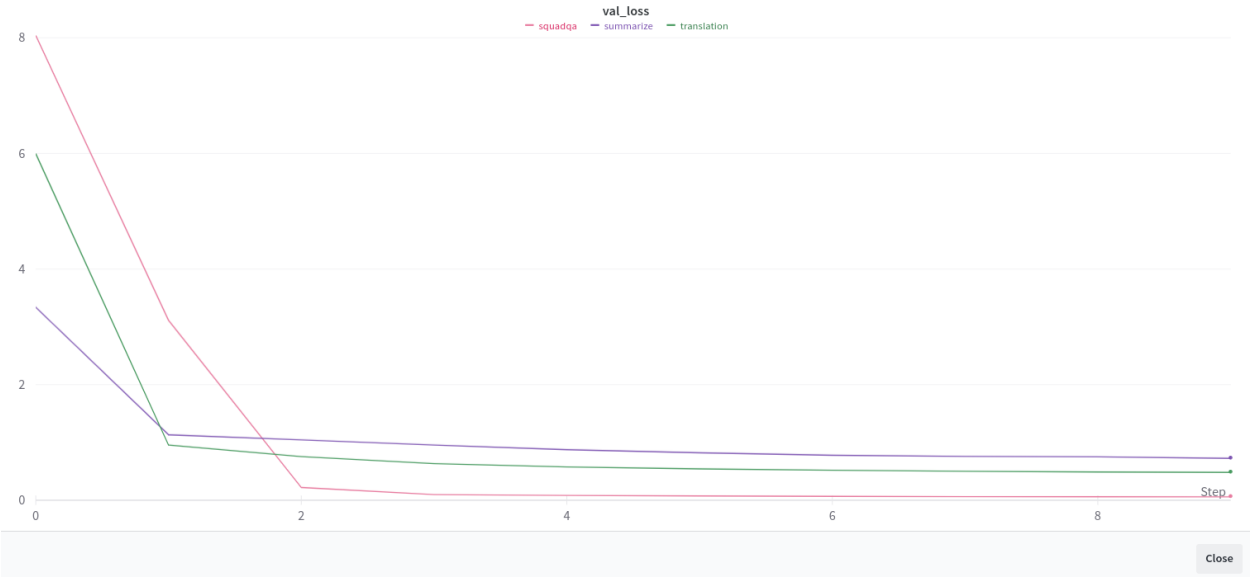
This exploration into the relationship between model scale and the efficacy of prompt tuning sheds light on both the potential and challenges in the realm of large-scale language model development, particularly concerning their adaptability and efficiency in various tasks.

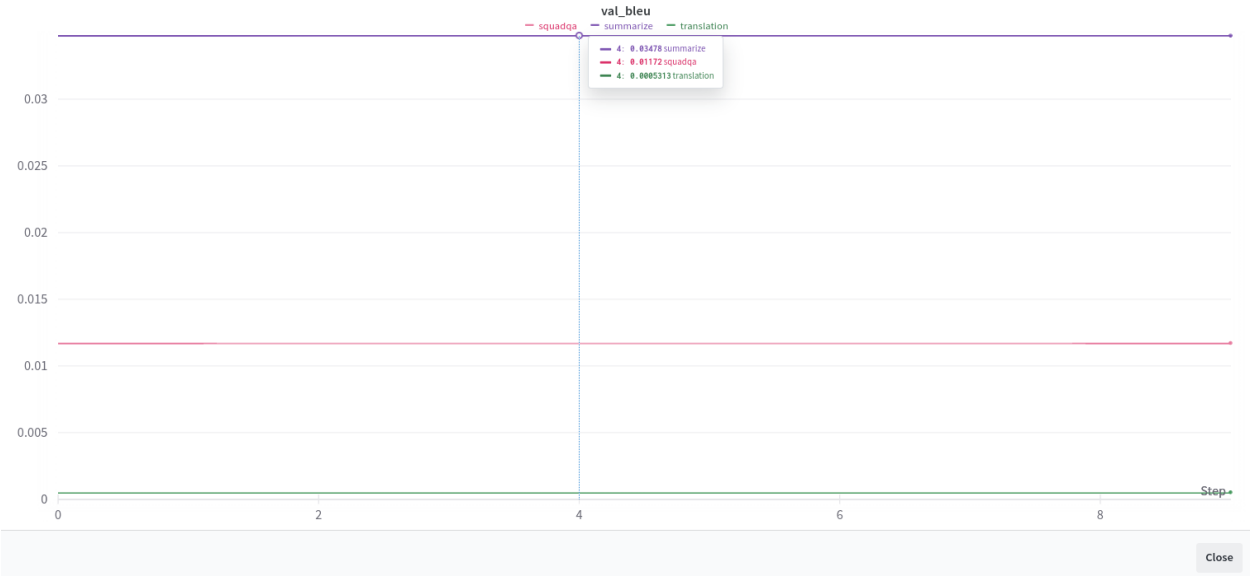
REPORT

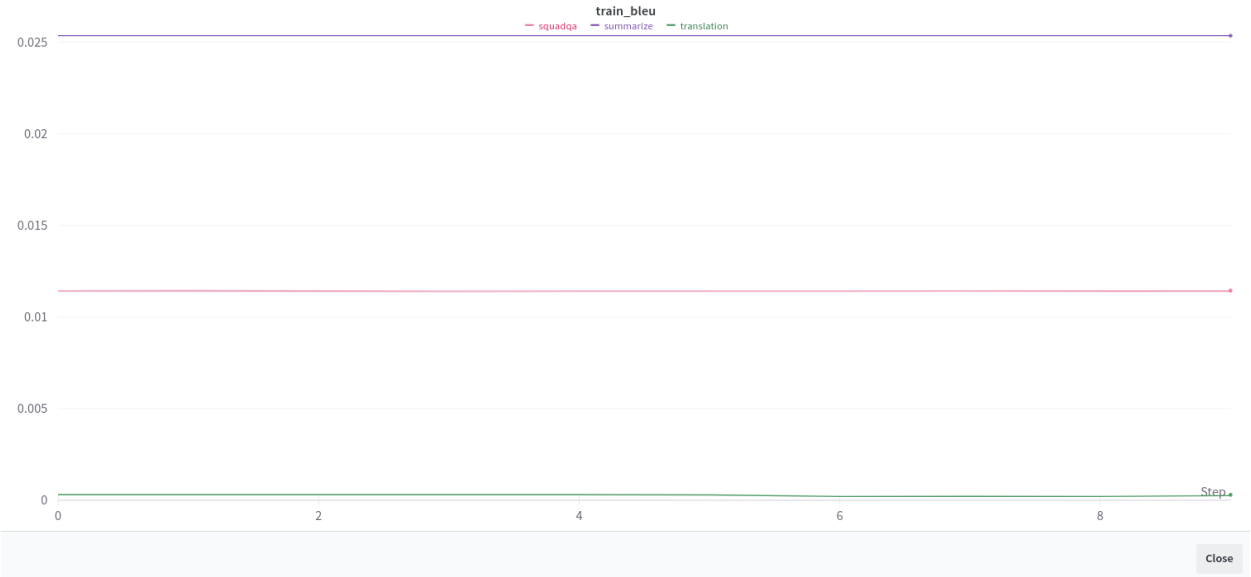
Here are the hyperparams, results and someother details about hyperparamerter tuning use for the task,

```yaml
num_tokens: 20
tasks:
  - squadqa
  - summarization
  - europal

task : squadqa
squadqa:
  soft_prompt: "QUESTION ANSWER"
  paths:
    train_df: "/ssd_scratch/cvit/kolubex/squad_data/Squad_train.csv"
    val_df: "/ssd_scratch/cvit/kolubex/squad_data/Squad_val.csv"
  wandb_run_name: "squadqa"
  model_save_path: "/ssd_scratch/cvit/kolubex/models_squadqa/"


summarization:
  soft_prompt: "SUMMARIZE"
  paths:
    train_df: "/ssd_scratch/cvit/kolubex/cnn_dailymail/train.csv"
    val_df: "/ssd_scratch/cvit/kolubex/cnn_dailymail/validation.csv"
  wandb_run_name: "summarize"
  model_save_path: "/ssd_scratch/cvit/kolubex/models_summarize/"

europal:
  soft_prompt: "TRANSLATE"
  paths:
    train_df: "/ssd_scratch/cvit/kolubex/europal/EuroPal_train.csv"
    val_df: "/ssd_scratch/cvit/kolubex/europal/EuroPal_val.csv"
  wandb_run_name: "translation"
  model_save_path: "/ssd_scratch/cvit/kolubex/models_europal/"

num_workers: 8
epochs: 10

training:
  batch_size: 1
  lr: 0.0001
validation:
  batch_size: 6

model_name: "gpt2"

wandb_logging: True
wandb_project: 'gptprompttune'
wandb_entity: 'kolubex'
model_save_freq: 1
model_save_path: "/ssd_scratch/cvit/kolubex/models_europal/"

clip_grads: True
early_stop:
  use: True
  patience: 4
num_accumulate: 4
gradient_accumulate: True
```


train_loss
squadqa
summarize
translation
30
20
10
0
0
2
4
6
8
Step
Close


val_loss
squadqa
summarize
translation
8
6
4
2
0
0
2
4
6
8
Step
Close





As you can observe the model is learning in the initial epochs, and no much increment is observed in bleu score due to its capacity, and the difficulty in task. Like in summarization though the model is has only 1024 as seq length, the dataset samples of input articles is much higher, and the model is not quite well trained for translation because we don't have a GPT2 tokenizer for German from the gpt2-small model we are using. And other possibilities will be discussed in the evaluations.

Other combinations of hyperparams have also been tried but only these are observed to be feasible to train based on the results the other combinations tried.

Task - HardPrompt	SoftPrompt	Result with hard prompt	Result with hard prompt
Summarize - `summarize the task`	Summarize	0.01739	0.02536
SquadQA - Answer the question	Question Answer	0.0928	0.1144
Translate - Translate to German	Translate	0.00034	0.00034

The above results can be explainable based on the task given for the assignment.

Link to wandb project - wandb
Link to the models and files - ass4