Link

- https://arxiv.org/pdf/2104.08691.pdf

Code

https://github.com/google-research/prompt-tuning

- Done in Jax

What is prompt tuning

- Helping model learn downstream tasks
- Prompts are learned through backpropagation

Approaches

- Freezing pretrained model and learning task specific weighting
- Currently dominant technique is model tuning aka finetuning
    - Model params tuned during adaptation
    - Adaptation is where you copy the model for a specific task
- Prompt design
    - Task description and examples
    - However very error prone
- In conclusion, prompt design lags behind fine tuning in results
- Prefix tuning
    - method freezes the model parameters and backpropagates the error during tuning to prefix activations prepended to each layer in the encoder stack, including the input layer.
- Paper introduces Prompt Tuning

Paper's Key Contributions

1. Proposing prompt tuning and showing its competitiveness with model tuning for LLMs
2. Ablating design choices and proving that quality and robustness improve with scale
3. Prompt tuning outperforms model tuning on domain shift problems
4. Prompt emsembling

T5 Models

- https://arxiv.org/abs/1910.10683
- Text to Text Transformer Transformer
- Rudimentary version of llm we know today

Example Query

Original input: Question: Where did Jebe die?
Sentence: Genghis Khan recalled Subutai back to Mongolia soon afterwards, and Jebe died on the road back to Samarkand.
Processed input: qnli
question: Where did Jebe die?
sentence: Genghis Khan recalled Subutai back to Mongolia soon afterwards, and Jebe died on the road back to Samarkand.
Original target: 0
Processed target: entailment

Prompt Tuning in Depth

- Output of T5 Model is Prθ(Y |X)
- X is a series of tokens
- Y is a sequence of tokens for ae class label
- *P* (italicized P) is the prompting tokens
  - Prompting is when you prepend *P* to an input X
  - This maximizes Y = Prθ(Y |[*P*; X])
    - Here model params are fixed
- In prompt tuning θ or the model's weights are not frozen
  - θ_P is updated (prompt parameters)
  - Resulting in this equation Prθ;θP (Y |[*P*; X])
    - Gradient updates are applied to θP

Design Decisions

- Here the paper discusses how to intialize the first prompt representation
- Simplest – use random initialization
- Better way - initailize each prompt from an embedding in model's vocabulary
- Verbalizer technique (Schick and Schutze 2021)

Span Corruption

- T5 is pretrained on span corruption objective
- T5 needs to fill in those spans

- Consecutive corrupted tokens are treated as a span, each span is then given a single unique mask token, which replaces the entire span. This results in shorter sequences.
    - Original text: One Piece is the greatest story ever told in human history.
    - Corrupted Spans: One Piece <X> story ever <Y> in human history.
    - Target: <X>is the greatest<Y>ever told<Z>
    - The sentinels are the <X>, <Y>, <Z>
- Authors were initially unsure whether this works with prompt finetuning (because prompts are free flowing text and don't contain sentinels)
    - They were right about this. Span Corruption with promtps dont work well
- They tested between
    - Span corruption
    - Span corruption + sentinel
    - LM Adaptation - gives natural text input text, model has to output natural text)
- It turns out that LM Adaptation is the most robust

Resilience to Domain Shift

- Since they freeze model parameters and prompt params are trained separately, it reduces model's ability to overfit (sorta)

Prompt Ensembling

- Before prompts, we had neural networking ensembling
    - Known to improve performance
    - But impractical when model size increases
- Ensembling with prompt tuning is more efficient