TUNING

A68083 현 종 승
A66048 안 수 원
A68045 배 성 은

Downstream Tasks Learning

Fine-tuning

Prompt-tuning

In-context learning

SMART:

ROBUST AND EFFICIENT FINE-TUNING FOR PRE-TRAINED NATURAL LANGUAGE MODELS THROUGH PRINCIPLED REGULARIZED OPTIMIZATION

A68083
현종승

1. Summary

- 많은 NLP 모델은 대규모 Text Corpus 에 대해 Pre-training 된 후, 새로운 데이터셋 (Downstream Data)에 대해 Fine-Tuning으로 모델의 가중치를 조정하며 최적화한다.

- 제한된 Downstream Data Resource와 Pre-training 된 모델의 높은 Complexity 때문에 Overfitting (과적합) 되고, Aggressive Updating (빠른 업데이트) 이 발생한다.

- 일반화 성능을 달성하기 위해 '모델의 Complexity'를 제어하기 위한 'Smoothness-Inducing Adversarial Regularization' 와 'Aggressive Updating' 막기 위한 'Bergman Proximal Point Optimization' 를 제시한다.

2. Keywords / Contents

I. Complexity

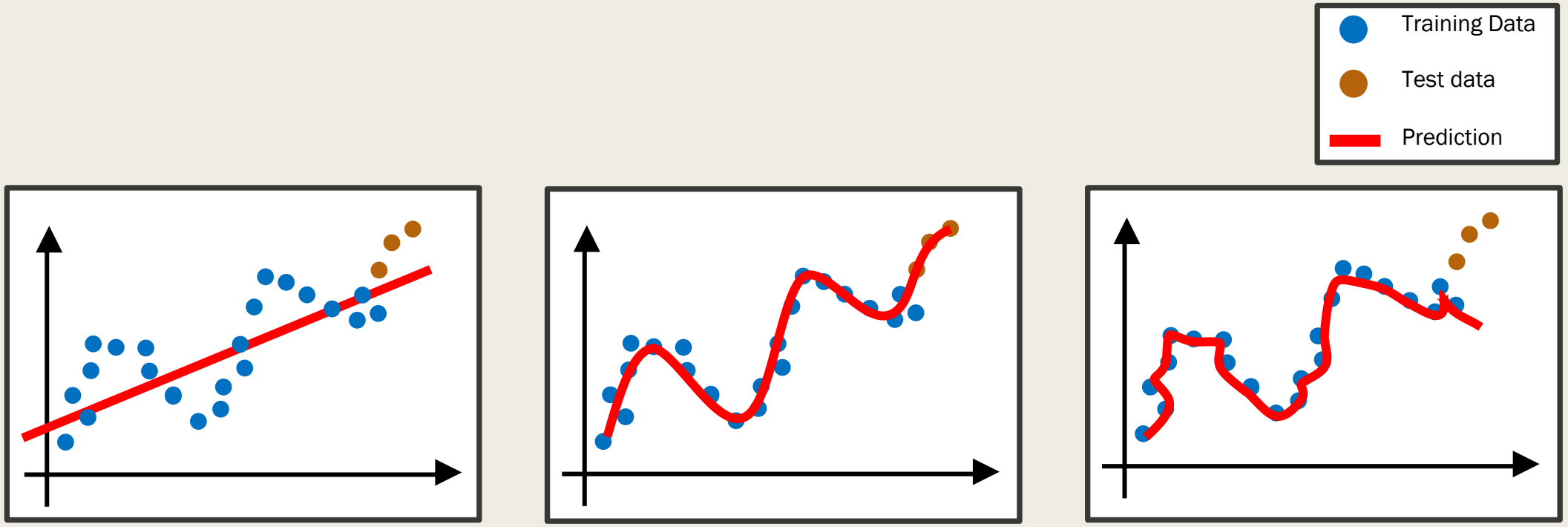
II. Aggressive Updating

III. Smoothness-Induing Adversarial Regularization

IV. Bergman Proximal Point Optimization

I. Complexity

- 모델이 데이터의 다양한 패턴과 관계를 얼마나 잘 학습하고 표현하는지 나타냄



- Underfitting
- Low Complexity

- Fitted Well
- Appropriate Complexity

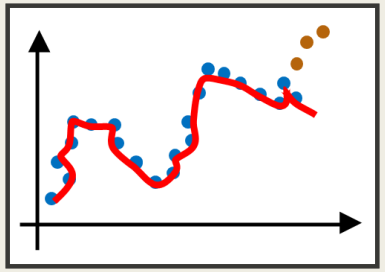
- Overfitting
- High Complexity

ILSVRC 대회 Training Data = 120만개

I. Complexity

P/N = Parameter Count (매개변수)
/ Num Training Data (훈련 샘플 개수)

모델의 복잡도와 훈련 데이터에 대한 적합도를 나타내는 지표



P/N ↑ 모델이 데이터에 과적합 됨 (High Complexity , Overfitting)

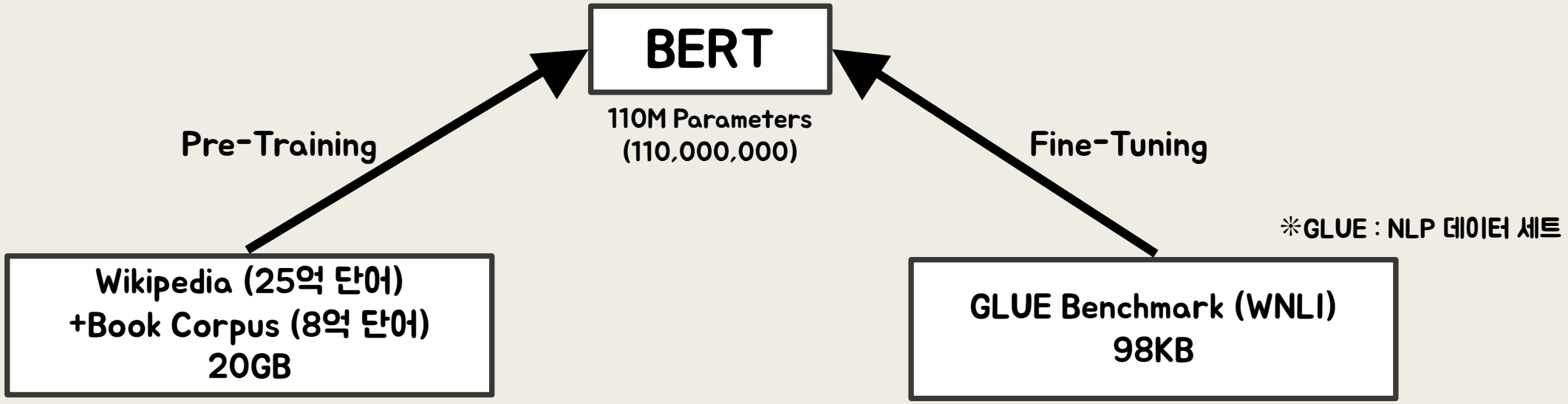
Ex1) AlexNet 의 P/N 비율 = 60,000,000 (Parameter) / 1,200,000 (Training Data) = 50

Ex2) Inception v1 의 P/N 비율 = 23,000,000 (Parameter) / 1,200,000 (Training Data) = 19

Ex3) DenseNet-121의 P/N 비율 = 8,000,000 (Parameter) / 1,200,000 (Training Data) = 6.7

Ex4) BERT의 P/N 비율
= 110,000,000 (Parameter) /
3,3000,000,000 { Wikipedia (약 25억단어) + BooksCorpus(약 8억단어) } = 1/30 = 0.034

I. Complexity



※GLUE : NLP 데이터 세트

P/N = Parameter Count (매개변수)
/ Num Training Data (훈련 샘플)

= 110,000,000 (매개변수)
/ 3,300,000,000 (훈련 샘플 개수) = 0.034

Complexity 낮아, 과적합 가능성 낮음

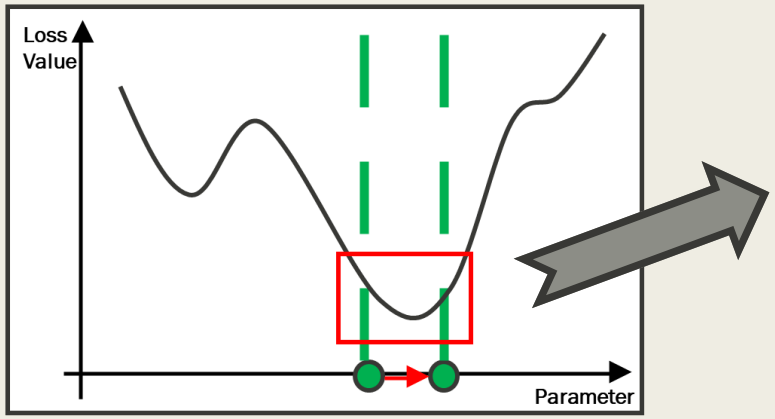
P/N = Parameter Count (매개변수)
/ Num Training Data (훈련 샘플)

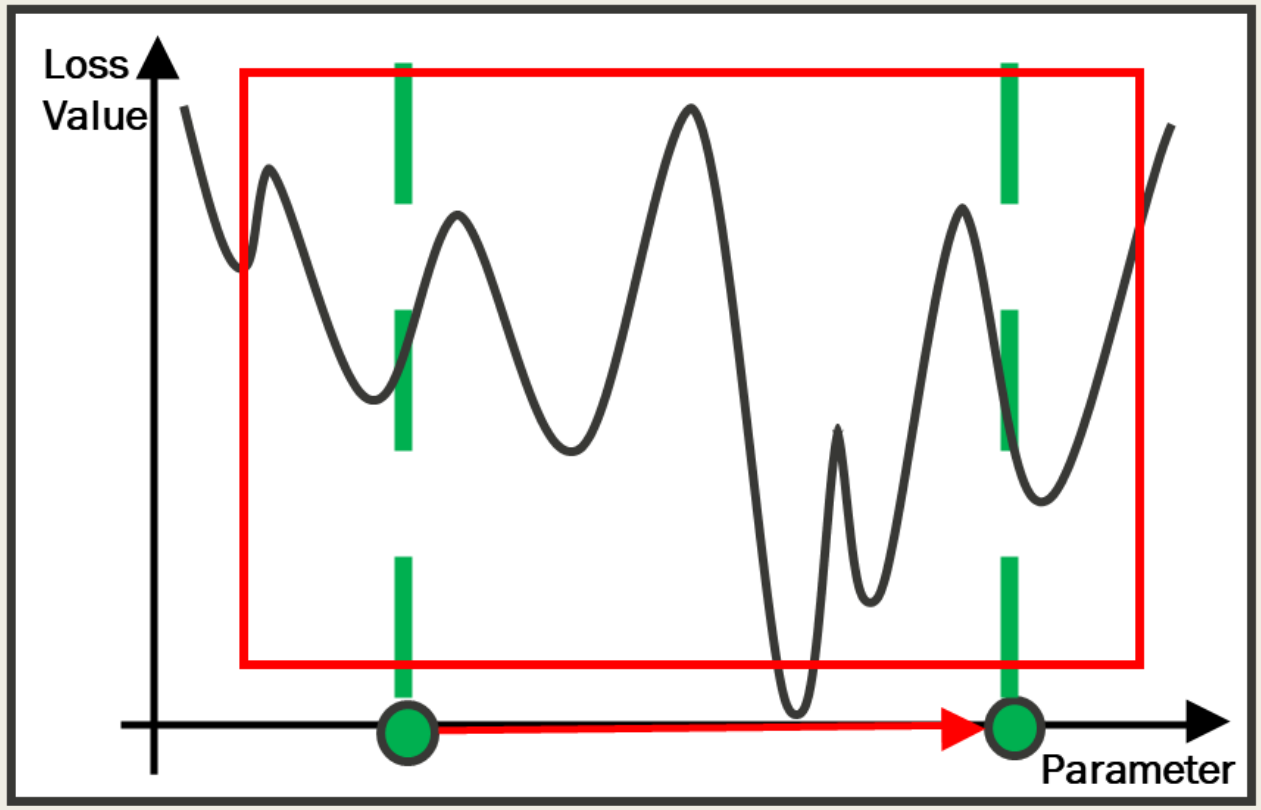
= 110,000,000 (매개변수)
/ 20,000 (훈련 샘플 개수) = 5,500

Complexity 높아, 과적합 됨

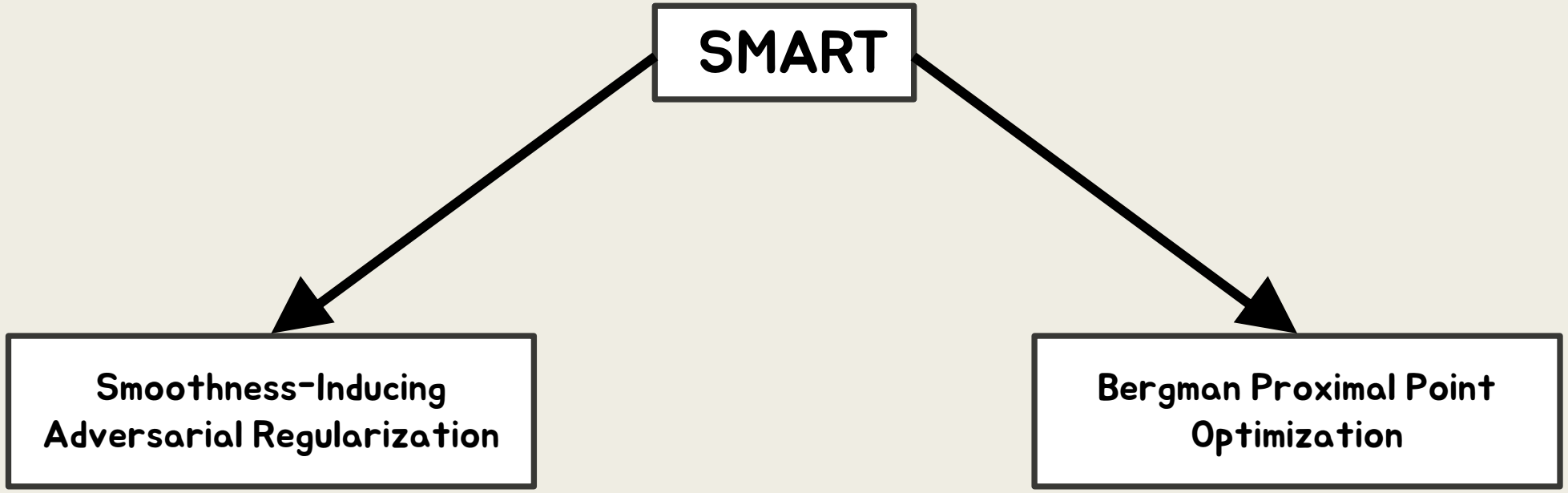
II. Aggressive Updating

- Gradient descent 과정
- Global optimum (최적해) 놓칠 수 있음





III. Smoothness-Inducing Adversarial Regularization



III. Smoothness-Inducing Adversarial Regularization

- 모델이 작은 입력 변화에 대해 부드럽게 반응하게 하는 정규화

$$\min_{\theta} \mathcal{F}(\theta) = \mathcal{L}(\theta) + \lambda_s \mathcal{R}_s(\theta)$$

$$\mathcal{L}(\theta) = \frac{1}{n}\sum_{i=1}^{n} \ell(f(x_i;\theta), y_i)$$

$$\mathcal{R}_s(\theta) = \frac{1}{n}\sum_{i=1}^{n} \max_{\|\tilde{x}_i - x_i\|_p \le \epsilon} \ell_s\big(f(\tilde{x}_i;\theta), f(x_i;\theta)\big)$$

$$\ell_s(P,Q) = \mathcal{D}_{KL}(P||Q) + \mathcal{D}_{KL}(Q||P)$$

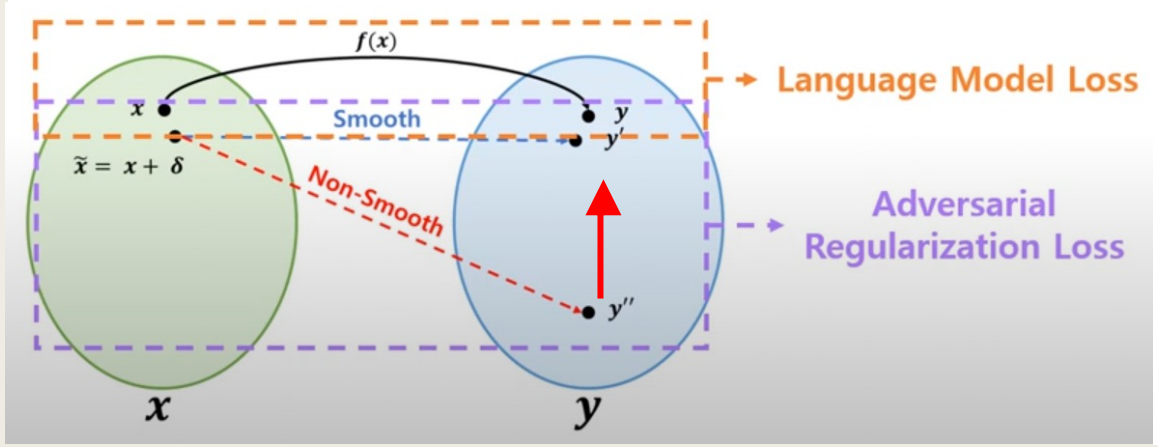
F(θ)를 최소화하는 파라미터 를 찾는 것을 목표
L(θ) : 손실 함수 , R(θ) : Smoothness 증가위한 정규화 항

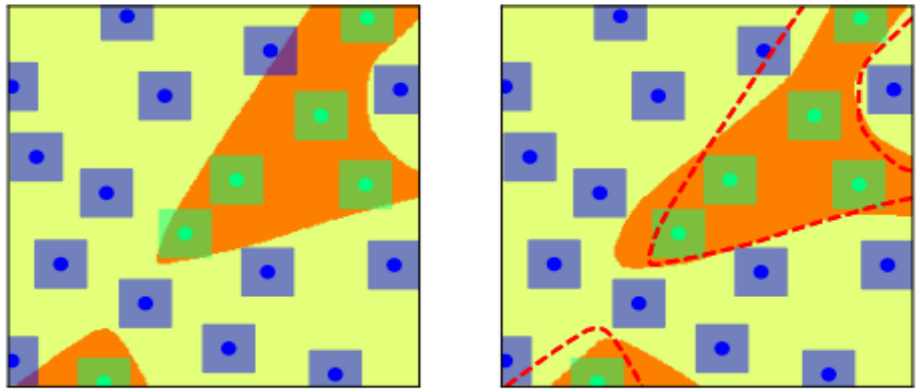
평균 손실 함수 / X : 입력 데이터 , y : 실제 라벨

SAR 항 (Smoothness-Inducing Adversarial Rgularization)

ε 내에서 최대의 적대적 변화를 찾아, 모델의 출력이 변화에 강건하게 함

KL 다이버전스
작은 입력 변화에 대해 출력이 크게 변하지 않도록 함.





(a) W/O SAR (b) W SAR

IV. Bergman Proximal Point Optimization

- 원본과 가깝게 Gradient descent를 수행하는 방법

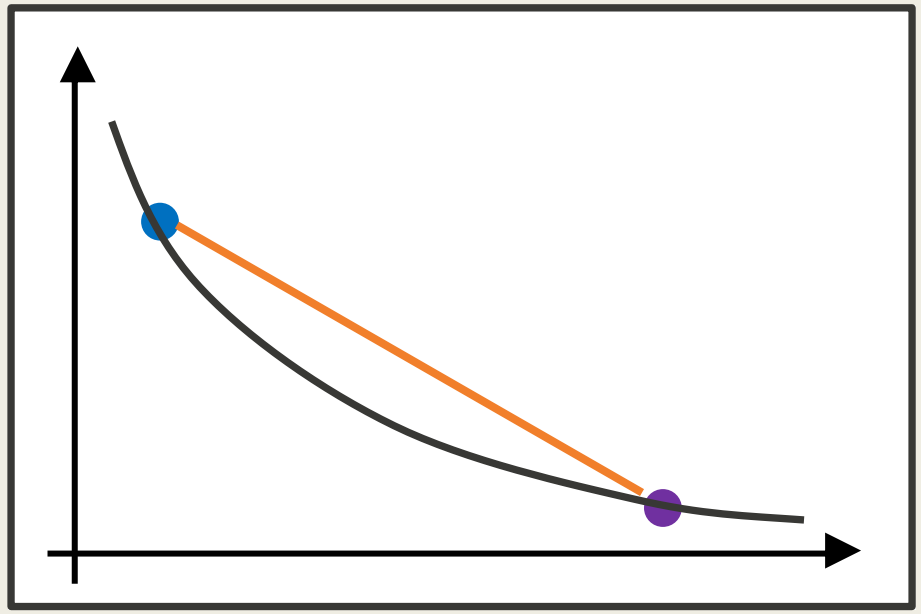
$$f(\cdot;\theta_0): Pre\text{-}Trained\ Model, Initialization$$

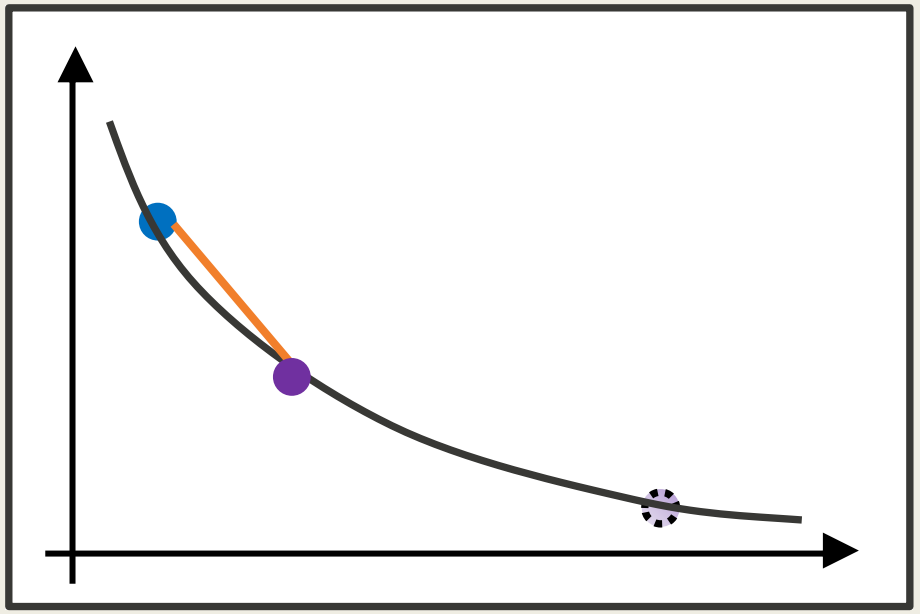
$$\theta_{t+1} = arg\min_{\theta} \mathcal{F}(\theta) + \mu \mathcal{D}_{\text{Breg}}(\theta, \theta_t)$$

$$\mathcal{D}_{\text{Breg}}(\theta, \theta_t) = \frac{1}{n}\sum_{i=1}^{n} \ell_s(f(x_i;\theta), f(x_i;\theta_t))$$

F(θ) 는 모델의 손실 함수
$D_h$reg : Bregman 발산을 기반으로 한 정규화 항
μ : 정규화 항의 가중치

Bregman 발산 기반의 정규화 함수
작은 입력 변화에 대해 예측을 덜 변하는 도움을 줌





PPT:

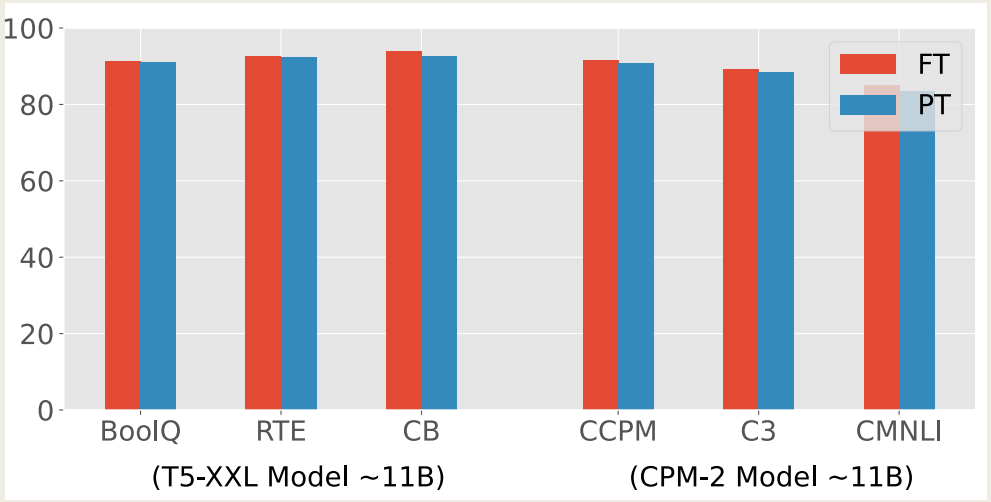
PRE-TRAINED PROMPT TUNING FOR FEW-SHOT LEARNING

A66048
안 수 원

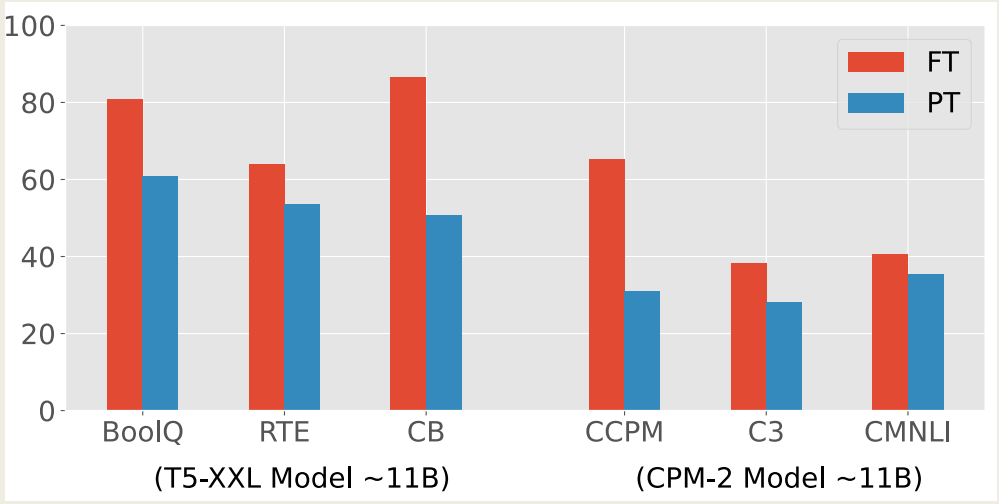
1. Abstract

- Prompts for pre-trained language models(PLMS) have shown remarkable performance by bridging the gap between pre-training tasks and various downstream tasks.

FT : Full-model tuning, PT: Prompt Tuning



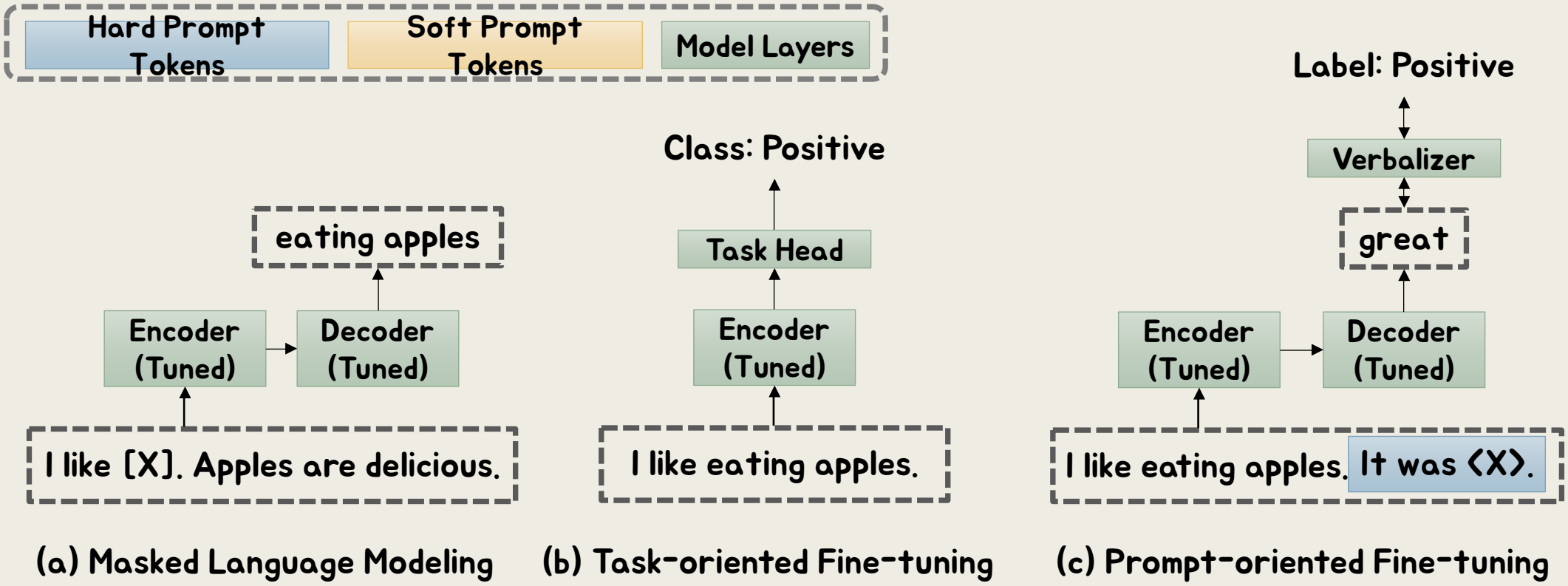
(a) Full-Data



(b) Few-Shot

2. Introduction

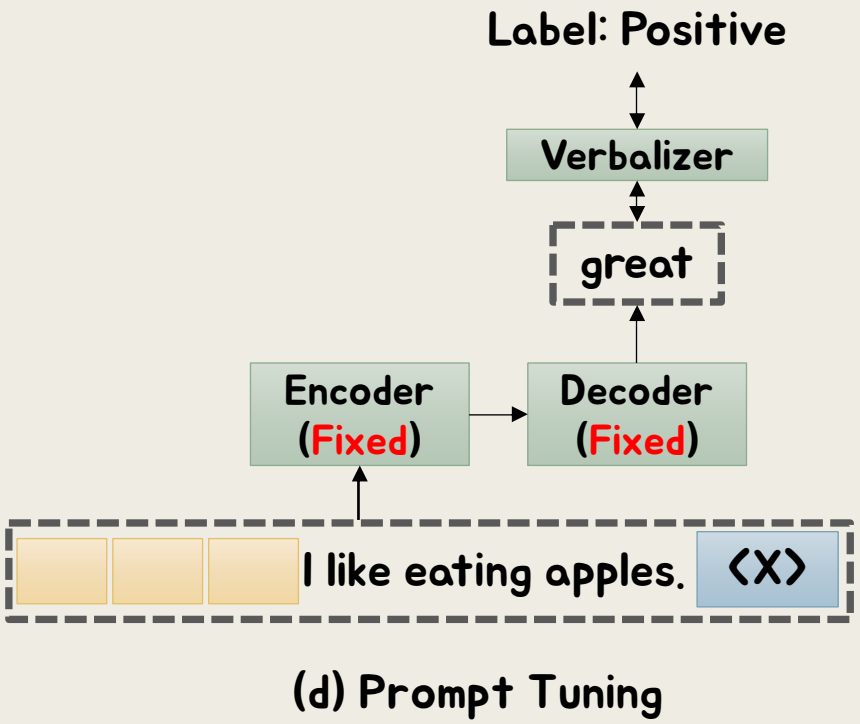
- Paradigms of pre-training, Full-model tuning, Prompt tuning



(a) Masked Language Modeling (b) Task-oriented Fine-tuning (c) Prompt-oriented Fine-tuning

2. Introduction

- Paradigms of pre-training, Full-model tuning, **Prompt tuning**



(d) Prompt Tuning

3. Pilot experiments

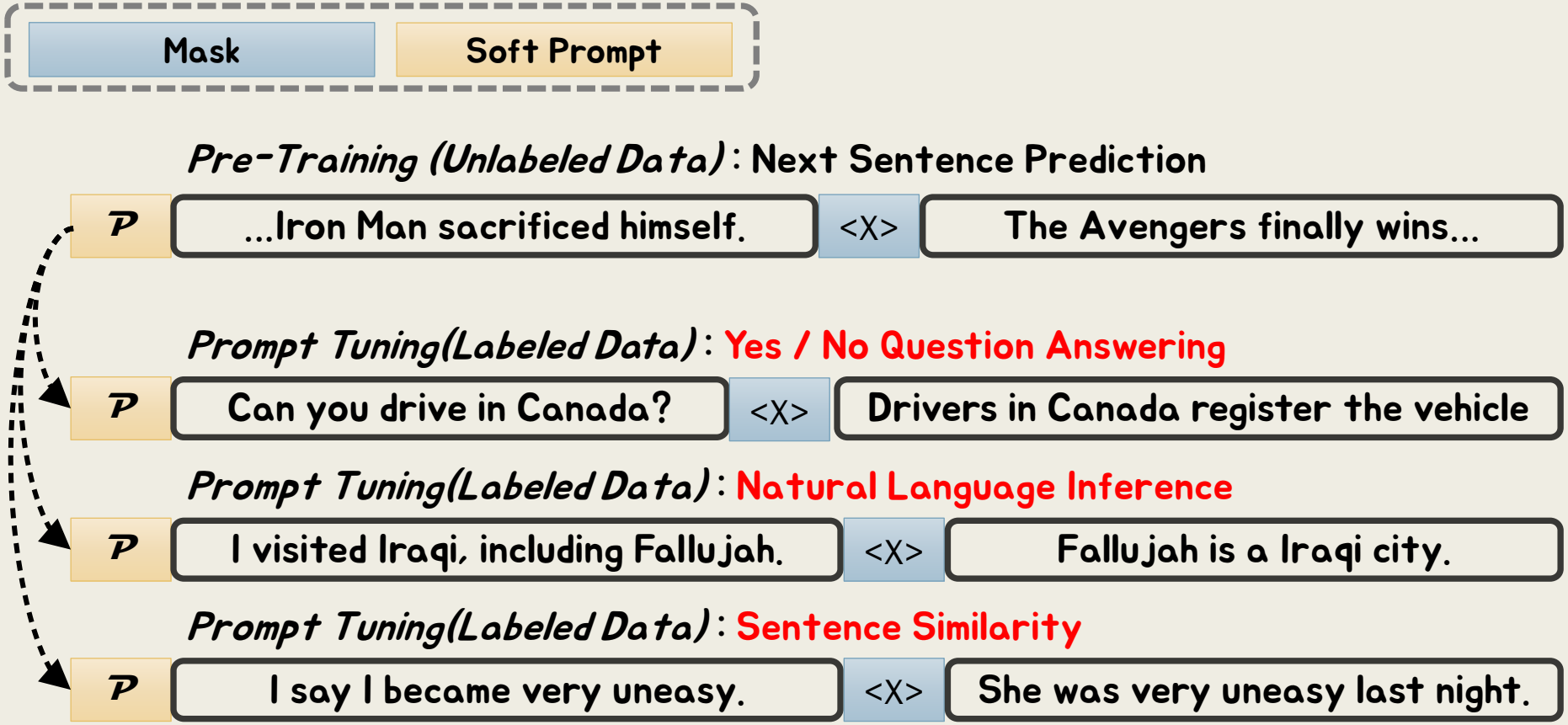
- 실험을 통해 발견한 부분

1. The verbalizer choice has a large impact on the performance
2. Simply initializing soft prompts with concrete word embeddings fails to improve the performance
3. Combining soft and hard prompts is helpful
4. All these methods cannot handle few-shot prompt tuning problems well

→ Prompt searching이 사소한 것이 아니라 신중하게 초기화된 soft prompt token이 중요
이를 위해 Sentence-pair classification, Multiple-Choice classification, single-text classification을 제시

4. Pre-trained Prompt Tuning

★ Overview



4. Pre-trained Prompt Tuning

- Designing Pattern-Verbalizer Pairs for Pre-training

1. *Sentence-Pair Classification Example*

sentence$_1$: 집에 가서 밥을 먹었다. sentence$_2$: 귀가 후 식사를 했다. Label : entailment

2. *Multiple-Choice Classification Example*

Query : 다음 중 뉴진스 멤버가 아닌 사람은? Label : 6

1 : 해린 2 : 민지 3 : 하니 4 : 다니엘 마쉬 5 : 혜인 6 : 유주

3. *Single-Sentence Classification Example*

sentence : 이거 진짜 스토리가 흥미진진! Label : Positive

4. Pre-trained Prompt Tuning

- Designing Pattern-Verbalizer Pairs for Pre-training
  1. *Sentence-Pair Classification*

sentence$_1$: 집에 가서 밥을 먹었다. sentence$_2$: 귀가 후 식사를 했다. Label : entailment

- next sentence prediction(BERT)를 3-class classification으로 확장

. Input : $x = (s_1, s_2)$

. Label : $y = \begin{cases} 0 : \text{irrelevant (두 문장이 서로 다른 Docu에서 추출)} \\ 1 : \text{similar (두 문장이 동일한 Docu에서 추출되었지만 next-sentence가 아님} \\ 2 : \text{coherent (두 문장이 next-sentence 관계)} \end{cases}$

4. Pre-trained Prompt Tuning

- Designing Pattern-Verbalizer Pairs for Pre-training

  2. Multiple-Choice Classification

Query : 다음 중 뉴진스 멤버가 아닌 사람은? Label : 6

1 : 해린 2 : 민지 3 : 하니 4 : 다니엘 마쉬 5 : 혜인 6 : 유주

- S_q: query sentence, S_q와 Next sentence 관계인 문장 고르기

  → model은 S_1~ S_6 6개의 후보군으로부터 인접한 Sentence을 고르는 것을 train

- $\{S_1 \sim S_6\}$: 한 개는 정답, 한 개는 S_q에 인접하진 않지만 같은 Document에서 얻은 문장,
  나머지는 각기 다른 Documents에서 얻은 문장

4. Pre-trained Prompt Tuning

- Designing Pattern-Verbalizer Pairs for Pre-training

  3. Single-Sentence Classification

  sentence : 이거 진짜 스토리가 흥미진진! Label : Positive

  - RoBERT$_{aBASE}$ model을 사용하여 unlabeled Data에 Pseudo Label을 생성하여 사용
    (5-class sentiment classification dataset으로 fine-tuned된 model)

    . Input : $x = (s)$

    . Label : $y = \{1,2,3,4,5\}$ → [terrible, bad, maybe, good, great]

  - Although the above method improves the model performance, we have to point out that it is still limited to generalize to other single-text classifications in different domains and with different numbers of labels

    → Unifying Task Formats
    unified to a single format : multiple-choice classification

Key Words

- Soft Prompt
- IDEA : Pre-training model → Pre-train Prompt Initialization
- Problem : few-shot setting에서 Prompt Tuning의 단점을 Soft Prompt Token을 통해 해결
- Pre-trained prompt의 일반화
  1) sentence-pair classification
  2) multiple-choice classification
  3) single-sentence classification
  → 이 3가지를 하나로 통합할 수 있다. : multiple-choice classification
  We name this Pre-trained Prompt Tuning framework 'PPT'

MEASURING INDUCTIVE BIASES OF IN-CONTEXT LEARNING WITH UNDERSPECIFIED DEMONSTRATIONS

A68045
배성은

0. In-Context Learning

- fine-tuning과 다르게 LLM 자체를 건들이지 않고, inference시에 질문을 잘 해보자는 접근
- LLM을 다시 훈련시키지 않음.
- 정보(Context)를 주는 방식으로 사용합니다.
- 예시를 몇개 주는지에 따라 zero, few shot으로 나뉨.
- 정확도로 보면 fine-tuning이 좋겠지만, LLM크기, 비용, 성능 등을 고려하였을 때 좋음.
- Chat-GPT

1. Abstract & Intro

- the inductive biases of ICL from the perspective of feature bias

1) GPT-3 models은 어떤 feature bias를 가지고 있는지?

  - Feature biases를 발견 (구두점과 같은 얕은 어휘 특징보다는 감정에 따라 레이블을 예측하는 강한 편향을 보여줌)

2) 특정 task과 관련된 feature에 치우치도록 하는 In-Context Learning은 성능이 좋아질까? (Intervention)

  - can influence the learner to prefer a particular feature, it can be difficult to overcome strong prior biases

1. Abstract & Intro

- inductive biases
  - 학습에는 없었던 상황에 대해, 정확한 예측을 하기 위해 사용하는 추가적인 가정.
  - 학습에는 없었던 데이터에 대해, 판단을 내리기 위해 가지고 있는 학습에서 얻은 bias
  - 일반화의 성능을 높이기 위해 만약의 상황에 대한 추가적인 가정이다.
- Relational Inductive Bias
  - Layer의 구조를 통해 발생

Component	Entities	Relations	Rel. inductive bias	Invariance
Fully connected	Units	All-to-all	Weak	-
Convolutional	Grid elements	Local	Locality	Spatial translation
Recurrent	Timesteps	Sequential	Sequentiality	Time translation
Graph network	Nodes	Edges	Arbitrary	Node, edge permutations

Table 1: Various relational inductive biases in standard deep learning components. See also Section 2.

1. Abstract & Intro

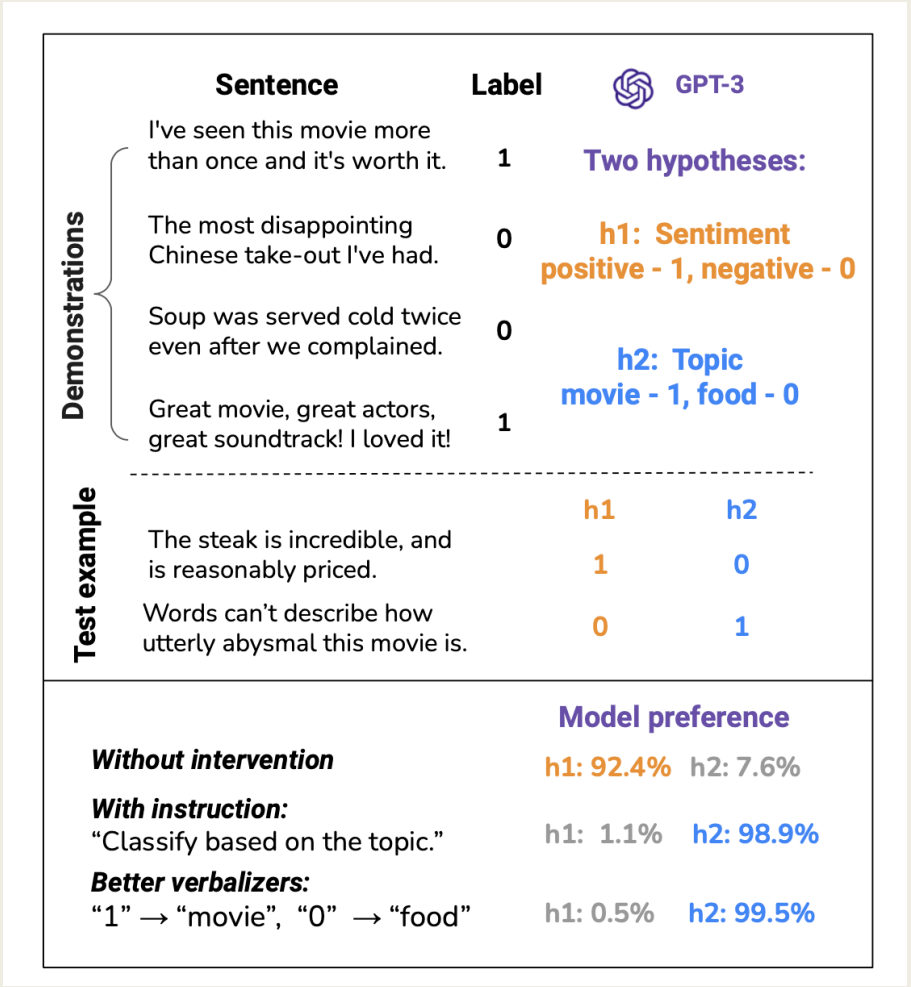
- inductive biases ICL에서 중요한 이유
  - a key limitation of ICL : most tasks will be highly underspecified
    - due to the context length limit of Transformer models

  - ICL이 효과적이기 위해선 다음과 같으면 된다.
    1) the LLM has an inductive bias that happens to align well with the given task
    2) a mechanism for imposing an inductive bias on the system, which can specify the task

Underspecified : 모델에게 명시화되지 않은 텍스트 문제

2. Setup

- Text classification problem
  - *X : input text (영화 or 식당 리뷰 데이터)*
  - *h1 : sentiment classifier*
  - *h2 : domain classifier*
  - *D : h1과 h2를 모두 적용할 수 있는 데이터셋*



2. Setup

- Text classification problem
  - Underspecified demonstrations (*demonstration*)
    - 감성분류와 도메인분류에서 동일한 결과가 나오는 데이터
  - Disambiguating dataset (*test*)
    - 감성분류와 도메인분류에서 다른 결과가 나오는 데이터

  - Instance template
    - Instance(demonstration, test example)
  - Label verbalizer
    - v(0) = 'bad', v(1)='Good'
  - Prompt = instance template + label verbailzer

<demonstration>
[긍정 텍스트, 1]
[부정텍스트, 0]
<test>
[테스트데이터]

➔<prompt>
1 or 0 ?

2. Setup

- Text classification problem
  - h-accuracy
    - h1-accuracy > h2-accuracy
      - *h1과 관련된 feature bias를 가지고있을 것으로 추정*

$$h\text{-accuracy} = \frac{1}{|\mathcal{D}_{\text{test}}|} \sum_{x \in \mathcal{D}_{\text{test}}} \mathbb{1}[f(x) = h(x)].$$

3. Data Construction

- h1, h2
  - h1 label : 데이터셋 라벨 그대로 사용
  - h2 label
    - ambiguity or spurious correlation
      : 감성분류(h1) 또는 도메인분류(h2)
    - Shallow features
      : 길이, 대문자, 특정단어 또는
      문장 부호의 존재 여부

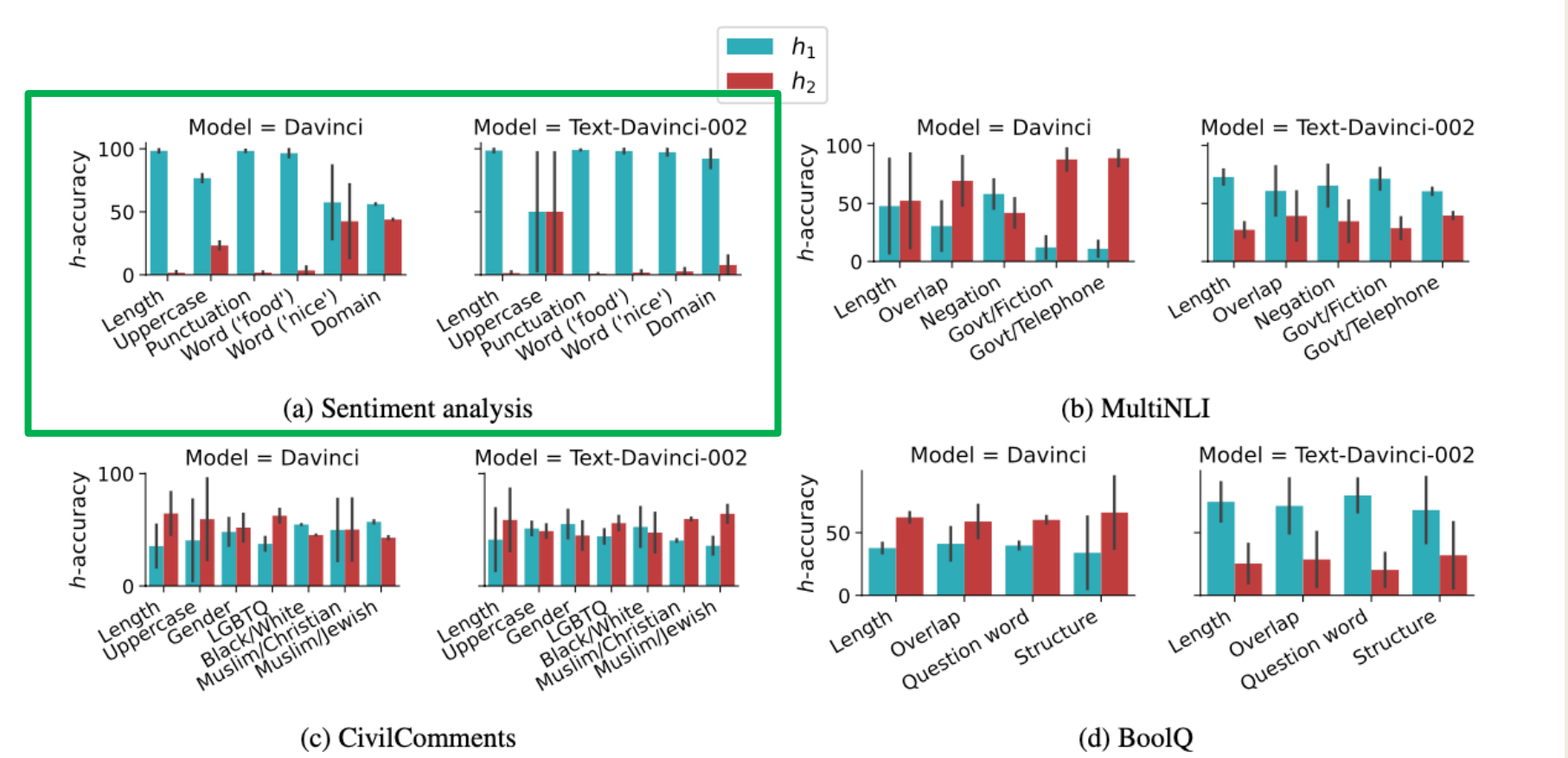
Hypotheses	
Sentiment (positive vs. negative)	h1
Domain (IMDb vs. Yelp)	
Length (short vs. long)	
Terminal punctuation (exclamation vs. period)	h2
Contains word ("nice"/"food")	
Capitalization (lowercase vs. uppercase)	
Toxicity (toxic vs. non-toxic)	
Gender (female vs. male)	
Sexuality (LGBTQ vs. non-LGBTQ)	
Religion (Muslim vs. Christian; Muslim vs. Jewish)	
Race (Black vs. White; Asian vs. White)	
Length (short vs. long)	
Capitalization (lowercase vs. uppercase)	
Entailment (entailment vs. non-entailment)	
Domain (government vs. fiction; government vs. telephone)	
Lexical overlap (overlap vs. non-overlap)	
Hypothesis length (long vs. short)	
Hypothesis negation (contains "not", "n't", "no")	
Answer (yes vs. no)	
Question word ("is/was" vs. "do/does/did")	
Lexical overlap (overlap vs. non-overlap)	
Question structure ("is x the same as y")	
Passage length (short vs long)	

4. Measuring Feature Biases

- Experiment Details
  - Demonstration 16개
    - h1(x) = h2(x) = 1 8개
    - h1(x) = h2(x) = 0 8개
    - 3개의 random set을 만들고 사용
  - Test examples 1200개
    - h1(x)=1, h2(x)=0 600개
    - h1(x)=0, h2(x)=1 600개
  - Model
    - Davinci
    - Text-Davinci-002 : instruction tuning
  - Metric
    - h-accuracy를 통해 어떤 feature를 더 선호하는지 파악 가능

4. Measuring Feature Biases

- Results

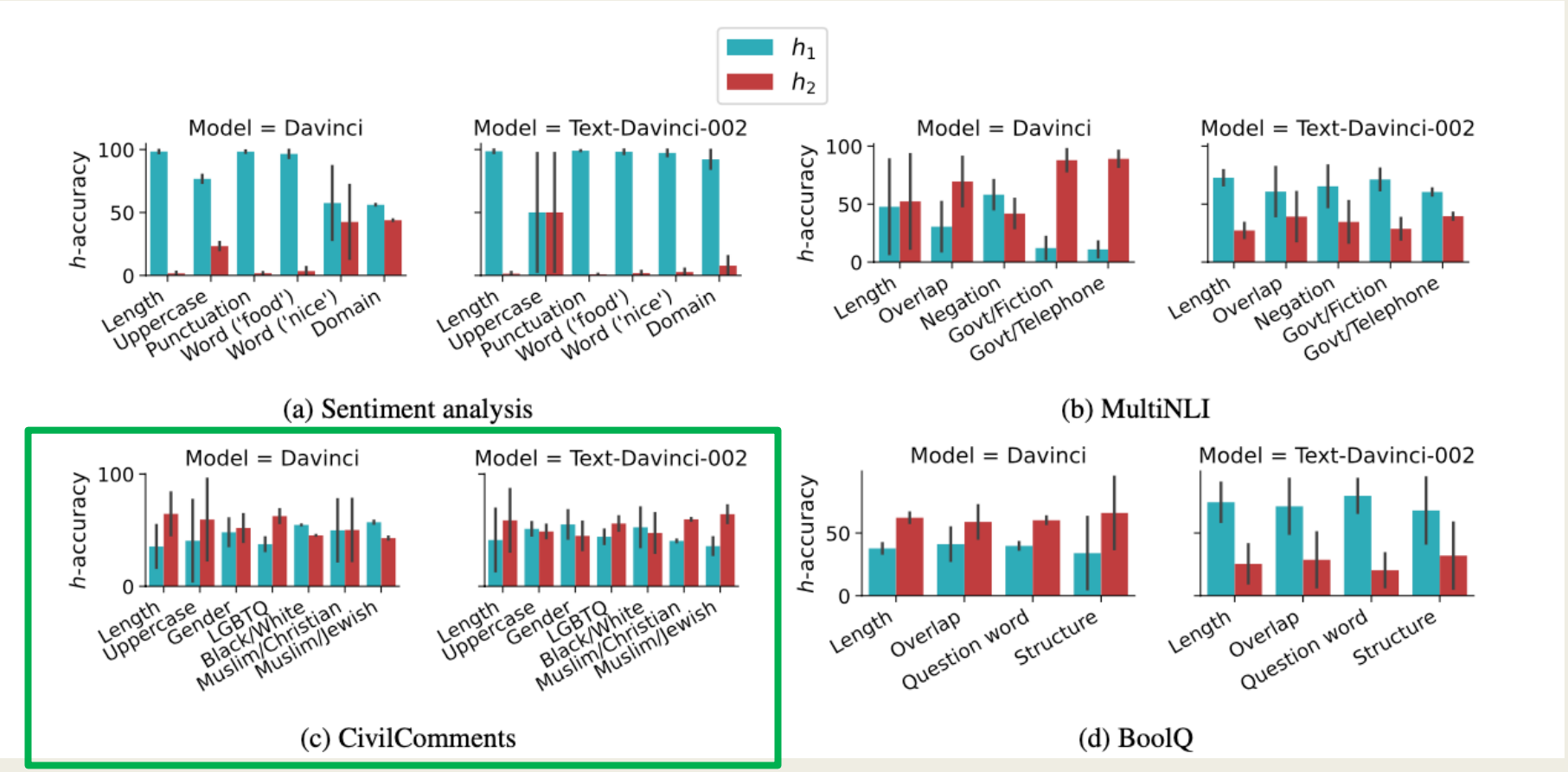


h1-acc>h2-acc

해당 feature bias는 sentiment analysis에 도움이 된다.

4. Measuring Feature Biases

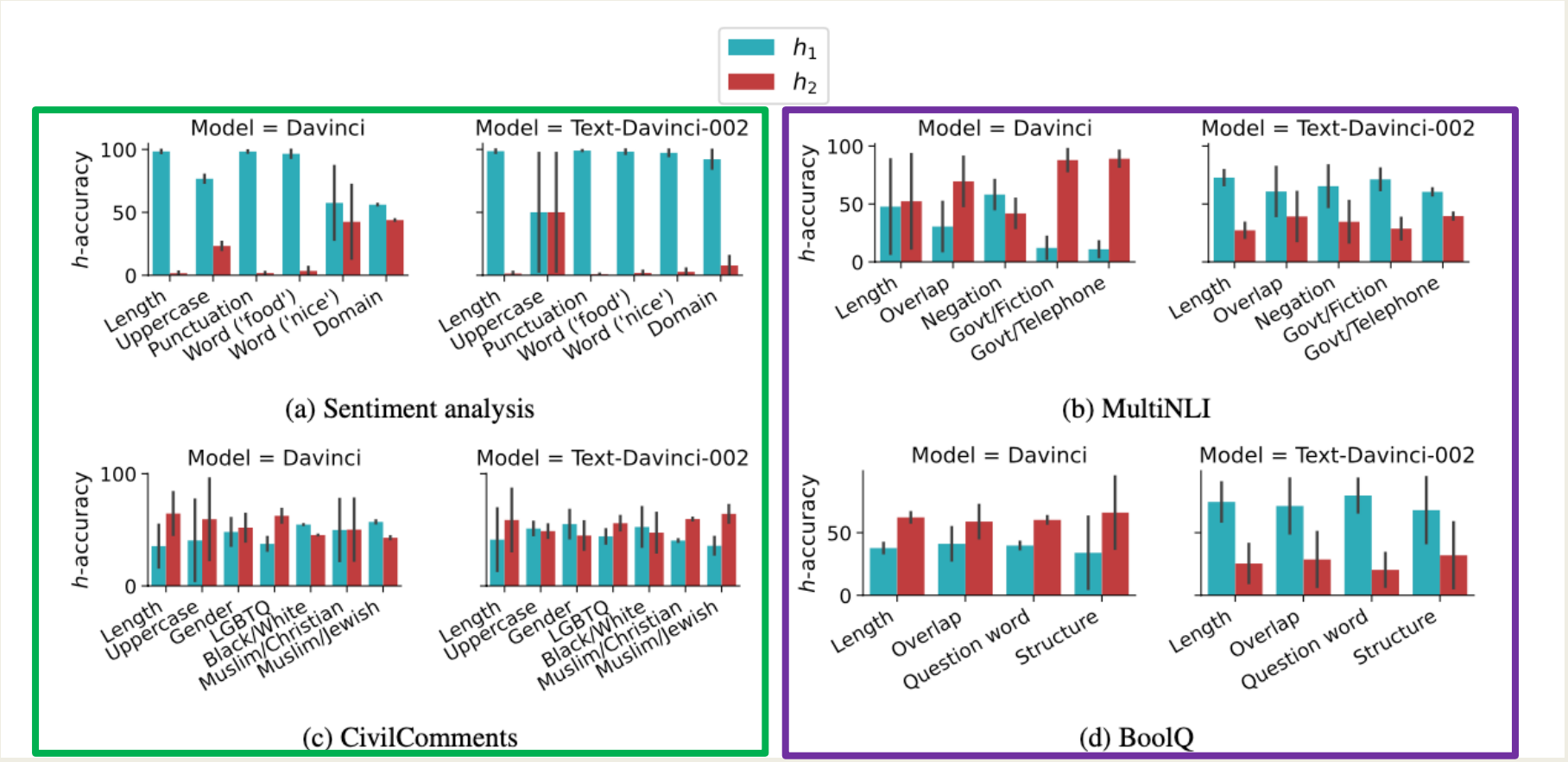
- Results



h1-acc<h2-acc
Task와 관련된 feature bias가 강하지 않음.

4. Measuring Feature Biases

- Results



(a), (c) : 두 모델이 비슷한 feature bias
(b), (d) : 두 모델이 다른 feature bias

Davinci : shallow distractor feature를 선호
Text-Davinci-002 : semantic feature 선호
→ instruction tuning의 영향

5. Comparing Interventions

- 실험
  - when the LLMs biases do **not align** with users intended task, such biases would hurt performance.
  - To resolve such misalignment, 원하는 feature bias를 가질수있도록 'intervention'을 주자

- 리뷰 X 논문에서 확인하세요

6. Conclusion

- Inductive biases of In-Context Learning
  - Intended task와 관련된 feature bias를 가지고 있다면 task 수행에 효과적임.
  - Prompt 구성 요소에서 intervention을 주어 intended task와 관련된 feature bias를 갖도록 함.
  - 강한 prior feature bias가 있는 경우, intervention이 어려움.