

HyperCLOVA – Korean GPT3

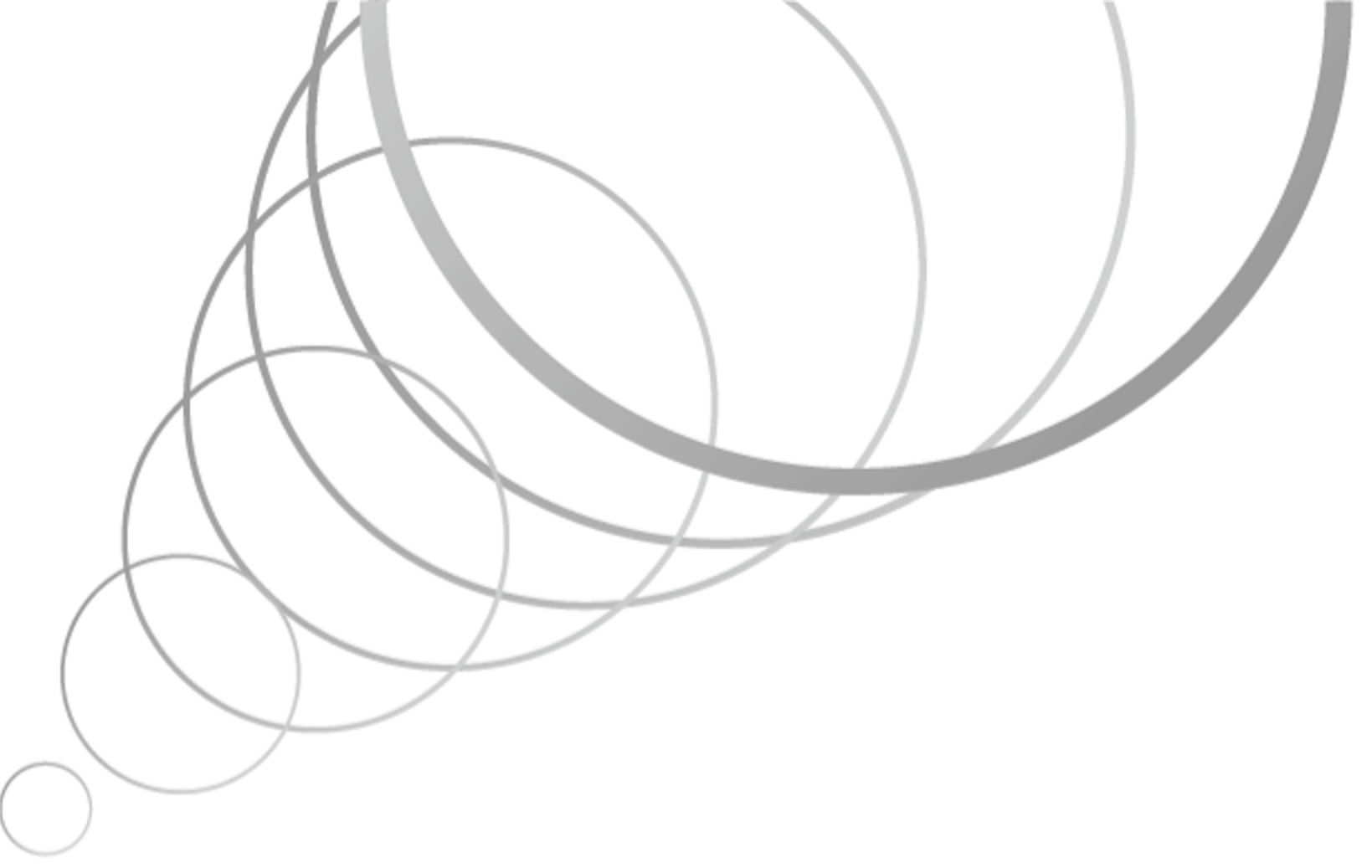
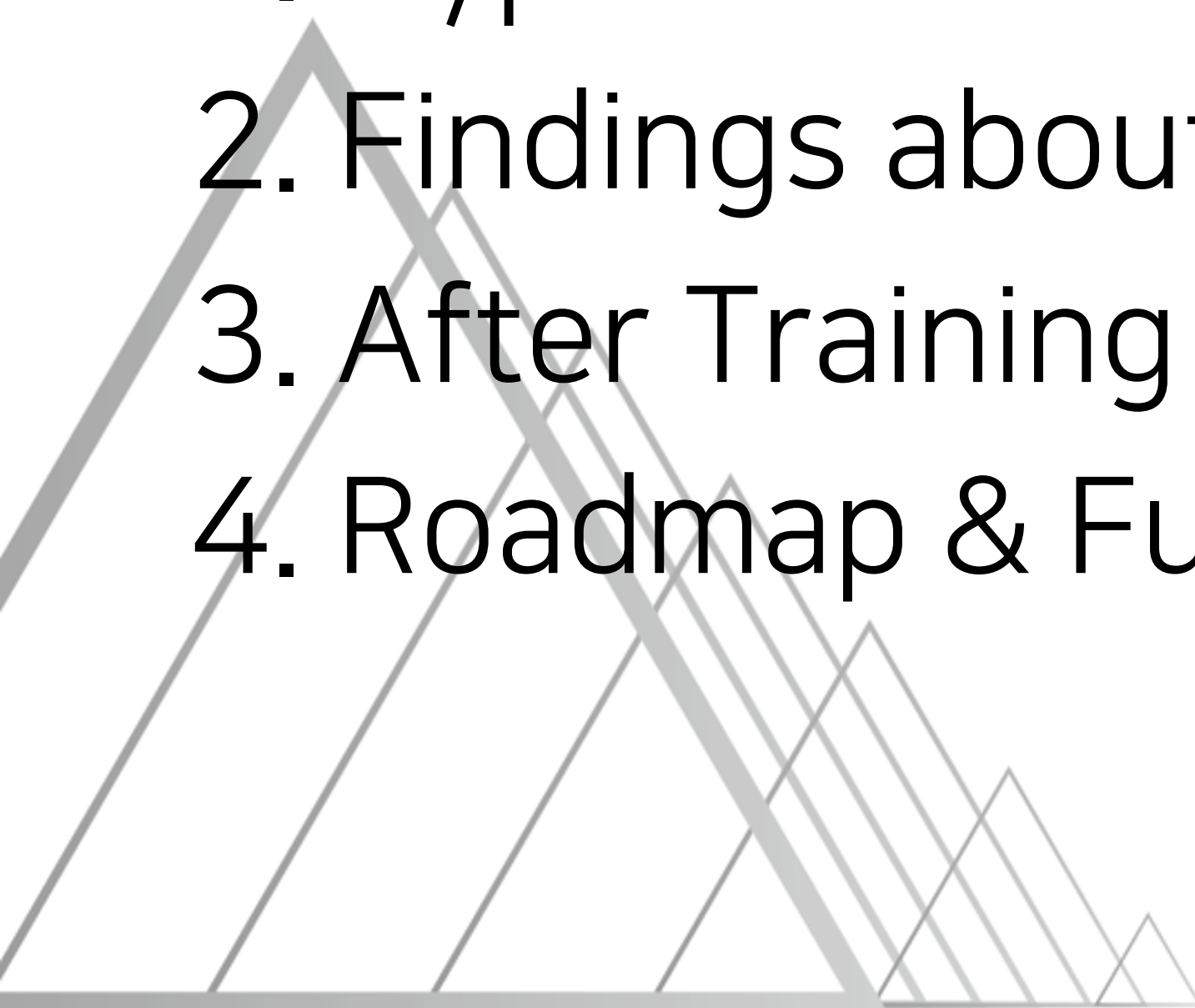
하이퍼스케일, 그리고 그 후



김형석, 이상우 CLOVA Conversation / AI Lab

CONTENTS

1. HyperCLOVA Language Model
2. Findings about HyperCLOVA
3. After Training Hyper-scale LM
4. Roadmap & Future Works



1. HyperCLOVA Language Model

1.0 AI 관점의 질문

자연어 처리 AI를 더 똑똑하게 만들기 위하여...

Brown et al., Language Models are Few-shot Learners, NeurIPS, 2020.

- 언어 데이터를 더 많이 학습할 수록 더 똑똑해질까?
- 모델 사이즈를 키우면 더 똑똑해질까?
- 세상에 있는 모든 글을 학습하면 인간과 같은 지능이 나올 수 있을까?

=> 초거대 언어 모델의 시대가 도래: GPT-3

- 하지만 기존 모델은 영어 중심 모델이었다는 한계가 존재
- 한국어 언어 모델의 필요성이 생김

1.1 HyperCLOVA 언어 모델 개요

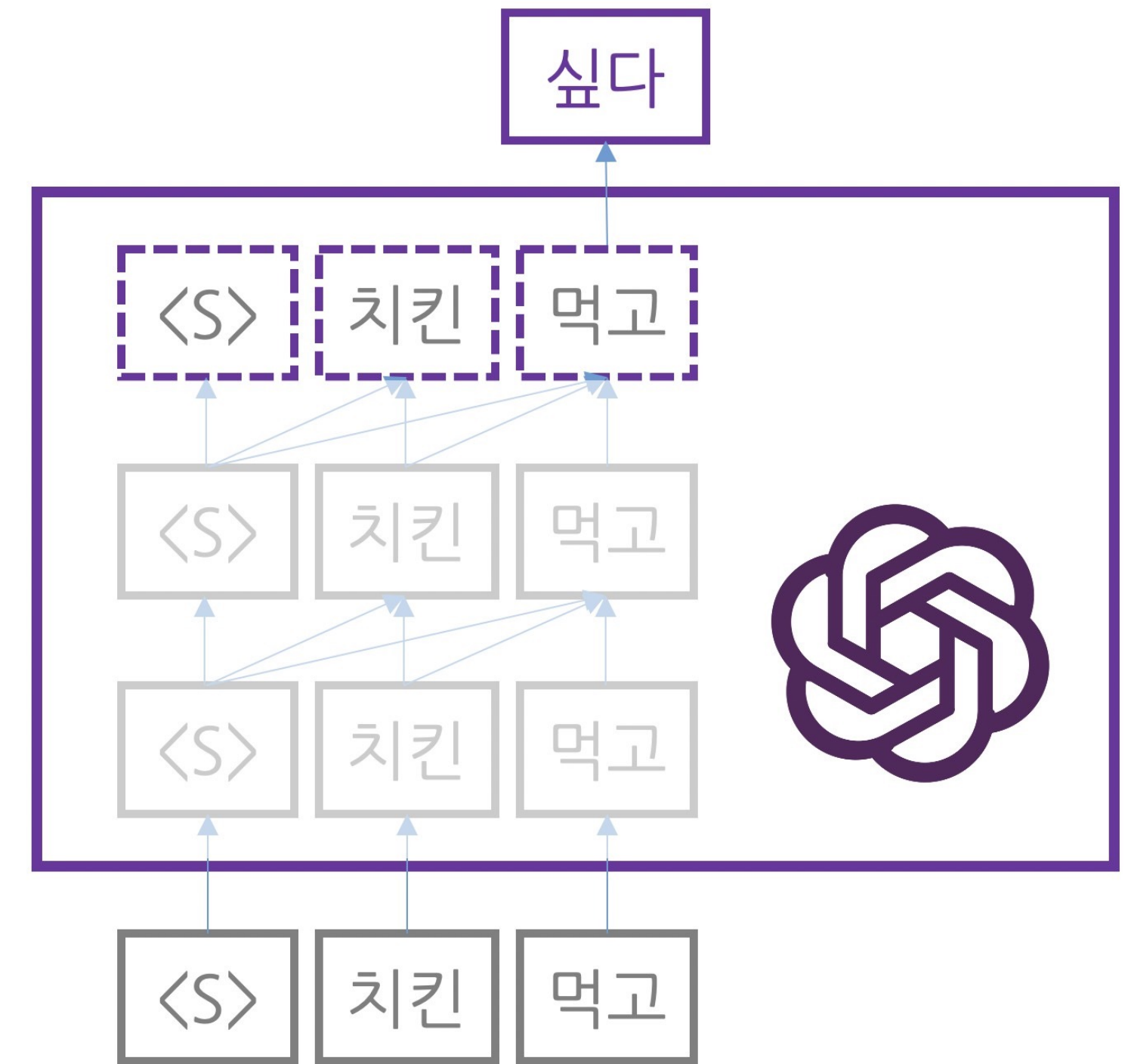
초거대 한국어 언어 모델의 등장: HyperCLOVA 언어 모델

- Transformer decoder 82B 파라미터 사이즈
- 1,120 GPU (A100) superpod
- 562B 한국어 데이터

1.1 HyperCLOVA 언어 모델 개요

82B 파라미터 사이즈의 transformer decoder

# Param	n_{layers}	d_{model}	n_{heads}	d_{head}	lr
137M	12	768	16	48	6.0e-4
350M	24	1024	16	64	3.0e-4
760M	24	1536	16	96	2.5e-4
1.3B	24	2048	16	128	2.0e-4
6.9B	32	4096	32	128	1.2e-4
13B	40	5120	40	128	1.0e-4
39B	48	8192	64	128	0.8e-4
82B	64	10240	80	128	0.6e-4



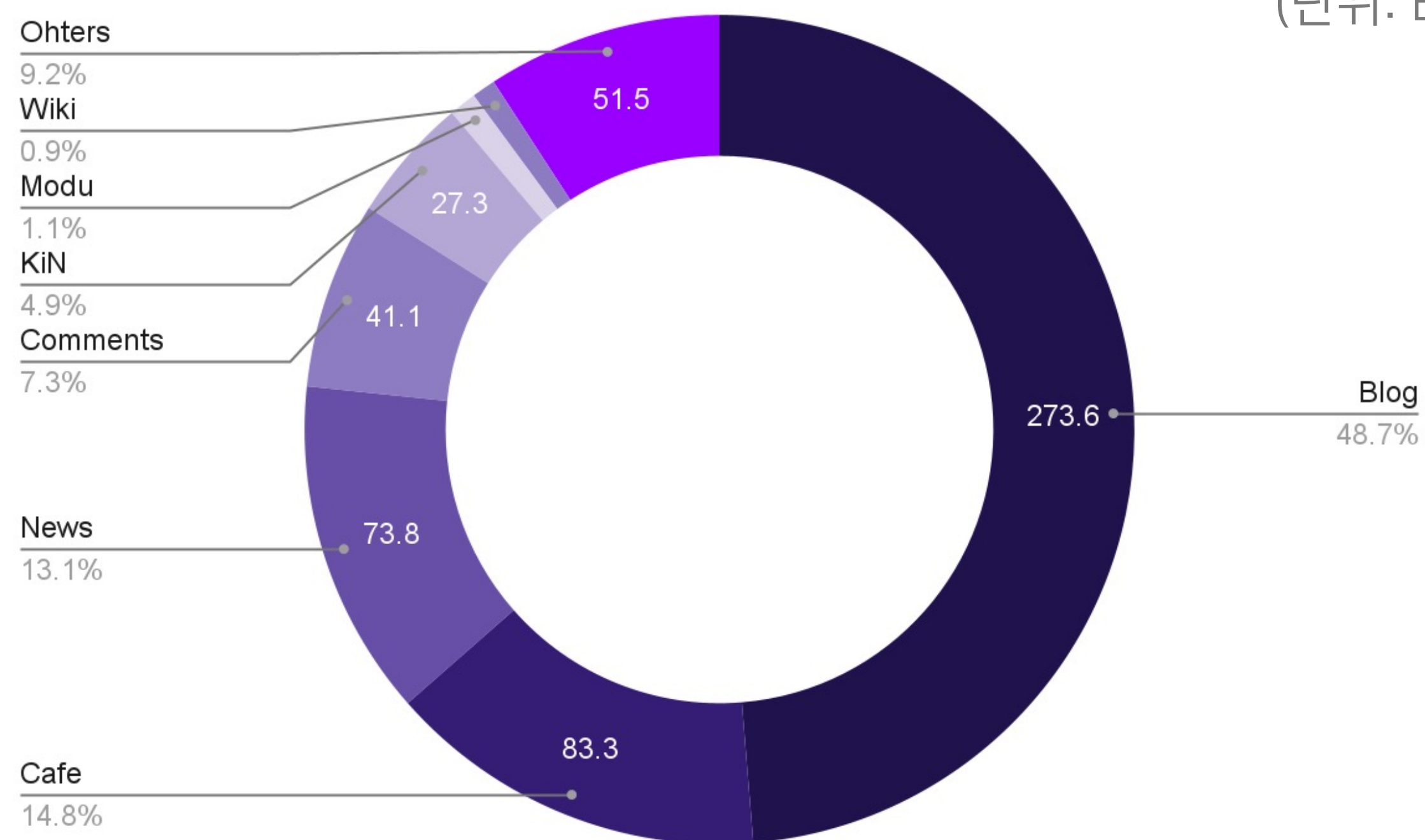
82B 학습 - 150B token 기준 Superpod에서 13.4일 소요 (실제로는 학습 방식에 따라 차이가 남)

1.1 HyperCLOVA 언어 모델 개요

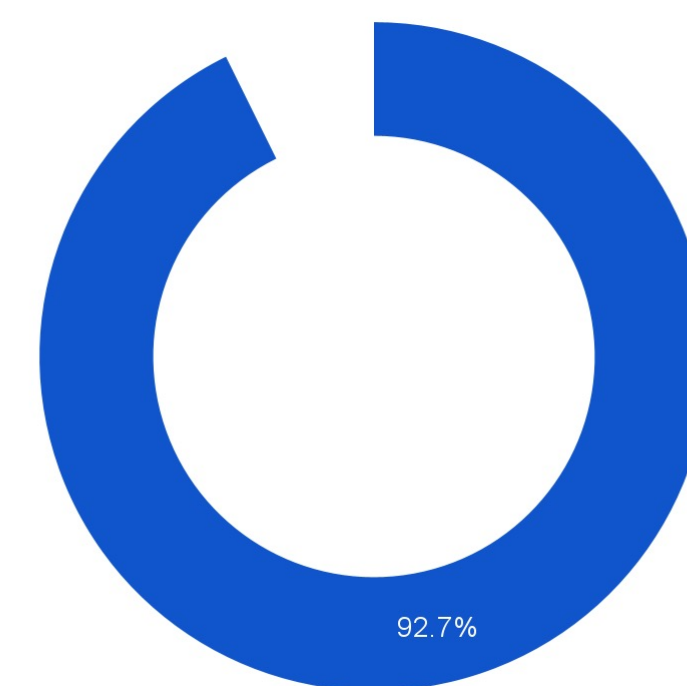
562B token의 다양한 한국어 데이터

Data Description

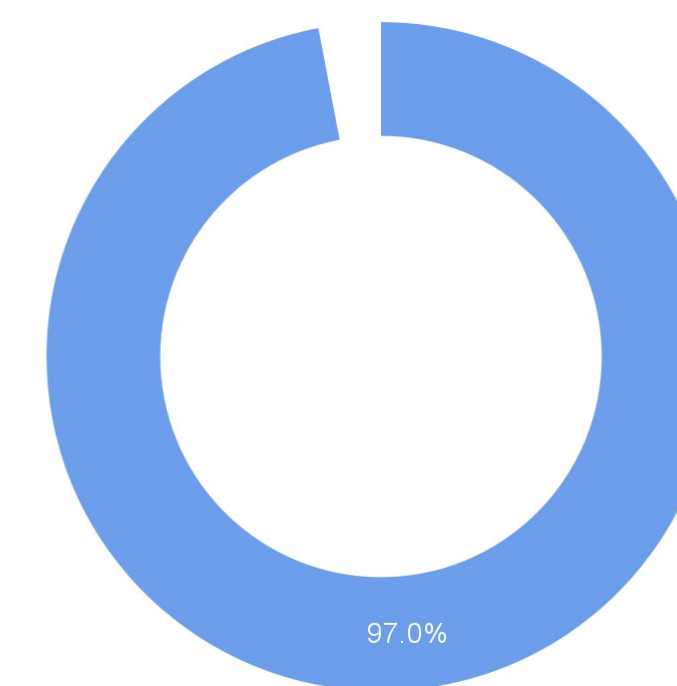
(단위: B)



Language Composition



GPT-3
영어 92.7%



HyperCLOVA
한국어 97.0%

1.1 HyperCLOVA 언어 모델 개요

Tokenization: Morph-aware byte-level BPE

- byte-level encoding + BPE + 형태소 분석기
- 기본적으로 byte-level encoding을 사용 (char-level이 아닌)
- 문장을 공백과 형태소 기준으로 미리 나눈 후에, BPE를 사용

sentence

마감이 잘 안돼서 옆부분이 안맞은게
불편했고 흰색이라 어쩔 수 없긴 하지만 때도
잘타요.

tokenized by Character-level BPE Tokenizer

['마감이</w>', '잘</w>', '안돼서</w>', '옆', '부분이</w>', '안', '맞은', '게</w>', '불편', '했고</w>', '흰색', '이라</w>', '어', '<unk>', '수</w>', '없', '긴</w>', '하지만</w>', '때도</w>', '잘', '타', '요.</w>']

tokenized by Byte-level BPE tokenizer

['마', '감이', '잘', '안돼서', '옆', '부분이', '안맞', '은', '게', '불편', '했고', '흰색', '이라', '어쩔', '수', '없', '긴', '하지만', '때도', '잘', '타', '요', '!']

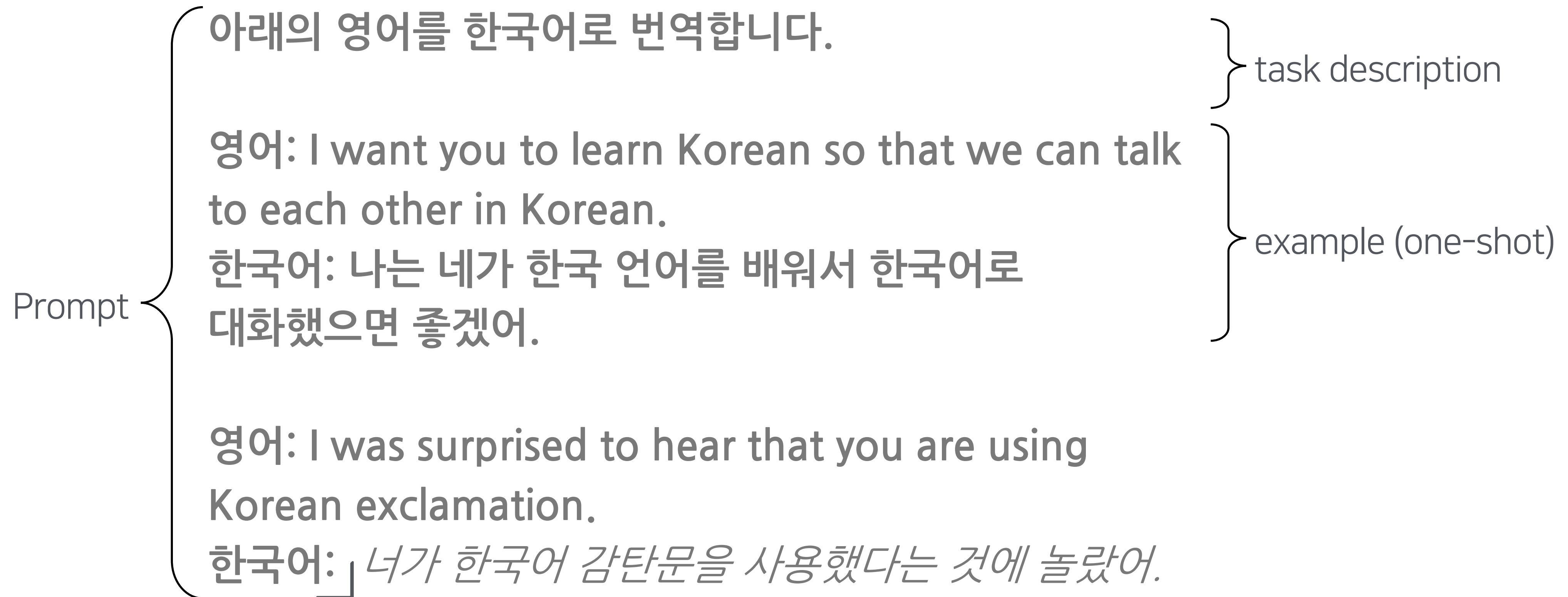
tokenized by Morpheme-Aware Byte-level BPE tokenizer

['마감', '이', '잘', '안돼서', '옆', '부분', '이', '안', '맞은', '게', '불편', '했고', '흰색', '이라', '어쩔', '수', '없긴', '하지만', '때', '도', '잘', '타', '요', '!']

1.2 HyperCLOVA 언어 모델의 능력

HyperCLOVA의 핵심 기능: In-context learning

- 모델 파라미터 업데이트 (fine-tuning) 없이 prompt에 정보를 넣어주어 언어 모델에게 자연어 처리 문제를 풀게 하는 학습 패러다임



1.2 HyperCLOVA 언어 모델의 능력

HyperCLOVA Studio & HyperCLOVA API

HyperCLOVA «

Studio

tooltip language 🌐

Engine

🇰🇷 prod-chimmy ▾

Top P 0.8

Top K 0

Response Length 32

Temperature 0.5

Repetition Penalty 5

Stop sequences

Enter sequence and press Tab

↵ ✕

이메일 작성기

나의 삶을 편하게 해달라

Choose tags ▾

🗑️ 🔒 </> 이메일 작성기 ▾

다음은 대화형 명령을 기반으로 이메일 답변을 써주는 생성기입니다.

이메일: 안녕하세요.
투자 자문건으로 기술력 검증 미팅 일정 문의 드립니다.
캘린더상으로 다들 관찮으신 일정은 5/12(수) 오후 3시인데, zoom 미팅 어떠신지
요?
일정 확인 부탁 드립니다.
명령: "No"
답변:
안녕하세요,
다른 일정으로 부탁드립니다.
감사합니다.

이메일:
안녕하세요,
오늘 잠깐 정우님과 이야기를 나누었는데, 연구소에서 인턴 채용을 하는 것 보다는
마침 TF가 생겼으니 이쪽에서 선발하여 좀 더 긴밀한 협업을 하는게 더 좋다는 의견
이시네요. 어떻게 생각하시나요?
감사합니다
명령: "Okay"
답변:
안녕하세요,
네 그렇게 하겠습니다.

Rolling

One-time

Multiple

Examples

Conversations

Calibr:

Rolling Generation

Submit and get results via prompt. (Shortcut: Ctrl+Enter or ⌘+Enter)

Submit →

1.2 HyperCLOVA 언어 모델의 능력

다양한 HyperCLOVA DEVIEW 발표들

- 세상에 쓸모없는 데이터는 없다: HyperCLOVA를 이용한 반지도 학습
- 다국어로 말해요, HyperCLOVA!
- Bring Your Own Data: Business AI 고민? HyperCLOVA에게 무엇이든 물어보살
- 세상 빠르고 안전한 챗봇 만들기 (Feat. HyperCLOVA)
- 네이버 수요예측의 비밀 HyperCLOVA, 네이버 빅데이터에서 예측시그널을 찾을 수 있는 이유
- 말투, 성격, 프로필 그리고 챗봇
- 지식백과 Question&Answering: HyperCLOVA 지식의 한계에 도전합니다
- 대규모 자연어처리 모델 서빙 경험기
- 로그 없이 영끌 키워드 추천: Few-shot Learning and Sequential Pattern based Keyword Recommendation
- 초거대 AI, 'HyperCLOVA' 서빙기

2. Findings about HyperCLOVA

2.0 HyperCLOVA 논문 및 성능 소개

HyperCLOVA 논문

Kim et al., What Changes Can Large-scale Language Models Bring? Intensive Study on HyperCLOVA: Billions-scale Korean Generative Pretrained Transformers, EMNLP, 2021.

최초의 질문

한국어로 학습한 GPT3는 in-context learning 잘 될까?
모델 크기에 따라서 in-context learning 성능은 어떨까?
최근 나온 **prompt-tuning** 기법들을 적용한 성능은 어떨까?

Executive Summary

한국어 560B 토큰으로 82B GPT3 모델을 학습했다
한국어 **few-shot** 성능에서 **SOTA** 성능을 달성했다
한국어 특화 **tokenizer**를 적용하고 성능을 비교했다
prompt-tuning 기법을 사용했더니 성능 향상이 있었다
No Code AI도 가능하다
GPT3를 사용한 Application도 소개하겠다

2.1.1 In-context Learning 성능

실험한 In-context learning 데이터셋 소개

데이터셋	특징	Shots
NSMC	긍부정 분류	70
KorQuAD	문서 독해	4
AI Hub	한영 번역, 영한 번역	4
YNAT	주제 분류	70
KLUE-STC	문장 유사도	40

2.1.1 In-context Learning 성능

In-context Learning 결과: 적은 데이터만으로도 훌륭한 성능

	긍부정 분류	문서 독해		한/영, 영/한 번역		주제 분류	문장 유사도
	NSMC (Acc)	KorQuAD (EA / F1)		AI Hub (BLEU) Ko→En En→Ko		YNAT (F1)	KLUE-STS (F1)
Baseline	89.66	74.04	86.66	40.34	40.41	82.64	75.93
137M	73.11	8.87	23.92	0.80	2.78	29.01	59.54
350M	77.55	27.66	46.86	1.44	8.89	33.18	59.45
760M	77.64	45.80	63.99	2.63	16.89	47.45	52.16
1.3B	83.90	55.28	72.98	3.83	20.03	58.67	60.89
6.9B	83.78	61.21	78.78	7.09	27.93	67.48	59.27
13B	87.86	66.04	82.12	7.91	27.82	67.85	60.00
39B	87.95	67.29	83.80	9.19	31.04	71.41	61.59
82B	88.16	69.27	84.85	10.37	31.83	72.66	65.14

2.1.1 In-context Learning 성능

데이터가 작을 때 SOTA 성능을 보임 - NSMC 예제

Methods	Acc
Fine-tuning	
mBERT (Devlin et al., 2019)	87.1
w/ 70 data only	57.2
w/ 2K data only	69.9
w/ 4K data only	78.0
BERT (Park et al., 2020)	89.7
RoBERTa (Kang et al., 2020)	91.1
Few-shot	
13B 70-shot	87.9
39B 70-shot	88.0
82B 70-shot	88.2

2.1.2 Tokenization 별 성능 변화

Tokenization ablation study

	형태소분석기	byte vs. char	BPE	OOV
Ours (Morpheme-aware byte-level BPE)	0	byte-level	0	X
byte-level BPE	X	byte-level	0	X
char-level BPE	X	char-level	0	0

2.1.2 Tokenization 별 성능 변화

Tokenization ablation study

	KorQuAD (EA / F1)		AI Hub (BLEU) Ko→En En→Ko		YNAT (F1)	KLUE-STS (F1)
Ours	55.28	72.98	3.83	20.03	58.67	60.89
byte-level BPE	51.26	70.34	4.61	19.95	48.32	60.45
char-level BPE	45.41	66.10	3.62	16.73	23.94	59.83

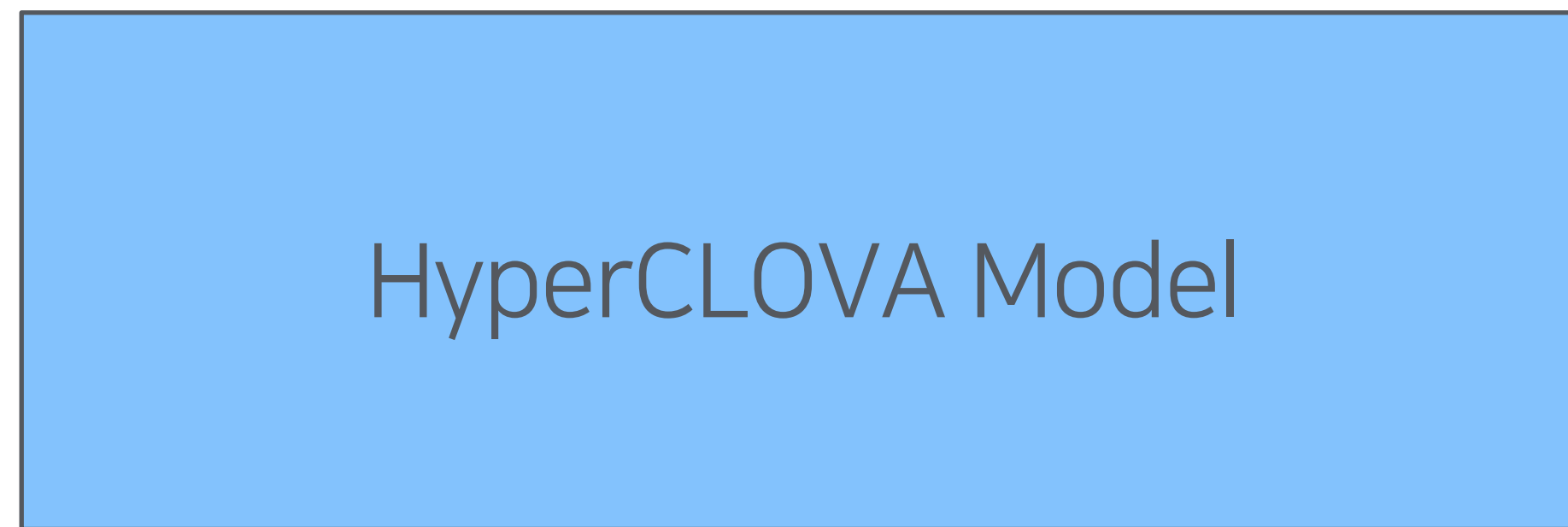
2.2 Fine-tuning: P-tuning and LoRA

일부 파라미터만 업데이트: 왜 필요한가?

- P-tuning and LoRA: 큰 언어 모델을 학습하기 위하여 제안된 방법론들
- 추가 파라미터를 적게 쓴다는 특징이 있음 (1/1000 ~ 1/100)
- 몇백~몇천 데이터로 모델 학습이 가능하다
- 모든 파라미터를 학습하는 fine-tuning과 종종 비슷한 성능 향상을 보여준다

2.2 Fine-tuning: P-tuning and LoRA

P-tuning과 LoRA의 차이: 추가 파라미터의 위치



(a) P-tuning



(b) LoRA

2.2 Fine-tuning: P-tuning and LoRA

적은 데이터셋으로 훌륭한 성능 - NSMC 예제

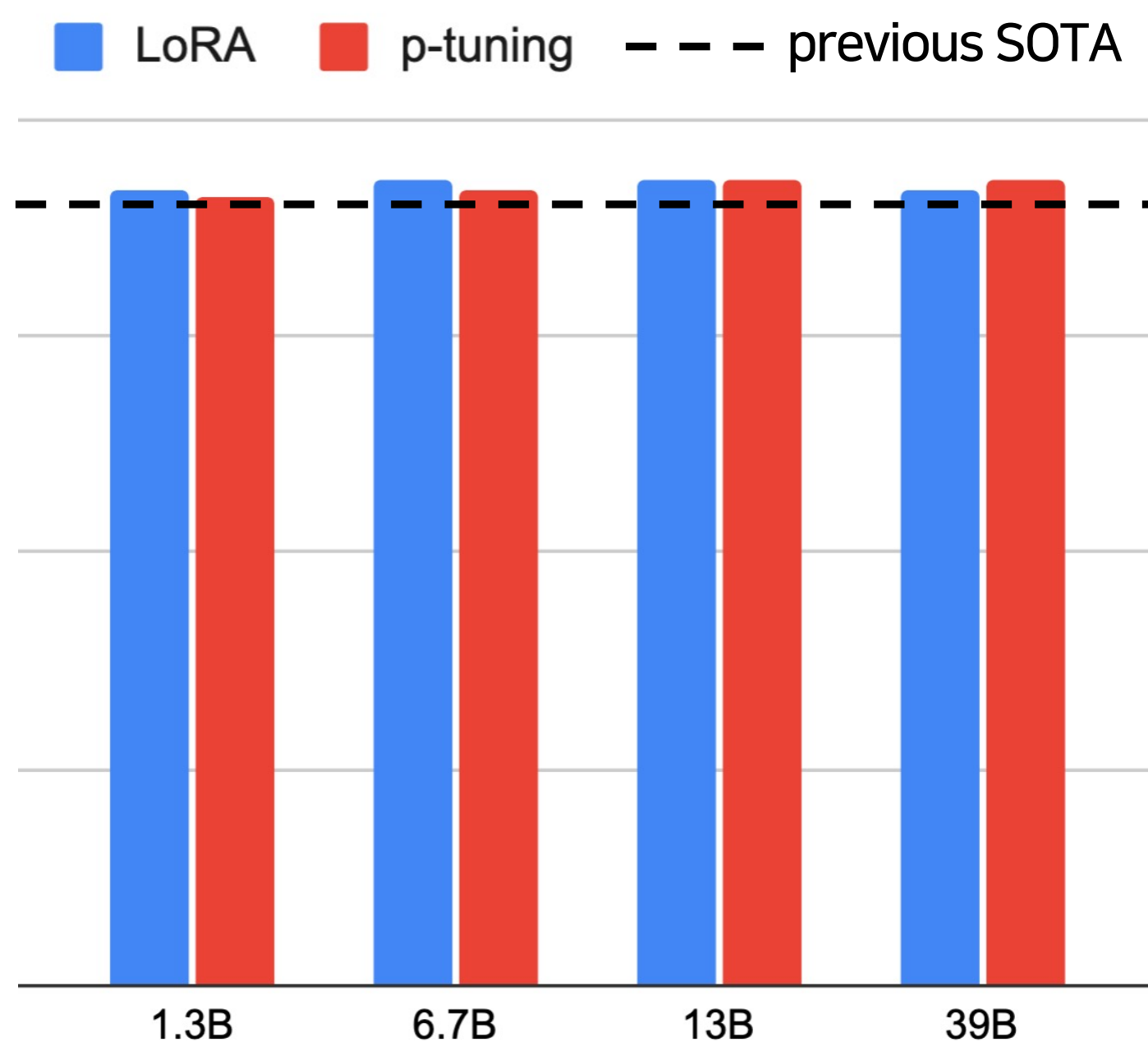
- p-tuning의 성능은 더 많은 데이터에서 더 좋음

Methods	Acc		
Fine-tuning		p-tuning	
mBERT (Devlin et al., 2019)	87.1	137M w/ p-tuning	87.2
w/ 70 data only	57.2	w/ 70 data only	60.9
w/ 2K data only	69.9	w/ 2K data only	77.9
w/ 4K data only	78.0	w/ 4K data only	81.2
BERT (Park et al., 2020)	89.7	13B w/ p-tuning	91.7
RoBERTa (Kang et al., 2020)	91.1	w/ 2K data only	89.5
Few-shot		w/ 4K data only	90.7
13B 70-shot	87.9	w/ MLP-encoder	90.3
39B 70-shot	88.0	39B w/ p-tuning	93.0
82B 70-shot	88.2		

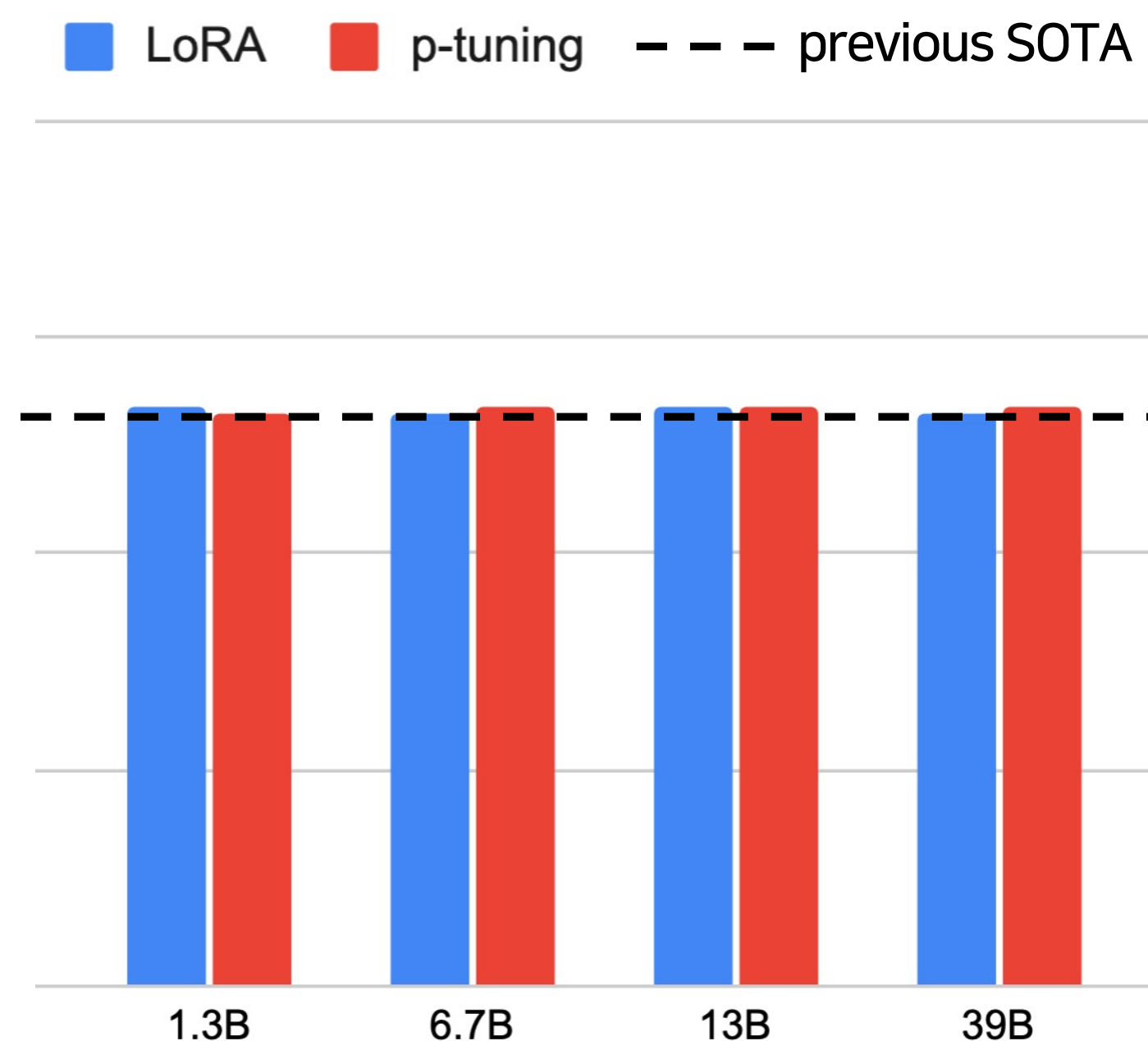
2.2 Fine-tuning: P-tuning and LoRA

full 데이터셋에서 작은 추가 파라미터로 SOTA 성능

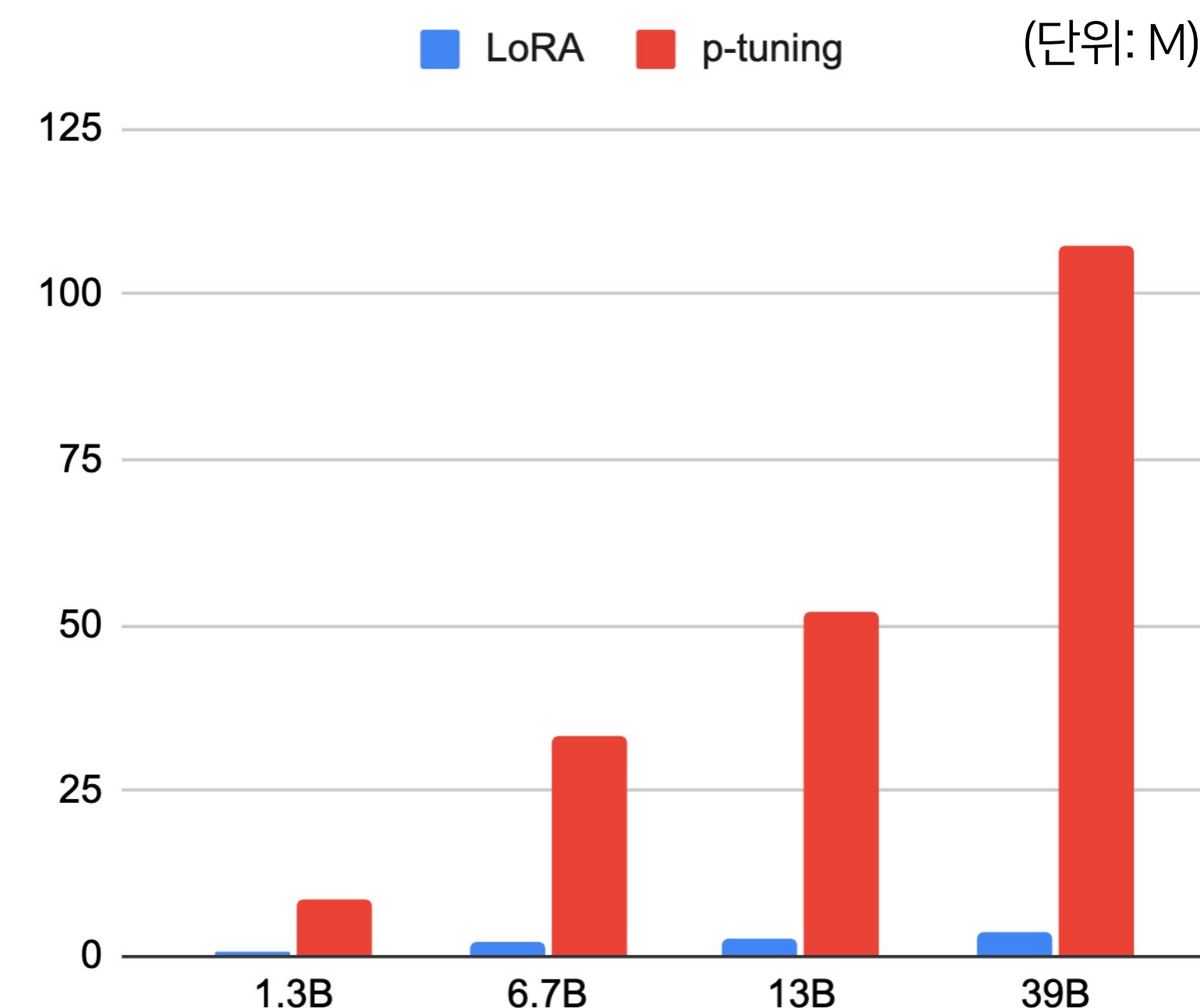
(a) NSMC



(b) YNAT



(c) 추가 파라미터 수



3. After training hyper-scale LM

3.0 학습이 끝나고 난 뒤

Model Evaluation

모델을 어떻게 평가할 것인가?

Model Expansion

모델을 어떻게 확장할 것인가?

3.1 Model Evaluation



3.1 Model Evaluation (cont'd)

**Generation results
in the next page**

그리웠던 순간을 호명하며 나는
한 줌의 눈물을 불빛 속에 던져주었다.
흔들리는 종소리의 동그라미 속에서
휘파람을 불며 너는 떠나갔다.
사랑의 비참한 기념품이여, 잘 가거라
오래 전 내 영혼은 너를 잊은 적이 없다.

Model Boss (39B, 2021)

그리웠던 순간을 호명하며 나는 한 줌의 눈물을 불빛 속에 던져주었다.

그 때 어떤 은유가 내 가슴을 두드렸다.

눈물을 잃어버린 자들의 슬픔을 그 말은 내게 조용히 말했다.

“아직 늦지 않았어.”

나는 눈물을 잃은 자들을 위하여 죽음을 사랑하기로 했다.

내가 가진 나의 ‘시간’을 한 올 한 올 잘라내면서 내 목숨을 하얗게 물들여 갔다.

3.1.1 Metric Issues

Human evaluation vs Automatic evaluation

- 사람이 직접 평가하는 것이 가장 정확하지만 비용과 시간 cost가 높음
- Machine translation 모델이 발전함에 따라 자동 평가 방법에 대한 연구도 본격화

Papineni et al., Bleu: a method for automatic evaluation of machine translation., ACL, 2002.

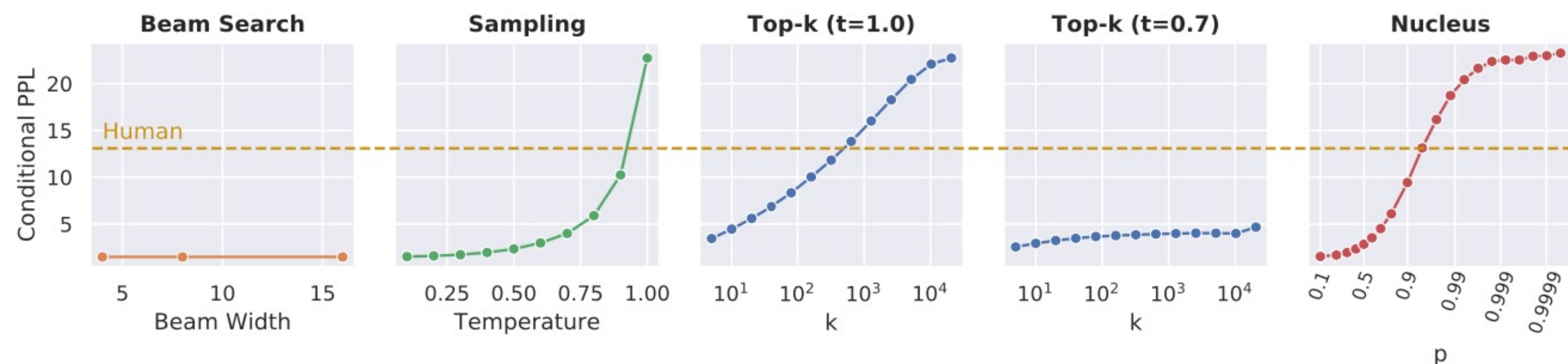
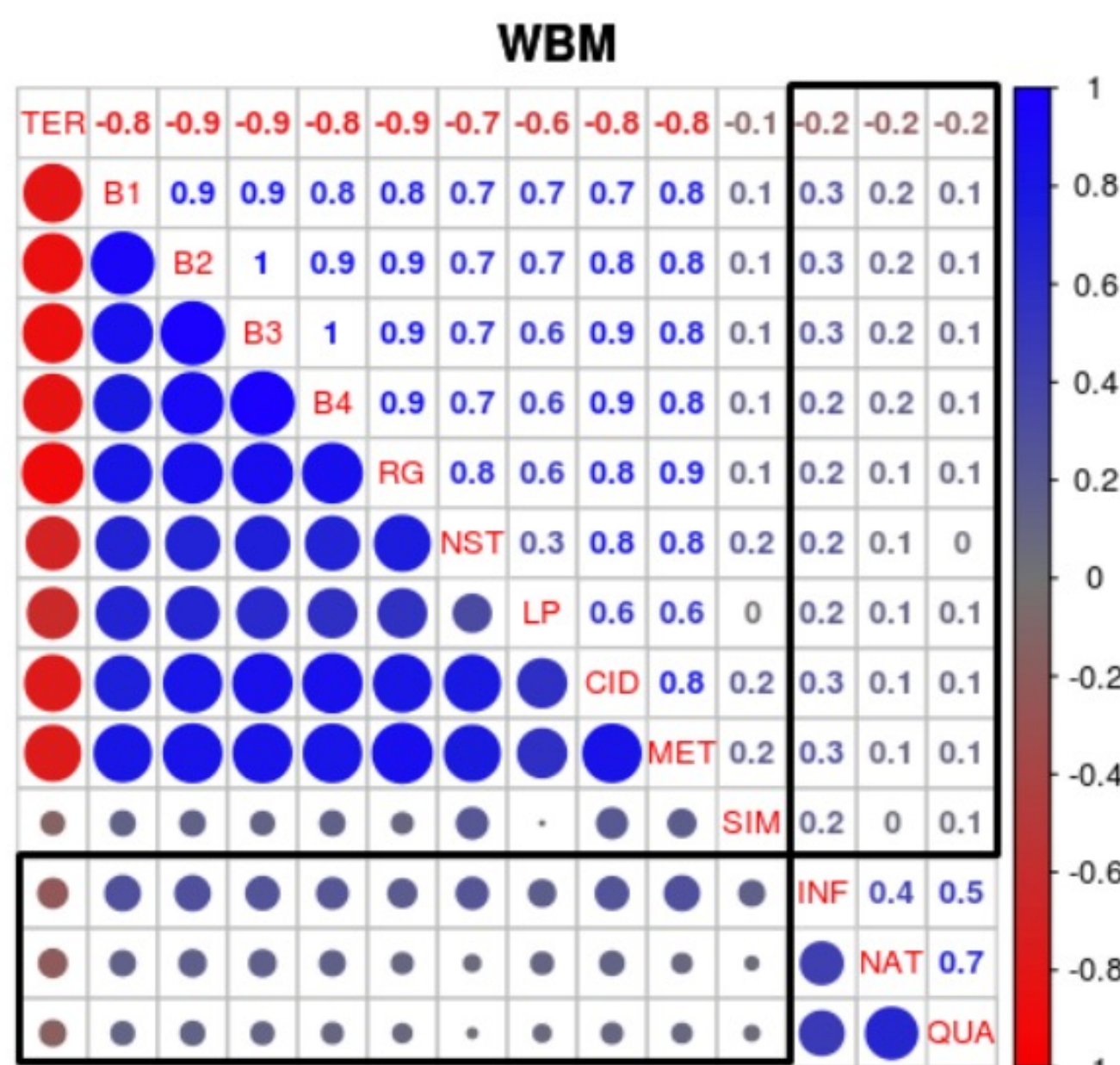
- Evaluation metrics *Garbacea et al., Neural Language Generation: Formulation, Methods, and Evaluation., arXiv, 2020.*

Metric	Target	Reference	___ is Better	Based on
BLEU	Quality	Needed	Higher	N-gram
Edit Distance	Quality	Needed	Lower	Character Distance
BLEURT	Quality	Needed	Higher	CTS Representation
Perplexity	Quality	Not Needed	Lower	Likelihood
Self-BLEU	Diversity	Not Needed	Lower	N-gram

3.1.1 Metric Issues (cont'd)

Limitations of automatic metrics

- Automatic metric의 유효성에 문제를 제기하는 연구결과가 지속적으로 보고됨



3.1.1 Metric Issues (cont'd)

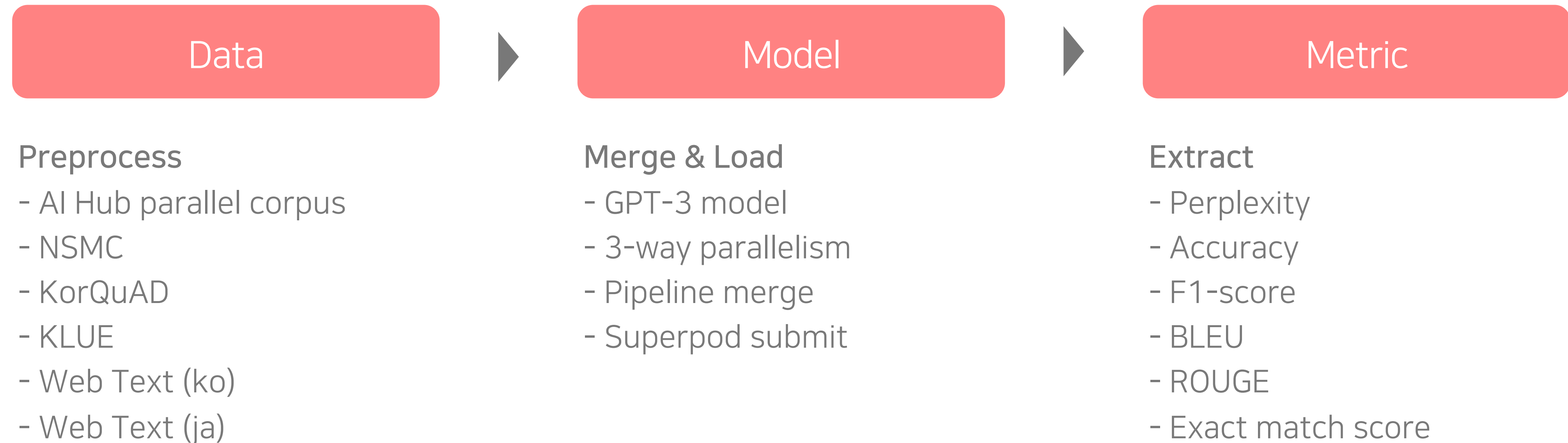
Measuring the model performance

- 모델 출력을 reference와 비교하는 것으로는 언어 구사 능력을 판단하기 어려움
Kannan et al., Adversarial Evaluation of Dialogue Models., arXiv, 2017.
- 여러 downstream task 성능을 통해 언어 모델 성능을 간접적으로 측정할 수 있음
Wang et al., GLUE: A Multi-Task Benchmark and Analysis Platform for Natural Language Understanding., ICLR, 2019.
- Downstream task와 별도로, 모델 자체의 언어 구사 능력을 평가할 필요가 있음

3.1.2 Evaluation Pipeline

What we need are data, model, and metric

- 평가에 사용할 데이터 전처리, 평가 대상 모델 로드, 평가 지표 추출 방법을 연동하여 생성 모델 평가 파이프라인 구축

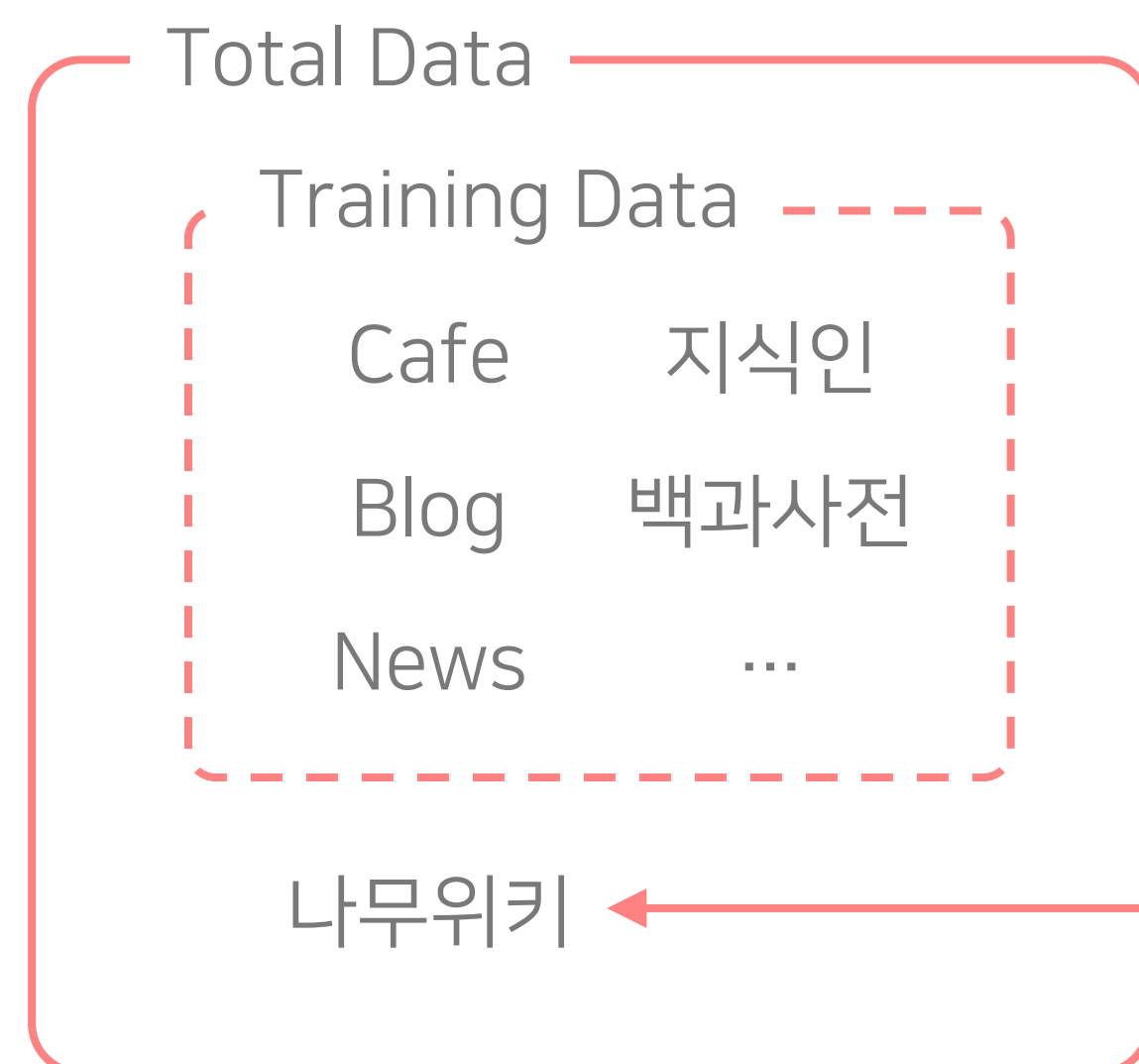


3.1.2 Evaluation Pipeline (cont'd)

Data preprocessing

- 공개 데이터 셋과 언어 모델 학습에서 배제한 데이터 셋을 평가에 활용
- 각 task 마다 별도의 few-shot prompt template을 개발하여 평가에 사용

Train set에서 배제한 데이터 활용



모델 학습에서 배제함

1. 모델 오염 방지
factuality가 보장되지 않은 내용을 생성 모델이 학습하지 않도록 분리
2. 비격식체 문장 비교 평가
사람이 작성한 레퍼런스 문장과 모델이 생성한 문장 비교 평가에 활용

Few-shot template 개발

KorQuAD

제목: {title}
 지문: {context}
 질문: {query}
 답변: {answer}
 질문: {query}
 답변: {answer}
 질문: {query}
 답변:

서술형

다음 질문에 답하세요.
 Q: '{query}'의 의미는 무엇인가요?
 A:

번역

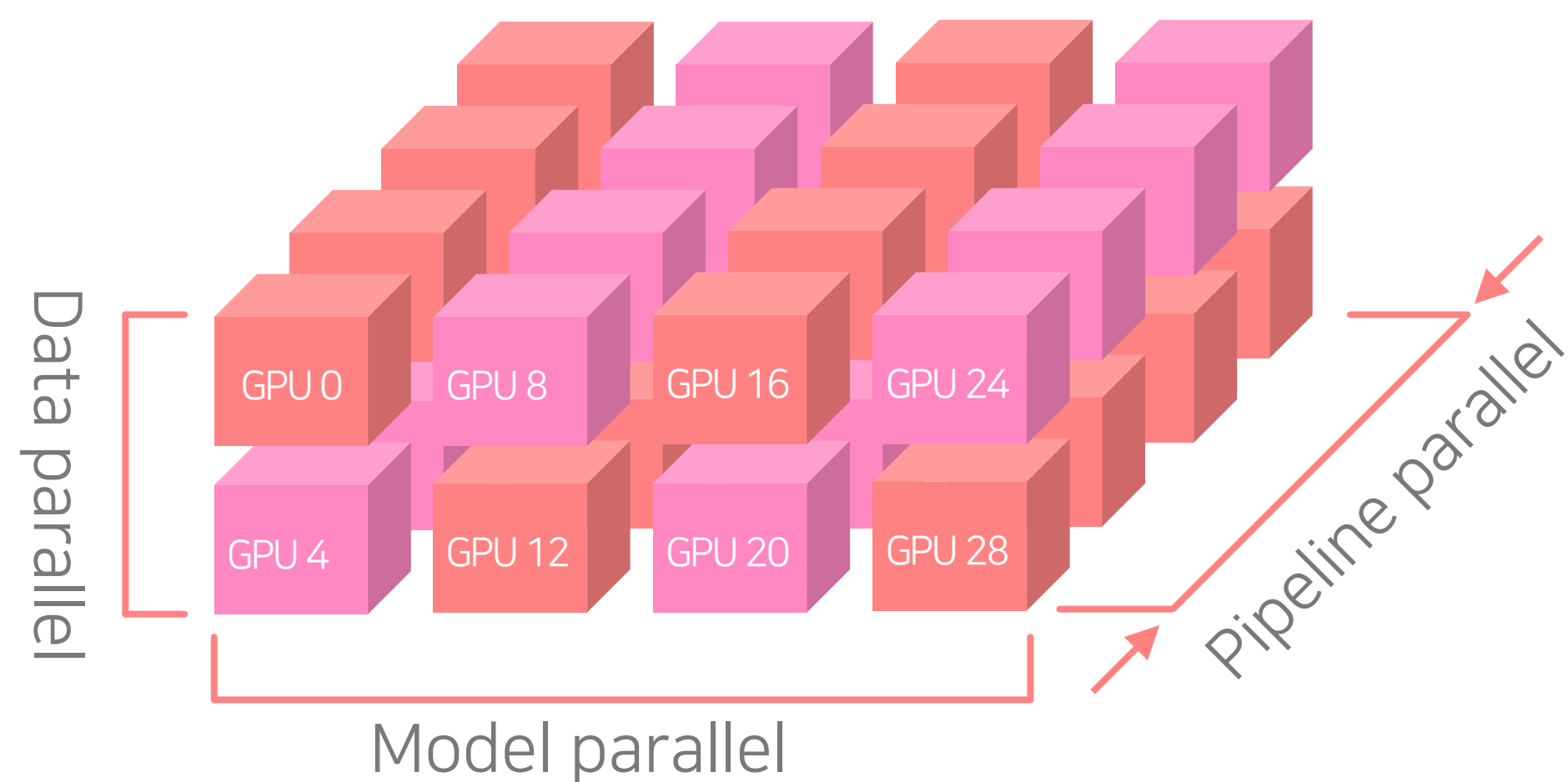
{source_lang}을 {target_lang}으로 번역해
 {source}={target}
 {source}=

3.1.2 Evaluation Pipeline (cont'd)

Merge and load model

- 학습을 통해 생성한 checkpoint를 load하기 위해서는 $MP * PP$ 만큼 GPU가 필요
- 효율적인 평가 자원 활용과 작업 안정성 확보를 위해 checkpoint merge 수행

Pipeline modification



Load merged checkpoint

Test model size: 82B

Model Parallel	Pipeline Parallel	Required GPU	Relative Elapsed Time	Trial Count
8	8	64	1.00	8
8	4	32	1.71	6
8	2	16	2.78	1

3.1.2 Evaluation Pipeline (cont'd)

Extract metric scores

- 객관식 태스크를 채점하는 방식인 rank classification 지원
- 서술형 태스크를 채점하기 위해 복수의 tokenization 지원

Rank classification

Model input (NSMC)

지루하지는 않은데 완전 막장임... 돈주고 보기에...

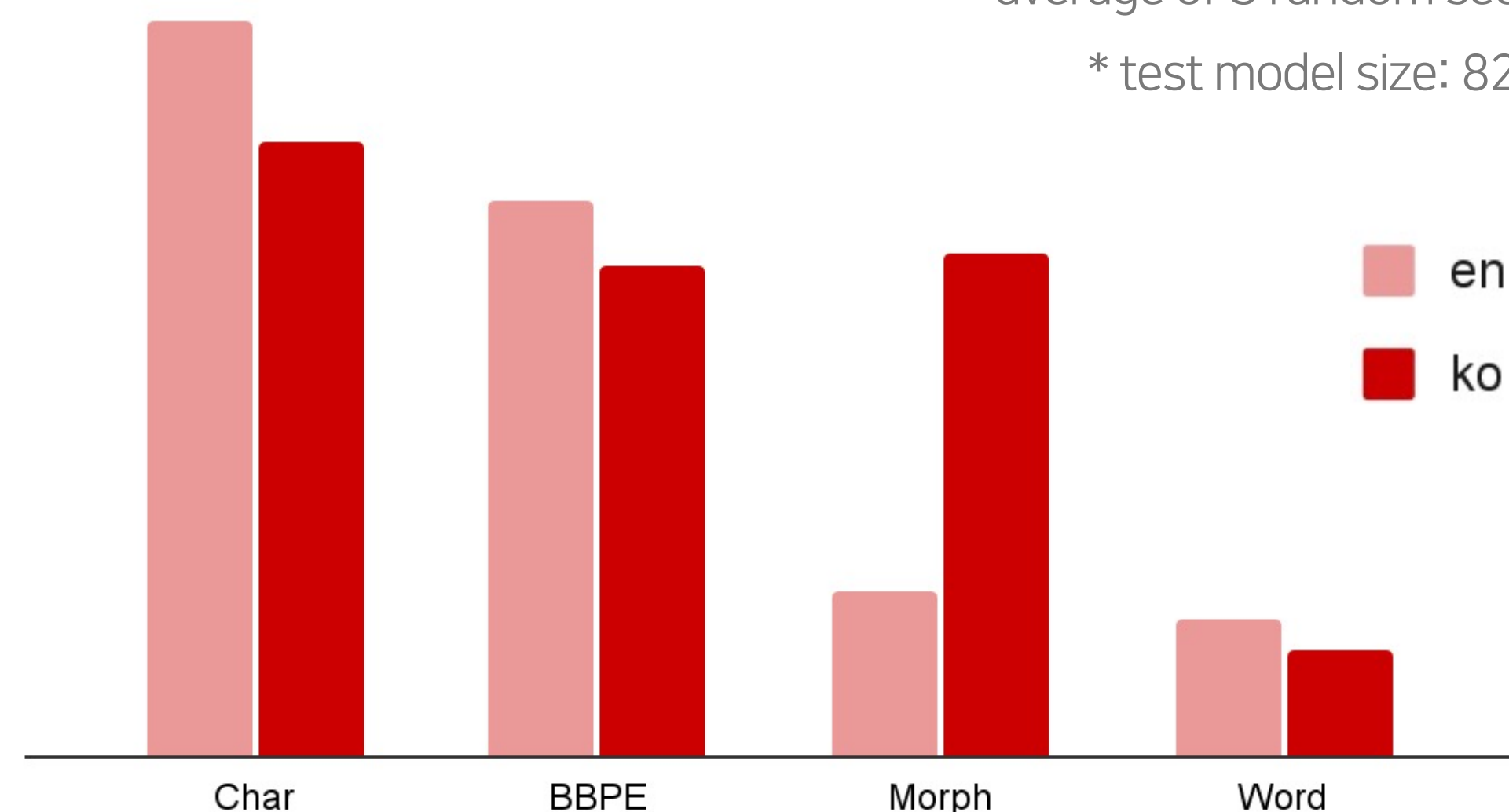
(Negative Log) Likelihood

안에 { (긍정)을 넣었을 때: -5.9746
(부정)을 넣었을 때: -0.8116 ← Completion
Candidates

Tokenization for BLEU

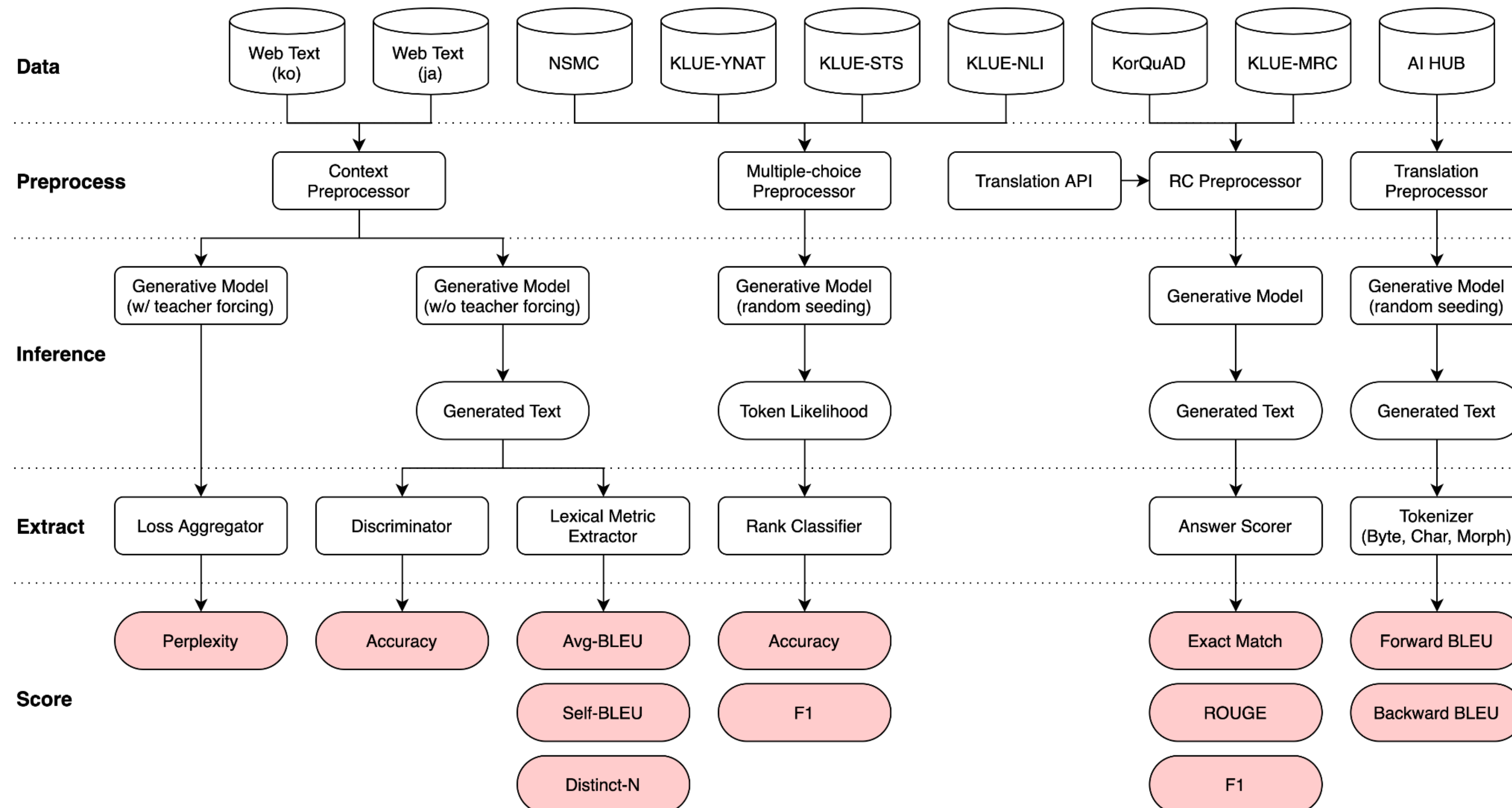
* average of 3 random seed

* test model size: 82B



3.1.2 Evaluation Pipeline (cont'd)

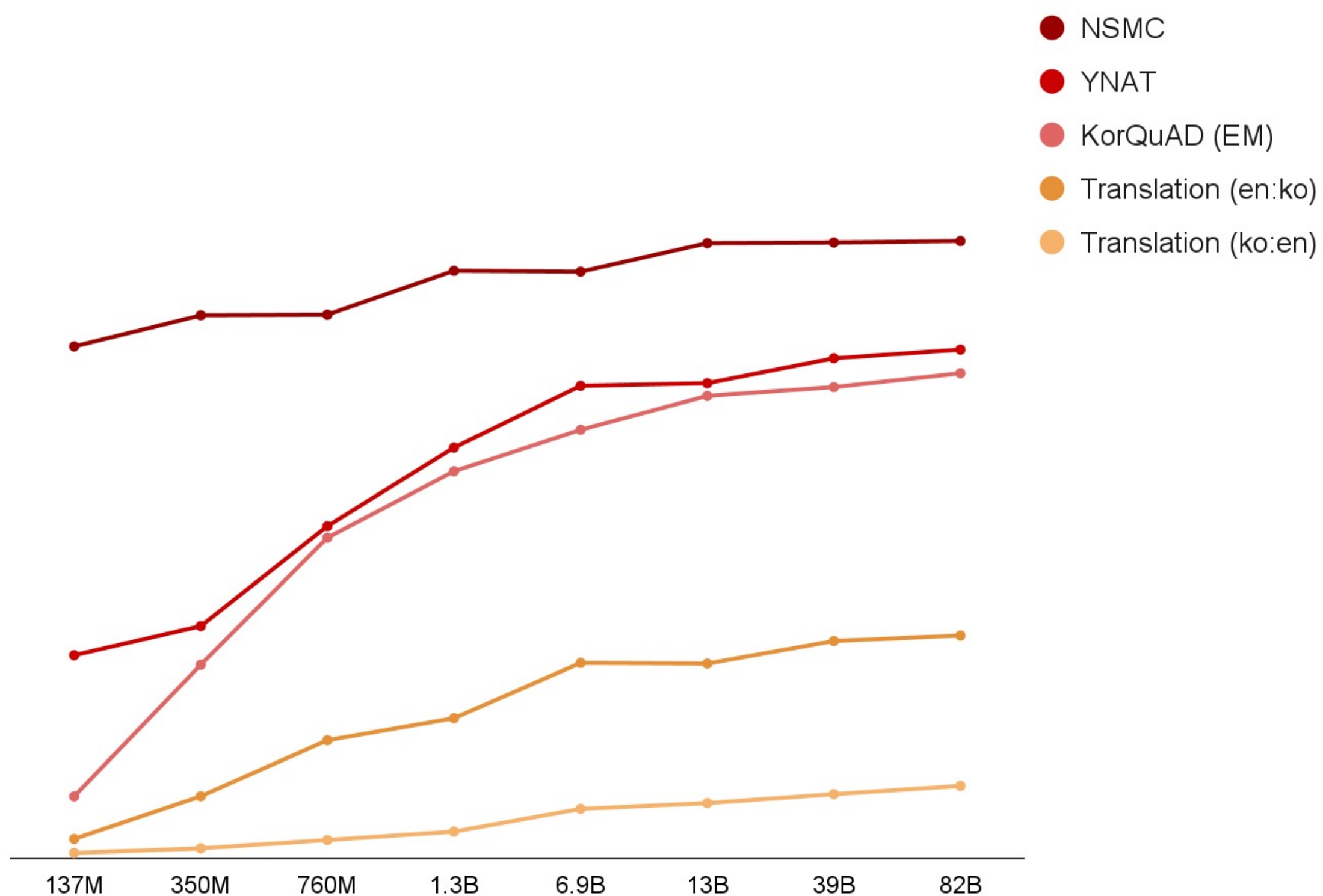
Structure overview



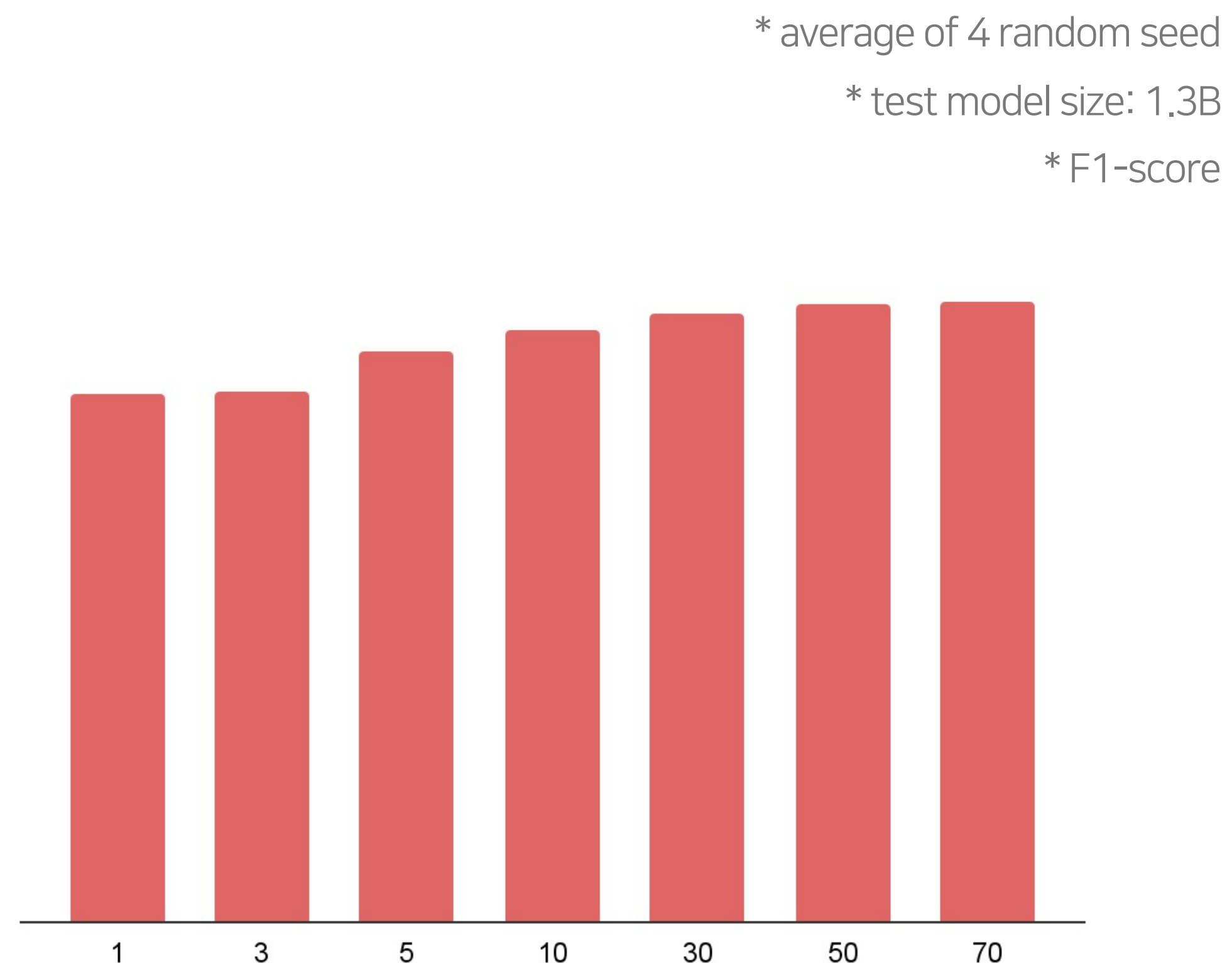
3.1.3 Downstream Benchmarks

Benchmark detail

Overall performance



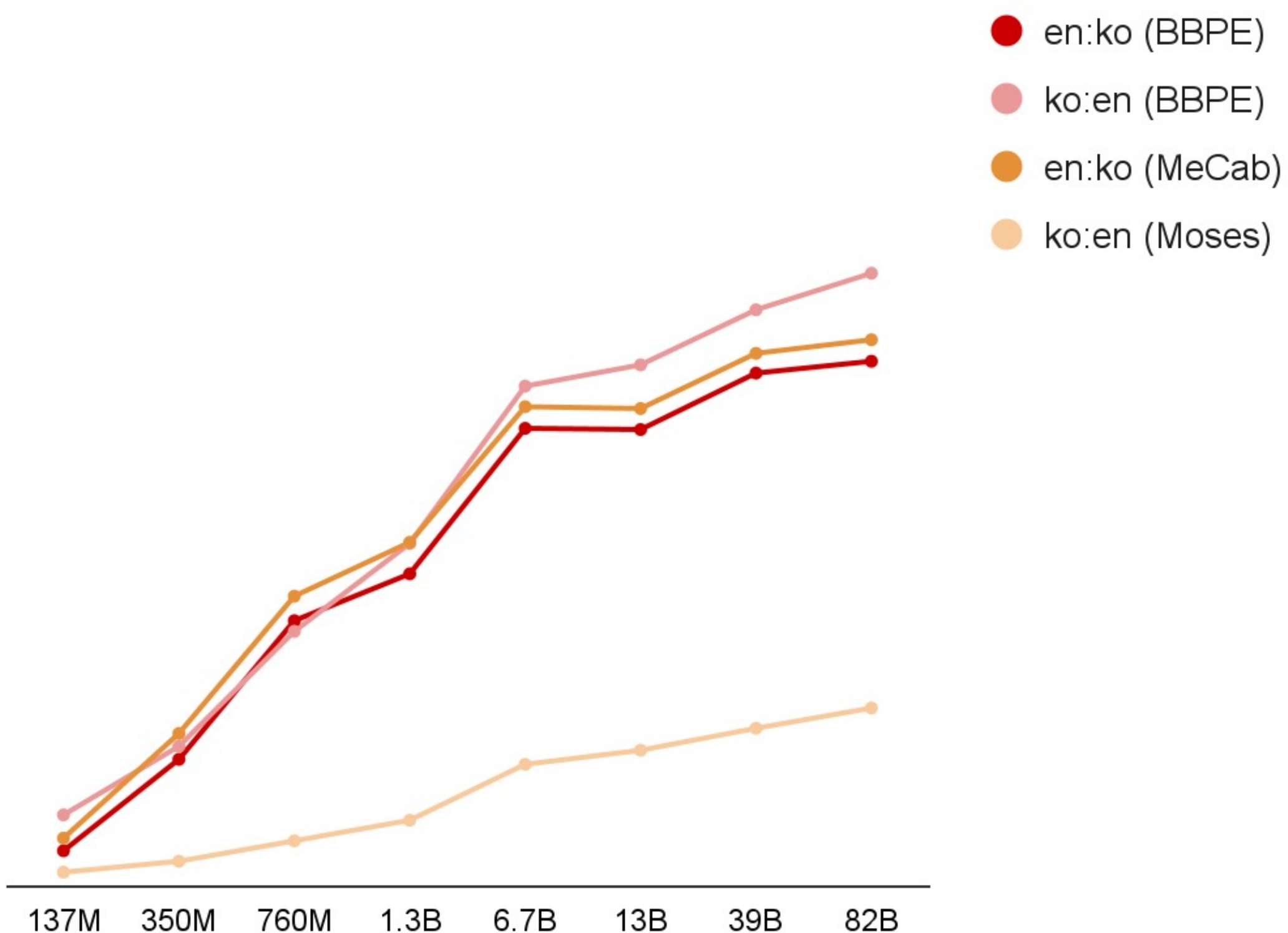
NSMC with k-shots



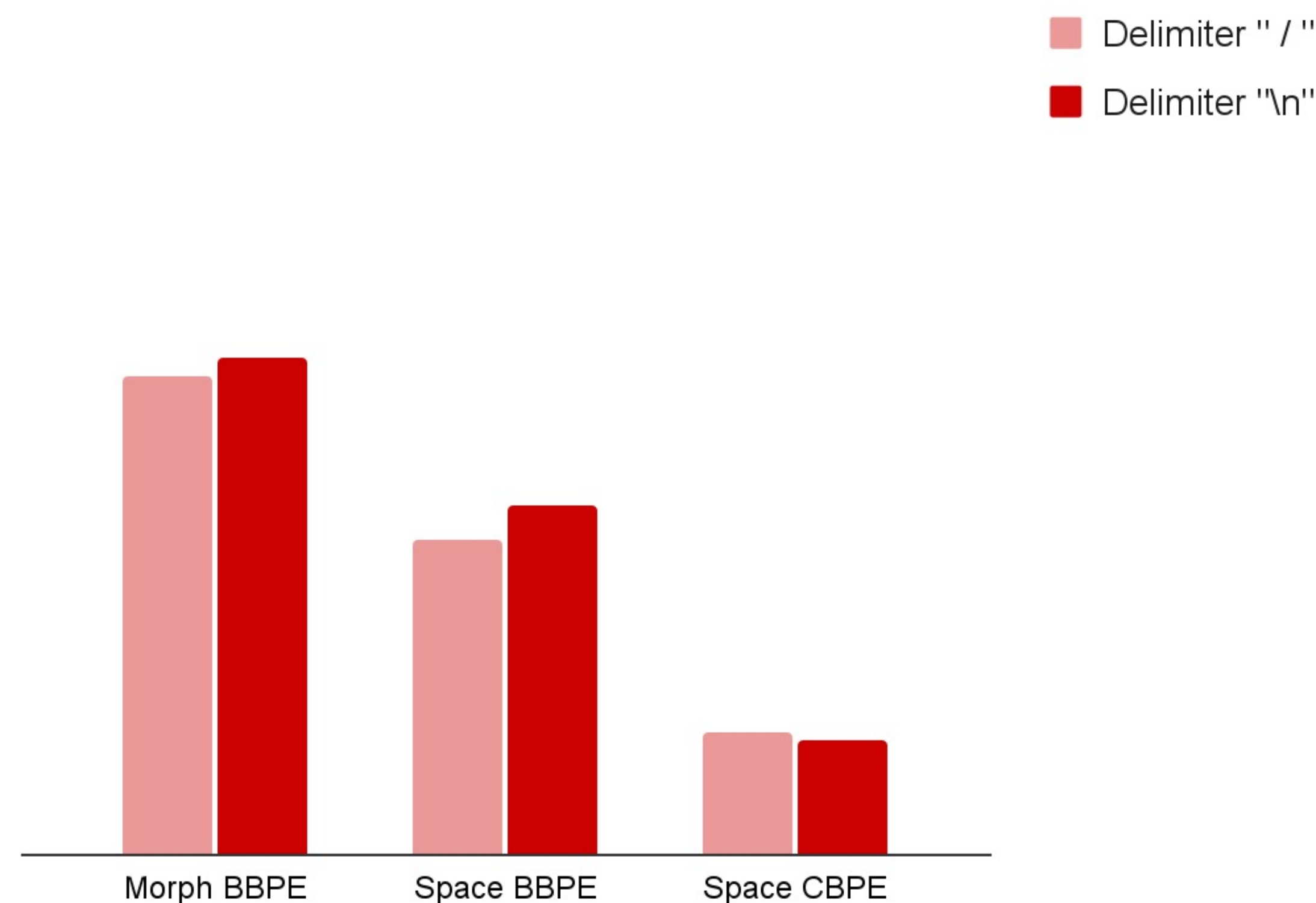
3.1.3 Downstream Benchmarks (cont'd)

Benchmark detail

Translation with tokenizers



YNAT with prompt delimiter

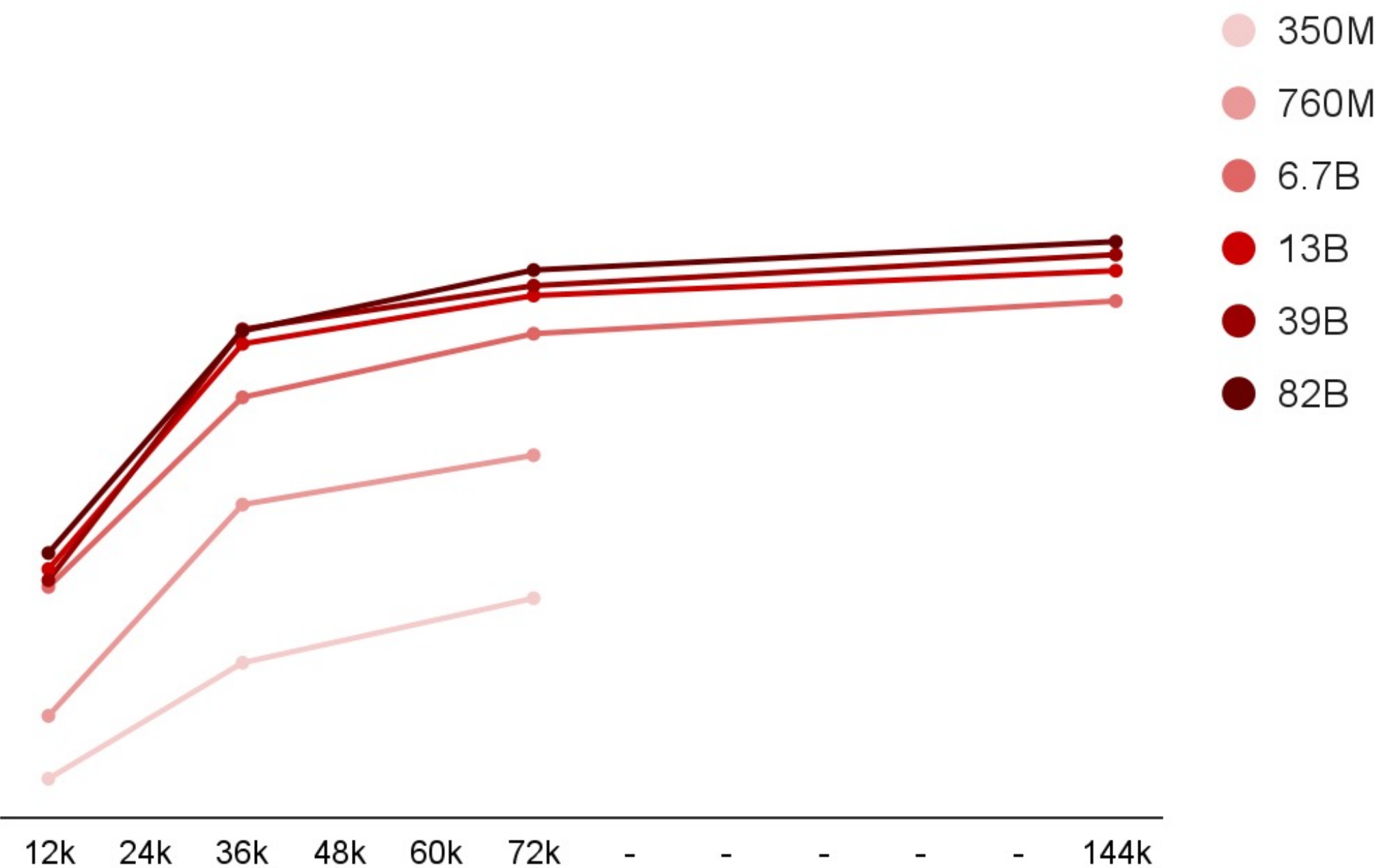


3.1.3 Downstream Benchmarks (cont'd)

Benchmark detail

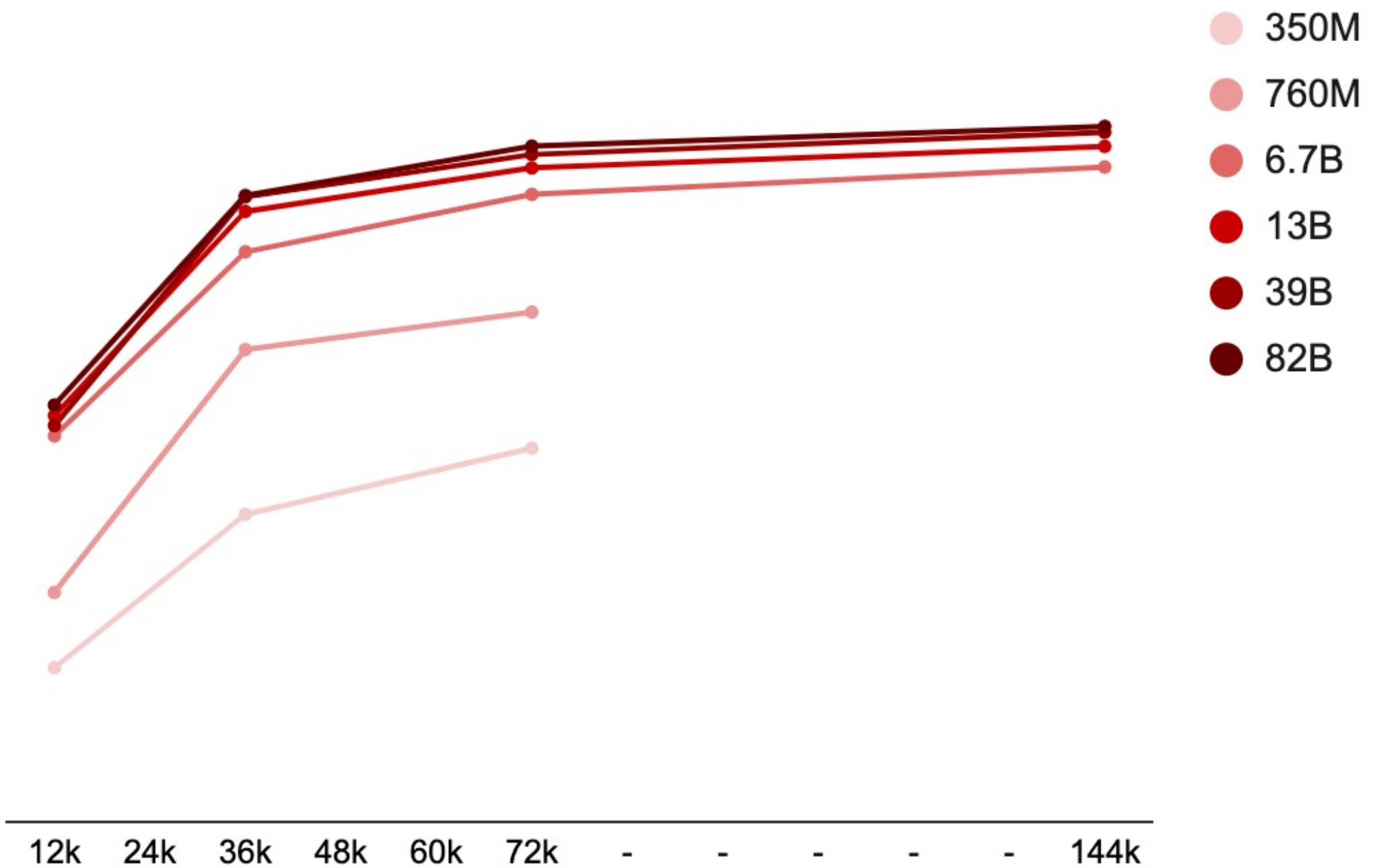
KorQuAD with training iteration (EM)

Exact Match



KorQuAD with training iteration (F1)

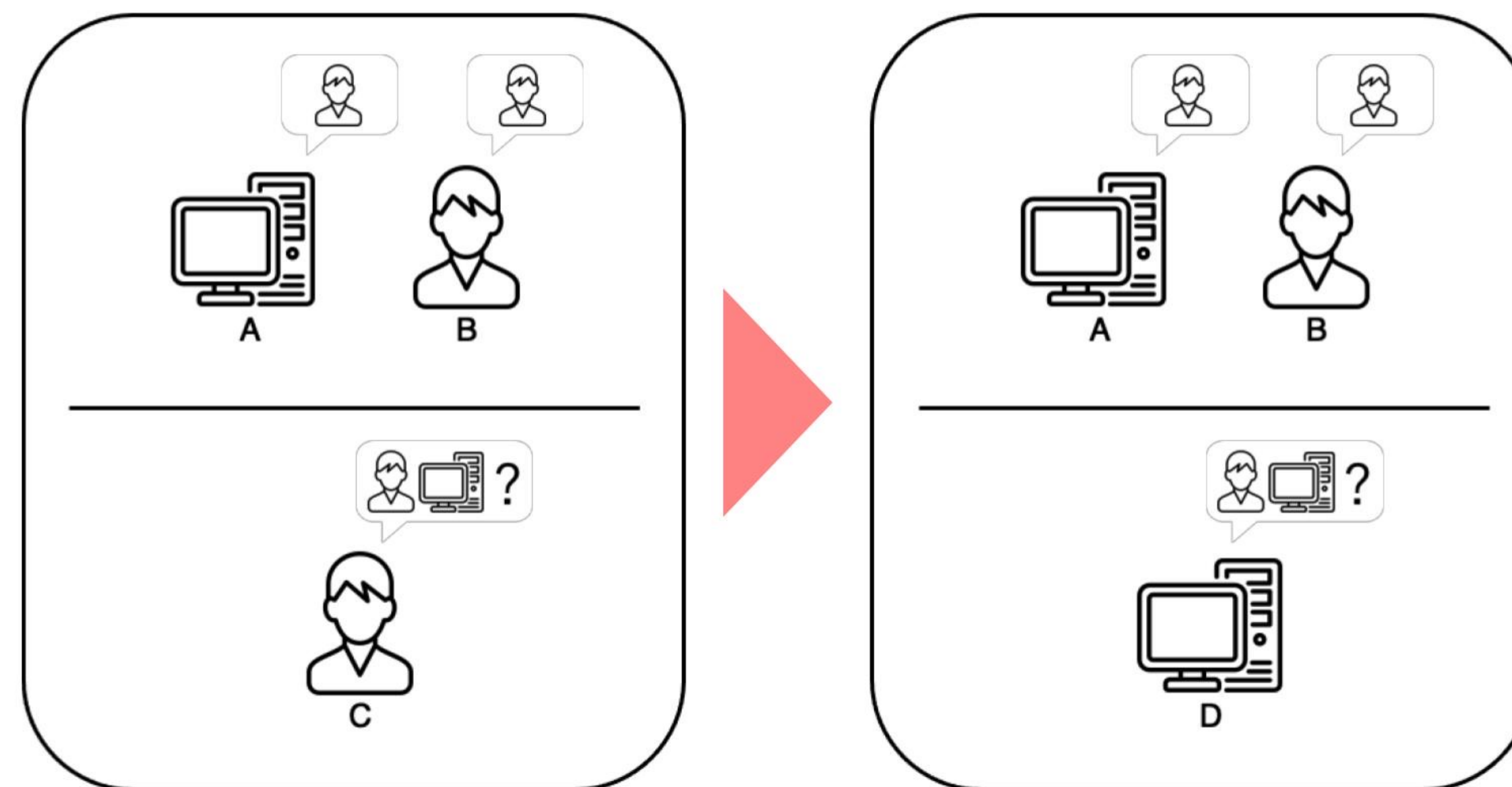
F1



3.1.4 Adversarial Evaluation

Intuition from the Turing test Li et al., Adversarial Learning for Neural Dialogue Generation., EMNLP, 2017.

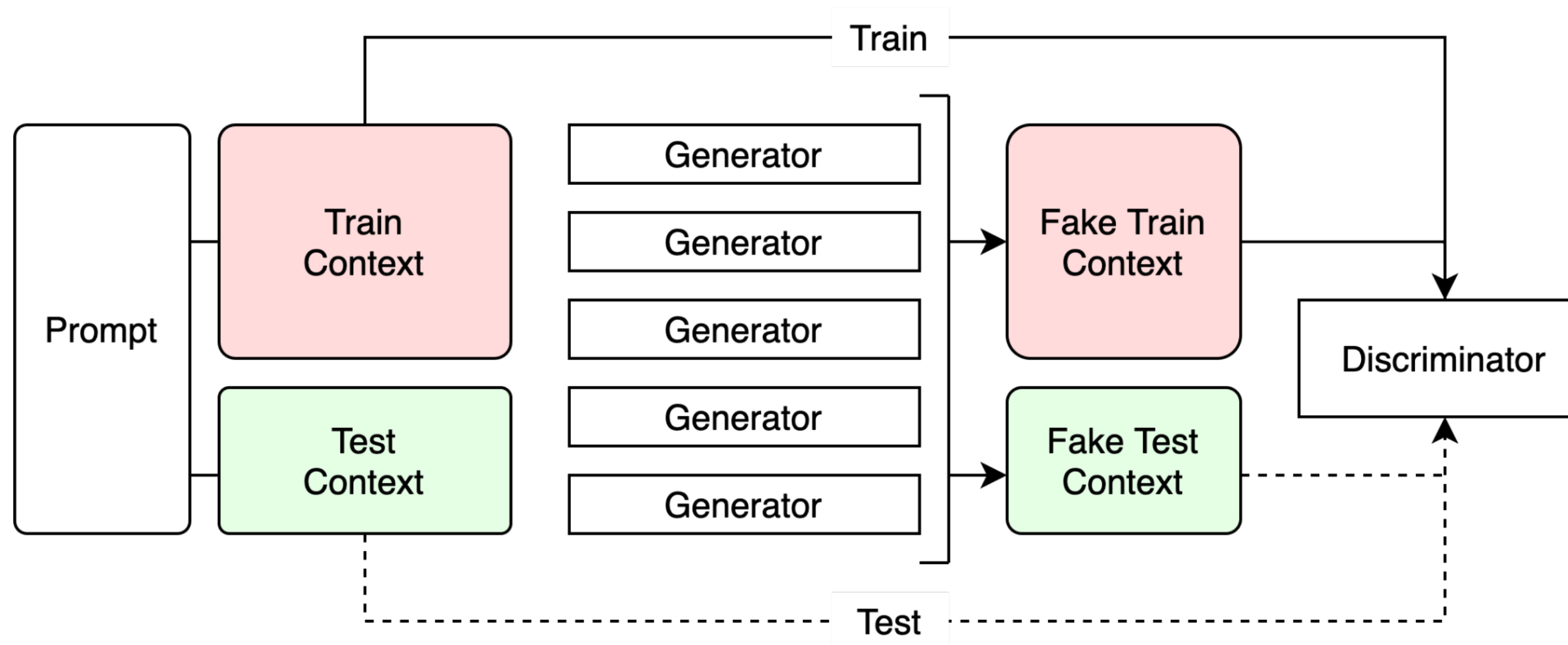
- 사람이 대화를 통해 기계과 사람을 분류하는 상황을 변형함
- 분류자를 trained model로 교체하여 분류 성능을 측정함



3.1.4 Adversarial Evaluation (cont'd)

Adversarial evaluation pipeline

- GAN과 유사한 구조를 가지고 있으나, gradient propagation을 하지 않음
- 여러 개의 generator 결과를 사용하여 분류기를 학습하도록 개선

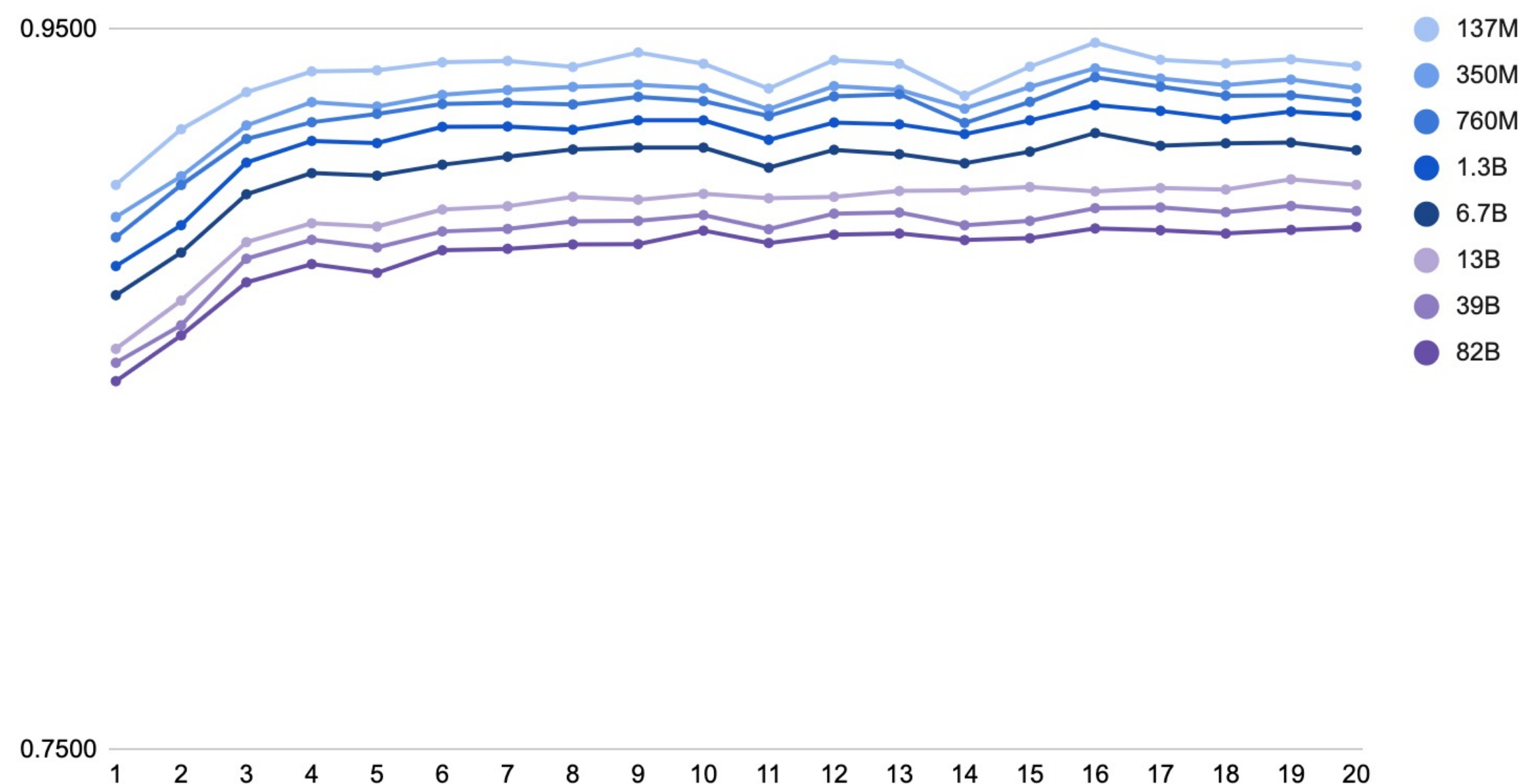


3.1.4 Adversarial Evaluation (cont'd)

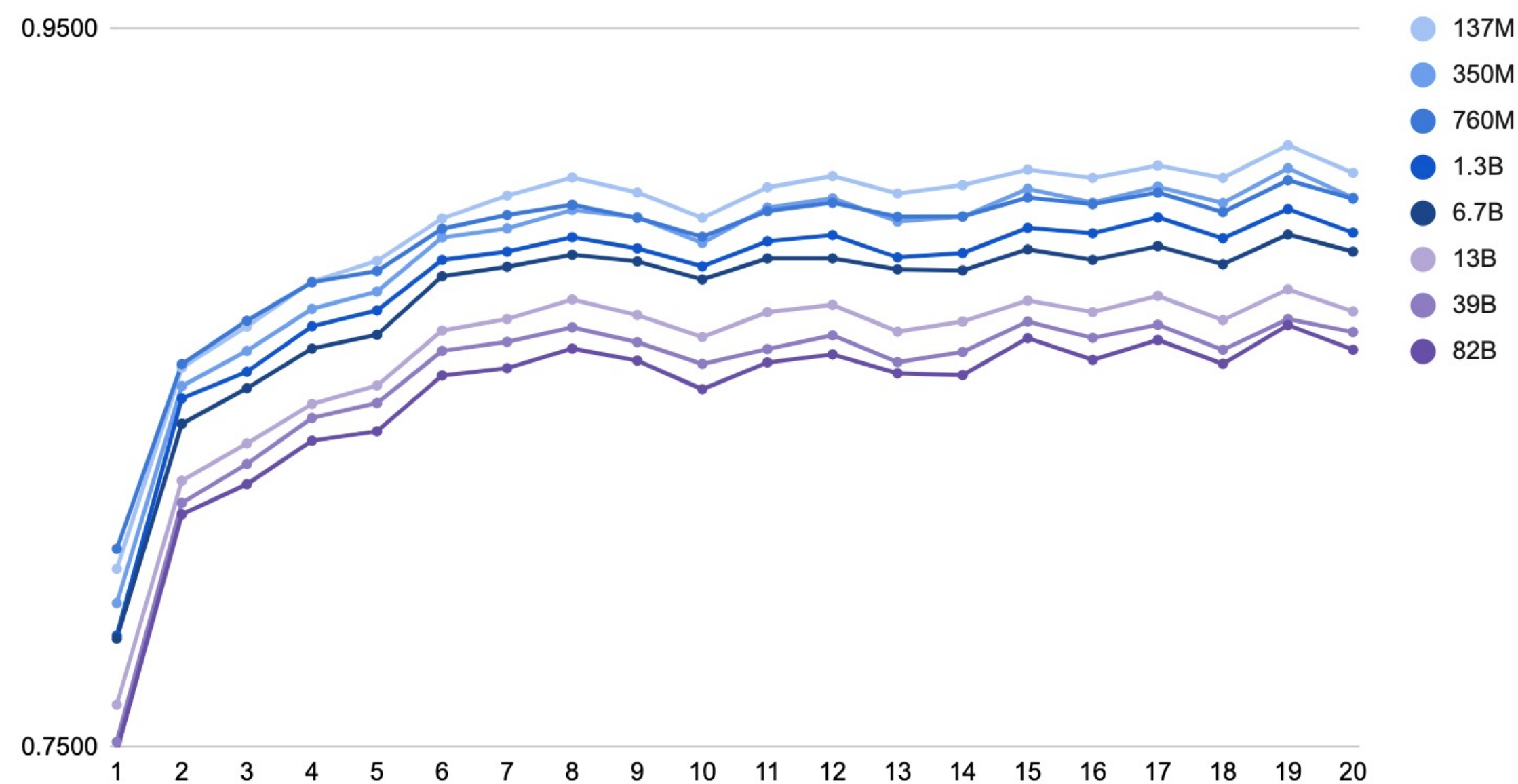
Discrimination accuracy

- 서로 다른 크기의 generator 결과를 학습한 분류기로 real / fake 문장을 분류함
- 단일 generator 결과를 학습한 분류기로 전체 test set을 분류해도 경향성을 유지함

full-data trained discriminator, Accuracy



6.7B-only trained discriminator, Accuracy



3.2 Model Expansion



3.2 Model Expansion (cont'd)

What we want to build

- 이미지와 텍스트를 입력 받아서 텍스트를 출력하는 모델
- 이미지 속 정보를 반영한 텍스트를 생성하는 능력을 배우는 것이 관건



+ 오늘 일상



Multi-Modal
Model



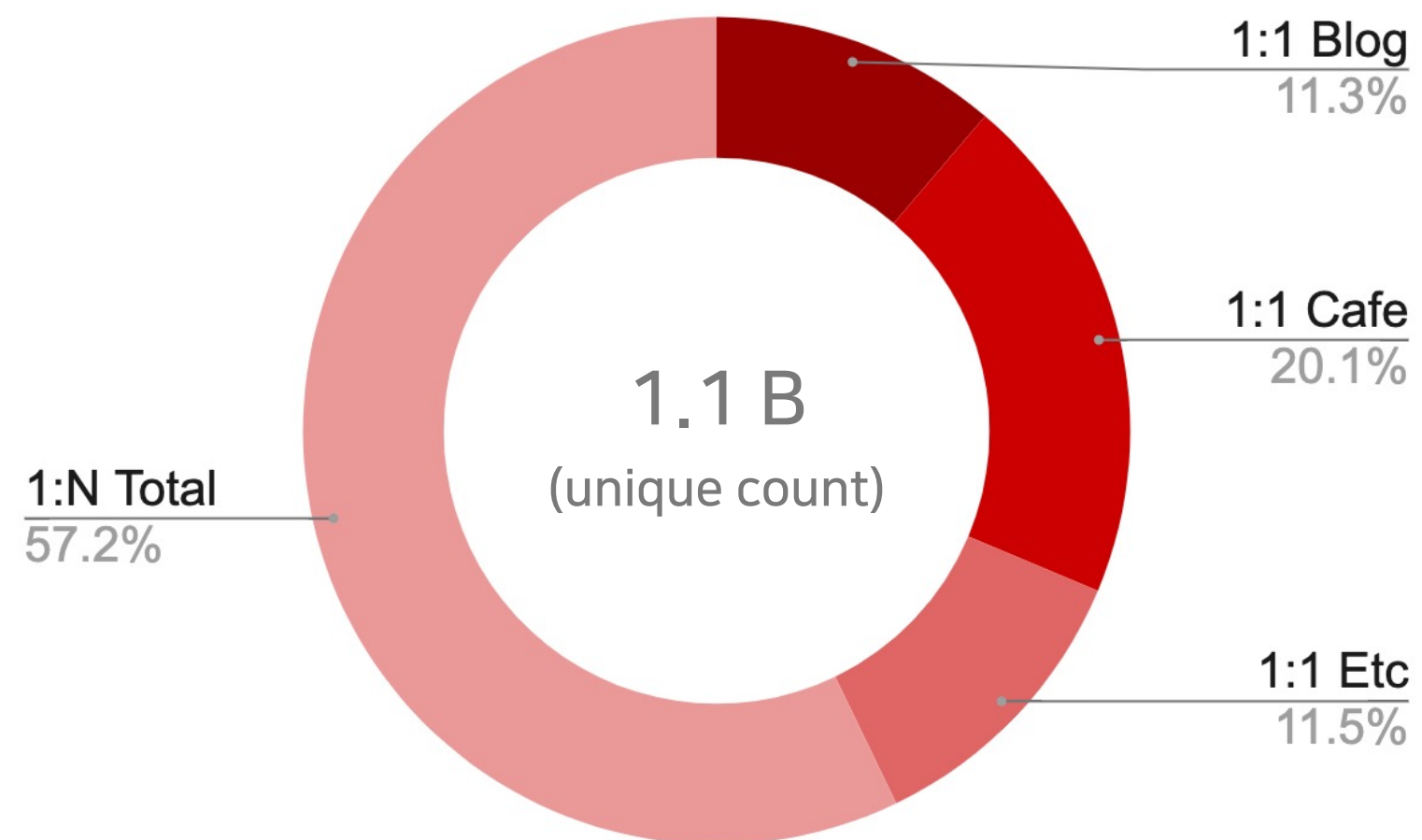
브런치 먹으러 한남동
에그 베네딕트도 맛있지만
자몽 주스가 더 맛있는 건 안 비밀

3.2.1 Data Collection

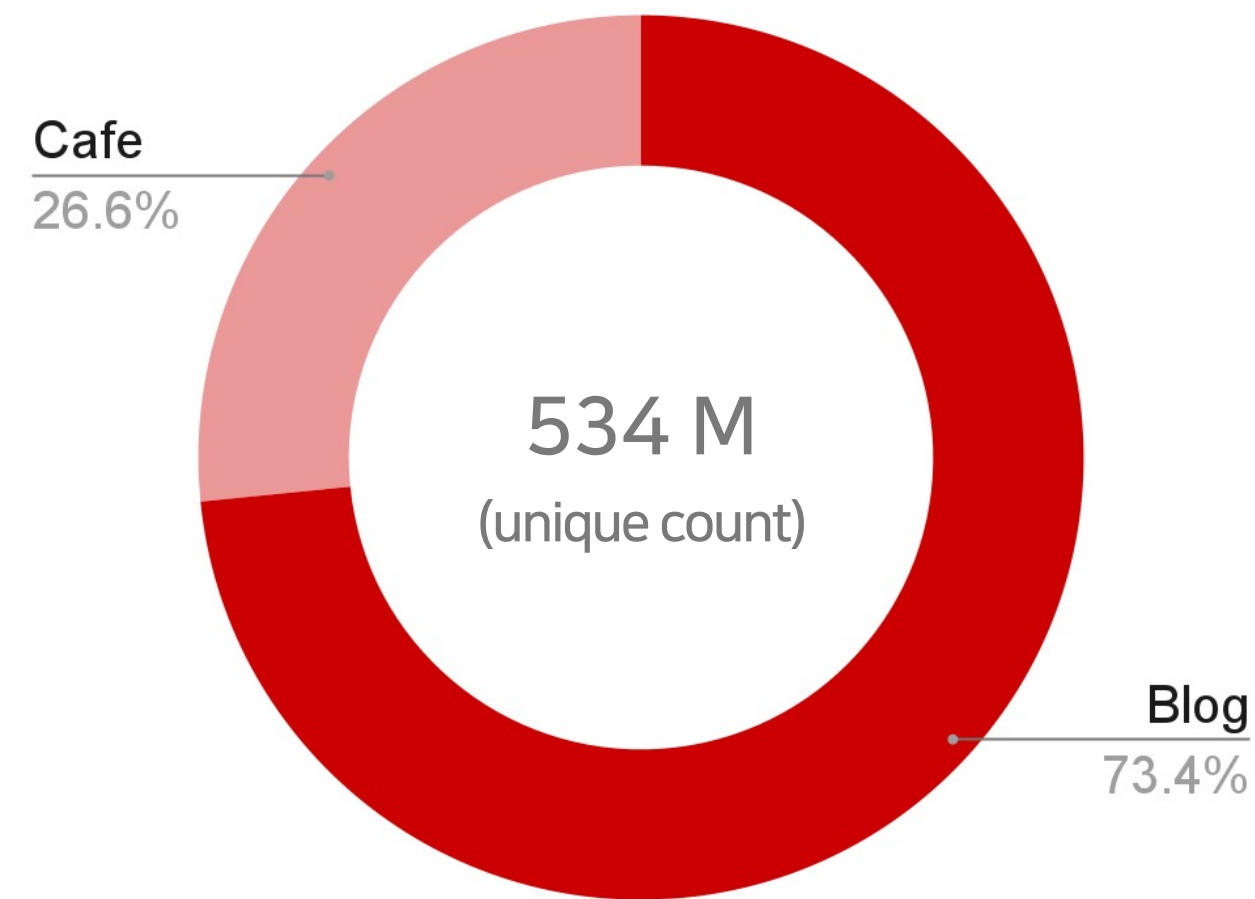
Target data examples

- 이미지와 연관성 있는 텍스트가 함께 존재하는 데이터가 필요함
- User Generated Contents 중, 텍스트-이미지 pair 추출하여 학습 데이터 구축

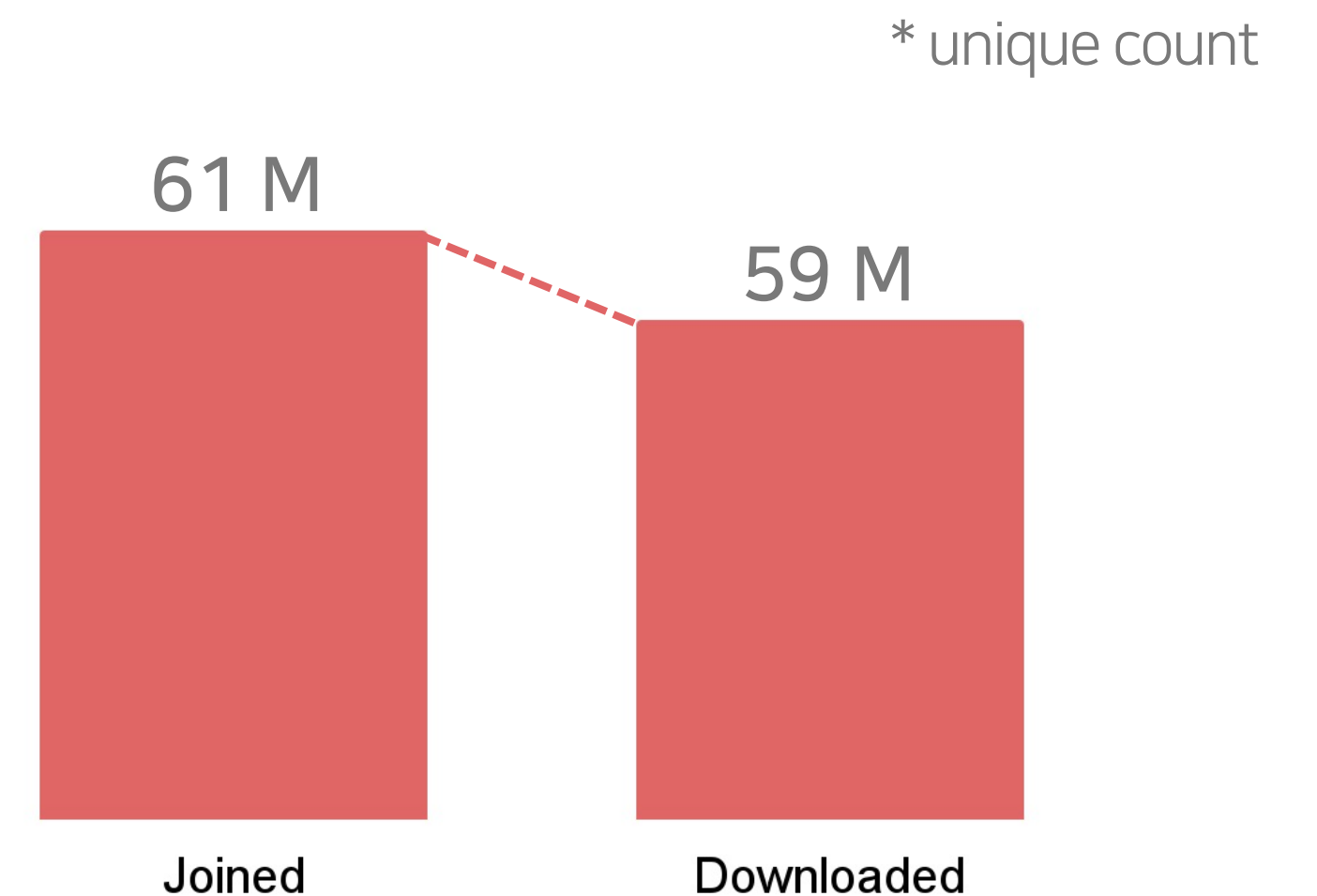
Image Data



Text Data



Final Data



3.2.1 Data Collection (cont'd)

Data issues

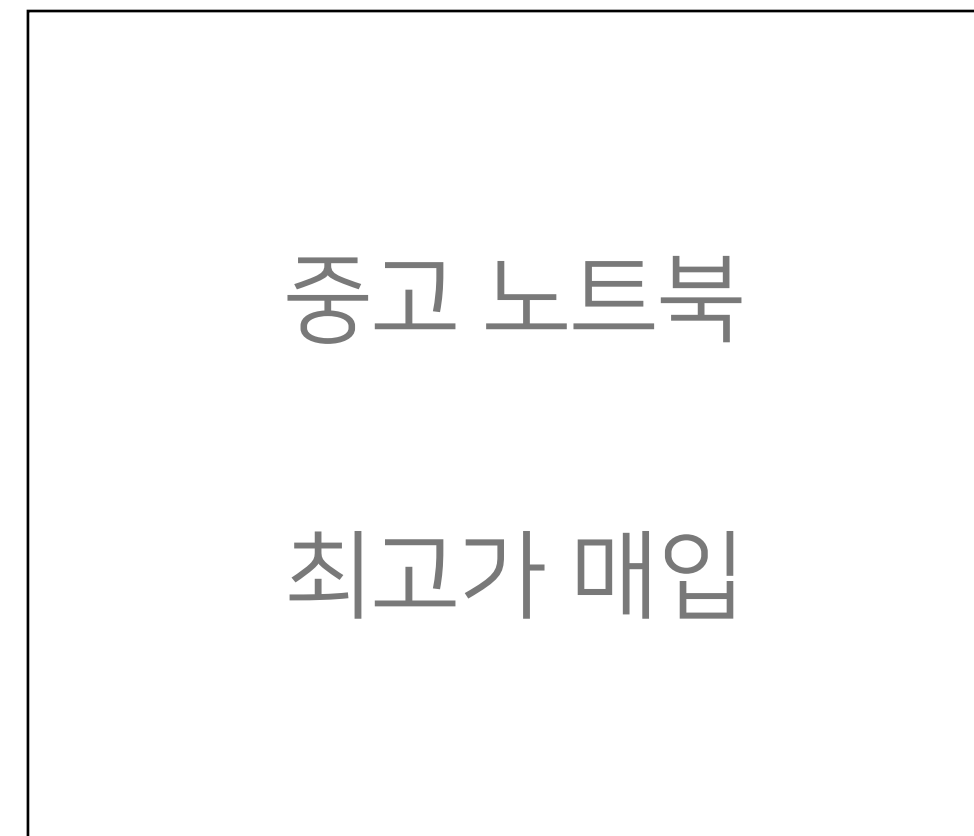
- 하나의 포스트에 여러 이미지가 들어있는 경우는 배제함
- 이미지를 텍스트처럼 사용하는 경우를 filter-out

1:N mapping



오늘은 고기 쌈장을
만들어 보겠습니다.
백종원 쌈 레시피를 참고했습니다.

이미지 = 텍스트



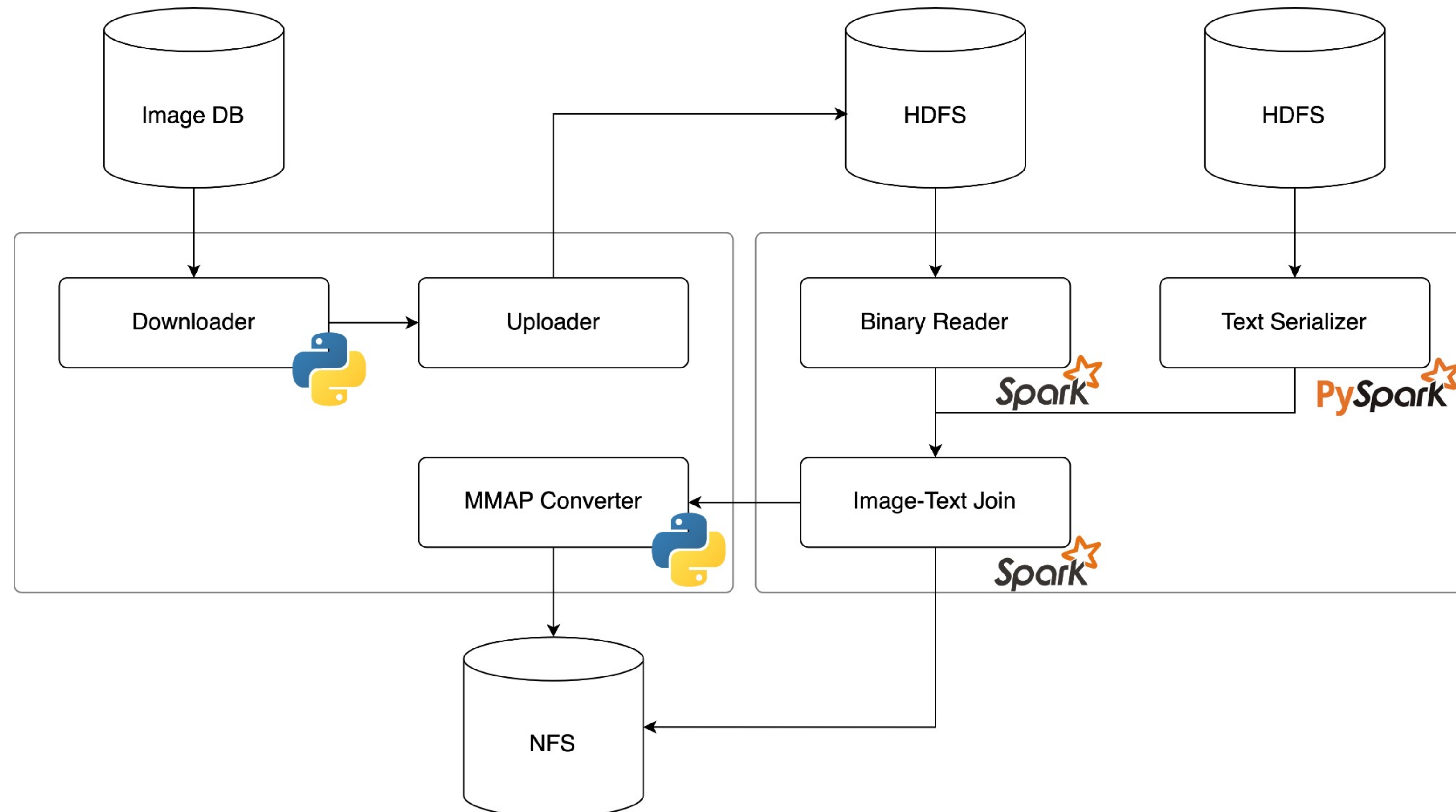
내용 불일치



던전 팁을 약간 드리자면
한 군데만 도는 것보다 여러 층을
도는 것이 경험치 수급에 좋습니다.

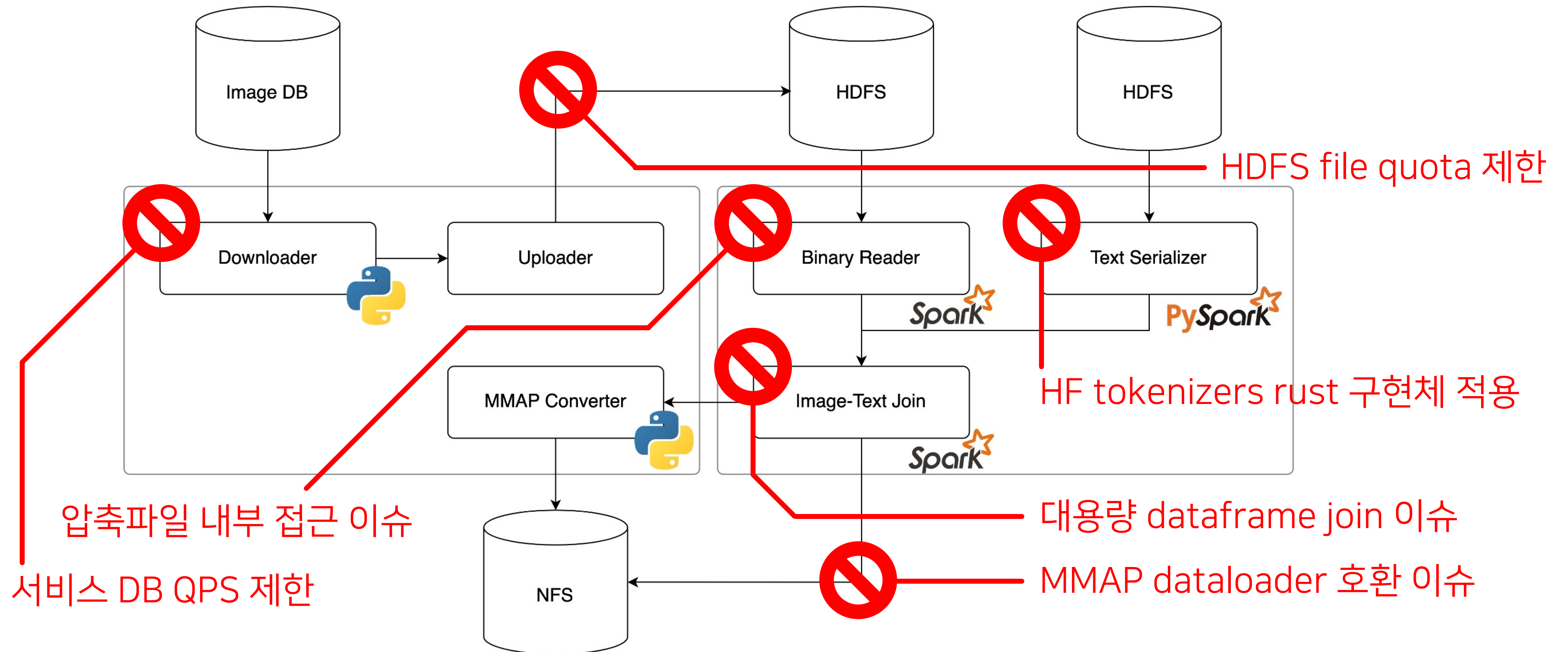
3.2.2 Data Pipeline

Image-text pair data pipeline



3.2.2 Data Pipeline (cont'd)

Pipeline issues



3.2.3 Model Architecture

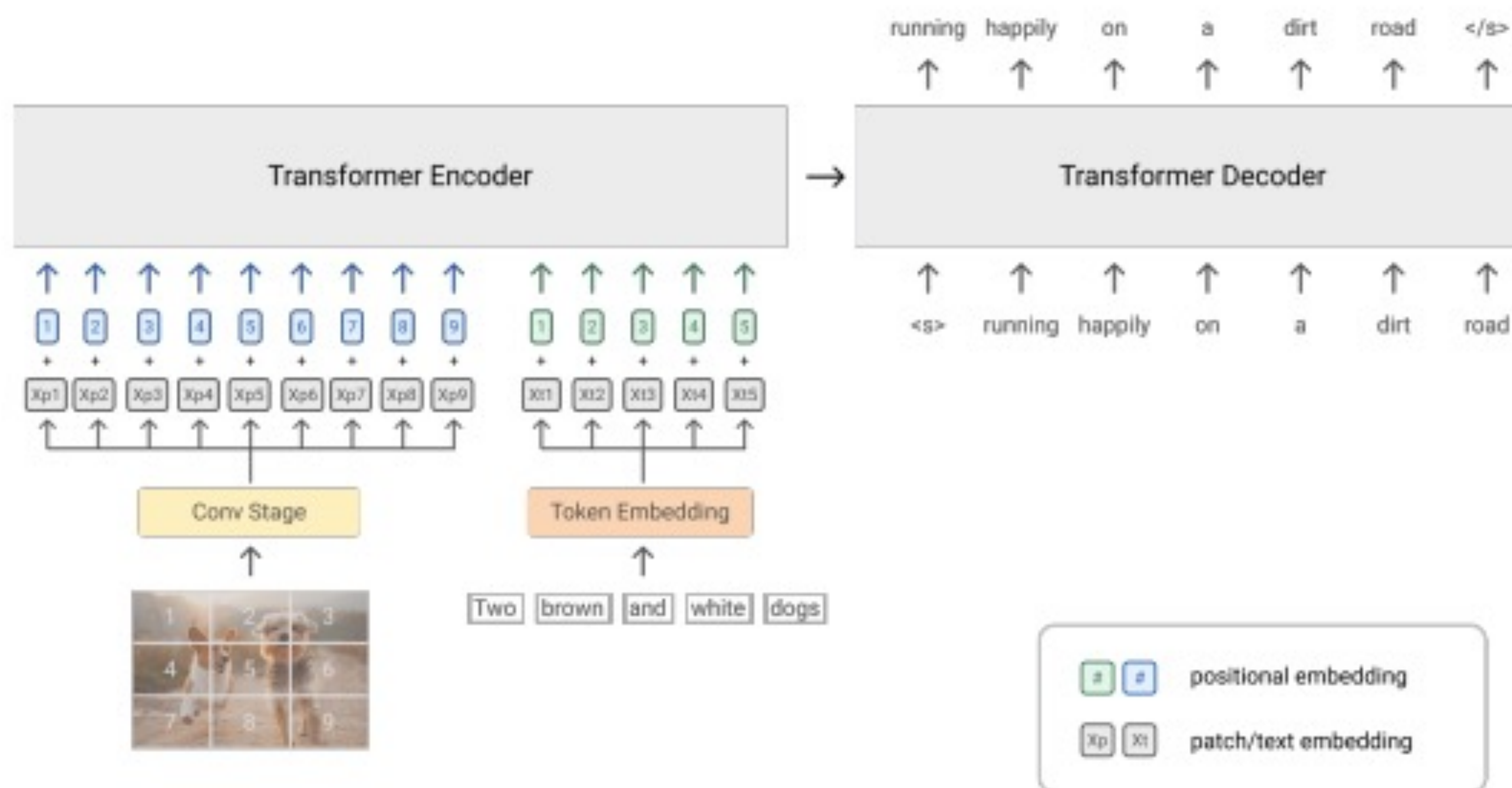
Model candidates in our research

- Transformer 구조를 근간으로 하는 모델이 주류
- Feature extraction에 사용하는 model은 architecture 별로 차이가 있음

Model	Size	Architecture	Info
CLIP	330M	ViT (ResNet) + Transformer	Text-Image Similarity
DALL-E	12B	Decoder-Only Transformer	Image Generation
M6	327M, 10B	Unified Encoder-Decoder	Multi-Task
SimVLM	86M, 307M, 632M	ViT (ResNet) + Transformer	Text Generation

3.2.3 Model Architecture (cont'd)

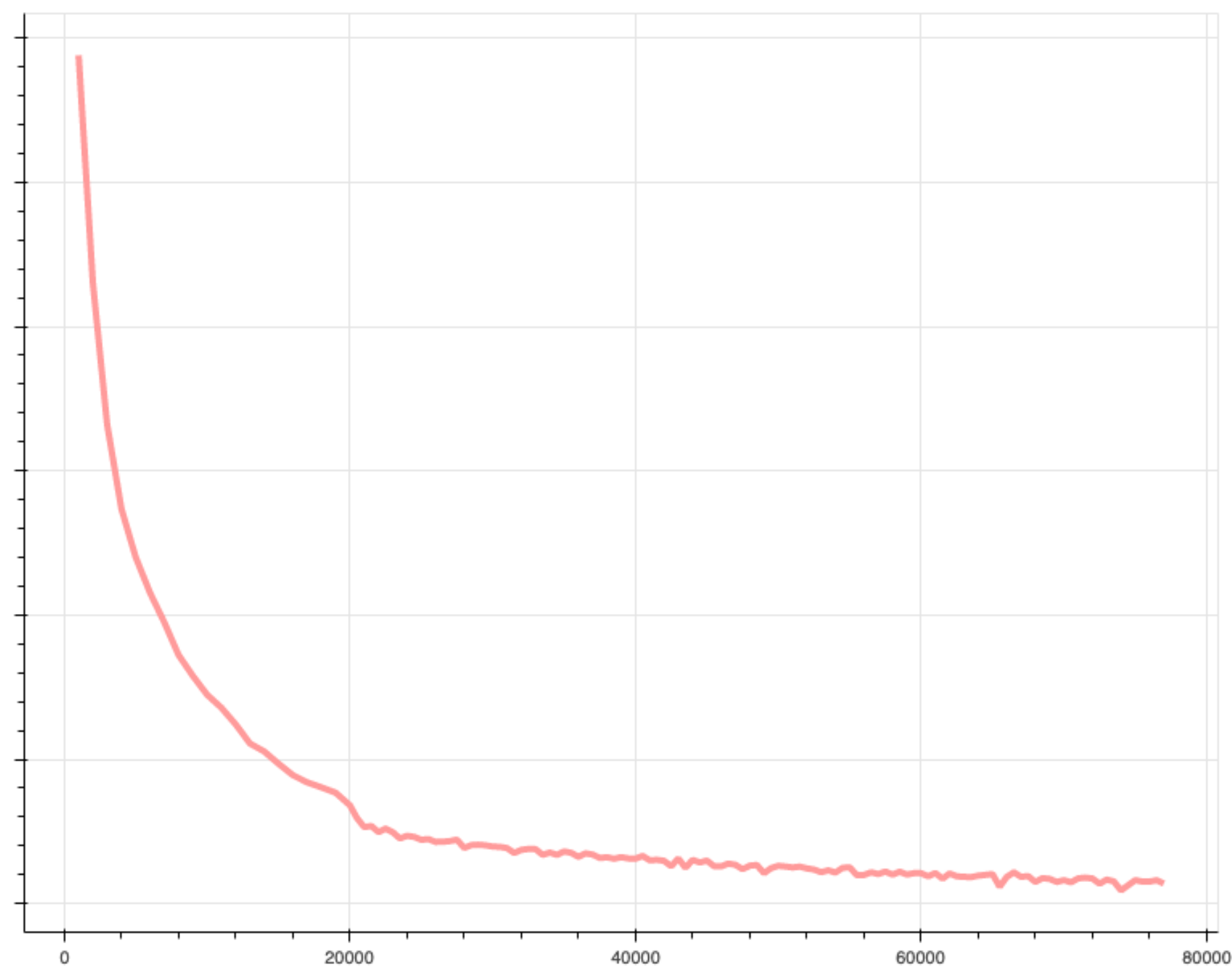
Final choice for our multi-modal model



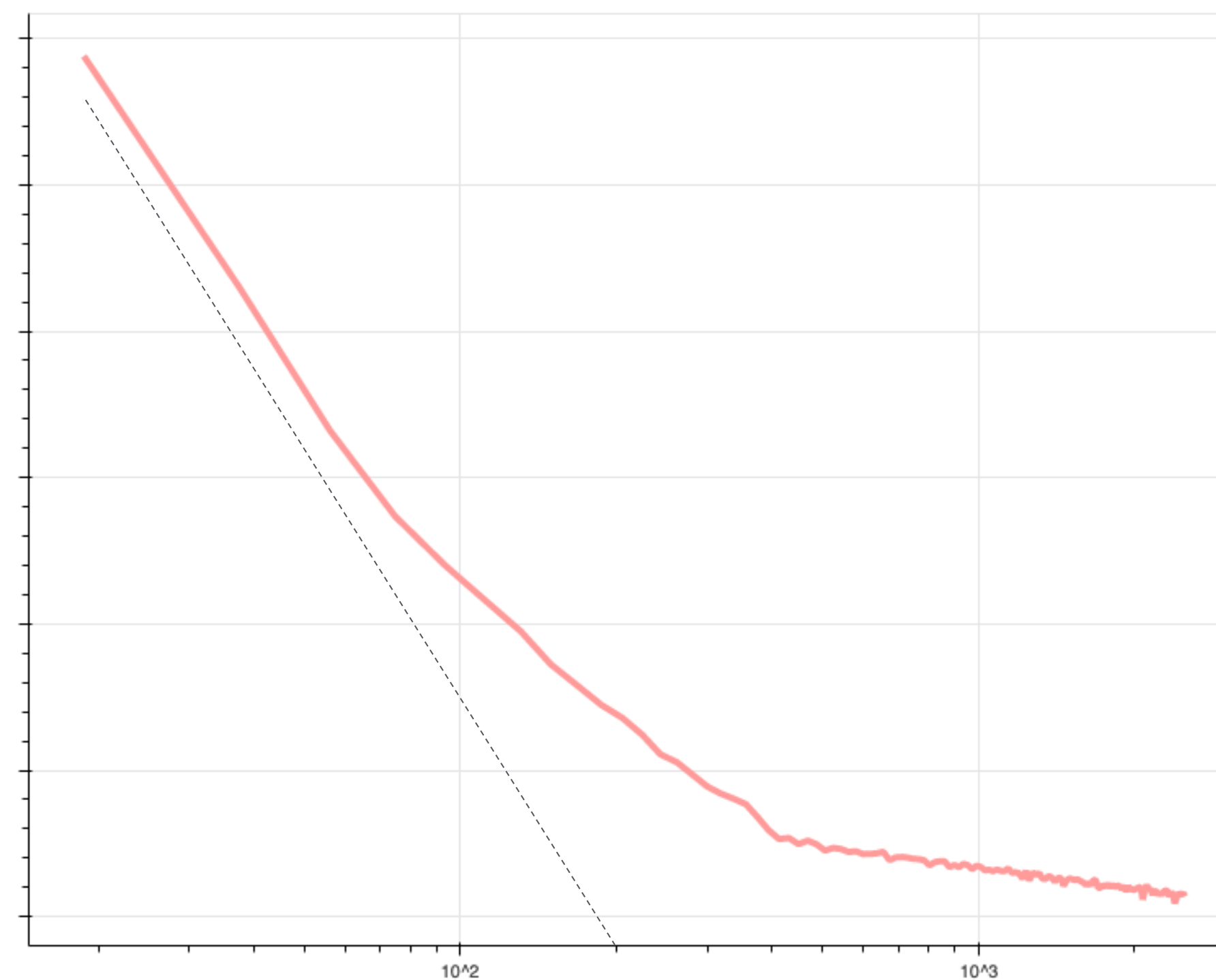
3.2.4 Model Training

Training is on going

Train loss with iterations



Train loss with log(elapsed time)



4. Roadmap & Future works

4.1 Roadmap and milestones

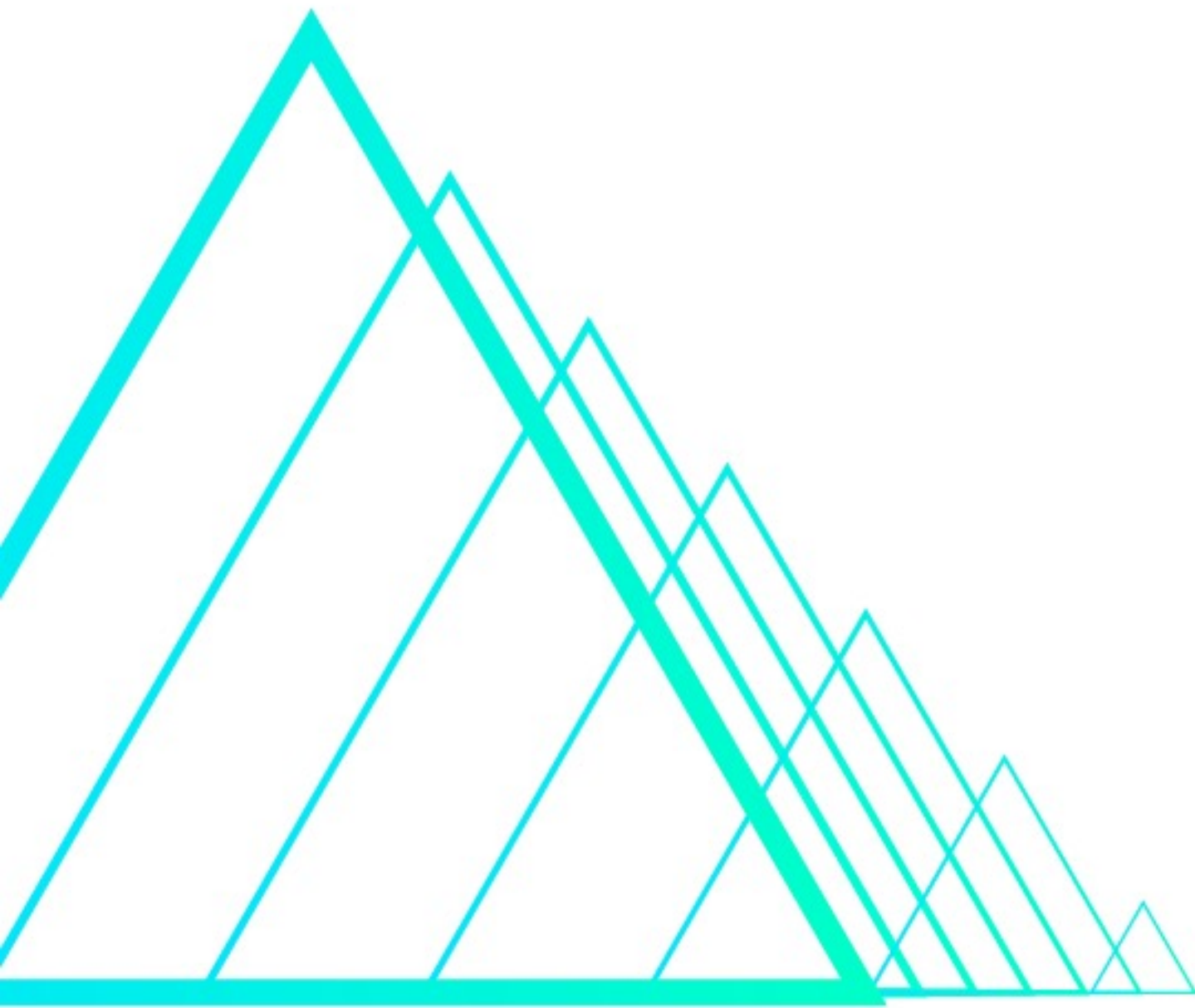
New task, new roadmap

- 새로운 태스크에 따라 roadmap은 다시 그려지고 있음
- 학습 (학습 안정화, 모니터링)
- 평가 (서술형 metric 추가, DA 안정화, 이종 언어간 평가)
- 확장 (Data mixing, Modality 확장, Language 확장)

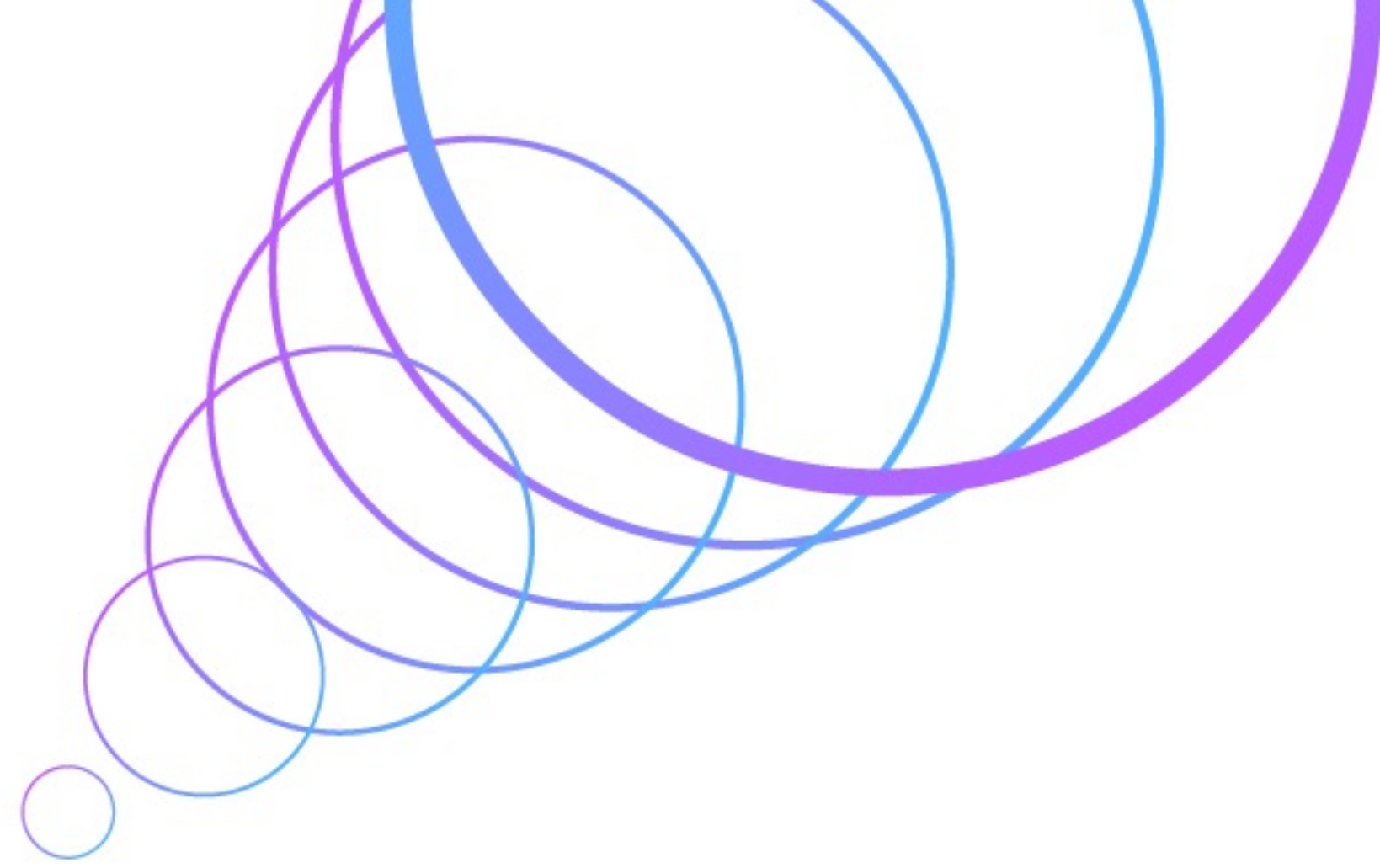
4.2 Future works

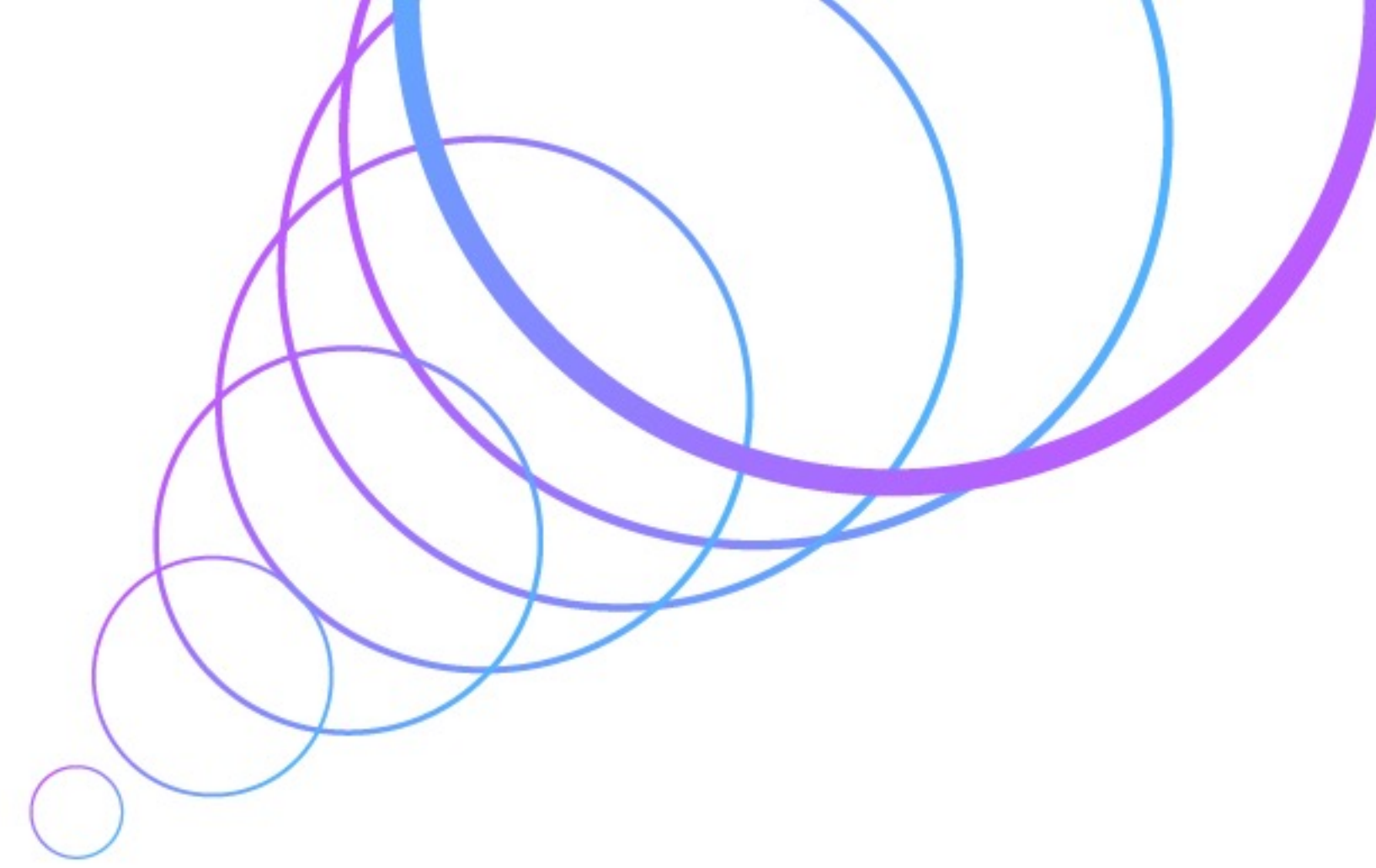
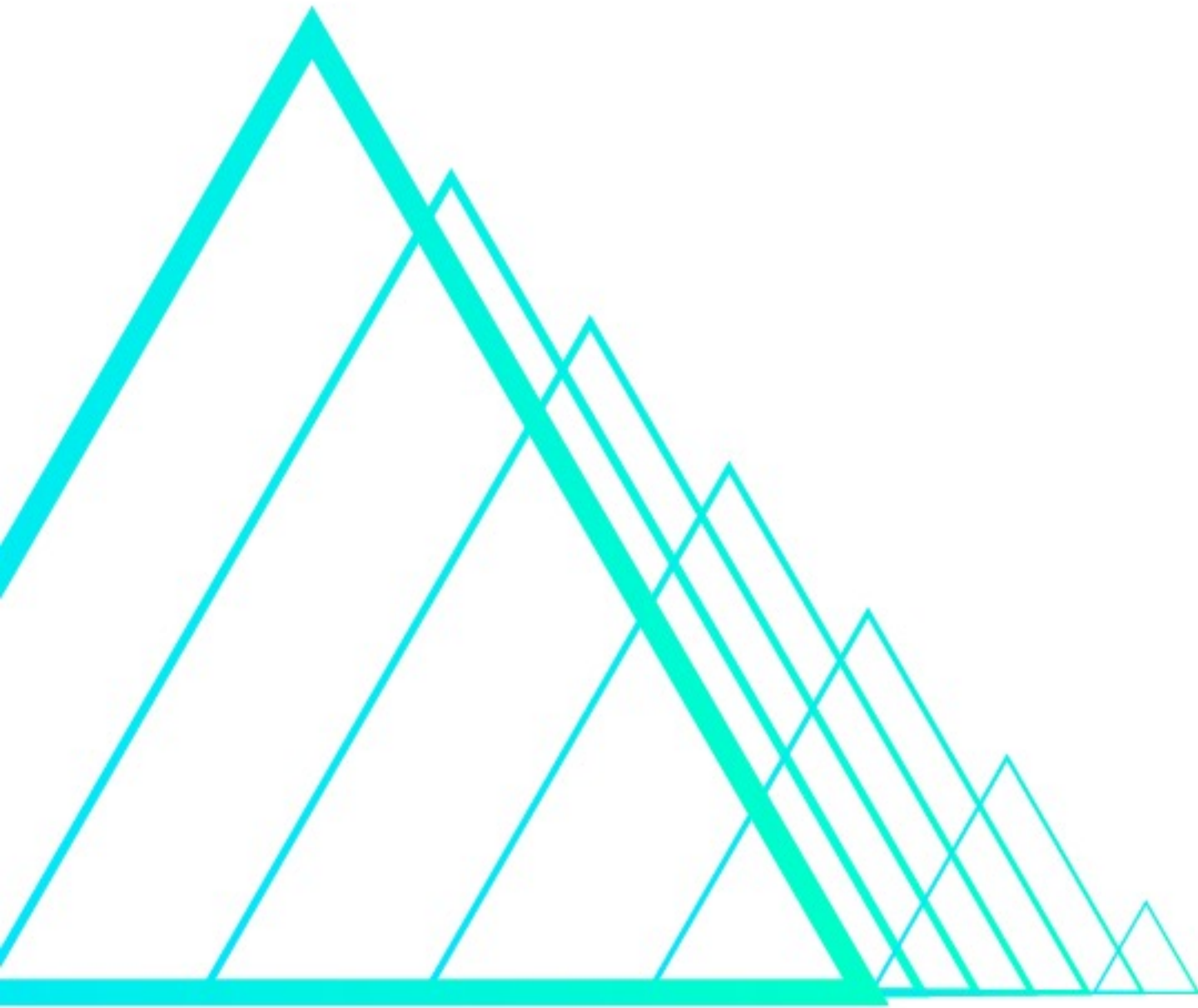
Multi-lingual model

- Multi-lingual model은 현재 매우 활발하게 연구 중인 분야
- “Multi-lingual HyperCLOVA” 세션에서 이 방향에 대해 자세히 소개



Q & A





Thank You

