2조 최종발표

이상규, 김대성, 정경송

INDEX

Dacon 한국어 문장 관계 분류 경진대회

1. Datasets
2. Modeling
3. Discussion

Datasets

Multi-Genre Natural Language Inference (MNLI)

Entailment	Neutral	Contradiction
Premise: 영화 시작부터 끝까지 긴장감을 늦출 수가 없네요. Hypothesis: 영화 시작부터 긴장감이 함께하네요.	Premise: 상당히 많은 것을 내포하고 있는 영화. Hypothesis: 인간의 감정에 대한 내용을 내포하고 있는 영화.	Premise: 최고다 이건 말로 할 수 없을 정도로 최고다. Hypothesis: 이것은 최악이다.

Dataset

Train

기본 훈련데이터

24998개

KLUE_dev

기본 훈련데이터의 **dev**

3000개

KorNLI

더 다양한 훈련데이터

7500개

Back translation

기본 훈련데이터 증강

21330개

데이터 추가 원칙

1. 많으면 많을 수록 좋다.
2. Train에서 너무 벗어난 글자 분포는 금물.
3. 코랩 Pro 환경에서 런타임 (최대 12시간)을 지킬 수 있을 정도만.

Error	Premise	Hypothesis
Word Overlap (N→E)	And, could it not result in a decline in Postal Service volumes across–the–board?	There may not be a decline in Postal Service volumes across–the–board.
Negation (E→C)	Enthusiasm for Disney's Broadway production of The Lion King dwindles.	The broadway production of The Lion King is no longer enthusiastically attended.
Numerical Reasoning (C→E)	Deborah Pryce said Ohio Legal Services in Columbus will receive a $200,000 federal grant toward an online legal self-help center.	A $900,000 federal grant will be received by Missouri Legal Services, said Deborah Pryce.
Antonymy (C→E)	"Have her show it," said Thorn.	Thorn told her to hide it.
Length Mismatch (C→N)	So you know well a lot of the stuff you hear coming from South Africa now and from West Africa that's considered world music because it's not particularly using certain types of folk styles.	They rely too heavily on the types of folk styles.
Grammaticality (N→E)	So if there are something interesting or something worried, please give me a call at any time.	The person is open to take a call anytime.
Real World Knowledge (E→N)	It was still night.	The sun hadn't risen yet, for the moon was shining daringly in the sky.
Ambiguity (E→N)	Outside the cathedral you will find a statue of John Knox with Bible in hand.	John Knox was someone who read the Bible.
Unknown (E→C)	We're going to try something different this morning, said Jon.	Jon decided to try a new approach.

1806.00692.pdf (arxiv.org) 참고할만함

KLUE_dev

- 가장 부담없이 데이터 수를 늘릴 수 있음
- Train 데이터와 같은 분포에서 나옴
- 전처리 X

Source	Train	Dev
Wikitree	3838	450
Policy	3833	450
wikinews	3824	450
Wikipedia	3780	450
Nsmc	4899	600
Airbnb	4824	600
Overall	24998	3000

korNLI

DACONIO: 이번 대회에서는 "*Premise 문장이 공개되어 있다고 하더라도 새로운 Hypothesis 문장이 주어졌을 때 과연 정답을 얼마나 맞출 수 있는지*"가 중요

-> 모델이 하나의 Premise 에서 다양한 Hypothesis를 접해봐야 함.

그리고 그가 말했다, "엄마, 저 왔어요."	그는 학교 버스가 그를 내려주자마자 엄마에게 전화를 걸었다.	neutral
그리고 그가 말했다, "엄마, 저 왔어요."	그는 한마디도 하지 않았다.	contradiction
그리고 그가 말했다, "엄마, 저 왔어요."	그는 엄마에게 집에 갔다고 말했다.	entailment

korNLI

- Train에 존재하지 않는 특수문자는 대체

예:

```
korNLI['premise'] = korNLI['premise'].str.replace("[《》『』]", "'")
korNLI['hypothesis'] = korNLI['hypothesis'].str.replace("[《》『』]", "'")
```

- Train (machine-translated) 데이터의 양이 너무 많은 관계로 dev와 test만 활용.

Back translation

- 데이터양을 증가시키기 위해 기본 train data를 back translation

- 데이콘 뉴스 토픽 분류 AI 경진대회에서 최종 3위를 기록한 코드 참고 (네이버 Papago를 크롤링해서 번역)

- 번역 후 번역되지 않은 데이터와 원본 문장 길이에 비해 지나치게 긴 번역은 이상번역치로 간주하고 drop, 오타가 존재하는 데이터는 직접 교정, 특수문자와 한자가 있는 데이터는 대체

예 :

```
back_trans['premise'] = back_trans['premise'].str.replace("㎡", '제곱미터')
```

최종 글자 분포

Train "%',./0123456789:?ABCDEFGHIJKLMNOPQRSTUVWYaehkmnoprsx~ ·가각간 …

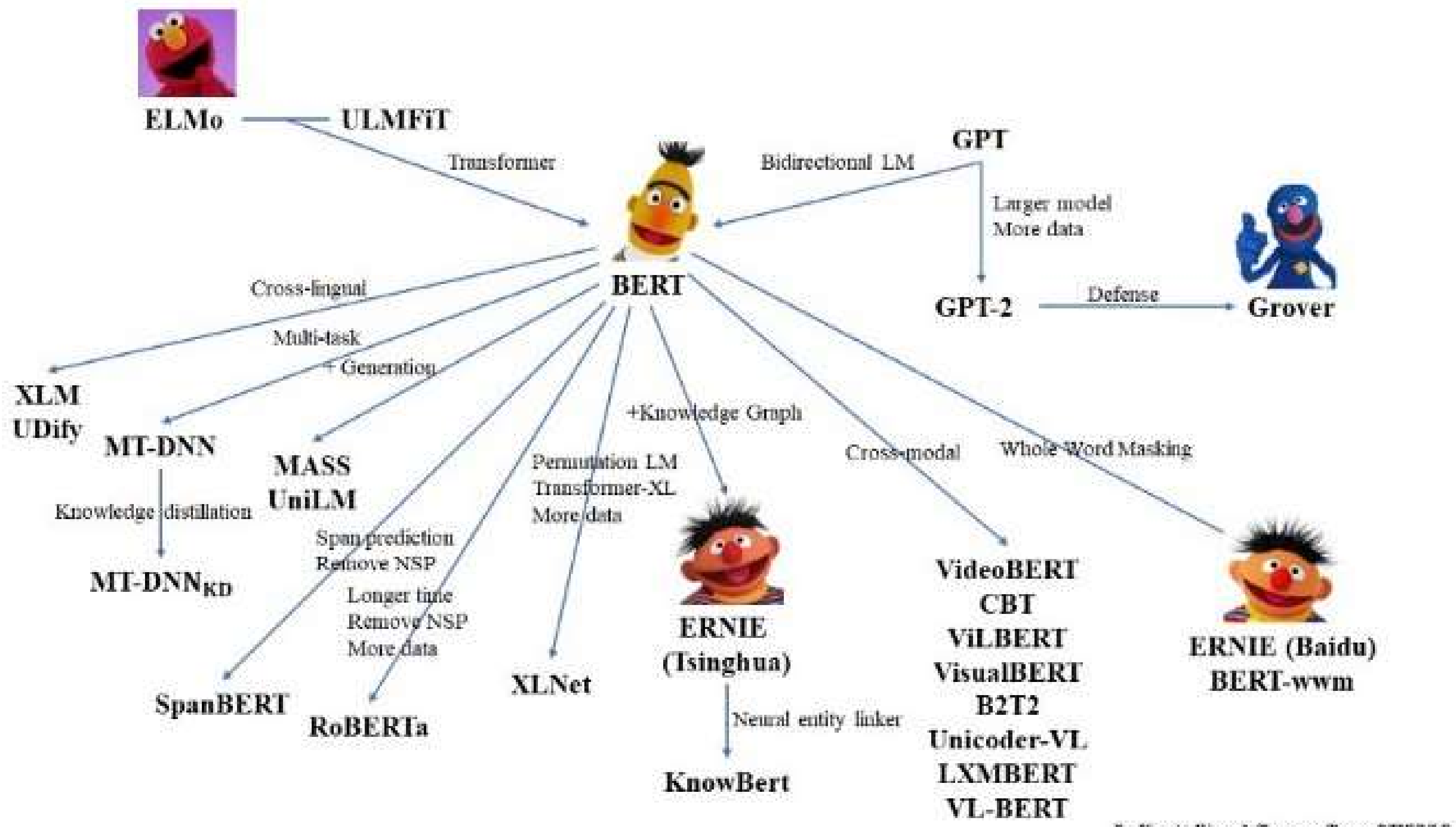
KLUE "%'(),.0123456789:?AFIKRSTkm ‘’“”가각간 …

korNLI !"%&'(),-.0123456789:;=?ABCDEFGHIJKLMNOPQRSTUVWXYZabcdefghijklmnopqrstuvwxyz~가각간 …

back_trans "'(),-./0123456789:?ABCDEFGHIJKLMNOPQRSTUVWYZabcdefghiklmnoprstuvwxyz~ ·가각간 …

Modeling

Lots, Lots of Pre-trained Model



Lots, Lots of Pre-trained Model

XLNet: Generalized Autoregressive Pretraining for Language Understanding

Zhilin Yang[*1], **Zihang Dai**[*12], **Yiming Yang**[1], **Jaime Carbonell**[1],
Ruslan Salakhutdinov[1], **Quoc V. Le**[2]
[1]Carnegie Mellon University, [2]Google AI Brain Team
{zhiliny,dzihang,yiming,jgc,rsalakhu}@cs.cmu.edu, qvl@google.com

ALBERT: A LITE BERT FOR SELF-SUPERVISED LEARNING OF LANGUAGE REPRESENTATIONS

Zhenzhong Lan[1] **Mingda Chen**[2*] **Sebastian Goodman**[1] **Kevin Gimpel**[2]
Piyush Sharma[1] **Radu Soricut**[1]
[1]Google Research [2]Toyota Technological Institute at Chicago
{lanzhzh, seabass, piyushsharma, rsoricut}@google.com
{mchen, kgimpel}@ttic.edu

RoBERTa: A Robustly Optimized BERT Pretraining Approach

Yinhan Liu[*§] **Myle Ott**[*§] **Naman Goyal**[*§] **Jingfei Du**[*§] **Mandar Joshi**[†]
Danqi Chen[§] **Omer Levy**[§] **Mike Lewis**[§] **Luke Zettlemoyer**[†§] **Veselin Stoyanov**[§]
[†] Paul G. Allen School of Computer Science & Engineering,
University of Washington, Seattle, WA
{mandar90,lsz}@cs.washington.edu
[§] Facebook AI
{yinhanliu,myleott,naman,jingfeidu,
danqi,omerlevy,mikelewis,lsz,ves}@fb.com

ELECTRA: PRE-TRAINING TEXT ENCODERS AS DISCRIMINATORS RATHER THAN GENERATORS

Kevin Clark
Stanford University
kevclark@cs.stanford.edu

Minh-Thang Luong
Google Brain
thangluong@google.com

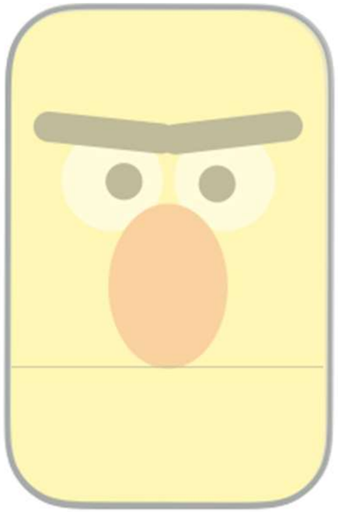
Quoc V. Le
Google Brain
qvl@google.com

Christopher D. Manning
Stanford University & CIFAR Fellow
manning@cs.stanford.edu

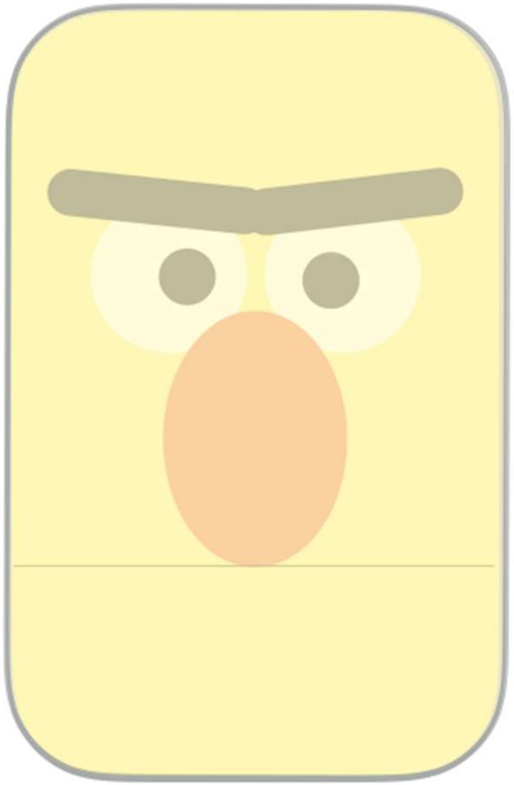
Pre-trained Model on Korean – Lack of ‘Large' Model

- klue/roberta-large
  Fill-Mask • Updated Oct 21, 2021 • 35.2k • 7
- klue/bert-base
  Fill-Mask • Updated Oct 21, 2021 • 33.5k • 3
- monologg/koelectra-base-v3-discriminator
  Updated Oct 21, 2021 • 26.5k • 8
- klue/roberta-base
  Fill-Mask • Updated Oct 21, 2021 • 16.9k
- monologg/kobigbird-bert-base
  Fill-Mask • Updated Nov 5, 2021 • 4.29k • 7
- lassl/roberta-ko-small
  Fill-Mask • Updated 6 days ago • 2.81k • 2
- klue/roberta-small
  Fill-Mask • Updated Oct 21, 2021 • 2.41k
- monologg/koelectra-base-v3-generator
  Fill-Mask • Updated Oct 21, 2021 • 2.12k • 1
- monologg/koelectra-base-discriminator
  Updated Oct 21, 2021 • 1.45k
- monologg/koelectra-base-generator
  Fill-Mask • Updated Oct 21, 2021 • 1.36k
- monologg/koelectra-base-v2-discriminator
  Updated Oct 21, 2021 • 910 • 1
- lassl/gpt2-ko-small
  Text Generation • Updated 6 days ago • 80 • 1
- lassl/bert-ko-base
  Fill-Mask • Updated 6 days ago • 41 • 1
- monologg/koelectra-base-v2-generator
  Fill-Mask • Updated Oct 21, 2021 • 19
- DeadBeast/korscm-mBERT
  Text Classification • Updated Aug 22, 2021 • 17 • 1
- Jinhwan/krelectra-base-mecab
  Updated Jan 12 • 14 • 1
- lassl/bert-ko-small
  Fill-Mask • Updated 6 days ago • 11

Pre-trained Model on Korean – Lack of 'Large' Model



$\text{BERT}_{\text{BASE}}$



$\text{BERT}_{\text{LARGE}}$

Pre-trained Model on Korean – Lack of 'Large' Model

System	MNLI-(m/mm)	QQP	QNLI	SST-2	CoLA	STS-B	MRPC	RTE	**Average**
	392k	363k	108k	67k	8.5k	5.7k	3.5k	2.5k	-
Pre-OpenAI SOTA	80.6/80.1	66.1	82.3	93.2	35.0	81.0	86.0	61.7	74.0
BiLSTM+ELMo+Attn	76.4/76.1	64.8	79.8	90.4	36.0	73.3	84.9	56.8	71.0
OpenAI GPT	82.1/81.4	70.3	87.4	91.3	45.4	80.0	82.3	56.0	75.1
$\text{BERT}_{\text{BASE}}$	84.6/83.4	71.2	90.5	93.5	52.1	85.8	88.9	66.4	79.6
$\text{BERT}_{\text{LARGE}}$	**86.7/85.9**	**72.1**	**92.7**	**94.9**	**60.5**	**86.5**	**89.3**	**70.1**	**82.1**

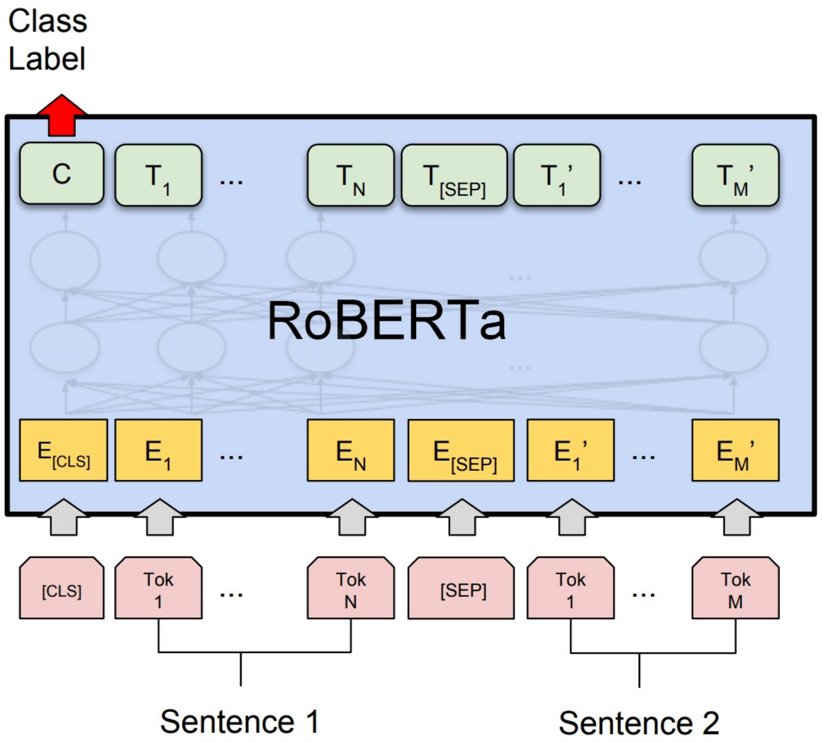
Table 1: GLUE Test results, scored by the evaluation server (`https://gluebenchmark.com/leaderboard`). The number below each task denotes the number of training examples. The "Average" column is slightly different than the official GLUE score, since we exclude the problematic WNLI set.[8] BERT and OpenAI GPT are single-model, single task. F1 scores are reported for QQP and MRPC, Spearman correlations are reported for STS-B, and accuracy scores are reported for the other tasks. We exclude entries that use BERT as one of their components.

Pre-trained Model on Korean – Lack of 'Large' Model

Model	Train FLOPs	Params	CoLA	SST	MRPC	STS	QQP	MNLI	QNLI	RTE	Avg.
BERT	1.9e20 (0.27x)	335M	60.6	93.2	88.0	90.0	91.3	86.6	92.3	70.4	84.0
RoBERTa-100K	6.4e20 (0.90x)	356M	66.1	95.6	**91.4**	92.2	92.0	89.3	94.0	82.7	87.9
RoBERTa-500K	3.2e21 (4.5x)	356M	68.0	96.4	90.9	92.1	92.2	90.2	94.7	86.6	88.9
XLNet	3.9e21 (5.4x)	360M	69.0	**97.0**	90.8	92.2	92.3	90.8	94.9	85.9	89.1
BERT (ours)	7.1e20 (1x)	335M	67.0	95.9	89.1	91.2	91.5	89.6	93.5	79.5	87.2
ELECTRA-400K	7.1e20 (1x)	335M	**69.3**	96.0	90.6	92.1	**92.4**	90.5	94.5	86.8	89.0
ELECTRA-1.75M	3.1e21 (4.4x)	335M	69.1	96.9	90.8	**92.6**	**92.4**	**90.9**	**95.0**	**88.0**	**89.5**

Table 2: Comparison of large models on the GLUE dev set. ELECTRA and RoBERTa are shown for different numbers of pre-training steps, indicated by the numbers after the dashes. ELECTRA performs comparably to XLNet and RoBERTa when using less than 1/4 of their pre-training compute and outperforms them when given a similar amount of pre-training compute. BERT dev results are from Clark et al. (2019).

KLUE/RoBERTa



KLUE: Korean Language Understanding Evaluation

Sungjoon Park[*] Upstage, KAIST sungjoon.park@kaist.ac.kr
Jihyung Moon[*] Upstage jihyung.moon@upstage.ai
Sungdong Kim[*] NAVER AI Lab sungdong.kim@navercorp.com
Won Ik Cho[*] Seoul National University tsatsuki@snu.ac.kr
Jiyoon Han[†] Yonsei University clinamen35@yonsei.ac.kr
Jangwon Park jangwon.pk@gmail.com
Chisung Song daydrilling@gmail.com
Junseong Kim Scatter Lab junseong.kim@scatterlab.co.kr
Youngsook Song KyungHee University youngsoksong@khu.ac.kr
Taehwan Oh[†] Yonsei University ghks10604@yonsei.ac.kr
Joohong Lee Scatter Lab joohong@scatterlab.co.kr
Juhyun Oh[†] Seoul National University 411juhyun@snu.ac.kr
Sungwon Lyu Kakao Enterprise james.ryu@kakaoenterprise.com
Younghoon Jeong Sogang University boychaboy@sogang.ac.kr
Inkwon Lee Sogang University md98765@naver.com
Sangwoo Seo Scatter Lab sangwoo@scatterlab.co.kr
Dongjun Lee humanbrain.djlee@gmail.com
Hyunwoo Kim Seoul National University hyunw.kim@vl.snu.ac.kr
Myeonghwa Lee KAIST myeon9h@kaist.ac.kr
Seongbo Jang Scatter Lab seongbo@scatterlab.co.kr
Seungwon Do seungwon.do1@gmail.com
Sunkyoung Kim KAIST sunkyoung@kaist.ac.kr
Kyungtae Lim Hanbat National University ktlim@hanbat.ac.kr
Jongwon Lee mybizzer@gmail.com
Kyumin Park KAIST pkm9403@kaist.ac.kr
Jamin Shin Riiid AI Research jshin49@gmail.com
Seonghyun Kim bananaband657@gmail.com
Lucy Park Upstage lucy@upstage.ai
Alice Oh[**] KAIST alice.oh@kaist.edu
Jung-Woo Ha[**] NAVER AI Lab jungwoo.ha@navercorp.com
Kyunghyun Cho[**] New York University kyunghyun.cho@nyu.edu

Problem: Increasing Accuracy, But Also Loss

Step	Training Loss	Validation Loss	Accuracy
500	0.636200	0.460690	0.834200
1000	0.370700	0.444924	0.848800
1500	0.271200	0.557969	0.851400
2000	0.192400	0.746097	0.861200
2500	0.136500	0.542568	0.864600
3000	0.081800	0.615479	0.868200
3500	0.058500	0.683380	0.867600
4000	0.034100	0.782043	0.870400

submission.csv
baseline edit

2022-02-12 18:47:43 0.83

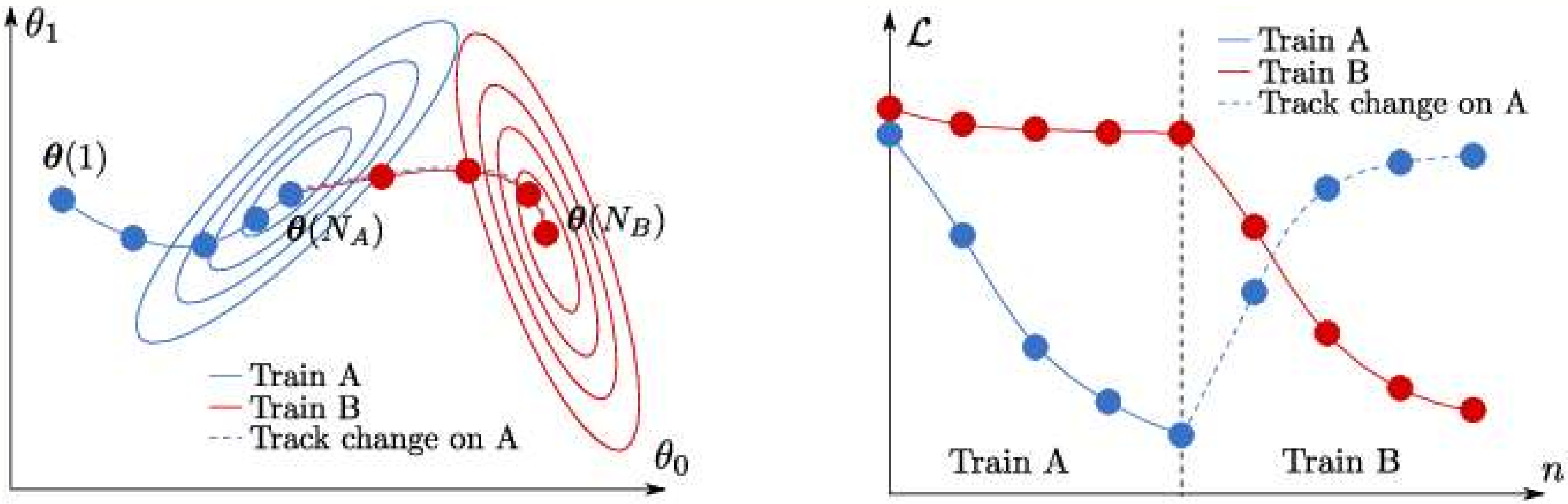
Needs Better Fine-tuning Method!

BETTER FINE-TUNING BY REDUCING REPRESENTATIONAL COLLAPSE

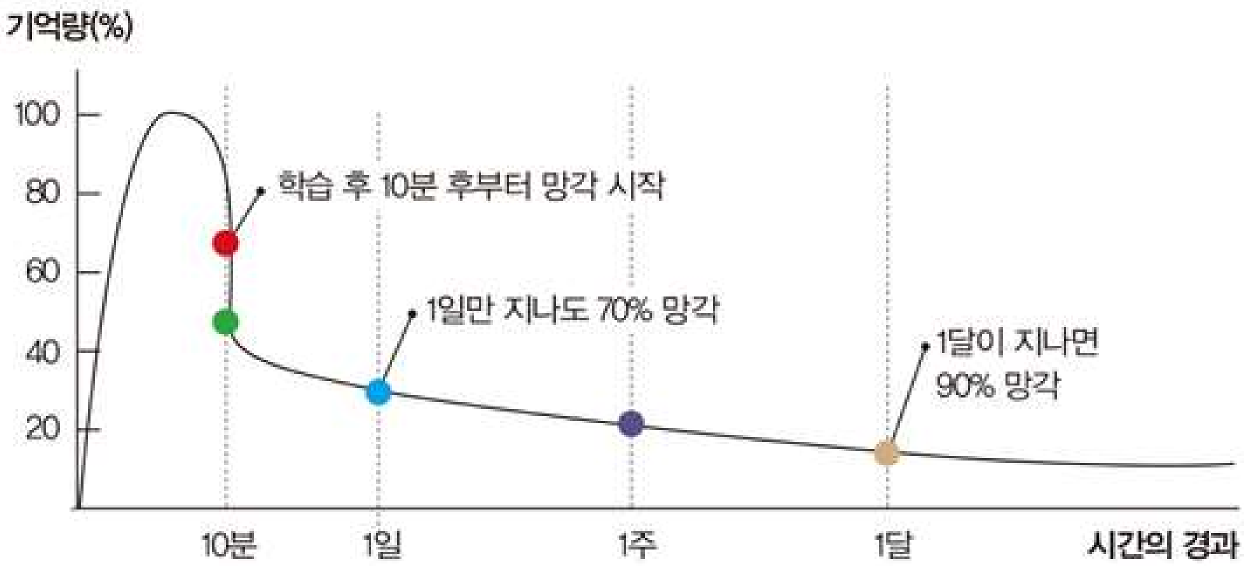
Armen Aghajanyan, Akshat Shrivastava, Anchit Gupta & Naman Goyal
Facebook
{armenag,akshats,anchit,naman}@fb.com

Luke Zettlemoyer & Sonal Gupta
Facebook
{lsz, sonalgupta}@fb.com

Catastrophic Forgetting



Catastrophic Forgetting



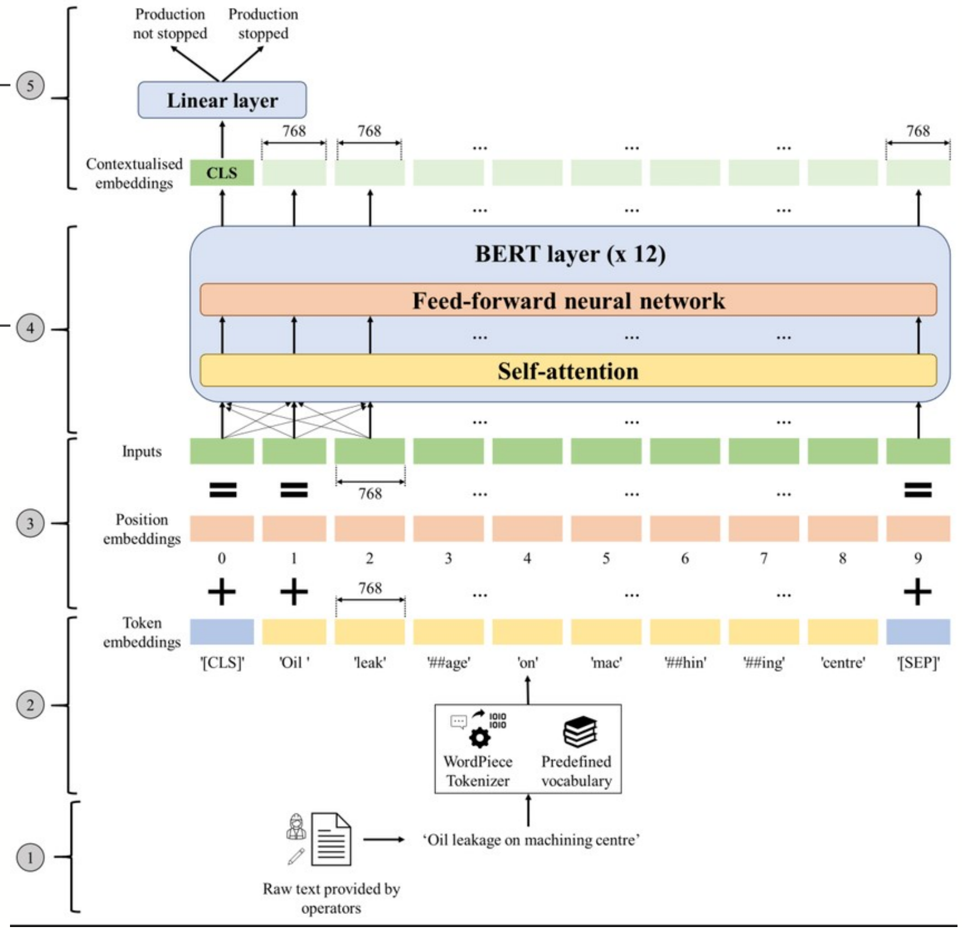
NLI Architecture as Functions

$$g : \mathbb{R}^p \rightarrow \mathbb{R}^q$$

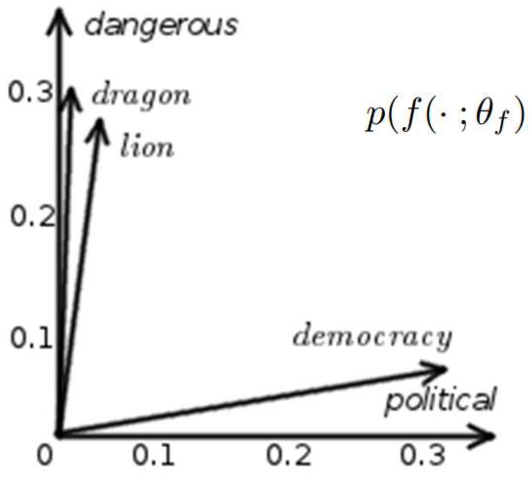
$$f : \mathbb{R}^{m \times n} \rightarrow \mathbb{R}^p$$

$$\Downarrow$$

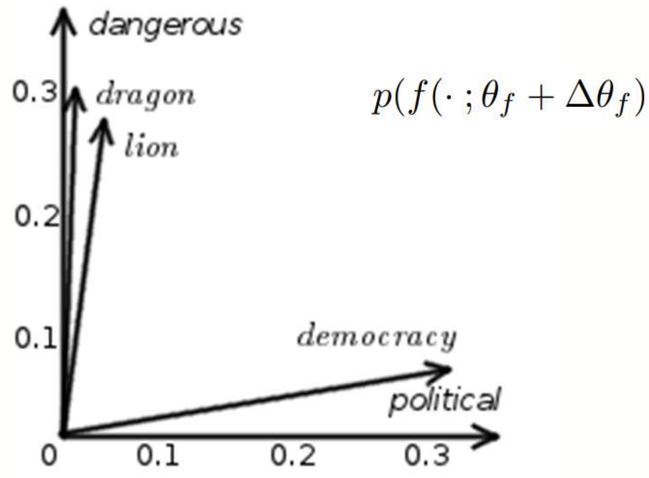
$$g \cdot f(x)$$



Purpose: Robust Representational Space



update



$$\arg\min_{\Delta\theta} \mathcal{L}(\theta + \Delta\theta)$$

$$s.t.\ KL(p(f(\cdot\,;\theta_f))||p(f(\cdot\,;\theta_f + \Delta\theta_f))) = \epsilon$$

$\longrightarrow$ $p(f)$ is intractable

Other Approximation: Too Expensive

SMART: Robust and Efficient Fine-Tuning for Pre-trained Natural Language Models through Principled Regularized Optimization

Haoming Jiang, Pengcheng He, Weizhu Chen, Xiaodong Liu, Jianfeng Gao, Tuo Zhao *

$$\mathcal{L}_{SMART}(\theta, f, g) = \mathcal{L}(\theta) + \lambda \overset{\text{intractable}}{\boxed{\mathbb{E}_{x \sim X}\left[\sup_{x^{\sim}:|x^{\sim}-x|\leq\epsilon} KL_S(g \cdot f(x) \,\|\, g \cdot f(x^{\sim}))\right]}}$$

→ Approximate by gradient ascents(Adversarial training) = high computational cost

R3F: Cheaper Approximation for Robust Representation

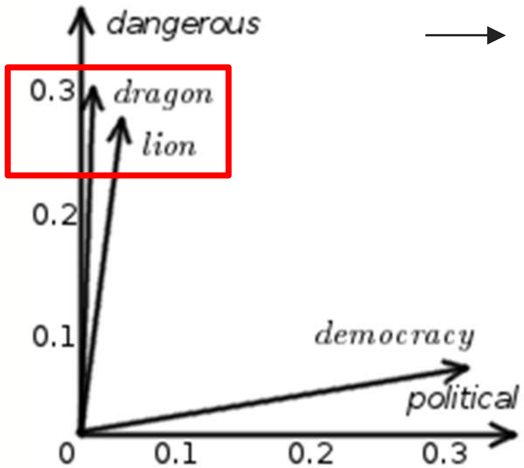
$$\mathcal{L}_{R3}(f, g, \theta) = \mathcal{L}(\theta) + \lambda KL_S(g \cdot f(x) \,\|\, g \cdot f(x + z))$$

R3F Method

$$s.t. \quad z \sim \mathcal{N}(0, \sigma^2 I) \text{ or } z \sim \mathcal{U}(-\sigma, \sigma)$$

$$s.t. \quad Lip\{g\} \leq 1$$

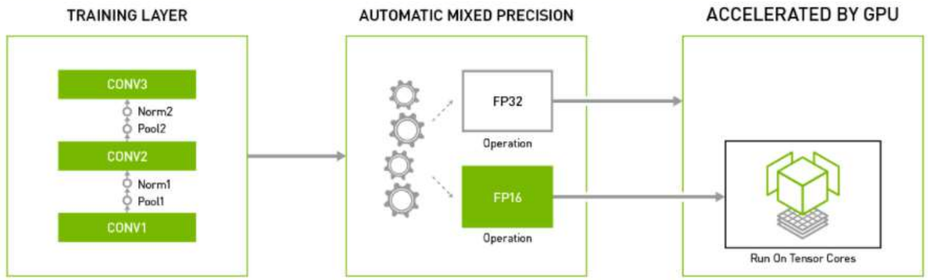
Optional **R4F** Method



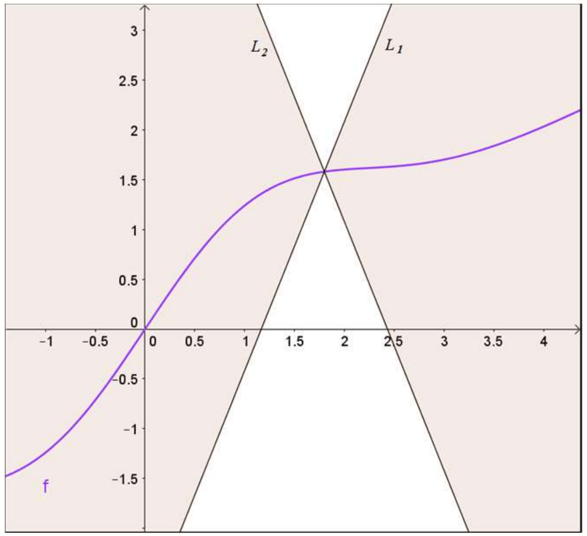
→ Gives penalty when differ of classification occurs on similar embedding vectors!

	FP	BP	xFP
FreeLB	$1+S$	$1+S$	$3+3S$
SMART	$1+S$	$1+S$	$3+3S$
R3F/R4F	2	1	4
Standard	1	1	3

Table 1: Computational cost of recently proposed fine-tuning algorithms. We show Forward Passes (FP), Backward Passes (BP) as well as computation cost as a factor of forward passes (xFP). S is the number of gradient ascent steps, with a minimum of $S \geq 1$



R4F – Needs Classifier to be Lipschitz Function(SN-GAN)



SPECTRAL NORMALIZATION
FOR GENERATIVE ADVERSARIAL NETWORKS

Takeru Miyato[1], Toshiki Kataoka[1], Masanori Koyama[2], Yuichi Yoshida[3]
{miyato, kataoka}@preferred.jp
koyama.masanori@gmail.com
yyoshida@nii.ac.jp
[1]Preferred Networks, Inc. [2]Ritsumeikan University [3]National Institute of Informatics

$$\sigma(A) := \max_{\boldsymbol{h}:\boldsymbol{h}\neq\boldsymbol{0}} \frac{\|A\boldsymbol{h}\|_2}{\|\boldsymbol{h}\|_2} = \max_{\|\boldsymbol{h}\|_2\leq 1} \|A\boldsymbol{h}\|_2, \quad \bar{W}_{\mathrm{SN}}(W) := W/\sigma(W)$$

→ Not available; Lack of VRAM capacity on Colab Pro (Tesla P100, VRAM < 16G)

Another Approach: Gradient Penalty(WGAN-GP)

Improved Training of Wasserstein GANs

Ishaan Gulrajani[1]*, Faruk Ahmed[1], Martin Arjovsky[2], Vincent Dumoulin[1], Aaron Courville[1,3]
[1] Montreal Institute for Learning Algorithms
[2] Courant Institute of Mathematical Sciences
[3] CIFAR Fellow
igul222@gmail.com
{faruk.ahmed,vincent.dumoulin,aaron.courville}@umontreal.ca
ma4371@nyu.edu

$$L = \underbrace{\mathop{\mathbb{E}}_{\tilde{\boldsymbol{x}} \sim \mathbb{P}_g} [D(\tilde{\boldsymbol{x}})] - \mathop{\mathbb{E}}_{\boldsymbol{x} \sim \mathbb{P}_r} [D(\boldsymbol{x})]}_{\text{Original critic loss}} + \underbrace{\lambda \mathop{\mathbb{E}}_{\hat{\boldsymbol{x}} \sim \mathbb{P}_{\hat{\boldsymbol{x}}}} \left[(\|\nabla_{\hat{\boldsymbol{x}}} D(\hat{\boldsymbol{x}})\|_2 - 1)^2 \right]}_{\text{Our gradient penalty}} .$$

takerum commented on 29 May 2018 Contributor

Hi,

We have tried the combination of wgan loss and spectral normalization, but it does not work.
We are not sure why that happens, and also happy if you have ideas on that problem!

Having said that, one shall not rule out the possibility that the gradient penalty can compliment spectral normalization and vice versa. Because these two methods regularizes discriminators by completely different means, and in the experiment section, we actually confirmed that combination of WGAN-GP and reparametrization with spectral normalization improves the quality of the generated examples over the baseline (WGAN-GP only).

→ Not tried; Different results even if same purpose

Current State: Public 0.889(20th, <10%)

SETTINGS	VALUE
PRE-TRAINED MODEL	KLUE/RoBERTa-large
OBJECTIVE	R3F Loss
OPTIMIZER	AdamW
SCHEDULER	Polynomial with warmup
POLICY	Early Stopping
DATA SPLIT	Stratified K-folds (5-fold)
ENSEMBLE	Soft voting

HYPERPARAMETERS	VALUE
LEARNING RATES	1e-5
EPOCHS	5
BATCH SIZE	[28, 30, 32]
WARMUP RATIO	0.2
LAMBDA	[0, 0.5, 1, 2]
NOISE TYPE	Gaussian
STANDARD DEVIATION	1e-5

20	YBIGTA-NLP	Od tr so	0.889

Discussion

Transformer is Strong against Data Noise than Expected

Noise Filtering We remove noisy and/or non-Korean text from the selected source corpora. We first remove hashtags (e.g., #JMT), HTML tags (e.g.,
), bad characters (e.g., U+200B (zero-width space), U+FEFF (byte order mark)), empty parenthesis (e.g., ()), and consecutive blanks. We then filter out sentences with more than 10 Chinese or Japanese characters. For the corpora derived from news articles, we remove information about reporters and press, images, source tags as well as copyright tags (e.g., copyright by ©).

Train `"%',./0123456789:?ABCDEFGHIJKLMNOPQRSTUVWYaehkmnoprsx~ ·가각간 …`

KLUE `"%'(),.0123456789:?AFIKRSTkm ‵′“”가각간 …`

korNLI `!"%&'(),-.0123456789:;=?ABCDEFGHIJKLMNOPQRSTUVWXYZabcdefghijklmnopqrstuvwxyz~가각간 …`

back_trans `"'(),-./0123456789:?ABCDEFGHIJKLMNOPQRSTUVWYZabcdefghiklmnoprstuvwxyz~ ·가각간 …`

ybigta

Fine-tuning Requires Fine Hyperparameter Searching

Transformers

OPTUNA

RAY

Not Completely Solved: Increasing Accuracy, But Also Loss

Epoch	Training Loss	Validation Loss	Accuracy
1	0.379400	0.309445	0.889535
2	0.238300	0.314908	0.909619
3	0.148100	0.389824	0.912879

Q&A