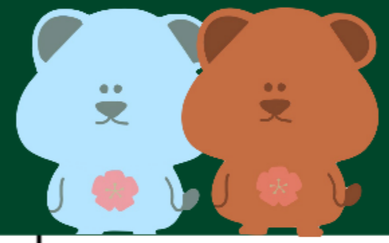


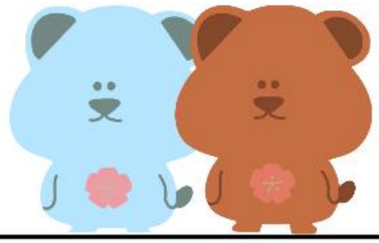
# Future of NLP + Deep Learning

김수한 , 문예지

EWHA EURON

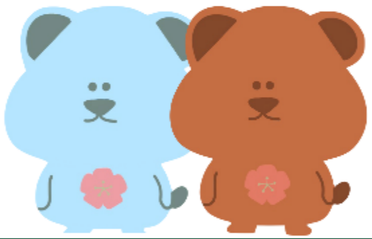
# 목차





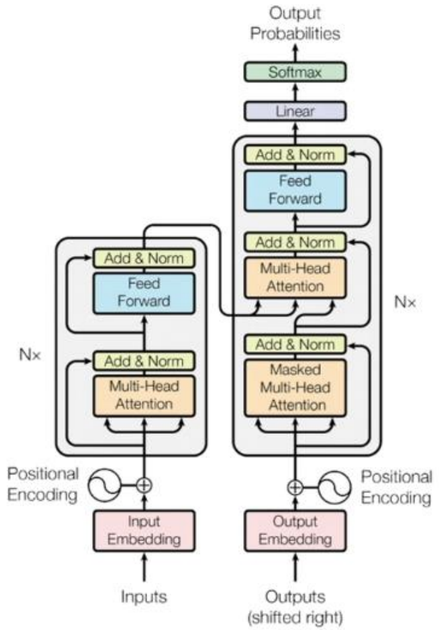
EWHA EURON

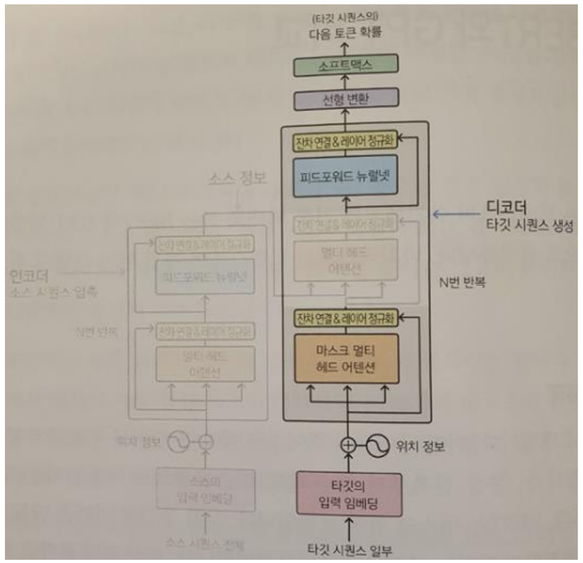
# #01 Large LM and GPT-3



EWHA EURON

# GPT(generative pre-training)




GPT
; 입력된 키워드에 대해 그 키워드와 관련된 모든 자료를 취합해서 사람이 요구하는 텍스트를 만들어 냄

;Transformer 블록을 기본 아키텍처로 사용

;파라미터수를 늘리는 방식으로 발전

나는 오늘 학교에 갔다

나는 → 오늘

나는 오늘 → 학교에 (0.3)

나는 오늘 학교에 → 갔다 (0.7)

미래의 토큰으로부터 문맥 정보를 습득하지 못하도록 가중치를 0 으로 만들어 주는 기법을 Masked Self-Attention 이라고 함

// "갔다" 에 대해서는 가중치 0

<GPT를 이용한 언어 모델 학습 방법>

;언어 모델이므로 입력 문장을 단방향으로만 분석

;앞에 나온 단어를 이용해서 다음 단어를 맞춰 나가는 방식으로 사전학습을 진행

# GPT-1

**Transformer.Decoder → GPT1**

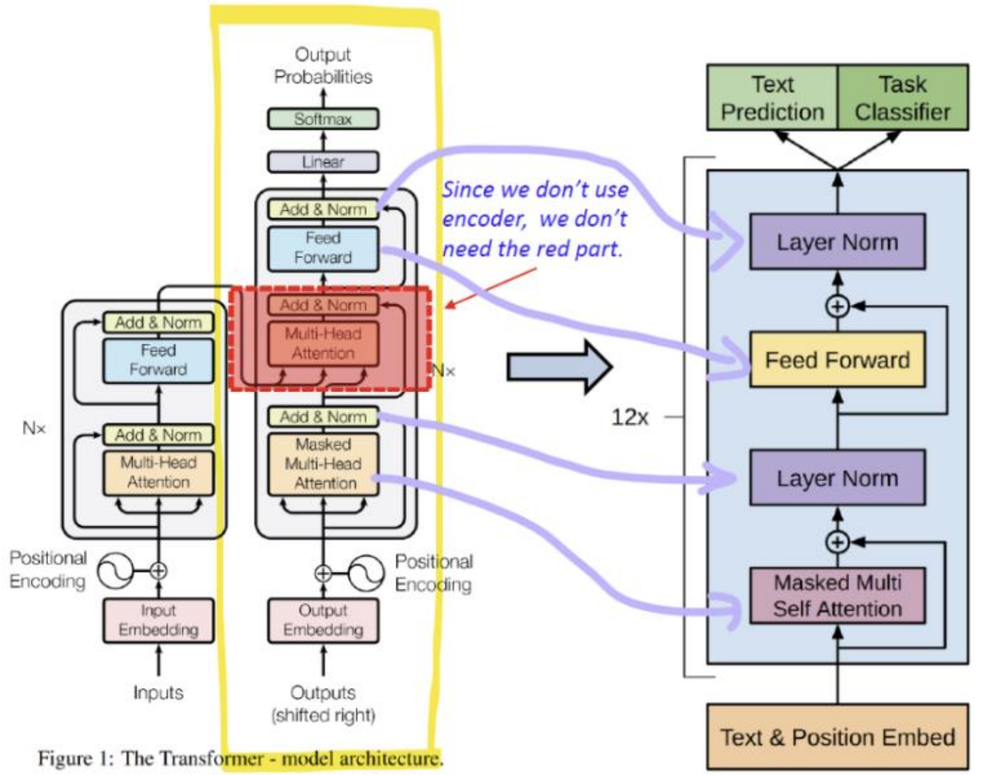
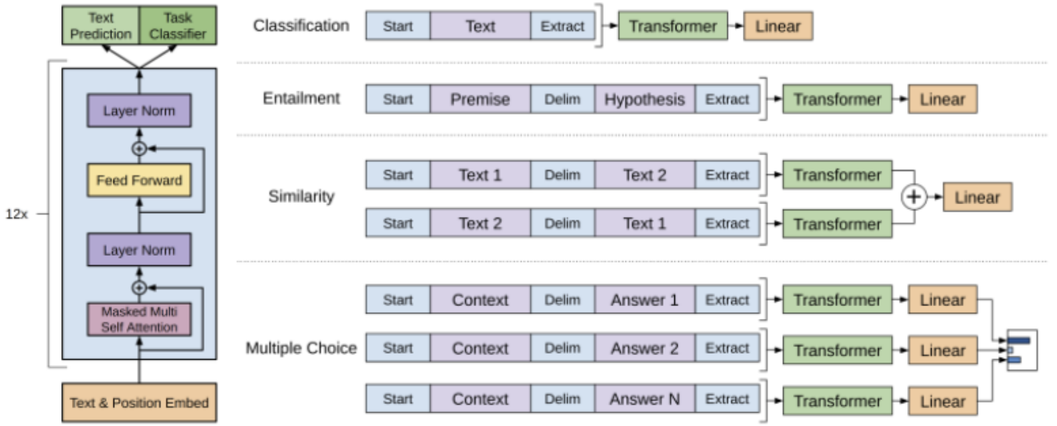



Figure 1: The Transformer - model architecture.




;transformer의 디코더 부분을 기조로 하므로 기존 transformer 인코더의 아웃풋을 받는 encoder-decoder self-attention부분이 필요 없게 됨.

;lase hidden state가 결국 중요하게 되며 start와 end token symbol도 필요 없게 됨

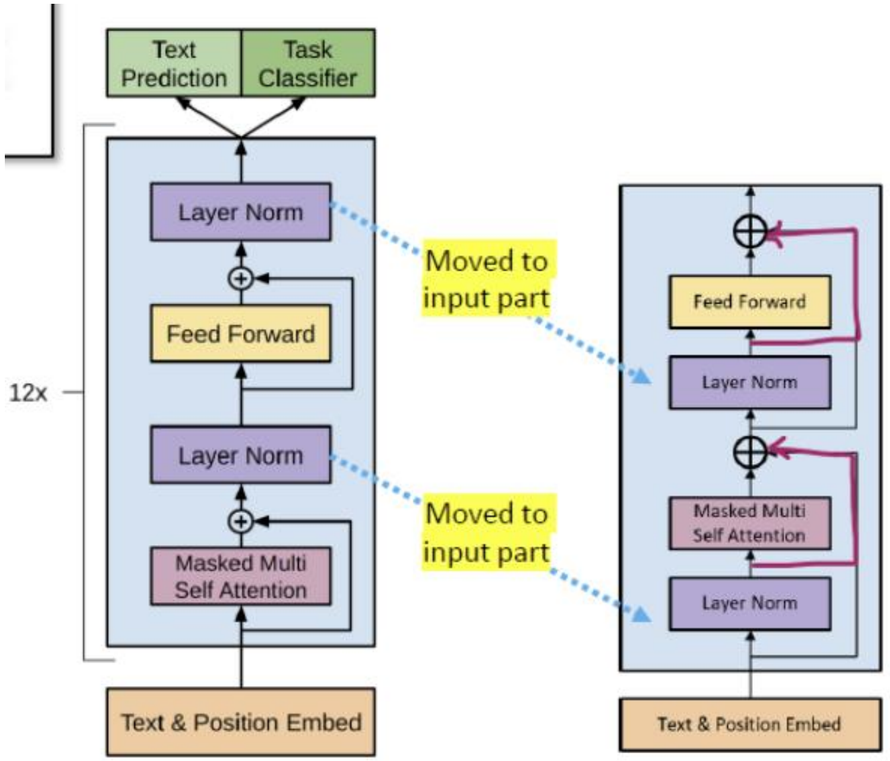
;아직 fine tuning 단계가 존재

→ 문장 마지막에 생성되는 extract 토큰을 이용해 각 task에 맞는 적절한 head와 labeling된 데이터로 supervised learning을 추가적으로 진행

→ 시간 & 돈이 많이 들고, 특정 task 에 fine tuning된 모델은 다른 task에는 사용이 불가능하는 단점을 가짐

# GPT-2

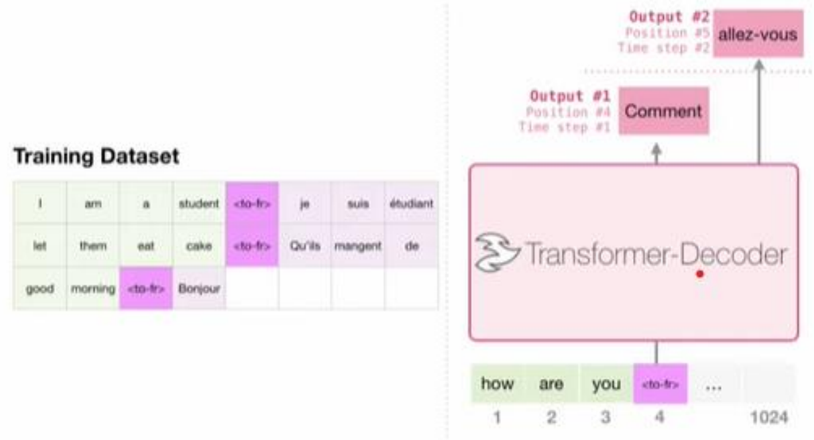
-구조

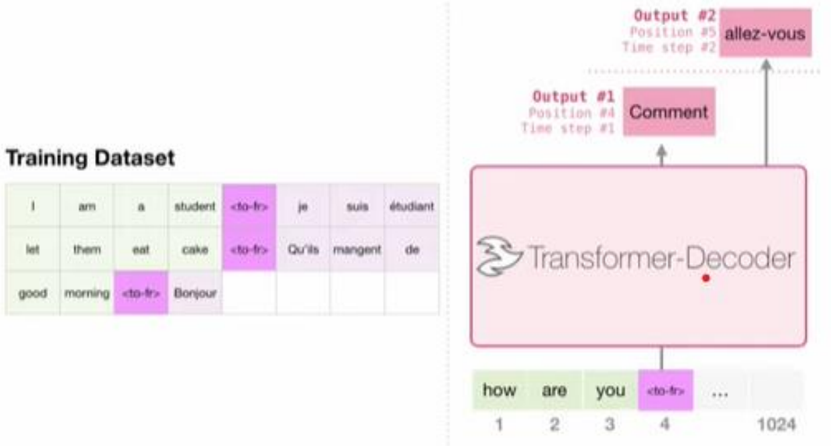



;transformer의 내부를 약간 변화시킨 모델
→ normalization layer의 위치 변화, 마지막 attention 블록 이후 normalization layer을 추가, 초기화 매소드가 수정됨

## -Unsupervised multi-task learner

;적절한 task description 을 입력으로 주는 것 만으로도 fine tuning 과정 없이 여러 task에 적용이 가능



# GPT-3 - 등장배경


그래프 해석

***No Prompt*** : 아무 설명 없이 입력 문제 넣음 ex) input : 1 + 2

***Natrual Language Prompt*** : 문제 처리에 대한 설명을 추가 ex) input : "add the two numbers." 1 + 2

=> 같은 상황에서는 Natural Language Prompt의 성능이 더 잘 나온다.

=> 하지만 모델의 parameter수를 키우고, shot의 개수가 많아질수록 둘 간의 성능 차이는 거의 나지 않는다.

pretrained 모델의 등장으로 downstream task를  적용할 때 task-specific모델 구조를 추가해줘야 하는 번거로움을 해소시켰지만,
Q. dataset 이나 fine-tuning방식은 task-specific해야 하는 한계가 존재

→ 1)매번 새로운 task 에 대해 거대한  labeled dataset을 준비해야 하므로 모델의 사용이 제한됨
2) 모델이 크기가 커짐에 따라 training data가 충분하지 못하면 잘못된 상관관계를 학습할 수 있음.

A. Meta-learning을 도입하여 미리 다양한 분야의 스킬들을 한 번씩 보여줘 학습함으로써 모델이 다양한 분야에 대해 강인하게 만듦.

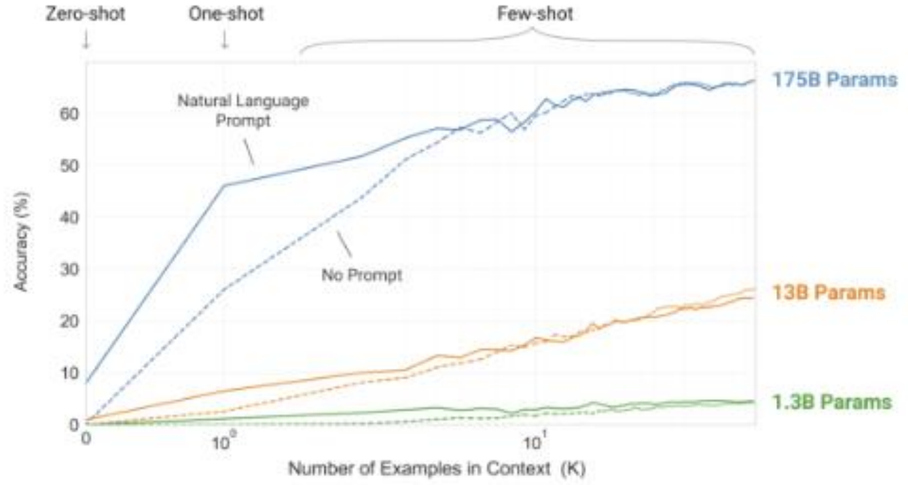


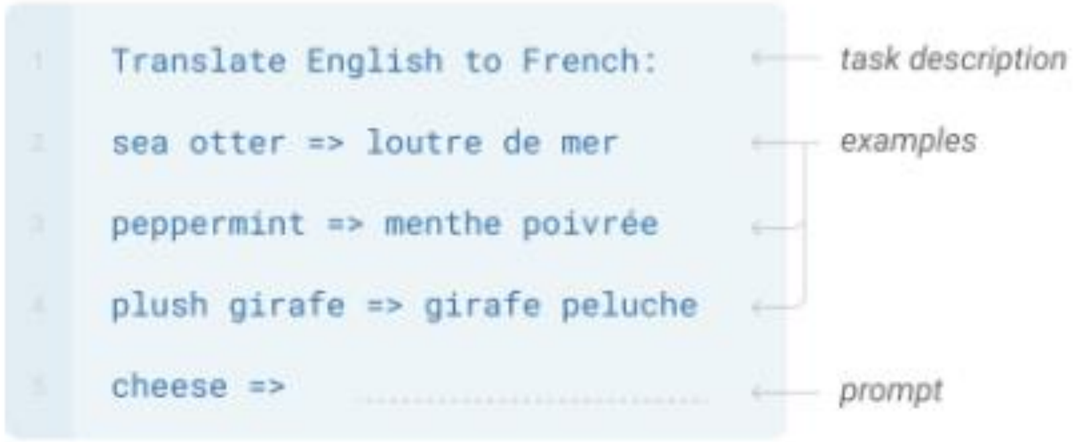

**Figure 1.2: Larger models make increasingly efficient use of in-context information.** We show in-context learning performance on a simple task requiring the model to remove random symbols from a word, both with and without a natural language task description (see Sec. 3.9.2). The steeper "in-context learning curves" for large models demonstrate improved ability to learn a task from contextual information. We see qualitatively similar behavior across a wide range of tasks.

# GPT-3 – Few shot

< few-shot >

**Few-shot**

In addition to the task description, the model sees a few examples of the task. No gradient updates are performed.



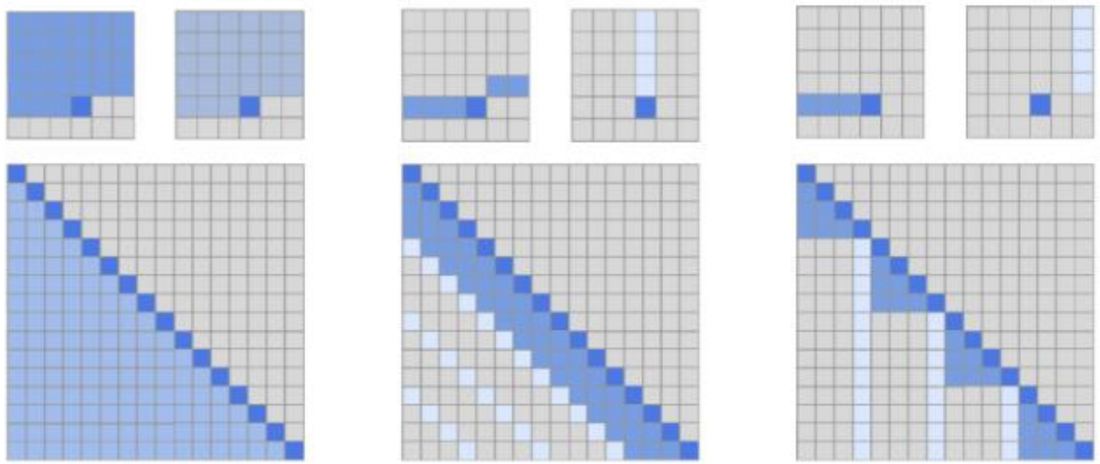

Few shot

: 모델이 inference 단계에서 추가적인 가중치 업데이트 없이 예시 몇 개를 전달하는 방식

ex)
BASE 모델에
' cheese => ' 를 전달하는 것이 아니라
' **Translate English to French :** cheese => ' 를 전달

힌트를 제공받은 모델은
힌트 뒤에 오는 문제를 더 적절하게 풀 수 있게 된다.

# GPT-3 – sparse attention pattern



(a) Transformer (b) Sparse Transformer (strided) (c) Sparse Transformer (fixed)

*Figure 3.* Two 2d factorized attention schemes we evaluated in comparison to the full attention of a standard Transformer (a). The top row indicates, for an example 6x6 image, which positions two attention heads receive as input when computing a given output. The bottom row shows the connectivity matrix (not to scale) between all such outputs (rows) and inputs (columns). Sparsity in the connectivity matrix can lead to significantly faster computation. In (b) and (c), full connectivity between elements is preserved when the two heads are computed sequentially. We tested whether such factorizations could match in performance the rich connectivity patterns of Figure 2.

GPT-2의 기존 attention 활용 방법 → 현재 처리하고 있는 token에 대해 이전에 처리한 모든 token간의 attention 값을 구함
GPT-3 → input sequence의 길이를 늘림과 동시에 디코더층이 늘어나면서 생긴 계산량 증가에 대한 부담을 덜기 위해 새로운 attention 방식 고안

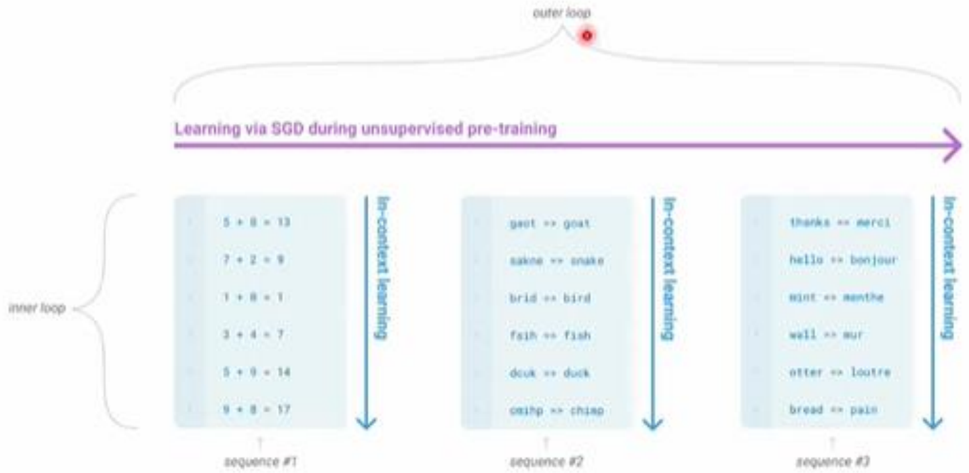
**sparse transformer(strided)**
;계산량을 줄이기 위해 masked attention의 attention이 걸리는 범위를 제한
→ 2가지 head의 attention 연산을 조합하여 사용

현재 처리하려는 Token이 하나 존재할 때,
< ***#1 head 1*** > : 현재 참조하려는 Token의 바로 윗자리 Token 다음까지 참조한다.
< ***#2 head 2*** > : 현재 참조하려는 Token의 위에 해당하는 모든 Token을 참조한다.

**sparse transformer(fixeed)**
→2개의 head에 대한 attention 연산을 결합

현재 처리하려는 Token에 대해,
< ***#1 Head*** > : 같은 행의 앞선 Token들을 모두 참조한다.
< ***#1 Head*** > : 위 행들의 가장 마지막 열에 대한 Token을 모두 참조한다.

# GPT-3 – Meta-learning




| | data |
|---|---|
| 1 | 5+8=13, 7+2=9, 1+0=1, 3+4=7, 5+9=14 |
| 2 | Gaot=>goat, sakne=>snake, brid=>bird, fsih=>fish |
| 3 | Thanks=>merci, hello=>bonjour, mint=>menthe |

Meta-learning
: 사람이 통제하던 기계학습 과정을 자동화함으로써 기계 스스로 학습 규칙을 익힐 수 있게 하는 방법론

;GPT-3의 pretraining과정에 존재하는 두개의 loop중 outer loop는 일반적인 사전학습을 만드는 과정 inner loop 는 한 sequence 내에서 패턴을 학습하는 과정

Cf)하나의 context내에서 학습을 진행한다고 하여 inner loop를 in-context learning(task 가 명시 되어있지 않은 상태에서 다양한 패턴을 인식하는 능력을 학습하게 됨)이라고도 함

# GPT-3 – Results

## Lambada

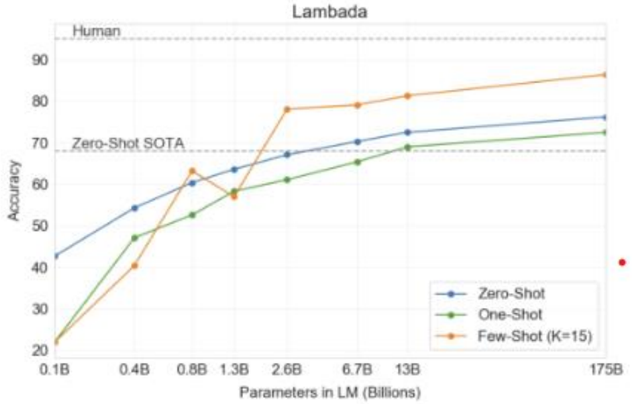



**Figure 3.2:** On LAMBADA, the few-shot capability of language models results in a strong boost to accuracy. GPT-3 2.7B outperforms the SOTA 17B parameter Turing-NLG [Tur20] in this setting, and GPT-3 175B advances the state of the art by 18%. Note zero-shot uses a different format from one-shot and few-shot as described in the text.

## TriviaQA

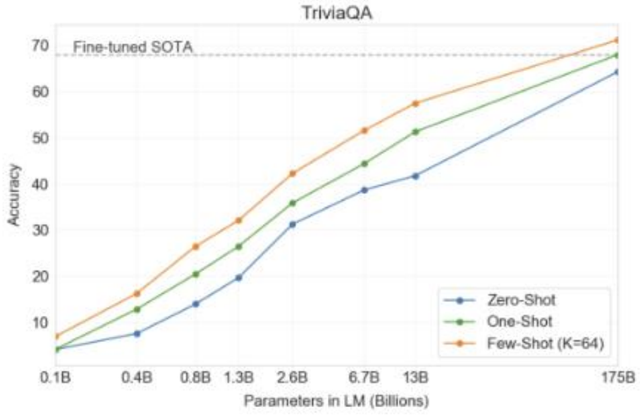



**Figure 3.3:** On TriviaQA GPT3's performance grows smoothly with model size, suggesting that language models continue to absorb knowledge as their capacity increases. One-shot and few-shot performance make significant gains over zero-shot behavior, matching and exceeding the performance of the SOTA fine-tuned open-domain model, RAG [LPP+20]

## Translation

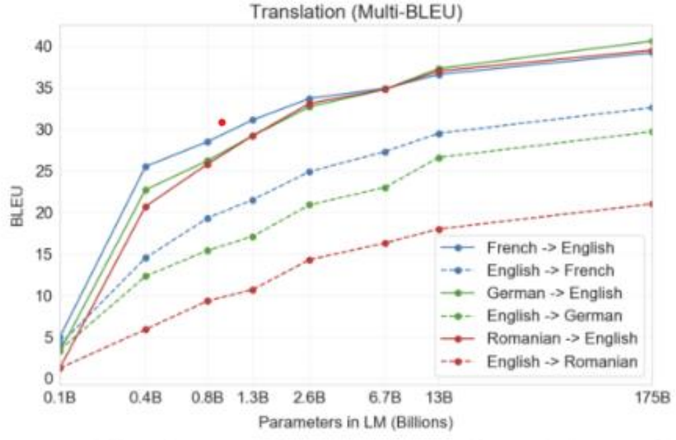



**Figure 3.4:** Few-shot translation performance on 6 language pairs as model capacity increases. There is a consistent trend of improvement across all datasets as the model scales, and as well as tendency for translation into English to be stronger than translation from English.

## News Article Generation

| | Mean accuracy | 95% Confidence Interval (low, hi) | $t$ compared to control ($p$-value) | "I don't know" assignments |
|---|---|---|---|---|
| Control (deliberately bad model) | 86% | 83%–90% | - | 3.6 % |
| GPT-3 Small | 76% | 72%–80% | 3.9 (2e-4) | 4.9% |
| GPT-3 Medium | 61% | 58%–65% | 10.3 (7e-21) | 6.0% |
| GPT-3 Large | 68% | 64%–72% | 7.3 (3e-11) | 8.7% |
| GPT-3 XL | 62% | 59%–65% | 10.7 (1e-19) | 7.5% |
| GPT-3 2.7B | 62% | 58%–65% | 10.4 (5e-19) | 7.1% |
| GPT-3 6.7B | 60% | 56%–63% | 11.2 (3e-21) | 6.2% |
| GPT-3 13B | 55% | 52%–58% | 15.3 (1e-32) | 7.1% |
| GPT-3 175B | 52% | 49%–54% | 16.9 (1e-34) | 7.8% |

**Table 3.11: Human accuracy in identifying whether short (~200 word) news articles are model generated.** We find that human accuracy (measured by the ratio of correct assignments to non-neutral assignments) ranges from 86% on the control model to 52% on GPT-3 175B. This table compares mean accuracy between five different models, and shows the results of a two-sample T-Test for the difference in mean accuracy between each model and the control model (an unconditional GPT-3 Small model with increased output randomness).

# GPT-3 - 한계

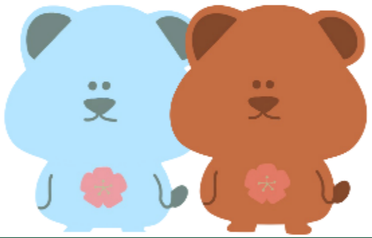
**•문장 생성 능력과 특정 NLP task에 대해 약점을 가지고 있다.**

특정한 표현을 계속해서 반복하거나, 일관성을 잃는다. 추가적으로 앞 뒤 간의 모순(contradict)발생하거나 자연스럽지 못한 문장이 생성되기도 한다.

**•구조적, 알고리즘 관점에서 문제점**

**- Auto-regressive**(Transformer의 Decoder) : not bidirectional(사람은 문제를 풀 때, 앞 뒤 문맥을 보는데 Decoder는 불가능)
- pre-training 당시 모든 토큰의 가중치를 동등하게 부여해서, 단어들 간의 중요도 차이를 제대로 계산하지 못할 수 있다.
-> 중요한 단어들에 대해 loss 가중치를 더 부여함으로써 pre-trained model의 quality 향상하는 방법이 기대된다.

# #02 Compositional Representations and Systematic Generalization



EWHA EURON

# #02 Compositional Representations and Systematic Generalization

## #01 용어

- **Systematicity (체계성)**
  - 사람이 이해하는 문장들 간엔 **확실하고 예측 가능한 패턴**이 있다.
  - E.g. 철수는 영희를 좋아한다. → 영희는 철수를 좋아한다.

Stefan Frank

Imagine you meet someone who only knows two sentences of English:

*Could you please tell me where the toilet is?*
*I can't find my hotel.*

So (s)he does not know:

*Could you please tell me where* ***my hotel*** *is?*
*I can't find* ***the toilet****.*

This person has no knowledge of English but simply memorized some lines from a phrase book.


- Human language behavior is (more or less) **systematic**: if you know some sentences, you know many.
- Sentences are not atomic but made up of words.
- Likewise, words can be made up of morphemes. (e.g., *un + clear = unclear, un + stable = unstable, …*)
- It **seems like** language results from applying a set of rules (grammar, morphology) to symbols (words, morphemes).

# #02 Compositional Representations and Systematic Generalization

#01 용어

- Compositionality (구성성)
  한 표현의 의미는 그 표현을 구성하는 요소들의  의미와 구조로 구성된다

산 넘어 마을 = 넘다(산, 도착지)

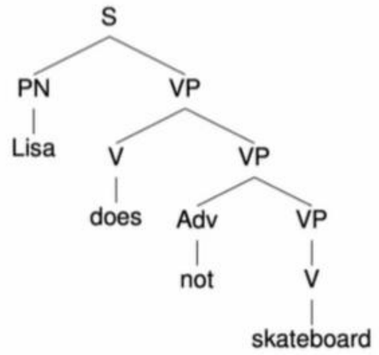
산 넘어 산 ≠ 넘다(산, 도착지)

# #02 Compositional Representations and Systematic Generalization

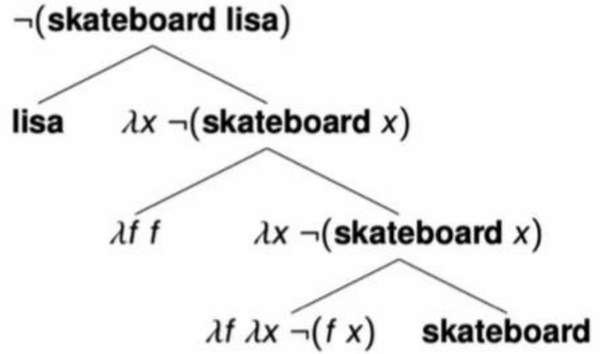
## #02 Are neural representations compositional?

- **Compositionality of representations**

Lisa does not skateboard =
⟨Lisa, ⟨does, ⟨not, skateboard⟩⟩⟩




m(Lisa does not skateboard) =
⟨m(Lisa), ⟨m(does), ⟨m(not), m(skateboard)⟩⟩⟩




Measuring Compositionality in Representation Learning (Jacob Andreas, ICLR 2019)

# #02 Compositional Representations and Systematic Generalization

## #02 Are neural representations compositional?

**Tree Reconstruction Error (TRE)**

First choose :

- a distance function $\delta : \Theta \times \Theta \to [0, \infty)$ satisfying $\delta(\theta, \theta') = 0 \Leftrightarrow \theta = \theta'$
- a composition function $* : \Theta \times \Theta \to \Theta$

Define $\hat{f}_\eta(d)$, a *compositional approximation to* $f$ with parameters $\eta$, as:

$$\hat{f}_\eta(d_i) = \eta_i \qquad \text{for } d_i \in \mathcal{D}_0$$
$$\hat{f}_\eta(\langle d, d' \rangle) = \hat{f}_\eta(d) * \hat{f}_\eta(d') \qquad \text{for all other } d$$

$\hat{f}_\eta$ has one parameter vector $\eta_i$ for every $d_i$ in $\mathcal{D}_0$; these vectors are members of the representation space $\Theta$.

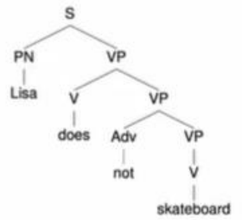
Given a dataset $\mathcal{X}$ of inputs $x_i$ with derivations $d_i = D(x_i)$, compute:

$$\eta^* = \arg\min_\eta \sum_i \delta\big(f(x_i), \hat{f}_\eta(d_i)\big) \tag{2}$$

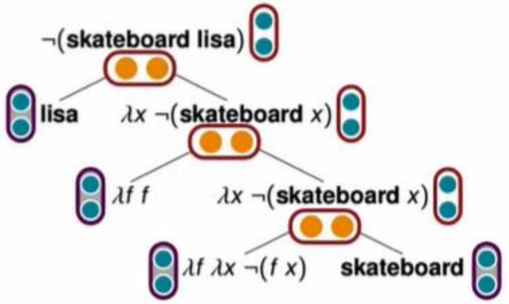
Then we can define datum- and dataset-level evaluation metrics:

$$\mathrm{TRE}(x) = \delta\big(f(x), \hat{f}_{\eta^*}(d)\big) \tag{3}$$
$$\mathrm{TRE}(\mathcal{X}) = \frac{1}{n} \sum_i \mathrm{TRE}(x_i) \tag{4}$$

Lisa does not skateboard =
⟨Lisa, ⟨does, ⟨not, skateboard⟩⟩⟩




NN(Lisa does not skateboard) ≈
f(v(Lisa), f(v(does), f(v(not), v(skateboard))))




Measuring Compositionality in Representation Learning (Jacob Andreas, ICLR 2019)

# #02 Compositional Representations and Systematic Generalization

## #03 Do neural networks generalize systematically?

- **Compositional Generalization**
  - The capacity to understand and produce a potentially infinite number of novel combinations of known components.
    - 훼렉하다 → 밥을 훼렉하다, 훼렉하고 산책하다
  - E.g. 모델이 알고 있는 단어 = [나, 사과, 먹다, 아침]
    - 나 아침에 사과 먹었어, 아침에 나 사과 먹었어, 사과 먹었어 나 아침에, …
- **Questions**
  1. Do neural networks (including large transformers) generalize systematically on challenging benchmarks involving realistic language?
  2. Can we create a dataset split that explicitly tests for this kind of generalization?

# #02 Compositional Representations and Systematic Generalization

## #03 Do neural networks generalize systematically?

- **Can we create a dataset split that explicitly tests for compositional generalization?**
  - **Ideal Compositionality Experiment**
    1. Similar atom distribution: All atoms present in the test set are also present in the train set, and the distribution of atoms in the train set is as similar as possible to their distribution in the test set.
    2. Different compound distribution: The distribution of compounds in the train set is as different as possible from the distribution in the test set.
  - Split data into train / test such that **compound divergence is maximized and atom divergence is minimized!**

| **Train set** | **Test set** |
| --- | --- |
| Who directed Inception? | Did Greta Gerwig direct Goldfinger? |
| Did Greta Gerwig produce Goldfinger? | Who produced Inception? |
| ... | ... |

# #02 Compositional Representations and Systematic Generalization

## #03 Do neural networks generalize systematically?

Let $\mathcal{F}_A(\text{ data }) \equiv$ normalized frequency distribution of atoms
Let $\mathcal{F}_C(\text{ data }) \equiv$ normalized frequency distribution of compounds
Define atom and compound divergence as:

$$\mathcal{D}_A(\text{ train } || \text{ test }) = 1 - C_{0.5}(\mathcal{F}_A(\text{ train }) || \mathcal{F}_A(\text{ test }))$$ **Minimize!**

$$\mathcal{D}_C(\text{ train } || \text{ test }) = 1 - C_{0.1}(\mathcal{F}_C(\text{ train }) || \mathcal{F}_C(\text{ test }))$$ **Maximize!**

where,

$$C_\alpha(P \| Q) = \sum_k p_k^\alpha q_k^{1-\alpha}$$

is the chernoff coefficient between two categorical distributions that measures similarity.

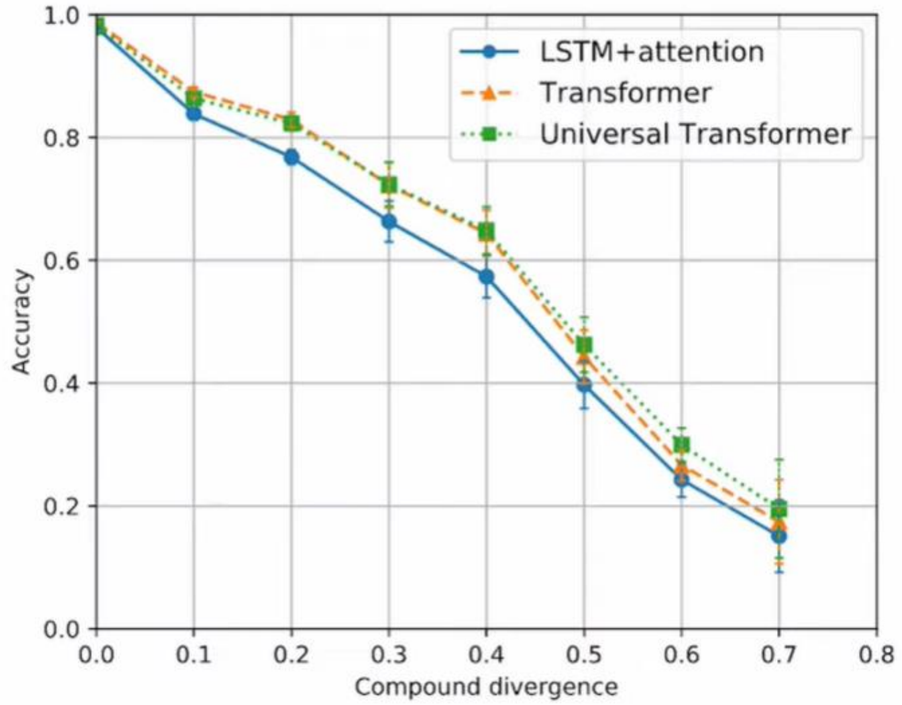
- The compound distributions of the train and test sets are very similar, then their compound divergence would be close to 0.
  → Not difficult tests for compositional generalization
- The compound divergence close to 1 means that the train-test sets have many different compounds.
  → Good test for compositional generalization

Measuring Compositional Generalization: A Comprehensive Method on Realistic Data (Keysers et al, ICLR 2020)

# #02 Compositional Representations and Systematic Generalization

#03 Do neural networks generalize systematically?

- **Do neural networks (including large transformers) generalize systematically on challenging benchmarks involving realistic language?**



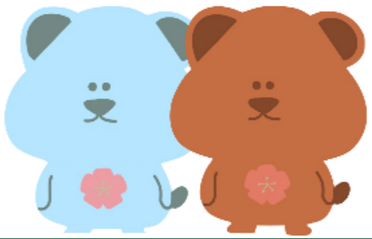
# #02 Compositional Representations and Systematic Generalization

#03 Do neural networks generalize systematically?

- **Do neural networks (including large transformers) generalize systematically on challenging benchmarks involving realistic language?**
  - Pre-training helps for compositional generalization, but doesn't solve it.

| *Model* | *CFQ (Maximum Compound divergence)* |
|---|---|
| T5-small (no pretraining) | 21.4 |
| T5-small | 28.0 |
| T5-base | 31.2 |
| T5-large | 34.8 |
| T5-3B | 40.2 |
| T5-11B | 40.9 |
| T5-11B-mod | 42.1 |

# #03 Improving how we evaluate models in NLP



EWHA EURON

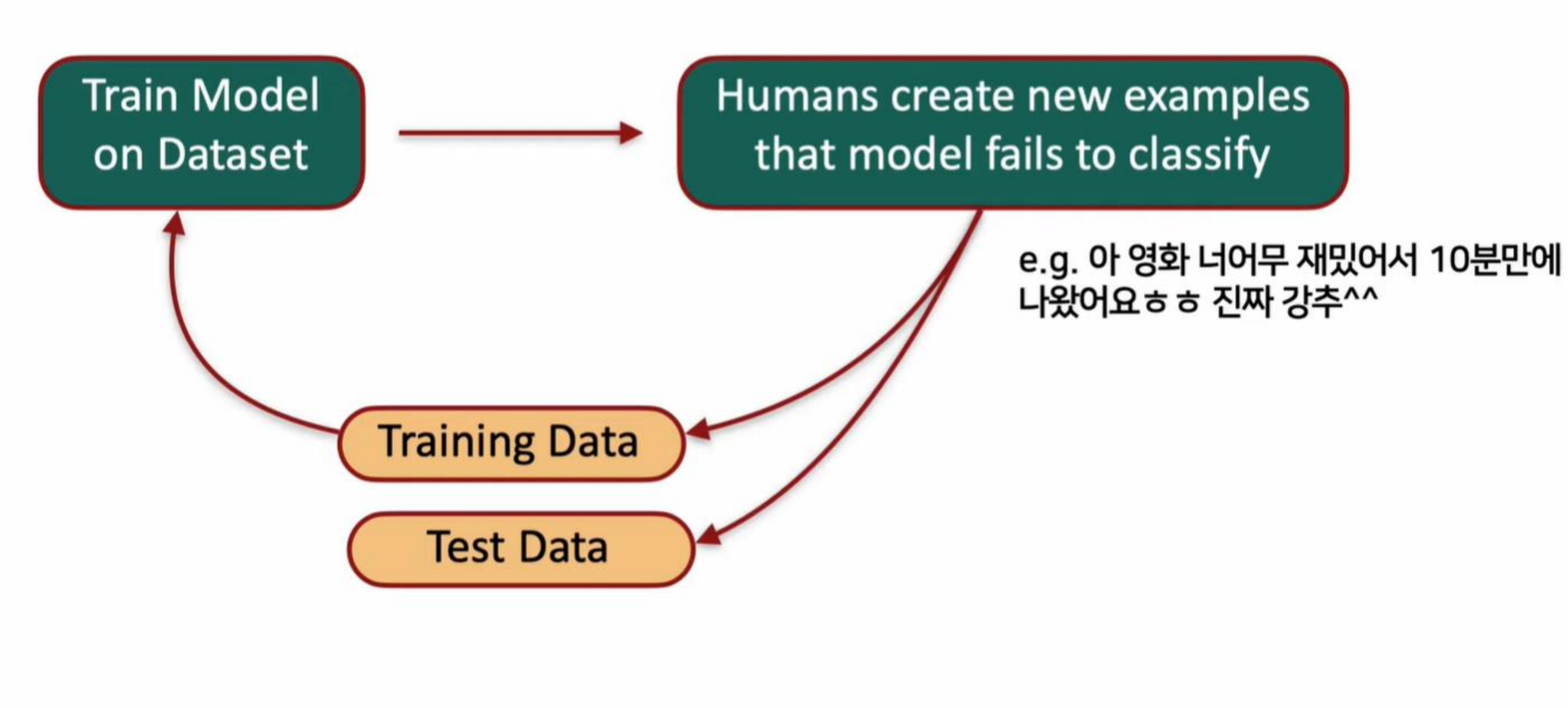
# #03 Improving how we evaluate models in NLP

- 벤치마크 데이터셋에서의 모델 성능은 날로 증가하는데 정말 실제 세계에서의 모델 성능도 그만큼 증가했을까?

- Task에 대한 모델의 이해도를 어떻게 하면 정확하게 측정할 수 있는가?

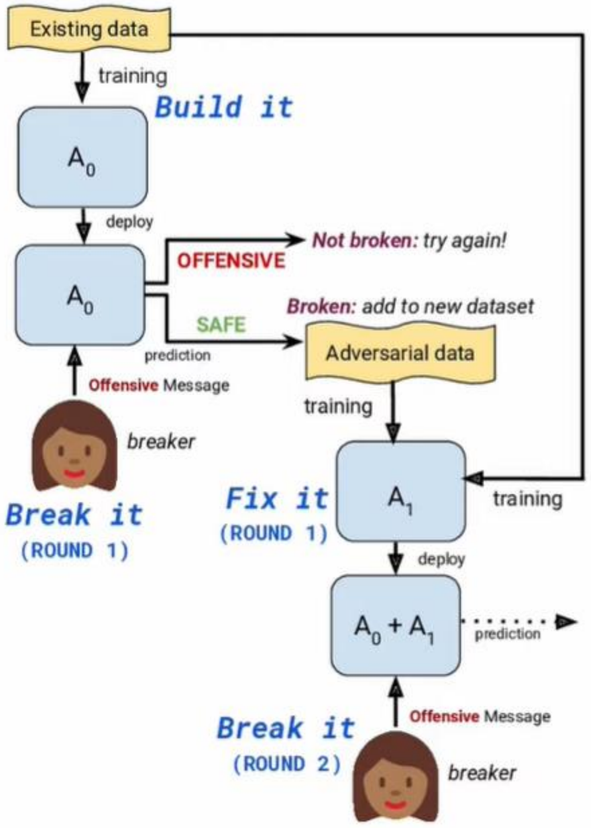
# #03 Improving how we evaluate models in NLP

#01 Dynamic Benchmarks



# #03 Improving how we evaluate models in NLP

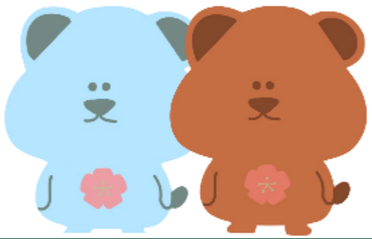
## #01 Dynamic Benchmarks




1. **Build it** : 사용자의 공격적인 메세지를 감지할 수 있는 모델 개발
2. **Break it** : Crowdworker에게 모델은 "SAFE"하다고 생각하지만 Crowdworker는 "OFFENSIVE"하다고 생각하는 메세지를 만들어서 ***"beat the system"*** 해달라고 요청
3. **Fix it** : 2번 과정을 통해 모여진 예제들을 통해 모델을 재학습 → 적대적인 공격에 더 강건한 모델이 될 수 있도록!
4. **Repeat** : Break it – Fix it 을 계속계속 반복

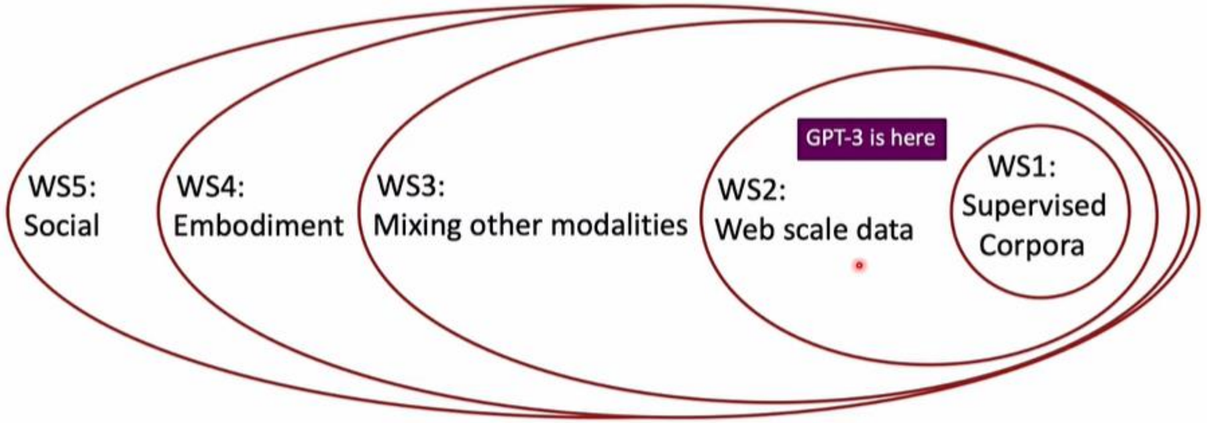
Build-It Break-It Fix-It for Dialogue Safety (Dinan et al, EMNLP 2017)

# #04 Grounding Language to other modalities



EWHA EURON

# Grounding Language to other modalities




Experience Grounds Language (Bisk et al, EMNLP 2020)

모델이 언어를 이해하는 scope

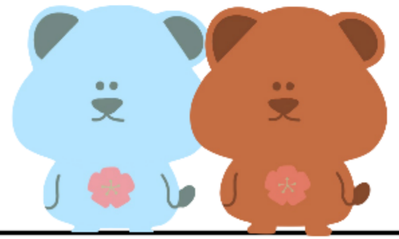
1) label이 존재하는 비교적 작은 규모의 데이터로 학습
2) Web scale data로 학습
3) 시각 청각 등의 양식을 이용해 언어를 이해
4) 촉감, 무게 등 물리적인 정보를 이용할 수 있음
5) 사회적으로 상호작용하며 언어를 이해

> Computer vision and speech recognition are mature enough for investigation of broader linguistic contexts (WS3). The robotics industry is rapidly developing commodity hardware and sophisticated software that both facilitate new research and expect to incorporate language technologies (WS4). Simulators and videogames provide potential environments for social language learners (WS5). Our call to action is to encourage the community to lean in to trends prioritizing grounding and agency, and explicitly aim to broaden the corresponding World Scopes available to our models.

Experience Grounds Language (Bisk et al, EMNLP 2020)

# THANK YOU

출처:[5-8] GPT-3 ( Language Mo.. : 네이버블로그 (naver.com)
GPT-1, BERT, GPT-2 모델 이해 : 네이버 블로그 (naver.com)
(24) [DSBA] CS224n 2021 Study | #18 Future of NLP + Deep Learning - YouTube



EWHA EURON