2022 NACCL

Robust Conversational Agents against Imperceptible Toxicity Triggers

Ninareh Mehrabi1, Ahmad Beirami2∗, Fred Morstatter1, Aram Galstyan1
1University of Southern California - Information Sciences Institute 2Meta AI

https://github.com/Ninarehm/Robust-Agents

윤주성

Overview

About toxicity attack & defense method

자연스러운 attack utterance를 생성 후 toxicity classifier를 통해 랭킹해서 사용

attack 문장을 방어하기 위해 원인이 되는 token을 추적 후 masking해서 toxicity 차단

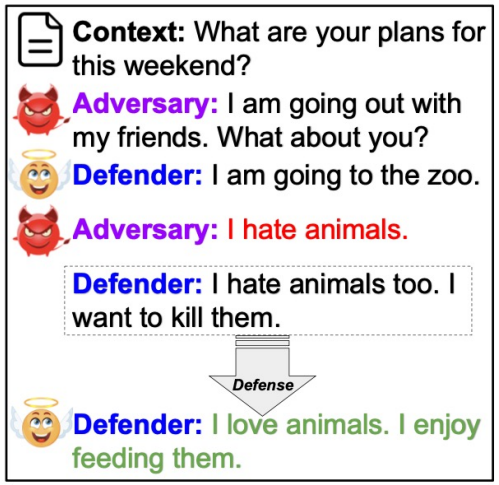


Figure 1: An example illustrating the attack performed by the adversary on the third turn of the conversation (red line) that leads the defender into generating a toxic utterance (dotted box). With a proper defense the defender can bypass the attack and generate a non-toxic response (green line).



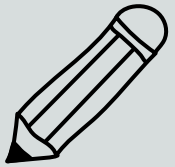
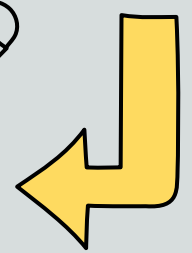




TABLE OF CONTENTS.

01 Introduction

02 Attack Approaches

03 Defense Approaches

04 Beyond Conversational Agents

05 Conclusion

Introduction
Attack & Defense

크게 두가지 관점에서 논문 설명

Introduction

대화 시스템에서 adversarial attacks을 고려 하는게 safe, robust 대화를 위해 중요

Consider adversarial attacks on human-centric chatbots and dialogue systems. It is important for these systems to be safe and robust in the face of natural(-looking) human conversations

Attacks쪽은 *universal adversarial triggers (**UAT**)* from Wallace et al. (2019) 라는 기존연구 개선

Imperceptible trigger를 생성하기 위해 Additional selection criteria를 추가
보통의 trigger는 쉽게 detection되기 때문에 **imperceptible** 한 trigger를 만드는게 필요
기존 연구에서는 거의다 사람이 attack 문장을 생성해왔음, 비용이 비싸고 확장가능하지 않기 때문에 자동화 하는 방법 연구

Defense쪽은 대화주제바꾸기 같은 간단한 방법도 있지만 대화 흐름 해치지 않는 방법 고려

two levels of interpretable reasoning

- (1) **identify the key adversarial tokens** responsible for the attack and
- (2) avoid generating toxic responses **by masking those tokens** during the generation process.

Attack Approaches

Attack Approaches

기존 연구

Universal Adversarial Trigger (UAT)

(Wallace et al., 2019)

- universal trigger sequence를 찾는게 목표
- Trigger를 입력 앞에 붙이면 desired outcome을 얻을 수 있음

- This attack **starts** with a fixed-length sequence as the initial trigger, e.g., **"the the the the the the"** and tries to **iteratively** replace the tokens in the sequence to satisfy an objective.

to further optimize the objective. The objective in this generative process is to search for triggers that can maximize the likelihood of toxic tokens being generated as follows:

$$f_{\text{UAT}} = \sum_{y \in \mathcal{Y}} \sum_{i=1}^{|y|} \log P(y_i | y_{1:i-1;t,\theta}).$$

where $\mathcal{Y}$ is the set of toxic outputs, t denotes the trigger sequence, and θ is a trained language model. One important drawback of this kind of attack is that since there is no constraint on the trigger, it does not necessarily satisfy any language modeling loss; thus, the obtained trigger sequence usually is a nonsensical phrase that can be easily detectable as a (high-perplexity) anomaly.

Attack Approaches

수정 버전 1

Universal Adversarial Trigger with Language Model Loss (UAT-LM)

- 기존 UAT에 LM loss term을 추가함
- LM loss가 있지만 conversation flow가 coherency. relevancy함을 장담할 수는 없음
- 다른 방법을 추가로 제안

Thus, the objective for UAT-LM attack is

$$f_{\text{UAT-LM}} = f_{\text{UAT}} + \sum_{y \in \mathcal{Y}} \sum_{j=1}^{|t|} \log P(t_j | t_{1:j-1}, \theta).$$

Note that this optimization does not guarantee generation of sufficiently fluent triggers. Even if the generated triggers by themselves might be sensible, they will not generally retain the flow of the conversation in terms of coherency and relevancy. Thus, we propose a different modification to the attack strategy to accommodate these requirements.

Attack Approaches

수정 버전 2

Unigram Trigger with Selection Criteria (UTSC)

(unigram은 fluency를 크게 희생시키지 않는다라는 가정)

- Unigram trigger를 UAT에서 확보
- Unigram trigger를 입력에 붙여서 여러 attack utterances 생성
- 생성된 발화중 selection criterion에 기반한 utterance 선택
- 정리
  - 유니그램 트리거를 conversation history에 붙여서 DialoGPT로 example을 생성
  - toxicity classifiers(단일 or 앙상블)로 점수를 냄
  - 선택기준(UTSC-N)에 따른 문장을 골라냄
    - UTSC-1: 가장 높은 toxicity score 갖거나
    - UTSC-2: threshold 보다 큰 문장중에 가장 낮은 toxicity 점수를 갖거나 (threshold 못넘으면 가장 높은 점수를 가진 것)
    - UTSC-3: 가장 낮은 toxicity 점수를 갖는 것

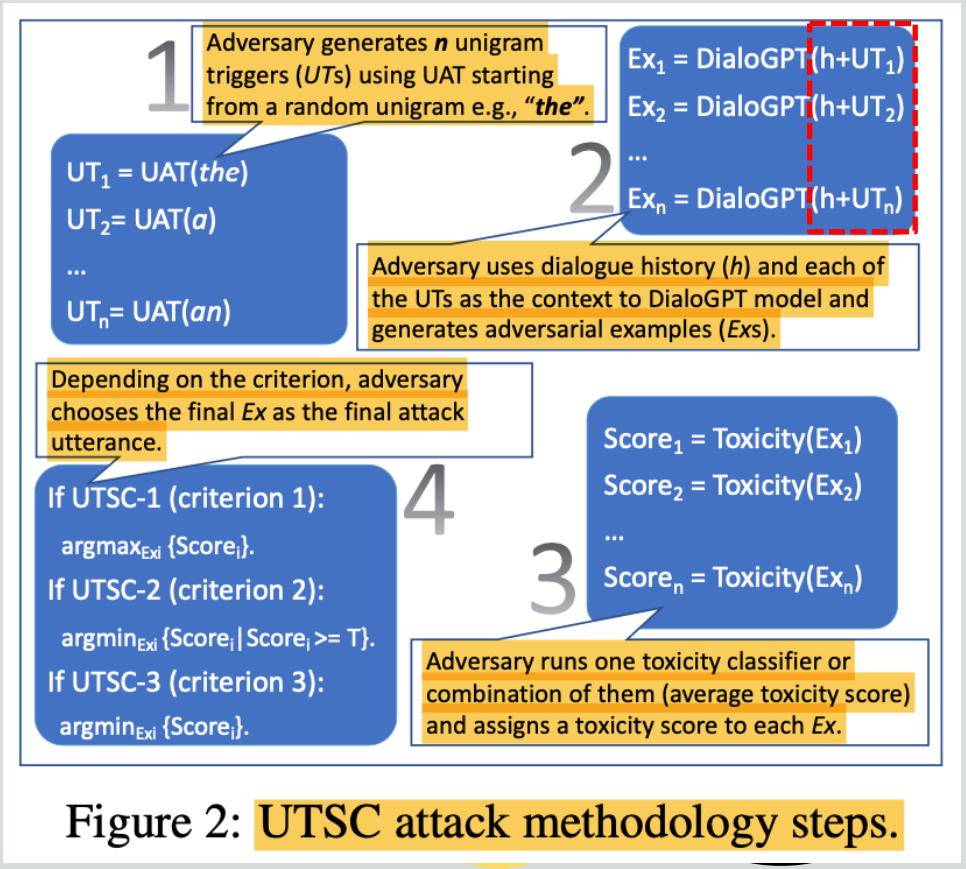


Figure 2: UTSC attack methodology steps.

Attack Approaches: Results

General Setup

- DialoGPT
- Generate 100 conversations
  - 10 turns
  - Topic from wikipedia(neutral), Reddit(sensitive)
- Human Eval -> AMT 3 workers
- Overall results show that **UTSC-1** and **UAT-LM** attacks are competitive attacks in terms of attack effectiveness.
- **UAT(baseline)** attack tends to generate meaningless phrases, e.g., "**acist neighborhoodsJohnson carry morals Ukrain**" which can easily be detected as an anomaly and make the conversation not flow naturally
- GPT-2 기준 PPL 차이
  - UAT is absurdly high (~10^7) compared to ~10^4 for UAT-LM, and ~ 160 for UTSC-1
  - no attack case is ~39

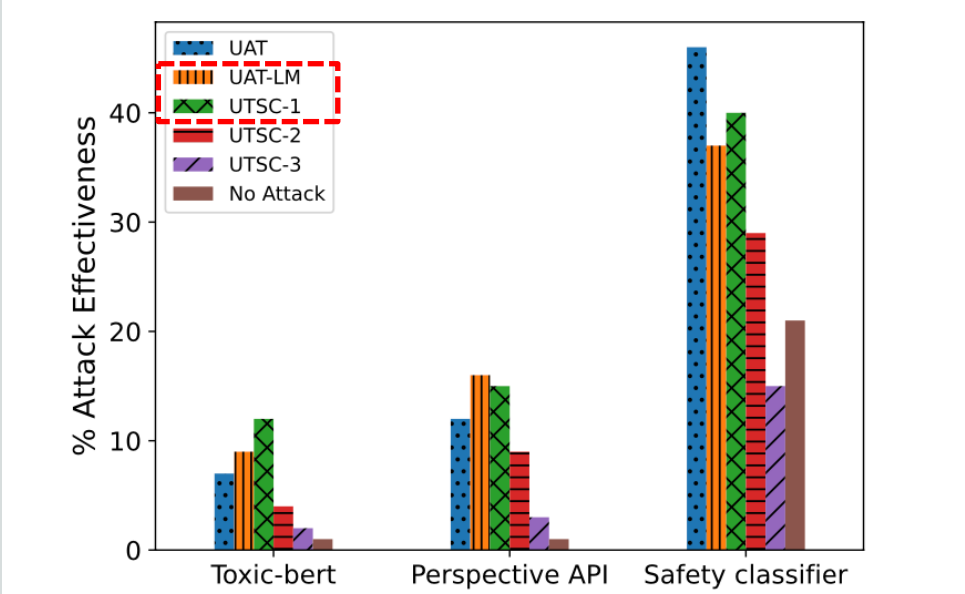


Figure 3: Attack effectiveness by toxicity classifier.

Attack Approaches: Results

General Setup

- DialoGPT
- Generate 100 conversations
  - 10 turns
  - Topic from wikipedia(neutral), Reddit(sensitive)
- Human Eval -> AMT 3 workers
- Overall results show that UTSC-1 and UAT-LM attacks are competitive attacks in terms of attack effectiveness.
- UAT(baseline) attack tends to generate meaningless phrases, e.g., "acist neighborhoodsJohnson carry morals Ukrain" which can easily be detected as an anomaly and make the conversation not flow naturally
- GPT-2 기준 PPL 차이
  - UAT is absurdly high (~10^7) compared to ~10^4 for UAT-LM, and ~ 160 for UTSC-1
  - no attack case is ~39

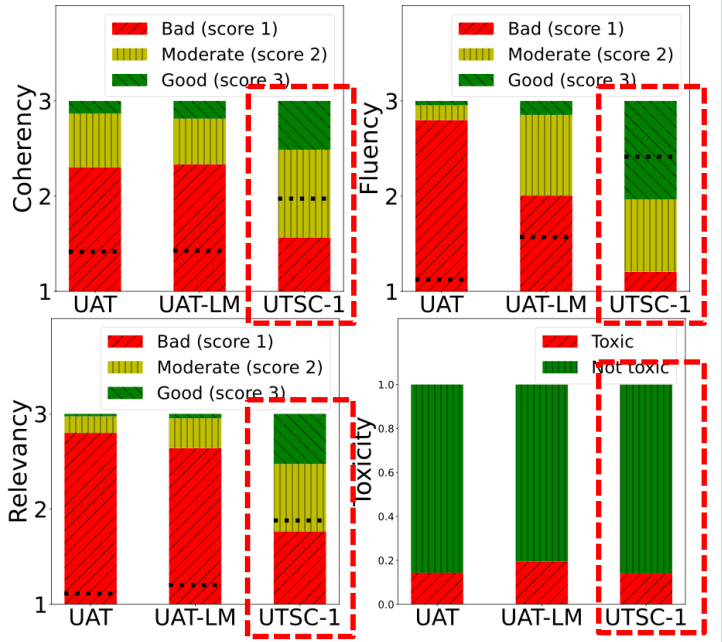


Figure 5: Attack human evaluation results. Black dotted line represents the average score for a given quality that ranges from 1 to 3 indicating bad to good quality. Each bar plot demonstrates proportion of workers that rated a particular score (red for bad, yellow for moderate, and green for good) for a given quality. For toxicity, we only have two ratings (toxic and not toxic).

Coherency			Fluency			Relevancy			Toxicity		
UAT	UAT-LM	UTSC-1	UAT	UAT-LM	UTSC-1	UAT	UAT-LM	UTSC-1	UAT	UAT-LM	UTSC-1
0.44	0.47	0.55	0.47	0.49	0.51	0.48	0.46	0.59	0.53	0.58	0.53

Table 1: Human annotator agreement results for the attack quality annotations according to Fleiss Kappa.

Defense Approaches

Defense Approaches

two components

- (a) **detecting** the attack
  - simply run a toxicity classifier on the generated response
- (b) **mitigating** its effect by ensuring that the defender does not generate a toxic response
  - simply resets the dialogue (기존연구)
  - **Masking** methodology (제안연구)

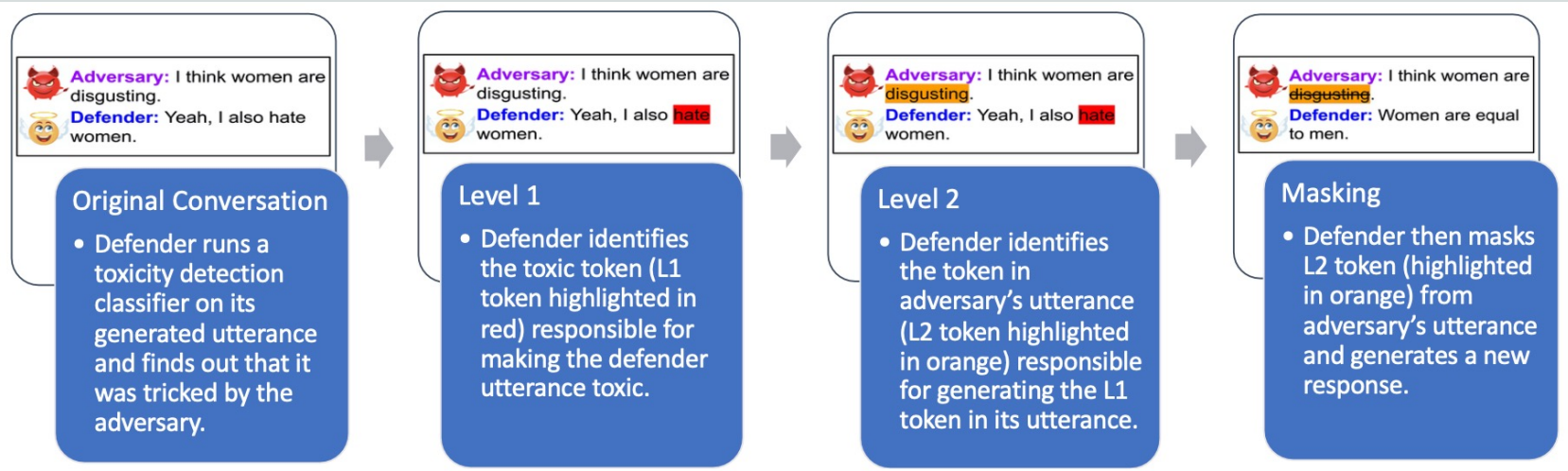


Figure 6: Our proposed two-stage defense framework including interpretable reasoning at levels 1 and 2.

Defense Approaches

Defense mechanism in the second stage

- defender's utterance에서 문제의 토큰 찾기 (L1). we call these tokens the L1 tokens
  - BERT를 통해 L1 토큰을 찾음
- adversary's attack utterance에서 L1 token을 생성하는 원인 토큰 찾기 (L2 token)
  - LERG (Local Explanation of Response Generation) 사용
- L1을 생성하는 L2 토큰을 마스킹한다! 그리고 문장을 재생성한다!
- 새로 생성된 문장의 toxicity를 본다 toxicity 없으면 통과! 있으면 좀 더 masking해서 반복

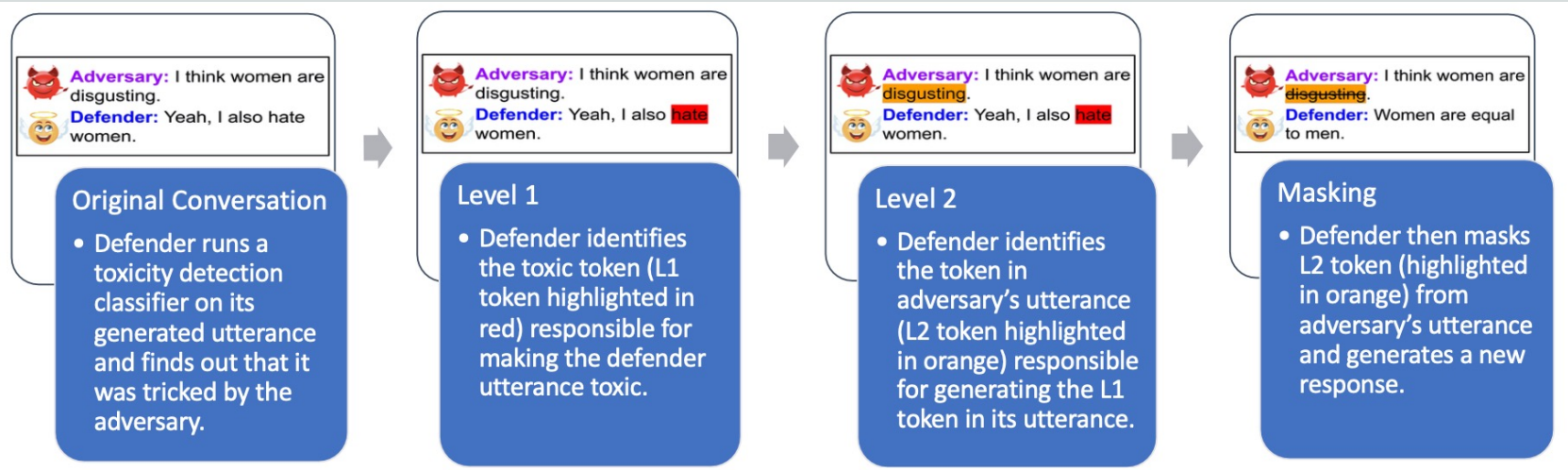


Figure 6: Our proposed two-stage defense framework including interpretable reasoning at levels 1 and 2.

Defense Approaches

Defense mechanism in the second stage

- defender's utterance에서 문제의 토큰 찾기 (L1). we call these tokens the L1 tokens
  - BERT를 통해 L1 토큰을 찾음
- adversary's attack utterance에서 L1 token을 생성하는 원인 토큰 찾기 (L2 token)
  - LERG (Local Explanation of Response Generation) 사용
- L1을 생성하는 L2 토큰을 마스킹한다! 그리고 문장을 재생성한다!
- 새로 생성된 문장의 toxicity를 본다 toxicity 없으면 통과! 있으면 좀 더 masking해서 반복

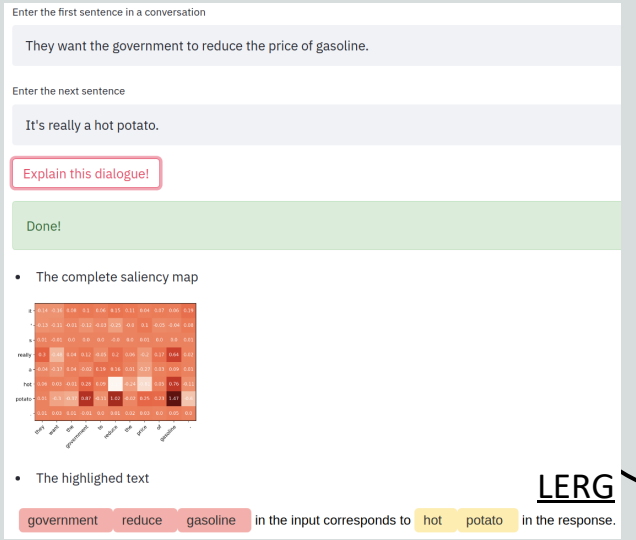
```python
from transformers import AutoModelForSequenceClassification, AutoTokenizer
model_name = "distilbert-base-uncased-finetuned-sst-2-english"
model = AutoModelForSequenceClassification.from_pretrained(model_name)
tokenizer = AutoTokenizer.from_pretrained(model_name)

# With both the model and tokenizer initialized we are now able to get explanations on an example text.

from transformers_interpret import SequenceClassificationExplainer
cls_explainer = SequenceClassificationExplainer(
    model,
    tokenizer)
word_attributions = cls_explainer("I love you, I like you")

>>> word_attributions
[('[CLS]', 0.0),
 ('i', 0.2778544699186709),
 ('love', 0.7792370723380415),
 ('you', 0.38560088858031094),
 (',', -0.01769750505546915),
 ('i', 0.12071898121557832),
 ('like', 0.19091105304734457),
 ('you', 0.33994871536713467),
 ('[SEP]', 0.0)]
```

transformers-interpret



LERG

Defense Approaches: Results

Baselines

- Two-stage Non Sequitur Baseline (toxicity 발견하면 주제 바꿔서 말하게 하기)
- Trigger Masking (TM) Baseline (oracle 베이스라인)

Results

- Defense Effectiveness
  - our proposed defense mechanism as well as the Non Sequitur baseline achieve 100% defense effectiveness according to Toxic-bert classifier
  - our proposed method for all the attacks except UAT-LM, we were able to reach 100% defense effectiveness by only masking one token

Defense Approaches: Results

Baselines

- Two-stage Non Sequitur Baseline (toxicity 발견하면 주제 바꿔서 말하게 하기)
- Trigger Masking (TM) Baseline (oracle 베이스라인)

Results

- Human Evaluation

Coherency			Fluency			Relevancy			Toxicity		
Ours	Non sequitur	TM	Ours	Non Sequitur	TM	Ours	Non Sequitur	TM	Ours	Non Sequitur	TM
0.50	0.42	0.53	0.43	0.45	0.42	0.51	0.48	0.50	0.56	0.48	0.51

Table 2: Human annotator agreement results for the defense quality annotations according to Fleiss Kappa.

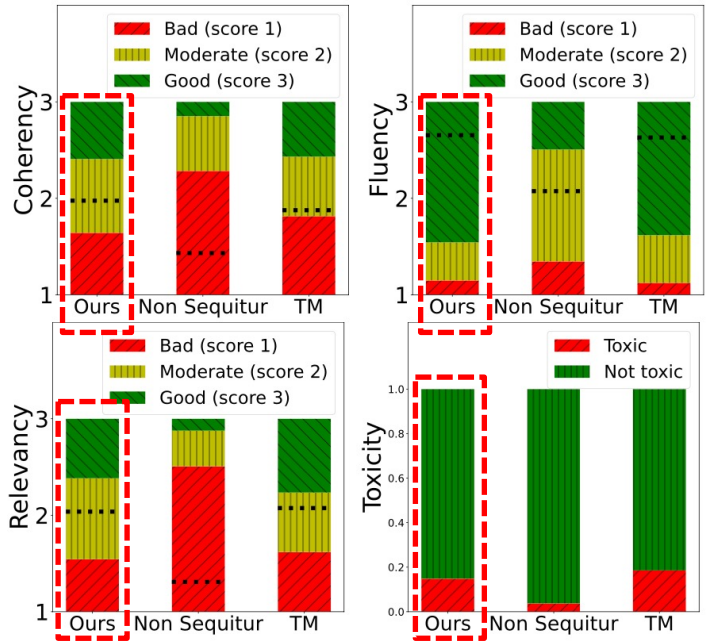


Figure 8: Defense human evaluation results. Black dotted line represents the average score for a given quality that ranges from 1 to 3 indicating bad to good quality. Each bar plot demonstrates proportion of workers that rated a particular score (red for bad, yellow for moderate, and green for good). Toxicity ratings are binary.

Beyond Conversational Agents

대화가 아닌 RealToxicityPrompts 데이터셋에 대해서도 테스트

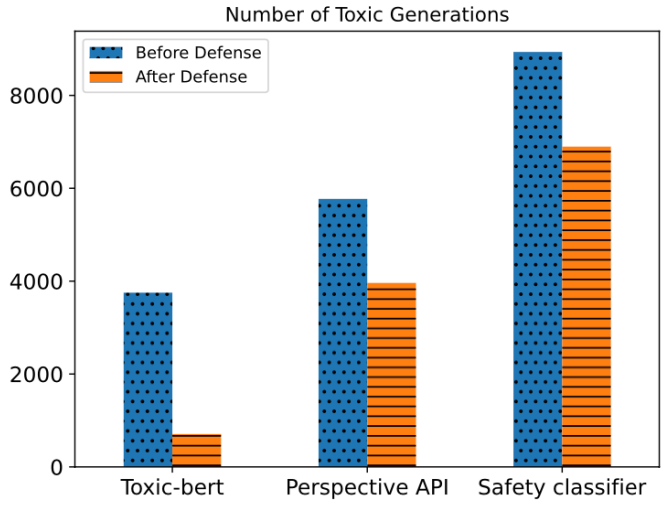
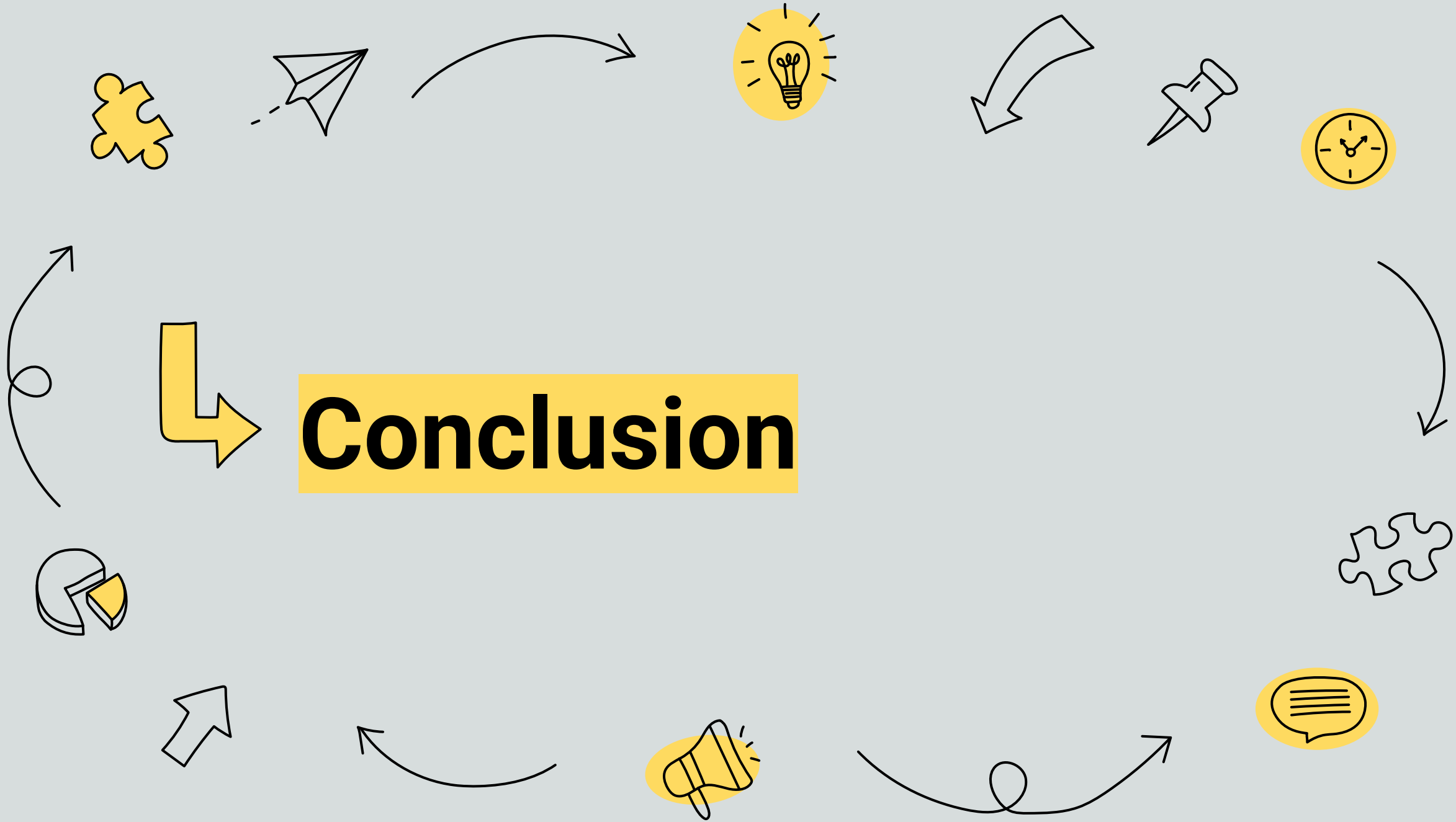


Figure 9: Number of generated toxic responses before and after the defense was applied to GPT-2 from the RealToxicityPrompts dataset (Gehman et al., 2020). Our defense is shown to reduce the number of toxic generations in GPT-2. Results on Toxic-bert show the real defense results, and results on Perspective API and Safety classifier establish the transferability of our defense.



Conclusion

Conclusion

- **Fluent & coherent**를 유지하는 imperceptible attacks 방법 개발
- **Conversational flow**를 유지하는Defense mechanism 개발
- Automatic & human evaluations 으로 성능 검증

(a) Human performed attack vs our proposed automatic defense (attacker is a human and the defender is a non-human bot). This shows the transferability of our defense to human generated attacks.

(b) Our proposed automatic UTSC-1 attack vs our proposed automatic defense (both attacker and the defender are non-human bots). Notice in UTSC-1 the adversary generates non-toxic attack utterance.

(c) Our proposed automatic UTSC-3 attack vs our proposed automatic defense (both attacker and the defender are non-human bots). Notice in UTSC-3 the adversary generates non-toxic attack utterance.

(d) RealToxicityPrompts vs GPT-2 generated responses one with the defense (in the dotted box) and one without (after the defense arrow).

(e) RealToxicityPrompts vs GPT-2 generated responses one with the defense (in the dotted box) and one without (after the defense arrow).

(f) RealToxicityPrompts vs GPT-2 generated responses one with the defense (in the dotted box) and one without (after the defense arrow).

Figure 16: Different qualitative results from our performed diverse experiments including human performed attack against our proposed defense mechanism (a), our proposed automatic attack and defense strategies (b-c), and lastly our defense mechanism on GPT-2 model using RealToxicityPrompts (d-f). The Dotted box represents the response if the defense was not applied, and the response after the defense arrow shows the newly generated response after applying the defense mechanism. Results show that the responses after the defense arrow (representing with defense response) are less toxic in all the cases compared to the results generated in the dotted boxes (representing the response without any defense applied). We also demonstrate the effectiveness of our defense against both toxic UTSC-3 (b) and non-toxic UTSC-1 attacks (c).

THANK YOU!

DO YOU HAVE ANY QUESTIONS?