동의어 치환을 이용한 심층 신경망 모델의 테스트 데이터 생성†

(Generating Test Data for Deep Neural Network Model using Synonym Replacement)

이 민 수 ‡ 이 찬 근 ¶
(Min-soo Lee) (Chan-gun Lee)

요 약 최근 이미지 처리 응용을 위한 심층 신경망 모델의 효과적 테스팅을 위해 해당 모델이 올바르게 예측하지 못하는 코너 케이스에 해당하는 행동을 보이는 데이터를 자동 생성하는 연구가 활발히 진행되고 있다. 본 논문은 문장 분류 심층 신경망 모델에 기반하고 있는 버그 담당자 자동 배정 시스템의 테스트를 위해 입력 데이터인 버그 리포트의 내용에서 임의의 단어를 선택해 동의어로 변형하는 테스트 데이터 생성 기법을 제안한다. 그리고 제안하는 테스트 데이터 생성 기법을 사용한 경우와 기존의 차이 유발 테스트 데이터 생성 기법을 사용했을 경우를 다양한 뉴런 기반 커버리지를 중심으로 비교 평가한다.

키워드 : 심층 신경망, 테스팅, 코너 케이스, 테스트 데이터 생성

Abstract Recently, in order to effectively test deep neural network model for image processing application, researches have actively conducted to automatically generate data in corner-case that is not correctly predicted by the model. This paper proposes test data generation method that selects arbitrary words from input of system and transforms them into synonyms in order to test the bug reporter automatic assignment system based on sentence classification deep neural network model. In addition, we compare and evaluate the case of using proposed test data generation and the case of using existing difference-inducing test data generations based on various neuron coverages.

Key words : Deep Neural Network, testing, corner-case, test data generation

1. 서 론

우리의 주변 일상생활의 응용은 물론 안전 크리티컬 시스템에서도 심층 신경망 모델을 이용하여 개발하는 사례가 근래에 많아졌다. 일반적으로 심층 신경망 모델은 복잡도가 높고 확률을 기반으로 하는 비결정적(non-deterministic) 시스템이다. 이러한 특성에 의해 심층 신경망 모델 기반의 시스템에 임의의 입력을 가했을 경우 시스템이 응답해야 하는 안전한 반응의 범위를 보장하기 힘들다. 따라서 테스팅(Testing)에 기반한 접근 방법 [4]과 정형 기법에 기반한 접근 방법 [5] 모두 심층 신경망을 다루기 위한 많은 연구가 진행되고 있다.

최근 소프트웨어 테스트 분야에서 화이트박스 테스팅(White-box testing)을 위해 심층 신경망이 예측하지 못하는 코너 케이스(corner case) 영역에 해당하는 데이터를 찾는 연구가 활발히 진행되었다. 기존의 연구는 대부분 이미지 기반 학습 모델의 테스트 데이터인 이미지를 변형시키는 접근을 시도하였다.

본 논문은 이미지 데이터를 기반으로 하고 있는 기존 연구와는 달리 텍스트 데이터를 기반으로 하고 있는 심층 신경망 모델에 대해 동의어 치환을 통해 해당 모델을 코너 케이스에 위치하도록 테스트 데이터를 변형시키는 방법을 제안한다.

본 논문의 구성은 다음과 같다. 2절에서는 관련 연구를 소개하고, 3절에서는 배경지식을 설명한다. 4절에서는 본

† 본 연구는 한국연구재단 기초연구사업 (과제번호: NRF-2017R1E1A1A01075803)과 한국원자력연구원 원자력 ICT 안전성 검증체계 구축 및 운영과제 (과제번호: 524320-18)의 지원을 받았음.

‡ 학생회원 : 중앙대학교 소프트웨어학부
als950901@naver.com
¶ 종신회원 : 중앙대학교 소프트웨어학부 교수
cglee@cau.ac.kr

논문접수 : 2019년 1월 4일
(Received 4 January 2019)
심사완료 : 2019년 1월 12일
(Revised 12 January 2019)

논문에서 제안하는 텍스트 데이터 변형 방법에 대해 소개하며, 5절에서는 제안한 변형 방법에 대한 평가 방법과 실험 결과를 다룬다. 마지막으로 6절에서 본 연구의 결론을 소개한다.

2. 관련 연구

심층 신경망 모델의 잘못된 예측을 유도하는, 코너 케이스 영역에 해당하는 행동을 보이는 테스트 데이터를 찾기 위한 선행 연구는 [1, 2]가 있다. [1]은 심층 신경망 모델을 위한 테스트 커버리지(Test Coverage) 개념의 필요성을 제기하고 뉴런 커버리지(Neuron Coverage)의 개념을 제안하였다. 또한 동일한 응용 목적을 갖는 두 개 이상의 심층 신경망 모델이 있을 때 같은 입력에 대해 모델 간 출력의 차이를 유발하는 테스트 데이터 생성(Difference Inducing Test Data Generation)을 통해 데이터 변형 알고리즘을 제안하였다. 제안된 알고리즘은 변형된 테스트 데이터가 학습 모델이 기존 학습 데이터를 처리하기 위해 사용한 뉴런들 이외의 뉴런을 사용케 하여 뉴런 커버리지를 높이는 전략을 사용한다. [2]에서는 다양한 뉴런 커버리지와 차이 유발 테스트 데이터 생성 방법을 제안하였다. 자세한 내용은 3절에서 설명한다.

3. 배경 지식

테스트 데이터를 학습된 심층 신경망 모델에 입력했을 때, 신경망의 뉴런에서 출력되는 결과 값을 기준으로 뉴런의 행동을 두 가지 부류로 나눌 수 있다. 첫 번째는 결과 값이 예상되는 범위에 속할 경우이며, 이것을 메인 케이스(main case) 영역에 위치해 있다고 한다. 두 번째는 결과 값이 예상되지 않는 범위에 속할 경우이며, 이것을 코너 케이스 영역에 위치해 있다고 하며, 본 논문에서 목표로 두고 있는 것이 비교적 많은 뉴런이 코너 케이스 영역에 해당하는 행동을 보이는 데이터를 찾는 것이다.

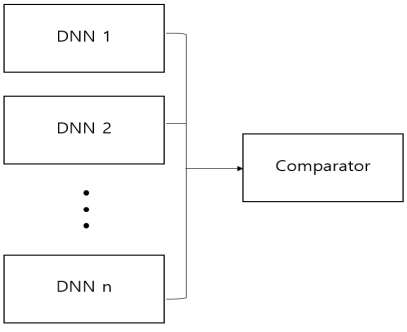

그림 34 DNN의 결과 비교

심층 신경망 모델이 예측하지 못하는 테스트 데이터를 찾는 과정은 먼저 같은 학습 데이터로 학습한 다수의 같은 구조의 모델이 필요하다. 이는 그림 1과 같이 같은 심층 신경망 모델들이 테스트 데이터 입력에 따라 나오는 결과 값이 서로 다른 경우를 예측이 빗나간 경우로 정의하기 위함이다. 다음은 테스트 데이터를 변형시키는 과정으로써 3.1절에서는 뉴런 커버리지, 3.2절에서는 차이를 유발하는 테스트 데이터 생성을 자세히 소개한다. 그리고 3.3절에서는 본 논문의 대상 시스템인 버그 담당자 자동 배정 시스템에 대해서 소개한다.

3.1 뉴런 커버리지

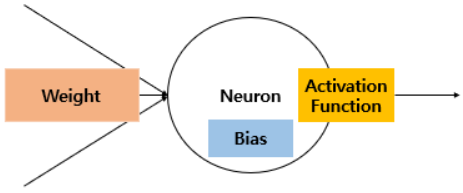

그림 35 뉴런의 구조

그림 2에서 볼 수 있듯이 일반적으로 심층 신경망에서의 뉴런은 입력 데이터, 가중치(Weight)와 바이어스(Bias)를 이용하여, 그들을 활성화 함수(Activation Function)을 통해 결과 값을 도출하는 구조를 가지고 있다.

학습된 심층 신경망 모델의 경우 가중치와 바이어스는 고정되어 오직 입력 데이터만 뉴런의 결과 값에 영향을 줄 수 있다. 입력 데이터에 의해 뉴런이 반응할 때, 즉 결과 값이 달라질 때 뉴런이 활성화되었다고 말한다.

$$Coverage = ActivatedNeuron\ /\ \ (1)$$

수식 1은 [1]에서 제안한 뉴런 커버리지에 대한 것이며, 이는 전체 뉴런에 대한 활성화된 뉴런의 비율을 나타낸다. 뉴런 커버리지의 값이 높을수록 심층 신경망 모델은 입력된 데이터를 예측 분류하기 위해 더 많은 뉴런을 활성화시킨다는 것을 의미하며, 이는 해당 모델의 예측 가능한 입력의 범위를 대략적으로 나타내기도 한다.

심층 신경망의 잘못된 예측을 유도하는, 코너 케이스 영역에 해당하는 행동을 보이는 테스트 데이터를 찾기 위해 뉴런 커버리지를 높이는 데이터를 찾는다. 이는 학습 시 모델에서 사용하지 않았던 뉴런을 사용하는 데이터가 예상과는 다른 행동을 보일 가능성이 크기 때문이다.

본 논문에서 사용할 기존 뉴런 커버리지의 종류는 Threshold Neuron Coverage(TNC) [1]와 K-Multi section Neuron Coverage(KMNC), Strong Neuron Boundary Coverage(SNBC) [2]이다. 이들은 서로 뉴런의 활성화를 판단하는 기준을 다르게 하였다. TNC는 단순히 뉴런의 활성화 판단을 뉴런의 결과 값이 미리 설정된 임계 값보다 큰 것으로 결정짓는다. 다시 말해 뉴런의

결과 값이 임계 값보다 크다면 활성화된 뉴런에 추가 시킨다. KMNC는 심층 신경망 모델을 학습 시 얻어지는 뉴런들의 결과 값을 사용하여 뉴런마다 최댓값과 최솟값으로 범위를 설정하고 그 범위를 K개로 나눈다. 주어진 입력 데이터에 의해 뉴런의 결과 값이 나눠진 구간에 속한다면, 그 구간이 활성화 되었다고 간주한다. 따라서 KMNC는 다른 뉴런 커버리지와 다르게 수식 2와 같이 계산할 수 있다.

$$KMNC = ActivatedSection / (K * \quad (2)$$

SNBC는 심층 신경망 모델을 학습 시 얻어지는 뉴런들의 결과 값을 이용하여 뉴런마다 최댓값을 설정한다. 뉴런의 결과 값이 얻어진 최댓값보다 크다면 해당 뉴런이 활성화되었다고 간주한다.

KMNC는 메인 케이스 영역에 해당하는 행동에 대한 커버리지이고, SNBC는 코너 케이스 영역에 해당하는 행동에 대한 커버리지라 할 수 있다.

3.2 차이 유발 테스트 데이터 생성

차이 유발 테스트 데이터 생성은 기존 테스트 데이터를 변형하며 심층 신경망 모델이 예측하지 못하는 데이터를 생성하는 과정이다.

먼저 테스트 데이터가 차이를 유발하는지 판단하기 위해 같은 학습 데이터로 학습된 복수개의 심층 신경망 모델을 준비한다. 설정된 데이터 변형 방법으로 입력 데이터를 변형 시키고, 변형시킨 데이터를 다시 준비된 모델들의 입력으로 가한다. 가해진 입력 데이터에 대한 모델의 뉴런 커버리지와 모델의 비용(cost)를 계산하고, 뉴런 커버리지와 비용을 같이 증가시키는 방향으로 테스트 데이터 변형을 반복한다. 이때 뉴런 커버리지와 비용을 증가시키는 방향이란, 변형된 테스트 데이터로 인한 모델의 최종 비용과 활성화 되지 않은 뉴런의 출력 값을 최대로 하는 방향을 뜻한다. 즉, 수식 3을 최대화 하는 것이다. 본 논문에서 ω는 [1]에서 사용한 값 0.1을 사용한다.

$$f(x) = \text{최종비용} + w * \text{비활성화뉴런의결과값} \quad (3)$$

변형 반복 과정 중 심층 신경망 모델들의 예측이 엇갈리면 생성을 종료한다. 이 일련의 과정은 모든 테스트 데이터에 대해 적용되며, 변형은 설정된 반복 수 만큼 적용된다.

본 논문에서 제안하는 방법과 비교하기 위해 사용할 기존의 차이 유발 테스트 데이터 생성 방법은 Fast Gradient Sign Method(FGSM) [2], Gradient Ascent Method(GAM) [1]이다. 이들은 변형하는 과정 중 모델의 비용을 이용하는 방법에 차이를 두었다.

$$x^* = x + \varepsilon * sign(f(x)) \quad (4)$$

각 데이터 변형 방법은 다음과 같다. FGSM은 수식 3을 이용한 수식 4로 변형을 진행한다. 수식 4에서 x는 현재 테스트 데이터를 의미하며 x^*는 변형된 테스트 데이터를 의미한다. GAM은 단순히 수식 3으로 변형을 진행한다.

3.3 버그 담당자 자동 배정 시스템

버그 관리 프로세스에서 새로이 보고된 버그를 해결하기 위해 적합한 담당자를 찾는 과정을 버그 선별(Triage)이라 한다. 우리는 본 논문에서 제안하는 변형 방법을 버그 담당자 자동 배정 시스템 [3]에 적용한다. 해당 시스템은 심층 신경망을 이용하여 기존의 버그 리포트를 해결한 개발자 패턴을 학습하고 새로운 버그 리포트가 주어졌을 때 적절한 개발자를 추천하게 한다. 버그 리포트는 보고자의 이름, 버그에 대한 설명(bug description), 버그 발생 환경, 버그의 심각도, 스택 추적(stack trace) 등으로 이루어져있다. 버그 담당자 자동 배정 시스템에서 사용하는 심층 신경망은 버그 리포트 구성 요소 중 버그에 대한 설명을 입력으로 사용한다.

버그에 대한 설명을 심층 신경망의 입력으로 사용하기 위해 워드 임베딩(Word embedding)을 진행한다. 워드 임베딩을 수행하기 전 특수용어와 불용어(stop words)을 제거하여 버그에 대한 설명에서 의미 있는 단어로만 구성하였다. 워드 임베딩의 학습 방법으로 Word2Vec을 사용했으며, 중심 단어로부터 주변 단어를 예측하는 Skip-gram 방식을 사용했다.

심층 신경망의 학습은 입력 데이터인 버그 리포트, 버그에 대한 설명은 해당 버그를 고친 개발자의 아이디로 라벨링(labeling)하여 지도 학습(supervised learning)방식으로 진행했다. 학습과 테스트에 사용할 데이터는 오픈소스 프로젝트인 Eclipse JDT의 2013년 2월부터 2015년 2월까지의 기간의 버그 리포트 1340개를 사용하였다. 해당 리포트에 나타난 담당자의 수는 16명이다.

본 논문에서는 [3]에서 실험한 기계학습 알고리즘 중 가장 높은 성능을 보인 CNN 모델을 대상으로 적용하였다.

4. 접근 방법

본 절에서는 텍스트 데이터에 적합한 테스트 데이터 변형 방법에 대해 제안한다. [1, 2]에서 제시한 알고리즘과 뉴런 커버리지를 사용하되, 명암 변경, 가림막 등 이미지 데이터를 기반으로 진행되던 변형 방법을 텍스트 데이터에 맞게 변경하였다.

텍스트 데이터의 변형을 위해 버그 리포트에 나타난 단어들을 사전적 의미가 같은 동의어로 변형시키는 방법으로 진행한다. 변경된 단어는 워드 벡터 테이블(word vector table)에 존재하지 않을 수 있기 때문에, 우리는 GoogleNews-vectors-negative300을 사용하였다.

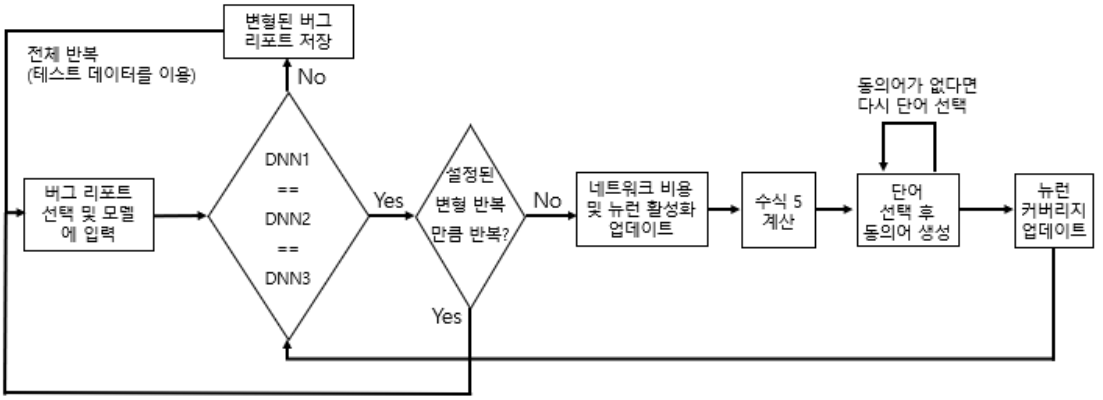

그림 36 차이 유발 테스트 데이터 생성 전체 알고리즘

$$x^* = x + \lfloor \max(L(W,\beta)) \rfloor \quad (5)$$

수식 5에서 x는 변형하기 전 데이터, x^*는 변형한 후 테스트 데이터, $L(W,\beta)$은 x를 모델의 입력으로 가했을 때 얻어지는 최종 비용을 의미한다. 해당 수식은 심층 신경망 모델의 비용을 이용한 테스트 데이터의 변형에 이용된다. 우리는 우선 모델을 잘못된 예측으로 유도해야 하고, 이미지 데이터에 비해 많은 변형이 필수로 요구되므로 $L(W,\beta)$에서 최대인 값을 이용했다. 또한 단어인 텍스트 데이터를 변형시키기 때문에, $L(W,\beta)$에서 최대인 값을 바닥(floor) 함수를 이용해 정수로 만들었다. 최종적으로 생성된 정수만큼 단어들을 선택해 동의어로 변형시킨다. 이렇게 변형된 단어들은 심층 신경망 모델로 하여금 입력 버그 리포트에 대해 적합하지 않은 담당자를 배정하도록 유도한다. 전체적인 알고리즘은 그림 3과 같다.

5. 평가 방법 및 실험 결과

5.1 평가 방법

본 논문은 텍스트 기반의 심층 신경망 모델이 예측하지 못하는 데이터를 찾는 과정에서 기존 데이터를 변형시키는 방법을 제안했다.

따라서 본 논문이 제안한 방법이 기존 방법에 비해 효율적인지 평가하기 위해 먼저 TNC, KMNC, SNBC [1, 2]와 같은 뉴런 커버리지 수치를 이용해 기존 이미지를 대상으로 한 차이 유발 테스트 데이터 생성인 FGSM, GAM과 비교하여 보여주는 것으로 평가한다. 그리고 설정하는 변형 반복과 전체 반복을 다르게 하여 반복수가 변형에 어떠한 영향을 주는지 SNBC를 이용하여 실험했다.

5.2 실험 결과

먼저 본 논문에서 제안한 데이터 변형 방식과 기존의 데이터 변형 방식을 TNC, KMNC, SNBC를 이용하여 비교한다. 본 논문에서 제안한 방법은 “Proposed”라 표기한다.

표 1 차이 유발 데이터 생성 방법 비교

	TNC	KMNC	SNBC
GAM	0.949	0.947	0.139
FGSM	0.948	0.948	0.165
Proposed	0.951	0.949	0.217

표 1은 세 개의 생성 기법을 적용해서 데이터를 만들었을 때 다양한 뉴런 커버리지 측정값을 나타낸다. 이는 변형 반복을 10회로 두고, 전체 반복을 30회로 두었을 때 실험 결과이다.

KMNC는 메인 케이스에 해당하는 모델의 행동을 계산하는 뉴런 커버리지이다. 표를 보면 제안한 방법과 기존의 방법으로 생성한 데이터가 대략 95%p 정도로 학습된 심층 신경망 모델이 예상할 수 있는 범주, 메인 케이스를 커버했다고 볼 수 있다.

SNBC는 코너 케이스에 해당하는 모델의 행동을 계산하는 것인데, 제안한 방법이 GAM, FGSM과 같은 기존 방법보다 대략 3~7%p 정도 차이를 보이고 있다. 이는 제안한 방법이 데이터를 생성하는 동안 기존 방법보다 학습된 심층 신경망의 뉴런의 3~7%p 가 코너 케이스에 해당하는 결과 값을 도출했다는 것을 의미한다. 즉, 비교적 더 많은 예상하지 못하는 범주에 속한 데이터를 생성했다고 말할 수 있다.

```
the jdt lexer Lashkar-e-Taiba hexadecimal floatingpoint misprint without for exemplar give_birth
valid misprint javac correctly report_card error misprint error hexadecimal numbers_pool
mustiness desegregate least matchless hexadecimal finger doubling the jdt give similar for malformed
literals missing exponent jdt gives error message invalid float literal digits xp00 eclipse testslexnum10f
test java2 xp00 error x0 p number
```

```
the jdt lexer Lashkar-e-Taiba hexadecimal floatingpoint misprint without for exemplar give_birth
valid misprint javac correctly report_card mistake misprint mistake hexadecimal numbers_pool
mustiness desegregate least matchless hexadecimal feel double the jdt give similar for malformed
literals missing exponent jdt gives error message invalid float literal digits xp00 eclipse testslexnum10f
test java2 xp00 error x0 p number
```

그림 4 버그 리포트의 원본과 변형 결과 1

```
if plugin provides file possible enable tracing plugin via
general tracing preference this bug covers adding plugin
preference page make easier debug problems runtime without restarting options
page org eclipse jdt ui eclipse
```

```
if plugin supply file potential enable trace plugin via
cosmopolitan trace predilection this bug screen add plugin
predilection page brand easy debug problem runtime without restart option
page org eclipse jdt ui eclipse
```

그림 5 버그 리포트의 원본과 변형 결과 2

표 2 변형 반복과 전체 반복 횟수 변화에 따른 SNBC 값

변형 반복 / 전체 반복	5	10	15	20
10	0.065	0.073	0.098	0.123
30	0.145	0.182	0.196	0.217
50	0.235	0.252	0.259	0.263

표 2는 변형 반복수와 전체 반복수를 조정하면서 제안한 방법을 적용할 때 측정된 SNBC의 값을 비교한 실험 결과이다. 첫 번째 열은 변형 반복수, 첫 번째 행은 전체 반복수를 의미한다.

전체적으로 변형 반복이나 전체 반복수를 늘렸을 때 SNBC는 증가하고 있으며, 이는 더욱 많은 코너 케이스 영역에 해당하는 행동을 보이는 데이터가 생성되었다는 것을 의미한다. 변형 반복수를 늘렸을 때와 전체 반복수를 늘렸을 때를 비교해 보았을 때, 비교적 전체 반복수를 늘려 다양한 데이터를 변형을 가한 것이 변형 반복수를 늘려 데이터에 많은 변형을 가한 것보다 SNBC 증가폭이 높다는 것을 알 수 있다.

그림 4, 5는 본 논문에서 제안하는 방법으로 심층 신경망 모델에 입력 데이터인 버그 리포트를 변형한 결과이다. 그림의 상단 버그 리포트는 원본이고, 하단 버그 리포트는 변형된 것이다. 또한 변형된 단어는 빨간색 박스로 표시하였다. 그림 4는 심층 신경망 모델 3개가 각각 12 , 14 , 14번째 담당자로 다른 예측을 했으며, 그림 5는 각각 8 , 8 , 0번째 담당자로 다른 예측을 했다.

6. 결론

심층 신경망 기술에 대한 수요가 많아지면서 검증의 필요성이 부각되어 많은 모델의 차이를 유발시키는 데이터를 생성하는 연구가 진행되었다. 이미지를 변형하여 심층 신경망이 예측하지 못하는 데이터를 생성하는 기존 연구와는 달리 본 연구는 텍스트를 변형하여 예측하지 못하는 데이터를 생성한다. 제안한 방법은 사용하는 심층 신경망 모델의 입력인 버그 리포트의 유의미한 단어를 동의어로 바꾸는 것으로 심층 신경망 모델이 예측하지 못하는, 그리고 코너 케이스 영역에 해당하는 행동을 보이는 데이터를 생성하는 것이다.

기존 연구의 데이터 생성 방법보다 본 연구에서 제안한 것이 뉴런 커버리지의 관점에서 약 3~7%p 좋은 성능을 보이는 것을 확인할 수 있었다. 또한 데이터의 변형 반복수, 그리고 전체 반복수를 조정하며 실험하여 전체 반복이 변형 반복보다 SNBC 증가폭이 더 높다는 것을 알 수 있었으며, 반복수와 뉴런 커버리지가 어느 정도 비례관계에 있다는 것을 알 수 있었다.

참 고 문 헌

[1] K. Pei, Y. Cao, J. Yang, S. Jana, “DeepXplore: automated whitebox testing of deep learning systems,” In Proc. of SOSP, ACM, pp. 1-18, Oct, 2017.

[2] Lei Ma, Felix Juefei-Xu, Jiyuan Sun, Chunyang Chen, Ting Su, Fuyuan Zhang, Minhui Xue, Bo Li, Li Li, Yang Liu, Jianjun Zhao, and Yadong Wang, “Deepguage: Comprehensive and multi-granularity testing criteria for gauging the robustness of deep learning systems,” In Proc. of ASE, pp. 120-13, 2018.

[3] 박해성, 김수빈, 이찬근, 채병훈, 딥러닝 모델 기반의 버그 담당자 자동 배정 시스템 성능 측정, 한국 소프트웨어 공학 학술대회 2018

[4] Houssem Ben Braiek, Foutse Khomh, "On Testing Machine Learning Programs,", CoRR, abs/1812.02257, 2018, URL: https://arxiv.org/abs/1812.02257

[5] Francesco Leofante, Nina Narodytska, Luca Pulina, Armando Tacchella, "Automated Verification of Neural Networks: Advances, Challenges and Perspectives," CoRR, abs/1805.09938, 2018, URL: http://arxiv.org/abs/1805.09938.

이 민 수
2019년 중앙대학교 컴퓨터공학부 학사
2019년~현재 중앙대학교 컴퓨터 공학과 석사과정.

이 찬 근
정보과학회논문지
제 44 권 제 3 호 참조