

다중언어 결함 벤치마크 개발

안가빈, 유신

1. 연구 개요

1단계에 ‘풀스택 소프트웨어 결함 벤치마크’의 구축을 수행함으로써, 향후 디버깅 기법 및 도구 개발의 기반을 확보하도록 한다. 풀스택 소프트웨어 결함 벤치마크는 실제 사용되는 오픈소스 프로젝트 중 풀스택 소프트웨어의 특징을 가지는 프로젝트를 선별하고, 해당 프로젝트에서 보고된 계층간 상호작용으로 인한 결함(bug)과 테스트케이스(test case)를 수집함으로써 구축하고, 이를 활용하여 풀스택 소프트웨어에 대한 정량적 분석이 가능하도록 한다.

2. 연구 내용

1) 벤치마크조사 대상 프로젝트 및 코드베이스 선정

- Tensorflow와 Numpy를 결함 사례 수집 대상 프로젝트로 선정하였다. 두 프로젝트 모두백엔드와 프론트엔드가 각기 다른 언어로 구현되어 있으며, 각 단을 구성하는 주 언어의 비율이 40%이상으로 일정 수준 이상의 비중을 차지하고 있어다중 언어로 구현된 프로그램으로 여겨지기에 적합하다고 판단하였다

프로젝트명	프로젝트 정보	
Tensorflow	버전	v1.8.0-rc1
	코드 저장소	https://github.com/tensorflow/tensorflow
	언어 구성	C++: 48%, Python: 40.6%, Other: 11.4%
	설명	Tensorflow는 데이터 플로우(data flow graphs)를 이용한 오픈소스 계산 라이브러리이다. 구글에서 개발되어 주로 기계학습 응용 프로그램에 사용한다. 핵심 부분은 C++과 CUDA로 작성되어 있지만, 사용의 편의를 위해 앞단(프론트)은 Python으로 작성되어 있다.
Numpy	버전	v1.14.2
	코드 저장소	https://github.com/numpy/numpy
	언어 구성	C: 53.9%, Python: 44.8%, Other: 1.3%
	설명	Numpy는 Python에서 고성능의 수치계산을 위해 만들어진 오픈소스 라이브러리이다. 계산 성능 향상을 위해 Numpy의 핵심 부분은 C로 작성되어 있고, 사용성의 편의를 위해 Python으로 인터페이스를 작성했다.

2) 실제 결함 사례 수집

가. 결함 수집 기준

- 이슈(Issue) 목록에서 닫혀있는 이슈(closed issue)이고 결함 태그(bug tag)가 붙어있을 것
- 초기 이슈는 또는 테스트 실패는 Python에서 발생하였지만, 실제 결함은 Python과 C 모두, 혹은 C/C++ 코드에서 고쳐졌을 것
- 결함과 관련된 테스트가 추가되었을 것

나. 수집 대상

- TensorFlow: 2016년 11월에서 2017년 5월 사이 발생한 다섯 개의 결함을 수집

- Numpy: 2016년 9월에서 2018년 3월 사이에 발생한 다섯 개의 결함을 수집

다. 결함 목록

프로젝트명	결함 ID	세부 정보	
Tensorflow	T1	설명	tf.reshape에서 올바르지 않은 shape를 return하는 버그이다.
		초기 이슈	https://github.com/tensorflow/tensorflow/issues/5886
		결함 위치	- tensorflow/core/kernels/transpose_op.cc - tensorflow/core/ops/array_ops.cc - tensorflow/core/ops/math_ops.cc - tensorflow/python/framework/ops.py
		결함 해결	https://github.com/tensorflow/tensorflow/commit/91ce95d497ec2957535b2ce6a965cd8269d723e5
		관련 테스트	- tensorflow/core/common_runtime/function_test.cc - tensorflow/core/ops/array_ops_test.cc - testDefaultAttrType, testDefaultListAttrType(tensorflow/python/framework/op_def_library_test.py) - testConvertToTensorPreferred, testConvertToInvalidTensorType(tensorflow/python/framework/ops_test.py)
	T2	설명	random number를 생성하는 서로다른 두 개의 session임에도 불구하고 같은 number를 return하는 버그.
		초기 이슈	https://github.com/tensorflow/tensorflow/issues/6854
		결함 위치	- tensorflow/compiler/jit/xla_local_launch_op.cc - tensorflow/compiler/tf2xla/op_registrations.cc - tensorflow/compiler/tf2xla/xla_compiler.cc - tensorflow/compiler/xla/service/elemental_ir_emitter.cc
		결함 해결	https://github.com/tensorflow/tensorflow/commit/bd371a33810ecb7957fa952b2af0e6dc33f9341c
		관련 테스트	- RandomOpsTest(tensorflow/compiler/tests/random_ops_test.py)
	T3	설명	split_v 연산자와 관련하여 shape shape inference에 오류가 존재하며, num_or_size_splits 변수가 constant tensor일 때 또한 IndexError가 발생한다.
		초기 이슈	https://github.com/tensorflow/tensorflow/issues/7754
		결함 위치	- tensorflow/core/kernels/split_v_op.cc - tensorflow/core/ops/array_ops.cc

			- tensorflow/python/ops/array_ops.py
		결함 해결	https://github.com/benoitsteiner/tensorflow/commit/53f68459f18fd9c707183511e1e58d03e2f367db
		관련 테스트	- testPhasedLSTMCell(tensorflow/contrib/rnn/python/kernel_tests/rnn_cell_test.py) - testShapeInference(tensorflow/python/kernel_tests/split_op_test.py)
	T4	설명	tensorflow.placeholder shape가 protobuf와 호환되지 않는다. (protobuf는 구조화된 데이터를 직렬화하기 위해 구글에서 개발된 메커니즘이다.)
		초기 이슈	https://github.com/tensorflow/tensorflow/issues/9103
		결함 위치	- tensorflow/cc/framework/cc_op_gen.cc - tensorflow/core/framework/partial_tensor_shape.cc - tensorflow/core/framework/partial_tensor_shape.h - tensorflow/core/ops/array_ops.cc - tensorflow/python/ops/array_ops.py
		결함 해결	https://github.com/tensorflow/tensorflow/commit/24a95ae389e1c76e771ac33d66e0ec40a236260f
		관련 테스트	- tensorflow/core/ops/array_ops_test.cc - tensorflow/cc/client/client_session_test.cc - testDtype, testShape, testUnknownShape, testScalarShape, testPartialShape, testControlDependency, testBadShape, testTensorStr, testOldGraph(tensorflow/python/kernel_tests/constant_op_test.py)
	T5	설명	tf.pow(x, y) 함수가 int 타입의 0보다 작은 값을 x 혹은 y의 인자로 전달 받았을 때 실행이 끝나지 않는다. float 타입일때는 정상적으로 동작한다.
		초기 이슈	https://github.com/tensorflow/tensorflow/issues/9560
		결함 위치	- tensorflow/core/framework/types.h - tensorflow/core/kernels/cwise_op_pow.cc - tensorflow/core/kernels/cwise_ops.h - tensorflow/core/kernels/cwise_ops_common.cc
		결함 해결	https://github.com/tensorflow/tensorflow/pull/15607
		관련 테스트	- testPowNegativeExponent(tensorflow/python/kernel_tests/cwise_ops_test.py)
Numpy	N1	설명	특정 input에 대해 ufunc.reduce가 틀린 결과를 반환한다. Outer dimension이 1일 때만 reduction이 일어나야 하는데, 0 초과일 때 reduction이 일어나도록 조건 검사가 잘못 작성되어 있어 발생하였다.

		초기 이슈	https://github.com/numpy/numpy/issues/8036
		결함 위치	- numpy/core/src/multiarray/nditer_api.c
		결함 해결	https://github.com/numpy/numpy/pull/8042/commits/21794ee1d62de5112ee981cffb9e9df5ccee6a16
		관련 테스트	- test_iter_buffering_reduction(numpy/core/tests/test_nditer.py) - test_reduce_noncontig_output(numpy/core/tests/test_ufunc.py)
	N2	설명	type_tuple_userloop_type_resolver에서 무한루프가 발생한다. Linked list를 traverse하는 동안 circuit이 존재하여 발생하는 문제이다.
		초기 이슈	https://github.com/numpy/numpy/issues/9351
		결함 위치	- numpy/core/src/umath/ufunc_type_resolution.c
		결함 해결	https://github.com/numpy/numpy/pull/9354
		관련 테스트	- test_custom_ufunc_forced_sig(numpy/core/tests/test_ufunc.py)
	N3	설명	np.double 에서 큰 값을 넣으면 error가 발생하는 버그이다. error handling 로직을 추가하여, 이럴 경우 inf를 return 하도록 변경한다.
		초기 이슈	https://github.com/numpy/numpy/issues/9924
		결함 위치	- numpy/core/src/multiarray/arraytypes.c.src
		결함 해결	https://github.com/numpy/numpy/pull/9961
		관련 테스트	- test_floating_overflow(numpy/core/tests/test_scalar_ctors.py)
	N4	설명	유효하지 않은 dtype 스트링을 파싱하려 할 때 파싱에러가 발생하는 대신 segmentation fault가 발생한다.
		초기 이슈	https://github.com/numpy/numpy/issues/10440
		결함 위치	- numpy/core/src/multiarray/descriptor.c
		결함 해결	https://github.com/numpy/numpy/pull/10623/commits/79697ad84fa73b49cca2b6380a74a0bfce4b32bc
		관련 테스트	- test_invalid_dtype_string(numpy/core/tests/test_dtype.py)
	N5	설명	Python2에서 numpy.float64 타입의 변수에 대해서, str과 print의 결과가 다르다. tp_str 메서드가 해당 타입에 대해 재정의되어있는 반면 tp_print 메서드는 built-in float를 그대로 상속받았기 때문에 발생한 문제이다. tp_print 메서드 또한 재정의해줌으로써 해결하였다.
		초기 이슈	https://github.com/numpy/numpy/issues/10753
		결함 위치	- numpy/core/src/multiarray/scalartypes.c.src
		결함 해결	https://github.com/numpy/numpy/pull/10763/commits/0d6296a660b2b05a8b364fbdf7c5c290de957b8c
		관련 테스트	- TestRealScalars::test_py2_float_print(numpy/core/tests/test_scalarprint.py)

3. 연구 결과

- 1) 세계 최초로 다중 언어 결함 벤치마크를 수집하였다.
- 2) 향후 2단계 과제 진행시 벤치마크로 수집한 결함을 지능형 자동 디버깅 기술의 개발 및 평가에 활용하도록 한다.