Compilers–1: CS3320 2019
Reading Assignment - 1:
TensorFlow/XLA and JIT

Sai Harsha Kottapalli
CS17BTECH11036

March 27, 2019

1

1. XLA (Accelerated Linear Algebra) is a domain-specific compiler for linear algebra that optimizes TensorFlow computations(as per tensorflow).

2. With the use of XLA for tensorflow graphs, we can accelerate Tensorflow ML models with minimal source code changes.

3. Tensorflow computations involve graphs which inturn relies on linear algebra.
   Therefore, Ops based on linear algebra are very important for ML algorithms, for which XLA optimizes the required computations.

4. Supports JIT compilation technique for optimize tensorflow computations during runtime which can potentially reduce memory bandwidth requirements and improve performance and AOT compilation technique to obtain a reduced executable file which can be run on devices with lower memory allocation.

5. XLA supports alternative backends and devices which is really helpful for new kind of computing devices.

6. Few objectives of XLA for tensorflow(source: tensorflow.org)

    - Improve execution speed
    - Improve memory usage
    - Reduce reliance on custom Ops
    - Reduce mobile footprint
    - Improve portability

2

- JIT stands for just-in-time compilation, which is resposible for using XLA to optimize the parts of Tensorflow graphs it runs.
- JIT compilation technique can optimize tensorflow computations during runtime which can potentially reduce memory bandwidth requirements and improve performance.
- It can be noted that during runtime we get get to know more about the state at which the program currently which can inturn help greatly in optimizing the compilation.
- TensorFlow also offers AOT compilation technique, which stands for ahead-of-time compilation, which can obtain a reduced executable file which can be run on devices with lower memory allocation.
- Using AOT compilation technique, avoids the runtime overhead which is why the total binary size is reduced making it quite favourable for mobile devices.

3

A compiler needs to focus on the following performance metrics(referred from tensorflow.org):

- Correctness of program
  This is obviously the most important metric as any user does not want to compromise on this.

- Execution speed
  optimization is not the only factor which user wants, there is a tradeoff between optimization and time required for it. Though the compiler should produce the best optimized code user should not wait too long to obtain the executable.

- Memory usage
  Especially helpful for lower memory devices or this allows for other processes to run parallely too.
  So, the compiler should not use too many intermediate storage buffers.

- Portability
  The intermediate code should be machine independent while also be able to support the different types of computers i.e. the compiler should not be specific to a particular type of configuration only as it forces users range of choices to lessen.