Papers

Steps Conferences

- Paper submission before a particular deadline
- Review
- Rebuttal
- Acceptance or rejection
- Camera ready
- Presentation

Steps in Journals

- Paper submission
- Review (more than 6-months is not rare)
- Accept/Reject/Minor Revision/ Major Revision
- Review
- Accept/Reject/Minor Revision/ Major Revision
- …
- +1 year can be possible

Conferences vs Journals in Computer Science

- Journals are way too slow for most practical fields
- Only relevant for ”stable” knowledge
- Readers are more interested in conferences (this is where the work has an impact)
- Current "acceleration" in certain areas (e.g., ML), even conferences are too slow (just preprints)

Spanish Accademia (ANECA)

- Most people who builds rankings is not aware of this

- Its changing …

Rankings

- IF (JCR)
  - Only journals
  - Q1/Q2/Q3/Q4 per field
- CORE/GRIN/SCIE
  - Effort from academy from Australia, Spain and Italy build to convince rank makers
- Unnecessary in most "serious" countries
- Beyond academia

SCIE/GRIN Top Conferences (A++, A+)

- Comp. Arch.
  - ISCA, HPCA, MICRO, …
- OS
  - OSDI, SOSP, ASPLOS, …
- Software
  - ICSE, PLDI, …
- …
- Machine Learning
  - ICML, NEURIPS, …

- 75 (A++, A+), 157 (A, A-), 319 (B,B-), 1811 (Uncategorized)

Review Process

- Conference
  - Program Chair(PChair) is named by steering committee
  - PChair invites Program Committee (PC) members (20-30 ppl)
  - PC (and external reviewers) prepare review
  - Authors resolves doubts of reviewers in rebuttal process
  - Paper merits are reevaluated and sorted
  - PC in plenary meeting decides what is accepted (<15% in top conf. is usual)
- Paper
  - Editor choses Associated Editor (according the topic)
  - Associated Editor as for external reviewers
  - Paper is accepted/asked for revision/rejected by AE (>50%)

Review #39B
==========

Overall merit
---
3. Weak accept

Reviewer expertise
---
2. Some familiarity

Paper summary

DNN has gained popularity and relevance in the last decade.
It requires large amount of computational resources.
GPUs and FPGAs are the preferred method of executing DNN applications.
The high availability and flexibility of general purpose CPUs makes the use of them, to execute DNN applications, also growing.
The focus of the paper is on CPUs.

Understanding hardware-software interaction is a key in uArch design.
The paper analyses the performance of a representative group of DNN applicatic from Intel's Model Zoo, on an Xeon Silver machine, and an Xeon Gold machine.
As a base line, the analysis use regular applications from SPEC CPU2017.

Section II gives a brief description of the DNNs used in the analysis, and the Top-Down profiling method.
The Top-Down methodology groups metrics from the Performance Monitoring Units in a multi-level hierarchy, allowing to identify bottlenecks in high levels, and diving into details by inspecting lower levels.

Section III describes the analysis settings.
The main machine is a 64-bit Intel Xeon Silver with two sockets of 16 out-of-order cores with SMT.
The main profiling tools are the Linux profiling command perf, and Intel's pmu-tools.

Section IV presents the analysis.
Most notably are the differences between DNN applications and regular programs, the effect of a second FMA in Xeon Gold, the memory bandwidth bottleneck, and the reduction of the DNN applications to just 8 representatives.

Comments for authors

The paper is easy to read and follow.

Strong points:
- Use of a popular suite (SPEC CPU2017) as a baseline to compare with.
- Relatively big suite of DNN models.
- Reduction of the DNN suite to a much smaller set of representative models.
- Identification of the memory bottleneck (adding more F.U.s quickly becomes less effective).

Main weak points:
- Only two systems were considered (Sliver and Gold), with only one significant difference between them.
- The analysis is straight forward interpretation of performance counters, using a preexisting methodology and tools.

Other weak points:
- Only a single architecture (x86-64) was analysed.
- Only a single memory configuration was analysed.
- Only a few conclusions about HW design, and almost nothing about SW design.
- No discussion about observed differences between PyTorch and TensorFlow models.
- Other SW parameters which were mentioned at the beginning, such as data types, are not discussed in the analysis later.

Notes:
- "Based on our evaluation, we show that the variability in terms of both performance and inherent characteristics between the different models is marginal."
  Maybe the selected group of models is too uniform? Probably not, but needs discussion.

- Table I. the alternation between single and double lines is uneasy on the eyes.

- The acronym IPC is used before it is explained.

- In some of the graphs the y access is 0-100, and in others 0-1, maybe make all of them the same.

- "As can be seen in Fig. 10, on average, DNN applications use one port per cycle"
  I'm not sure I can see that, if it's true maybe just rephrase. (I think I can see that in Fig. 3 looking at AVG-DNN?)

- Typo: "Each graph includes he previous (Silver) results", fix "he" -> "the".

- "Fig. 17 shows [...] performance gains [...]", can't really see that from this kind of graph.

- "most of the models fit into Cluster 1", it's the biggest cluster, but I wouldn't say "most".

- In S1 "choosing a smaller subset (7) of benchmarks representative of all of them (32)", but then later in S4, table II shows 8 highlighted.