

LArSoft - Support #22504

Request to upgrade the tensorflow version

05/02/2019 03:27 PM - Tingjun Yang

Status:	Closed	Start date:	05/02/2019
Priority:	Normal	Due date:	
Assignee:	Kyle Knoepfel	% Done:	100%
Category:		Estimated time:	0.00 hour
Target version:		Spent time:	0.00 hour
Experiment:	ArgoNeut, DUNE	Co-Assignees:	
Description			
Dear LArSoft experts, Would it be possible to upgrade tensorflow in larsoft? The current version is v1_3_0 and it would be great if it can be upgraded to v1_8_0 or higher. This will help the deep learning development in DUNE and ArgoNeuT and possibly other experiments. Thanks, Tingjun			
Related issues:			
Related to SciSoft - Support #23361: Abseil not c++17 compliant, causing prob...		Closed	10/02/2019

History

#1 - 05/06/2019 10:28 AM - Kyle Knoepfel

- Status changed from New to Feedback

We are accepting this, but we would like assistance from your tensorflow experts. With whom should we setup a meeting?

#2 - 05/07/2019 11:35 AM - Tingjun Yang

Hi Kyle,

You can setup a meeting with Saul and Leigh. I would like to join the meeting but they are the tensorflow experts.

Thanks,
Tingjun

#3 - 05/07/2019 12:36 PM - Lynn Garren

We also have a request from MicroBooNE for a working python interface. I would like have a joint meeting. Tentatively scheduled for 10 am May 8.

#4 - 05/07/2019 12:55 PM - Tingjun Yang

Lynn Garren wrote:

We also have a request from MicroBooNE for a working python interface. I would like have a joint meeting. Tentatively scheduled for 10 am May 8.

This does not work for us as there will be a ProtoDUNE meeting at the same time.

#5 - 07/12/2019 11:28 AM - Tingjun Yang

- Status changed from Feedback to Work in progress

From Lynn's email:

I had a talk with Marc P about the tensorflow build. We came up with two possible options. Fortunately, the cleanest option has worked and I have provided new patches for the c++17 build. I also had a look at the headers that are installed in the tensorflow include directory and refined the set.

tensorflow v1_12_0b is available for testing on cvmfs.

Please let us know if you have problems using this release. We presume that you will supply a larsim feature branch for use with tensorflow v1_12_0b. We should alert the larsoft mailing list and make sure there are no objections before making a release with the new build.

This release is only available for e17 (and e17:py3).

I remain concerned about tensorflow going forward. With newer releases we will have to use the bazel build, which appears to be problematic for spack. Also, tensorflow seems to be making its own copy of some utilities that are usually provided by the system. As long as everything is completely contained, that should be fine. Given that tensorflow maintains its own ecosystem, it may be wise to consider running it inside a container and taking the output in some fashion.

Lynn

#6 - 07/12/2019 11:38 AM - Tingjun Yang

I have started testing tensorflow v1_12_0b. The upgrade seems to be quite straightforward. The only issue is the following error:

```
/cvmfs/larsoft.opensciencegrid.org/products/tensorflow/v1_12_0b/Linux64bit+3.10-2.17-e17-prof/include/tensorflow/core/public/session.h:24,
      from /data/tjyang/dune/larsoft_em/srcs/dunetpc/dune/CVN/tf/tf_graph.cc:12:
/cvmfs/larsoft.opensciencegrid.org/products/tensorflow/v1_12_0b/Linux64bit+3.10-2.17-e17-prof/include/tensorflow/core/lib/core/stringpiece.h:29:10: fatal error: absl/strings/string_view.h: No such file or directory
#include "absl/strings/string_view.h"
      ^~~~~~
```

The solution is to add include_directories(\$ENV{TensorFlow_INC}/absl) to the CMakeLists.txt file wherever session.h is included.

I have created feature branches feature/team_for_tensorflow_v1_12_0b in both larreco and dunetpc. I am going to test if the tensorflow results remain the same for DUNE. I will deal with goneoutcode later.

#7 - 07/12/2019 12:16 PM - Tingjun Yang

- File errors.txt added

Tried to run tensorflow on an existing DUNE MC file:

```
lar -c select_ana_dune10kt_nu.fcl xroot://fndca1.fnal.gov:1094/pnfs/fnal.gov/usr/dune/tape_backed/dunepro/mccl1/protodune/mc/full-reconstructed/07/51/34/20/nue_dune10kt_1x2x6_12855888_0_20181104T211321_gen_g4_detsim_reco.root -n -1
```

There are lots of errors like this

```
2019-07-12 12:04:58.201673: E tensorflow/core/framework/op_kernel.cc:1197] OpKernel ('op: "MutableDenseHashTableV2" device_type: "CPU" constraint { name: "key_dtype" allowed_values { list { type: DT_STRING } } } constraint { name: "value_dtype" allowed_values { list { type: DT_INT64 } } }') for unknown op: MutableDenseHashTableV2
```

The full error log is attached.
The program hangs on the first event:

```
Classifier summary:
Output 0: 0.0257705,
Output 1: 2.55598e-05, 0.999172, 0.000753308, 4.93325e-05,
Output 2: 0.00586059, 0.367086, 0.627042, 1.20333e-05,
Output 3: 6.90421e-05, 3.75733e-05, 0.000108019, 0.999785,
Output 4: 0.997856, 0.0021328, 1.05676e-05, 3.6611e-07,
Output 5: 0.999907, 9.18793e-05, 7.18959e-07, 3.51992e-08,
Output 6: 0.999743, 0.000255833, 9.39033e-07, 3.14338e-07,
```

I guess the conclusion is that we cannot use the old networks with the new version of tensorflow.

#8 - 07/15/2019 10:33 AM - Kyle Knoepfel

There appears to be a conflict between the old and new tensorflow-generated data schema. Unless tensorflow supports schema evolution, you may need to regenerate your tensorflow-formatted data.

#9 - 07/15/2019 10:34 AM - Kyle Knoepfel

- Status changed from Work in progress to Feedback

Do you think the feature branches you are working on will be ready for this week's release?

#10 - 07/15/2019 10:53 AM - Tingjun Yang

Kyle Knoepfel wrote:

Do you think the feature branches you are working on will be ready for this week's release?

Hi Kyle,

No, we need to train new networks in order to use the new version of tensorflow. By the way, does MicroBooNE use tensorflow in their code? If so they also need to retrain in order to use the new tensorflow.

#11 - 07/17/2019 11:09 AM - Lynn Garren

MicroBooNE uses tensorflow via a container. They do not use our build. We expect to have a report from them at the next coordination meeting.

#12 - 08/28/2019 07:56 AM - Aiden Reynolds

- File error.txt added

Tried to run a larsoft job to do the CNN based hit tagging with a new network trained with tensorflow v1.12.

I get a lot of errors similar to those mentioned by Tingjun above, e.g.

```
2019-08-28 07:49:27.703868: E tensorflow/core/framework/op_kernel.cc:1197] OpKernel ('op: "
MutableDenseHashTableV2" device_type: "CPU" constraint { name: "key_dtype" allowed_values { list { type:
DT_STRING } } } constraint { name: "value_dtype" allowed_values { list { type: DT_INT32 } } }') for
unknown op: MutableDenseHashTableV2
```

After opening the input file the job stalls but does not return a failure code, full log attached.

#13 - 08/28/2019 10:03 AM - Lynn Garren

Aiden, Tingjun is still working on the training to use tensorflow 1.12. The good news is that this might be ready next week.

#14 - 08/28/2019 10:18 AM - Tingjun Yang

Lynn Garren wrote:

Aiden, Tingjun is still working on the training to use tensorflow 1.12. The good news is that this might be ready next week.

Lynn, Aiden is the person who is training the CNN using tensorflow 1.12 and he saw the same error with the retrained network.

#15 - 08/28/2019 10:37 AM - Kyle Knoepfel

Aiden, can you give us a sample workflow to test, including setup instructions?

#16 - 08/28/2019 11:44 AM - Tingjun Yang

I have created feature branch feature/team_for_tensorflow_v1_12_0b in larreco and dunetpc. It works with larsoft v08_29_00.

To build it, you need to unsetup tensorflow and protobuf first.

After you build and setup dunetpc, you can run the following command to reproduce the problem:

```
lar -c protoDUNE_SP_keepup_decoder_reco.fcl
xroot://fndca1.fnal.gov:1094/pnfs/fnal.gov/usr/dune/tape_backed/dunepro/protodune/np04/beam/detector/None/raw/07/74/37/39/np04_raw_run00580
9_0023_dl6.root -n 1
```

#17 - 09/03/2019 01:54 PM - Kyle Knoepfel

I find that if I use the debug build, I am able to proceed without any issues, albeit more slowly. However, there does appear to be some type of hang in the prof build. That will take some more investigation.

#18 - 09/03/2019 04:09 PM - Kyle Knoepfel

- File tensorflow_stacks.txt added

Using the 'pstack' command, I get the attached printout. Looking at the lowest stack frame for each of the threads launched by tensorflow:


```

// Compute new capacity by repeatedly doubling current capacity

// TODO(psrc): Check and avoid overflow?

size_type new_capacity = capacity(); // KJK: new_capacity is 0
while (new_capacity < target) {      // KJK: comparison returns true
    new_capacity <<= 1;               // KJK: new_capacity is shifted by 1 bit (i.e. doubled); still 0.
}                                     // KJK: new_capacity still 0, repeat loop infinitely

Allocation new_allocation(allocator(), new_capacity);

UninitializedCopy(std::make_move_iterator(data()),
                  std::make_move_iterator(data() + s),
                  new_allocation.buffer());

ResetAllocation(new_allocation, s);
}

```

Although this is a bug in Abseil's library, it's possible it is triggered by incorrect use of TensorFlow. I recommend two approaches:

- Switch to using TensorFlow's newer C++ API (some of the TensorFlow code in LArSoft is using older API)--that may help get around the problem
- Switch to a newer version of TensorFlow, which does not use this fragile bit of code. It would take some investigation to figure out which version of TensorFlow would be adequate.

Please let us know how you would like to proceed.

#21 - 09/16/2019 12:20 PM - Tingjun Yang

Dear Kyle,

Thank you for the investigation and locating the problem. It is probably easier to update the C++ API as building a newer version of tensorflow can be time consuming. Do you expect us to update the API or are you available to help us with that?

Thanks,
Tingjun

#22 - 09/16/2019 12:44 PM - Kyle Knoepfel

We should probably do it together. Are you guys available sometime this week to meet?

#23 - 09/30/2019 10:49 AM - Kyle Knoepfel

Pinging to keep this alive.

#24 - 10/01/2019 05:09 AM - Saul Alonso Monsalve

- File *tensorflow_errors.txt* added

I am getting similar errors when trying to run a network trained with Tensorflow 1.12.0 (I have attached a file with the errors). As Kyle said, it might be an issue related to the Tensorflow C++ API. I will investigate it.

#25 - 10/10/2019 12:38 PM - Tingjun Yang

Hi Kyle and all,

We would like to setup a meeting with you next week to discuss a plan to upgrade the tensorflow C++ API. Could you please fill this doodle pool?
<https://doodle.com/poll/pt2fw2f26ntac5bw>

Thanks,
Tingjun

#26 - 10/15/2019 10:30 AM - Tingjun Yang

Lynn, Leigh, Saul and myself just had a meeting. We agreed to try ClientSession as suggested by Kyle and see if that solves the problem.

#27 - 10/15/2019 10:50 AM - Leigh Whitehead

Saúl and I have done a bit of research and found the following: the C++ interface in larreco (and the one we based on it in dunetpc), which was written by Robert Sulej, actually goes straight to the core of Tensorflow and bypasses the entire C++ API itself. This certainly doesn't sound like the best approach! As such, we will modify the interface in dunetpc to test things using the C++ API properly with the ClientSession class suggested by Kyle and some other necessary changes. Hopefully this will solve the issues that we have seen.

#28 - 10/21/2019 09:43 AM - Kyle Knoepfel

- Related to Support #23361: Abseil not c++17 compliant, causing problems in Tensorflow v1_12_0b e17 build added

#29 - 03/16/2020 05:48 PM - Tingjun Yang

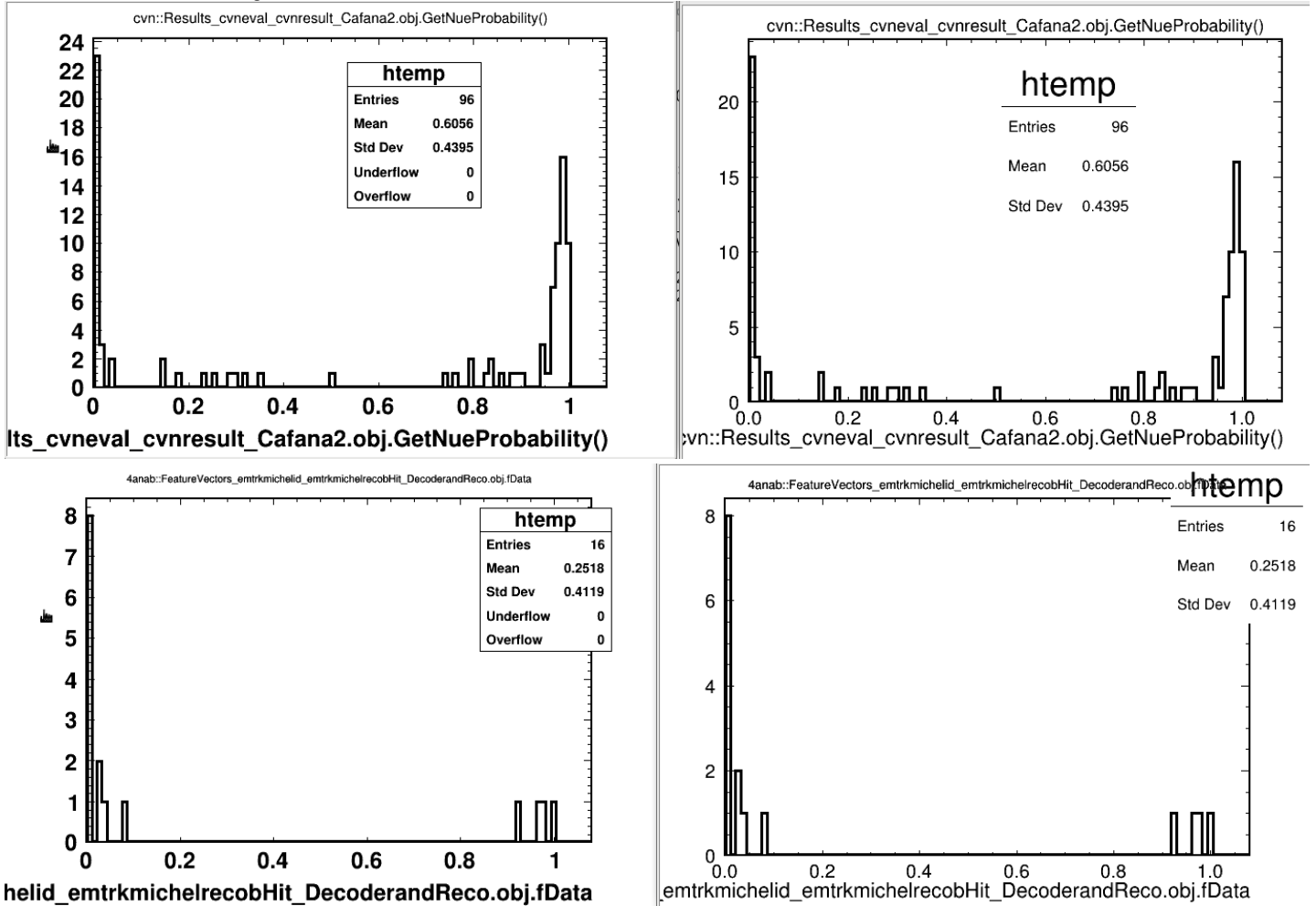
- File cvn.png added

- File cnn.png added

There is good progress on this. Pengfei made tensorflow 1.12.0b for SLF7 (both py3 and py2):

<https://home.fnal.gov/~dingpf/tensorflow/>

I tested the prof build and I was able to build larrecodnn and dunetpc against it with some minor changes. The results on DUNEFD cvn and ProtoDUNE cnn are unchanged:



Left is with tensorflow 1.12.0b and right is with tensorflow v1.3.0i.

Lynn, would it be OK to install the tensorflow 1.12.0b builds to larsoft cvmfs? I will provide pull request to larrecodnn and feature branches to experiment code.

Tingjun

#30 - 03/16/2020 06:02 PM - Lynn Garren

In order to install, we must have a source code tarball and be able to build tensorflow on jenkins. These are requirements. We already have a v1_12_0b tag for the tensorflow build that does not include any modifications from Pengfei. There will need to be a new tag (v1_12_0c) with an updated build script.

#31 - 03/16/2020 06:55 PM - Pengfei Ding

Here is the link to the source tarball:

<https://home.fnal.gov/~dingpf/tensorflow/tensorflow-v1.12.0c-source.tar.bz2>

It has the tag v1_12_0c, and have a several fixes for:

1. latest protobuf changed package layout (using lib64 as opposed to lib);
2. one of tensorflow's header files used a deprecated declaration which did not stop the build of tensorflow itself as "-Wno-deprecated-declarations"

is set, but it causes errors for dunetpc as that compiler flag is not set by default in e19;

#32 - 03/16/2020 08:43 PM - Lynn Garren

Would you commit the required changes to <ssh://p-build-framework@cdcvs.fnal.gov/cvs/projects/build-framework-tensorflow-ssi-build> please.

#33 - 03/16/2020 10:12 PM - Lynn Garren

And I have a question. There are old entries in the table file for the e17 qualifier. Is there any reason to keep these entries in the table file? In other words, are you intending to build and use this release of tensorflow with something other than the current head of larsoft?

#34 - 03/16/2020 10:13 PM - Tingjun Yang

We only need e19 builds.

#35 - 03/16/2020 10:37 PM - Lynn Garren

Thanks for clarifying that. How soon can you provide the PR and experiment code feature branches? I realize we can't run the CI tests until tensorflow v1_12_0c is on cvmfs.

#36 - 03/16/2020 10:38 PM - Tingjun Yang

I can provide them early tomorrow.

#37 - 03/16/2020 10:51 PM - Pengfei Ding

I've committed my changes to the repo. I forgot to mention that I added a "compile_nsync.sh" into the patch directory.

I also updated the source tarball linked here:

<https://home.fnal.gov/~dingpf/tensorflow/tensorflow-v1.12.0c-source.tar.bz2>

The build script in the source tarball is now in sync with the version in the git repo. The previous version was using ``-std=c++14`` for e17 and e19 build. It is fixed in the latest tarball now. Please download the tarball again.

#38 - 03/17/2020 09:45 AM - Tingjun Yang

Hi Lynn,

I have submitted larrecodnn PR#1 and created feature branch feature/team_for_tensorflow_v1_12_0b in dunetpc. Note this name of branch says v1_12_0b but everything depends on v1_12_0c now.

Please let me know if you have questions.

#39 - 03/17/2020 12:08 PM - Tingjun Yang

I have renamed the dunetpc branch to feature/tjyang_tensorflow_v1_12_0

#40 - 03/17/2020 03:09 PM - Lynn Garren

tensorflow v1_12_0c is now on cvmfs. I'll give it time to propagate before triggering a CI.

#41 - 03/20/2020 12:21 AM - Lynn Garren

- % Done changed from 0 to 100

- Status changed from Feedback to Resolved

Thanks to everyone who contributed. larsoft v08_47_00 now uses tensorflow v1_12_0c.

#42 - 03/23/2020 10:28 AM - Kyle Knoepfel

- Status changed from Resolved to Closed

Files

errors.txt	49.4 KB	07/12/2019	Tingjun Yang
error.txt	53.9 KB	08/28/2019	Aiden Reynolds
tensorflow_stacks.txt	77.4 KB	09/03/2019	Kyle Knoepfel
tensorflow_errors.txt	49.5 KB	10/01/2019	Saul Alonso Monsalve
cnr.png	48.5 KB	03/16/2020	Tingjun Yang
cvn.png	52.6 KB	03/16/2020	Tingjun Yang