Project: tensorflow

2015-11-09T20:17:58Z keon 245comments

Node.js (JavaScript) Wrapper API

Web url: https://github.com/tensorflow/tensorflow/issues/37

API url: https://api.github.com/repos/tensorflow/tensorflow/issues/37

Because JavaScript is Awesome

-------------------------------------------------------------------------

2015-11-09T20:52:58Z ZECTBynmo

+1!

-------------------------------------------------------------------------

2015-11-09T22:01:57Z dhritzkiv

+1

-------------------------------------------------------------------------

2015-11-09T23:20:33Z fiws

:+1:

-------------------------------------------------------------------------

2015-11-10T01:03:57Z miguelalche

+1

-------------------------------------------------------------------------

2015-11-10T01:19:39Z jagandecapri

Just what I was searching for. :+1:

As quoted from the offical website http://www.tensorflow.org/

> we‚Äôre hoping to entice you to contribute SWIG interfaces to your favorite language -- be it Go, Java, Lua, Javascript, or R.

I am new to this whole SWIG thing but searched around and found this. http://www.swig.org/Doc3.0/Javascript.html

Not really sure how this works. Do we need to write swig interface file specifically for Javascript or is it auto-generated when running some commands or is somebody already working on this (this would be awesome) ?

-------------------------------------------------------------------------

2015-11-10T01:47:27Z imgntn

+1 :+1:

-------------------------------------------------------------------------

2015-11-10T02:19:26Z prathamesh7pute

+1

-------------------------------------------------------------------------

2015-11-10T03:44:17Z Luna4

+1

-------------------------------------------------------------------------

2015-11-10T03:57:51Z dmitriykharchenko

+1

-------------------------------------------------------------------------

2015-11-10T04:35:40Z marcbaetica

+1

-------------------------------------------------------------------------

2015-11-10T05:06:27Z lowe0292

üëç

-------------------------------------------------------------------------

2015-11-10T05:22:36Z mattkosoy

:+1:

-------------------------------------------------------------------------

2015-11-10T10:37:59Z nikhilk

+1!

Just starting out on one, but new to writing a nodejs addon. Checking out the swig interface files to see if they're going to be helpful, or if I should just use the c++ API.

-------------------------------------------------------------------------

2015-11-10T10:48:40Z vnglst

+1

-------------------------------------------------------------------------

2015-11-10T11:00:44Z jalona

+1

-------------------------------------------------------------------------

2015-11-10T14:46:37Z Foorack

+1

-------------------------------------------------------------------------

2015-11-10T15:00:50Z tngan

+1

-------------------------------------------------------------------------

2015-11-10T23:47:34Z vincentvanhoucke

This is something the core TensorFlow team is unlikely to tackle in the near future, so if you want to contribute it, please go ahead! I would recommend circulating a proposed implementation on the discuss mailing list early on, so that a consensus about where such API might live (in repo / off repo / in 'contrib' directory) can be reached ahead of time.

-------------------------------------------------------------------------

2015-11-11T05:08:28Z Foorack

Anyone up to write a NodeJS library? :+1:

I think it would be better with a official NodeJS API however a community one will be as (if not more) interesting in my opinion. I know there are multiple ways of approaching this however I strongly recommend node-gyp for performance. I will gladly contribute in any way I can, however, this is something I will not be able to do alone. Would be best if a few other people is interested as well, specially someone with C++ knowledge.

-------------------------------------------------------------------------

2015-11-11T07:52:19Z augbog

:+1:

-------------------------------------------------------------------------

2015-11-11T07:53:06Z tngan

@Foorack I am willing to contribute it if some people are interested as well. Is it possible to move the discussion to a slack channel ? (see #31)

-------------------------------------------------------------------------

2015-11-11T08:00:05Z Foorack

@tngan The slack channel is private, however I was able to join with the herokuapp link. :+1:

-------------------------------------------------------------------------

2015-11-11T08:51:27Z tngan

We hope more developers to discuss and contribute. Now we have a slack channel named as nodejs (see #31), and a Github repository [node-tensorflow](https://github.com/node-tensorflow/node-tensorflow) is reserved. Thanks @Foorack !

-------------------------------------------------------------------------

2015-11-11T08:56:58Z marcbaetica

I am willing to contribute. Thanks for the initiative guys!

-------------------------------------------------------------------------

2015-11-11T10:40:34Z Foorack

@miguelalche Glad to see you're interested! Please join the slack channel and someone will add you to the repository. ^^

-------------------------------------------------------------------------

2015-11-11T11:29:17Z cauerego

I look forward for contributing ( specially along with #132 ) !!

-------------------------------------------------------------------------

2015-11-28T20:41:10Z gpresland

+1

-------------------------------------------------------------------------

2015-12-01T15:17:23Z mpj

Hooray for node! Let's do this.

-------------------------------------------------------------------------

2015-12-03T01:59:25Z miql

+1

-------------------------------------------------------------------------

2015-12-04T16:54:31Z g3ortega

+1

-------------------------------------------------------------------------

2015-12-04T18:11:37Z jimfleming

Here's a writeup on how to load and execute TensorFlow graphs using the C API: https://medium.com/jim-fleming/loading-tensorflow-graphs-via-host-languages-be10fd81876f (source code included)

-------------------------------------------------------------------------

2015-12-05T17:22:41Z tngan

The proposal will be released in the next week. Keep tracking on https://github.com/node-tensorflow/node-tensorflow.

-------------------------------------------------------------------------

2015-12-06T06:43:37Z duyetdev

+1

-------------------------------------------------------------------------

2015-12-07T03:27:29Z nikhilk

I have published my starting point -- https://github.com/nikhilk/node-tensorflow that will be published to npm later.

@jimfleming - like your approach (we're both using ffi ... I did it to get started quickly). Are you going to take on building higher level framework-style APIs to replicate the python experience? Thats my next step.

-------------------------------------------------------------------------

2015-12-07T05:37:21Z jimfleming

@nikhilk Thanks. Something like `new tf.Tensor()` instead of `tf.NewTensor()` might be a nice addition but I'm not planning on expanding it at the moment. I'm only interested in loading graphs created in python and I think I like the minimalism.

-------------------------------------------------------------------------

2015-12-15T09:42:39Z robbinhan

+1

-------------------------------------------------------------------------

2015-12-15T18:26:44Z tngan

The proposal is released here with current progress.

https://github.com/node-tensorflow/node-tensorflow/tree/1.0.0

There is an open issue for the discussion.

https://github.com/node-tensorflow/node-tensorflow/issues/2

-------------------------------------------------------------------------

2015-12-16T07:45:39Z alexzaporozhets

+1

-------------------------------------------------------------------------

2015-12-19T21:34:22Z gmosx

+1

-------------------------------------------------------------------------

2015-12-21T09:50:02Z please-wait

+1

-------------------------------------------------------------------------

2015-12-21T12:05:10Z haduythuan

+1

-------------------------------------------------------------------------

2015-12-21T15:13:47Z qrpike

+1

-------------------------------------------------------------------------

2015-12-22T16:44:10Z kenokabe

+1

-------------------------------------------------------------------------

2015-12-28T00:11:06Z buholzer

+1

-------------------------------------------------------------------------

2016-01-06T14:16:59Z ovrmrw

+1

-------------------------------------------------------------------------

2016-01-06T17:20:20Z Konard

+1

-------------------------------------------------------------------------

2016-01-06T21:08:56Z pavelchristof

+1

It'd be great to be able to easily run tensorflow networks in node.js based servers.

-------------------------------------------------------------------------

2016-01-07T13:57:00Z nikhildaga

+1

-------------------------------------------------------------------------

2016-01-09T02:18:03Z taveras

+1

-------------------------------------------------------------------------

2016-01-09T19:57:36Z alexahdp

+1

-------------------------------------------------------------------------

2016-01-11T09:09:17Z pavel-kudinov

+1

-------------------------------------------------------------------------

2016-01-14T10:34:25Z muka

+1

-------------------------------------------------------------------------

2016-01-14T19:38:17Z jasonmayes

+1

-------------------------------------------------------------------------

2016-01-22T01:30:51Z andrewjaykeller

I am very willing to contribute. @Foorack please add me to what ever you can! I have a real life use case for this so it will be great to develop and test it at the same time!!

-------------------------------------------------------------------------

2016-01-22T07:02:32Z Foorack

@pushtheworldllc I'm glad you are interested. :)

The repository and the proposal can be found here: https://github.com/node-tensorflow/node-tensorflow/

You can use this link to join our slack channel: https://tensor-flow-talk-invite.herokuapp.com/

-------------------------------------------------------------------------

2016-01-22T18:59:40Z ericmuyser

+1

-------------------------------------------------------------------------

2016-01-23T04:14:46Z teagup

+1

-------------------------------------------------------------------------

2016-01-23T06:16:51Z helxsz

+1

-------------------------------------------------------------------------

2016-01-27T12:24:22Z aralroca

+1 :+1:

-------------------------------------------------------------------------

2016-02-01T13:34:00Z pavelkukov

+1 :+1:

-------------------------------------------------------------------------

2016-02-15T10:16:30Z peterbraden

I have a working prototype using SWIG here: https://github.com/node-tensorflow/node-tensorflow/pull/13

The next steps would be to define the areas that the bindings would initially cover (must be within the C++ API ) and start implementing the SWIG interface files for these.

If anyone has experience with SWIG, I'd love to collaborate, as it seems like a huge amount of the python SWIG interfaces are custom overrides etc. and I'm keen not to reproduce their work. Additionally, would be great to get some clarity from the tensorflow team on what API's would be good to initially cover as I'm sure their roadmap has many changes on the way, and I wouldn't want to conflict. (cc @martinwicke ? )

-------------------------------------------------------------------------

2016-02-19T02:52:44Z vsantosu

+1

-------------------------------------------------------------------------

2016-02-20T20:30:21Z ghost

+1 :+1:

-------------------------------------------------------------------------

2016-02-28T05:04:10Z kristoffernolgren

+1

-------------------------------------------------------------------------

2016-03-12T13:17:04Z maltyxx

+1

-------------------------------------------------------------------------

2016-03-20T09:29:14Z ramsundhar20

+1

-------------------------------------------------------------------------

2016-03-23T21:33:39Z wajeehullah

+1

-------------------------------------------------------------------------

2016-03-27T20:10:05Z sbechet

+1

-------------------------------------------------------------------------

2016-03-28T23:46:26Z ripper2hl

please try use reactions button --> http://www.geekwire.com/2016/github-adds-reactions-keep-comments-track/ no more +1 comments xD

-------------------------------------------------------------------------

2016-04-14T12:27:17Z qrpike

+1

-------------------------------------------------------------------------

2016-04-14T18:46:20Z allanjuma

:+1:

-------------------------------------------------------------------------

2016-04-28T06:54:33Z adamvresky

+1

-------------------------------------------------------------------------

2016-05-01T18:02:50Z claudio4

+1

-------------------------------------------------------------------------

2016-05-02T11:52:43Z Loksly

+1

-------------------------------------------------------------------------

2016-05-02T15:44:53Z martinwicke

@peterbraden sorry for the prolonged silence. We are building out the C++ API, and it will grow over time. I expect the most useful bits to be the parts that are needed to run an existing graph. The C++ graph building API is being redone right now, so it's not particularly useful to spend much time on it.

I'd love it if you prepared a PR to put these bindings into contrib/nodejs.

-------------------------------------------------------------------------

2016-05-03T19:44:07Z peterbraden

@martinwicke thanks for the encouragement. I made an initial stab at it here: https://github.com/tensorflow/tensorflow/pull/2206 - this is just a proof of concept that gets the version string into nodejs. I'll start work on adding the swig interfaces for the graph running stuff.

-------------------------------------------------------------------------

2016-06-09T21:00:51Z jonchenn

+10000

-------------------------------------------------------------------------

2016-06-11T01:23:26Z progdude

+1

-------------------------------------------------------------------------

2016-06-26T14:13:02Z vunb

+1

-------------------------------------------------------------------------

2016-07-02T14:41:20Z bertolo1988

# +1

-------------------------------------------------------------------------

2016-07-20T14:42:10Z teckays

+1

-------------------------------------------------------------------------

2016-07-28T13:39:29Z lucked

+1

-------------------------------------------------------------------------

2016-07-30T07:42:09Z mredmundto

+1

-------------------------------------------------------------------------

2016-08-06T19:41:22Z guillaume-chevalier

This would be interesting for pure front-end graph exportation for direct usage on web clients. Even if the desired inputs/outputs asked from the graph would be hard-coded in the exported JS "sess.run" equivalent function.

-------------------------------------------------------------------------

2016-08-10T00:36:57Z jadewon

+1

-------------------------------------------------------------------------

2016-08-18T16:23:53Z rodrigo-garcia-olx

:+1:

-------------------------------------------------------------------------

2016-08-25T01:45:40Z prajankya

+1

-------------------------------------------------------------------------

2016-09-06T07:41:43Z heineiuo

+1

-------------------------------------------------------------------------

2016-09-13T12:33:30Z sequoiar

+1

-------------------------------------------------------------------------

2016-09-27T08:35:15Z vjk2005

+1

-------------------------------------------------------------------------

2016-09-27T15:58:19Z code-guru

+1

-------------------------------------------------------------------------

2016-09-28T17:37:34Z dmcmorris

Found this while looking into wether or not bindings existed already. Going to learn some tensor flow via the current python API before researching more, but I have built nodejs bindings for C++ libs before and can tell you from experience that swig is the wrong way.

If you simply use the swig bindings then you will have synchronous blocking code in an async environment. The swig bindings do not run things on IO threads, they execute on the main event loop from what I understand/experienced.

Once I have some basic tensor flow experience under my belt I'll likely be interested in building out proper bindings. So if anyone is serious about that and/or wants more details on working with v8 modules, let me know.

-------------------------------------------------------------------------

2016-10-05T11:53:12Z himanshurobo

Is there any link related to node-gyp binding for tensor flow API ??

-------------------------------------------------------------------------

2016-10-05T16:05:05Z marcbaetica

@dmcmorris I am seriously interested in lending a hand! What resources do you recommend for working with v8 modules? We can assemble a team here and start diving into materials asap as this project is way overdue :)

-------------------------------------------------------------------------

2016-10-14T15:13:38Z cedricve

+1

-------------------------------------------------------------------------

2016-10-19T22:34:40Z keyvez

+1

-------------------------------------------------------------------------

2016-10-23T02:46:20Z gastrodia

+1

-------------------------------------------------------------------------

2016-10-31T06:34:36Z willshion

+1s

-------------------------------------------------------------------------

2016-11-04T11:46:02Z himanshurobo

Is there any update ??

On 31-Oct-2016 12:06 pm, "willshion" notifications@github.com wrote:

> +1s

>

> ‚Äî

> You are receiving this because you commented.

> Reply to this email directly, view it on GitHub

> https://github.com/tensorflow/tensorflow/issues/37#issuecomment-257222936,

> or mute the thread

> https://github.com/notifications/unsubscribe-auth/ANrvwrbmShC9c0\_wsNpTudKSCb0-8qt8ks5q5Yx3gaJpZM4Ge5NV

> .

-------------------------------------------------------------------------

2016-11-10T07:53:53Z nikhildaga

Happy Anniversary TensorFlow !

https://research.googleblog.com/2016/11/celebrating-tensorflows-first-year.html

I notice TensorFlow is now accessible from Go, Rust and Haskell. Why ignore JavaScript ?

Really waiting for a machine library in JavaScript.

-------------------------------------------------------------------------

2016-11-17T01:42:04Z newshorts

+1

-------------------------------------------------------------------------

2016-11-23T12:24:38Z slinto

+1

-------------------------------------------------------------------------

2016-12-04T15:36:28Z boehlerlukas

üíØ üëç

-------------------------------------------------------------------------

2016-12-10T13:19:33Z deanshub

+1

-------------------------------------------------------------------------

2016-12-16T13:43:24Z samuelnub

+1.0000000000000000000000001

-------------------------------------------------------------------------

2016-12-16T15:05:44Z ernestasromeika

+1

-------------------------------------------------------------------------

2016-12-17T18:03:57Z madwiki

+1

-------------------------------------------------------------------------

2016-12-20T22:10:48Z mi-g

+1

-------------------------------------------------------------------------

2016-12-22T23:13:32Z Leomund

+1

-------------------------------------------------------------------------

2016-12-23T03:08:10Z stackOverMind

I am looking forward to see a official Node.js API.But I think there are some problems.

1. JavaScript have only 1 thread ,trainning can block the whole process unless using callbacks or other tricks.

2. lack of other science labs, like numpy

3. JavaScript only support 53bit precision.

anyway, JavaScript is awesome!

-------------------------------------------------------------------------

2017-01-05T15:30:07Z jconway45

Is anybody working on this?

-------------------------------------------------------------------------

2017-01-16T07:37:37Z warmhug

Looks very difficult.

-------------------------------------------------------------------------

2017-01-25T15:43:17Z junos

+1

-------------------------------------------------------------------------

2017-01-26T17:27:57Z maxbaluev

very need it!

-------------------------------------------------------------------------

2017-01-30T01:13:24Z 7ammer

@stackOverMind. Did a little search on those bullet points. I've not tried any of these and they might not be efficient to use / run but it looks like there are things that exist to potentially solve those issues.

\* [taking advantage of multi processor environments in node js](http://blog.carbonfive.com/2014/02/28/taking-advantage-of-multi-processor-environments-in-node-js/)

\* [node-lapack](https://github.com/NaturalNode/node-lapack)

\* [numjs](https://github.com/nicolaspanel/numjs)

\* [Long.js](https://github.com/dcodeIO/Long.js)

\* [mljs](https://github.com/mljs)

\* [WebMonkeys - node GPU processing](https://github.com/MaiaVictor/WebMonkeys)

-------------------------------------------------------------------------

2017-02-15T18:29:39Z ryanlangton

+1

-------------------------------------------------------------------------

2017-02-16T02:39:42Z zetos

+1

-------------------------------------------------------------------------

2017-02-16T20:06:32Z mranosa

+1 pweeettyyy pwease!!!

-------------------------------------------------------------------------

2017-02-18T01:33:49Z greenspand

+1

-------------------------------------------------------------------------

2017-02-24T05:59:52Z amorist

+1

-------------------------------------------------------------------------

2017-03-02T07:07:19Z naeplus

Looking forward to it. üëç

-------------------------------------------------------------------------

2017-03-02T13:08:58Z simonmilz

+1

-------------------------------------------------------------------------

2017-03-03T03:05:22Z Pro-YY

Looking forward to it. :+1:

-------------------------------------------------------------------------

2017-03-06T10:21:03Z laverix

üëç

-------------------------------------------------------------------------

2017-03-06T14:14:31Z mpowell90

+1

-------------------------------------------------------------------------

2017-03-06T14:28:39Z Tim15

+1. I have some experience in Node, and will take a look at this.

-------------------------------------------------------------------------

2017-03-07T02:58:49Z longnv-viosoft

+1

-------------------------------------------------------------------------

2017-03-16T22:32:12Z elvismt

+1 Can't wait, Python is great, Node and JS is great too

-------------------------------------------------------------------------

2017-03-27T20:22:55Z snowcxt

+1

-------------------------------------------------------------------------

2017-03-28T19:49:50Z davidawad

+1 just for the sake of it

-------------------------------------------------------------------------

2017-03-29T20:39:13Z gnuns

+1!

-------------------------------------------------------------------------

2017-04-06T10:00:48Z Illyrix

+1

-------------------------------------------------------------------------

2017-04-07T11:06:58Z cattuan

![image](https://cloud.githubusercontent.com/assets/16026183/24797768/f966e5c6-1bbc-11e7-9ce9-32b26607c3c2.png)

-------------------------------------------------------------------------

2017-04-07T17:27:33Z cameronroe

+1 üëç

-------------------------------------------------------------------------

2017-04-11T16:26:25Z land007

+1

-------------------------------------------------------------------------

2017-04-12T09:13:46Z sjovall

+1 would be cool af

-------------------------------------------------------------------------

2017-04-13T19:54:11Z NicolasNarvaez

+1 <3

-------------------------------------------------------------------------

2017-04-14T15:07:56Z kyuwoo-choi

üëç

-------------------------------------------------------------------------

2017-04-15T22:34:35Z George35mk

üëç

-------------------------------------------------------------------------

2017-04-18T16:37:07Z Pummelchen

+100

-------------------------------------------------------------------------

2017-04-22T12:57:09Z danielwatson6

+1

-------------------------------------------------------------------------

2017-04-24T18:51:26Z mtanjir

+1

-------------------------------------------------------------------------

2017-04-28T19:32:36Z zixia

:+1:

-------------------------------------------------------------------------

2017-04-29T08:06:34Z EddieOne

+2

-------------------------------------------------------------------------

2017-05-02T12:57:01Z bogoquiz

+1 ;)

-------------------------------------------------------------------------

2017-05-05T05:45:02Z yeliex

mark

-------------------------------------------------------------------------

2017-05-11T01:21:30Z Czechh

+1

-------------------------------------------------------------------------

2017-05-17T17:28:45Z i-love-code

+1

-------------------------------------------------------------------------

2017-05-18T18:19:01Z coutureJ

+1

-------------------------------------------------------------------------

2017-05-18T19:22:24Z Czechh

+1

-------------------------------------------------------------------------

2017-05-19T17:49:15Z vekexasia

+1

-------------------------------------------------------------------------

2017-05-22T15:53:00Z lsiric

+1

-------------------------------------------------------------------------

2017-05-22T17:47:10Z oas

+1

-------------------------------------------------------------------------

2017-05-23T01:07:59Z sequoiar

+1

-------------------------------------------------------------------------

2017-05-23T13:14:03Z r0ber70

+1

-------------------------------------------------------------------------

2017-05-24T03:14:16Z mrtannerjones

+1

-------------------------------------------------------------------------

2017-05-31T01:30:46Z Bondifrench

:+1:

-------------------------------------------------------------------------

2017-06-01T09:24:36Z jcimoch

+1

-------------------------------------------------------------------------

2017-06-02T03:18:38Z hynra

+1

-------------------------------------------------------------------------

2017-06-02T11:03:25Z skynode

+1

-------------------------------------------------------------------------

2017-06-06T13:33:47Z destromas1

üëç

Please !

-------------------------------------------------------------------------

2017-06-06T22:01:36Z horrigan

+1

-------------------------------------------------------------------------

2017-06-07T00:49:16Z djimoh5

+1

-------------------------------------------------------------------------

2017-06-07T01:03:04Z 7ammer

The OP's date was from 2015, its now 2017 and it's not really been picked up by anyone on the project. I think I might just be getting the hint that nothing will be happening regarding this issue.

Does anyone know if there has been any healthy discussion regarding tensorflow and node anywhere else as these +1's don't seem to be doing much :(

-------------------------------------------------------------------------

2017-06-07T05:15:09Z lynkxyz

+1

-------------------------------------------------------------------------

2017-06-07T20:13:00Z k1ng440

# +1

-------------------------------------------------------------------------

2017-06-09T09:59:41Z DevHossamHassan

üëç

-------------------------------------------------------------------------

2017-06-15T13:48:12Z mx601595686

+1

-------------------------------------------------------------------------

2017-06-18T01:20:47Z beauremus

+1

-------------------------------------------------------------------------

2017-06-25T17:15:33Z ofou

:+1:

-------------------------------------------------------------------------

2017-06-26T22:55:56Z sarkistlt

it's been 2 years and still no luck?

-------------------------------------------------------------------------

2017-06-26T23:03:08Z Riamse

Good. Fuck Javascript.

-------------------------------------------------------------------------

2017-06-27T00:48:27Z chai2010

+1.0/0.0

-------------------------------------------------------------------------

2017-06-27T16:52:05Z George35mk

Use synaptic https://github.com/cazala/synaptic

-------------------------------------------------------------------------

2017-06-27T18:41:51Z tngan

Inspired from [deep\_recommend\_system](https://github.com/tobegit3hub/deep\_recommend\_system), I am now trying another way in order to access the model with Node.js, through tensorflow serving. This is still in progress.

https://github.com/tngan/tensornode

-------------------------------------------------------------------------

2017-06-27T23:06:03Z sarkistlt

this could be useful [keras-js](https://github.com/transcranial/keras-js)

-------------------------------------------------------------------------

2017-06-27T23:55:14Z Bondifrench

@sarkistlt Unfortunately no, we are NOT talking of doing a javascript implementation of Tensorflow, which `keras.js` does, but we are talking having a nodejs \*\*wrapper\*\* around the C++ api of Tensorflow, so it requires understanding how to [write native add-ons for NodeJs](https://blog.risingstack.com/writing-native-node-js-modules/).

Other useful [source](https://github.com/freezer333/nodecpp-demo).

Unfortunately my day job is not coding and it requires proficiency in C++, which I personally don't have.

By the way `keras.js` does only inference, no training, so no backpropagation.

Google developers implemented a small portion of Tensorflow in Javascript in their [playground](https://github.com/tensorflow/playground), the neural network implementation is [here](https://github.com/tensorflow/playground/blob/master/src/nn.ts) and does include [back propagation](https://github.com/tensorflow/playground/blob/master/src/nn.ts#L282).

-------------------------------------------------------------------------

2017-07-02T13:39:06Z JIoJIaJIu

There is I started work on native nodejs Tensorflow implementation, would be great if anybody joins

https://github.com/nodejs-tensorflow/nodejs-tensorflow

-------------------------------------------------------------------------

2017-07-08T02:36:29Z ehhthing

+1

-------------------------------------------------------------------------

2017-07-09T12:24:47Z kimown

+1

-------------------------------------------------------------------------

2017-07-12T11:22:58Z neelkadia

\*\*+1\*\*

-------------------------------------------------------------------------

2017-07-13T09:41:04Z 7elven

+1

-------------------------------------------------------------------------

2017-07-14T02:03:58Z jart

I'm so happy to hear you're giving this a shot @JIoJIaJIu. The potential for impact in solving this issue is huge. It's our most upvoted issue.

At cursory glance, so far you seem to be doing the right thing. You created this in a separate project and are using the TensorFlow C API, as @martinwicke recommended earlier.

A good way to attract contributors to your project would be by sharing a design doc with the [TensorFlow mailing list](https://groups.google.com/a/tensorflow.org/forum/#!forum/discuss), as Vincent [recommended](https://github.com/tensorflow/tensorflow/issues/37#issuecomment-155605035) a few years back. That way we can build consensus around your vision and help it be the best vision possible.

The TensorFlow team wants the NodeJS community to benefit from TensorFlow. So we're absolutely interested in helping the individual devoted to making that happen be successful.

-------------------------------------------------------------------------

2017-07-16T20:19:06Z Loksly

If someone out there is listening, this seems to be the most requested feature for Tensorflow:

https://github.com/tensorflow/tensorflow/issues?utf8=%E2%9C%93&q=is%3Aissue%20is%3Aopen%20sort%3Areactions-%2B1-desc

-------------------------------------------------------------------------

2017-07-17T04:51:29Z nexteve

+1

-------------------------------------------------------------------------

2017-07-19T16:59:57Z yorkie

Hi all, I created the Node.js bridging library for Tensorflow at: https://github.com/yorkie/tensorflow-nodejs without SWIG, it has supported "predefined graph running" and very simple "graph construction", I'm also planing to support more client features in the future :)

At the same time, I would very happy to make this be merged by Tensorflow official group, that would be a zero-cost PR to me :)

-------------------------------------------------------------------------

2017-07-19T17:45:01Z ry

@yorkie It looks interesting, I will try it out! However we cannot merge GPL code into TensorFlow.

-------------------------------------------------------------------------

2017-07-19T18:28:03Z JIoJIaJIu

@yorkie it looks awesome, would you like to join [to the project](https://github.com/nodejs-tensorflow/) and join forces?

-------------------------------------------------------------------------

2017-07-19T19:36:03Z yorkie

@ry sure I can change the license surely :)

@JIoJIaJIu I dunno what's the best place to move this repo for now, if this repo is not suitable for moving tensorflow org, I think nodejs-tensorflow is the good place :)

-------------------------------------------------------------------------

2017-07-19T20:23:04Z yorkie

> However we cannot merge GPL code into TensorFlow.

@ry Updated the license to MIT and @JIoJIaJIu joined the group, thanks for the invitation :)

-------------------------------------------------------------------------

2017-08-02T04:34:49Z gagan-bansal

+1

-------------------------------------------------------------------------

2017-08-17T08:32:36Z qertis

+1

-------------------------------------------------------------------------

2017-08-19T22:52:28Z ohtangza

+1

-------------------------------------------------------------------------

2017-08-20T00:41:06Z sequoiar

+1

-------------------------------------------------------------------------

2017-08-20T20:53:07Z phellipeandrade

+1

-------------------------------------------------------------------------

2017-08-21T23:32:57Z vincentvanhoucke

JavaScript APIs for TensorFlow were [announced](https://research.googleblog.com/2017/08/harness-power-of-machine-learning-in.html) earlier this month. See details on the [deeplearn.js](https://pair-code.github.io/deeplearnjs/) homepage.

I'll close this (broad) bug now. Feel free to open other more specific FRs.

-------------------------------------------------------------------------

2017-08-22T01:43:51Z yeliex

`deeplearn.js` is for browser with webgl, not a sdk for node.js

-------------------------------------------------------------------------

2017-08-22T01:58:57Z vincentvanhoucke

Fair enough. I see now that the bug title references Node.js directly.

-------------------------------------------------------------------------

2017-08-26T15:18:58Z thefill

@vincentvanhoucke it's not even about node.js in the title. Talking about "deeplearn.js" and "Tensorflow API for Javascript" is like talking about apples & pears.

First of all - deeplearn.js is a library that only mirrors to some extent "the style of TensorFlow API" and operates purely in the browser and the other would be a direct API to whole Tensorflow goodness. Also, its not even remotely close to being called an alternative to Tensorflow... maybe for hobbyists but not for commercial use, where one would need clusters of machines to aim the computing process. I think of it as a demo what you can achieve with JavaScript and neural networks... a taste of things to come... ;-)

-------------------------------------------------------------------------

2017-09-01T14:49:51Z robert197

+1

-------------------------------------------------------------------------

2017-09-03T08:08:16Z shahen94

+999

-------------------------------------------------------------------------

2017-09-03T08:12:08Z k1sul1

Jesus christ, stop sending people useless notifications, there's a reason GitHub introduced üëç and üëé reactions.

+1 and +999 just annoys people and adds no value whatsoever.

-------------------------------------------------------------------------

2017-09-03T08:37:27Z rgaidot

+1000

-------------------------------------------------------------------------

2017-09-03T08:37:55Z rgaidot

:+1:

-------------------------------------------------------------------------

2017-09-03T08:44:16Z chanlito

+1

-------------------------------------------------------------------------

2017-09-03T09:45:08Z younesmln

+1

-------------------------------------------------------------------------

2017-09-03T09:50:00Z shahen94

# ATTENTION

üòÑ Guys please before commenting +1 or +whatever - Please take a look at @k1sul1 's comment

> Jesus christ, stop sending people useless notifications, there's a reason GitHub introduced üëç and üëé reactions. +1 and +999 just annoys people and adds no value whatsoever.

-------------------------------------------------------------------------

2017-09-03T09:52:07Z chanlito

@shahen94 we all saw that but still... we are js dev.

-------------------------------------------------------------------------

2017-09-03T10:19:34Z gorjan-mishevski

+1

-------------------------------------------------------------------------

2017-09-03T10:33:37Z thefill

@BruceHem not really sure how being a js dev correlates with blindly pushing unnecessary spam to the feed... üòÑ

You all are aware that "+1" just makes this topic unreadable? I understand that we all have the desire to support this case but can only deduct that on github's closest thing to "vote" functionality is implemented with "reactions" not with a count of comments in the thread... or am I missing something? ;-P

-------------------------------------------------------------------------

2017-09-03T13:54:50Z 7ammer

Lol :P this thread died years ago.

.

+1 Googolplex!

-------------------------------------------------------------------------

2017-09-03T13:57:25Z Alaboudi1

+1

-------------------------------------------------------------------------

2017-09-03T15:30:23Z yorkie

Agreed with @thefill absolutely "+1" just makes this topic unreadable, and actually we had community implementations then if anyone wants to use TensorFlow with Node.js or JavaScript, just have a try with the above one or two, I think this might be a good start than comment votes here.

As for me, I was working on https://github.com/yorkie/tensorflow-nodejs to make it enough functional like Python possible, and it still have such long time to go, but it basically has full implementation for official [language\_bindings](https://www.tensorflow.org/extend/language\_bindings) API. That means you could still use Python to build graph, but load models and run it within a Node.js runtime. Other languages except for Python also can do this currently.

I was also asking for the help from @ry to make my personal repository to be supported officially, there are few things we have to do like building some example models especially RNN cases, but unfortunately I'm got to work on other fields and have no time for these few months, if someone is interested in making this be happened, email to me, I'd love to guide you how to start. Let's the do something useful for community in best wishes :)

-------------------------------------------------------------------------

2017-09-03T16:41:33Z djimoh5

In regards to the current projects that have been started, and specifically the challenges of working with the C API, I have a suggestion on implementation that has worked well for me.

Since python is still the most robust, developer-friendly, and full featured wrapper around the Tensorflow API, rather than trying to "re-create" the python API for js, why not create bindings directly TO the python API? This would still require first creating a node C++ add-on, but rather than binding to the C API directly, you can employ "embedded" python to run python methods directly from C++.

This is unlike other JS => Python solutions out there that suggest simply spawning a python script...a solution not viable for any reasonable size learning problems because of the extensive data transfer cost (time) between the processes. With embedded python however, memory accessed by your python script / numpy arrays directly point to your js Float32Array buffers.

This solution is working very well for me (though admittedly getting the initial js => C++ => python flow working was kind of a pain). Since I have specific needs, I have not gone through the task of binding to each individual python TensorFlow method, and instead just pass my data and hyper parameters to a few methods that build most of the graph. The full individual binding wouldnt be too bad from my current starting point.

I welcome any thoughts or suggestions on the approach outlined above. Thanks.

-------------------------------------------------------------------------

2017-09-03T17:01:40Z yorkie

@djimoh5 Awesome thoughts on JavaScript to Python full-featured APIs! The other hand, we also could put an implementation of a RPC server for TensorFlow Python APIs with introspection feature, so that JavaScript and other language clients could access the real-time Python. (I will do this when I'm available, aha)

But here is something about why re-creating some Python features for JavaScript, because they are written in JavaScript, they are more friend to JavaScript developers, and it's easy for that developers to modify the source code to check if something different is possible, not just the get feeds from upstream :)

-------------------------------------------------------------------------

2017-09-06T14:07:11Z DecafManiac

+1

-------------------------------------------------------------------------

2017-10-19T11:46:19Z khelkun

I'm also interested in nodejs tensorflow API to be able to use it in a node-red flows that would chain tensorflows graphs and may be other kinds of data analysis nodes. I still don't know much about tensorflow. It may not be the right place to ask but I'd like to know why others developers look for a nodejs api/add-on for tensorflow ? What would be your use cases ?

-------------------------------------------------------------------------

2017-10-19T19:16:46Z thefill

@khelkun answer is rather simple: providing mature JavaScript package that allows easy interaction with Tensorflow opens myriad new possibilities. JavaScript operates on every mobile platform, all major desktop Operating systems & in all the browsers so possibilities are endless.

Biggest benefits would come for sure from server-side applications that operate on node.js that could directly interact with Tensorflow, but also node-webkit (desktop applications) could potentially spawn dozens of interesting projects.

-------------------------------------------------------------------------

2017-10-25T22:11:57Z somombo

Are the community organizers /admin of this thread not able to simply delete the posts of the those people that intensionally trolling with all the "+1's‚Äù maybe even ban them? Lol

-------------------------------------------------------------------------

2017-11-18T00:02:50Z lastmjs

Has anyone working on this integration considered using WebAssembly (wasm)? It is potentially the most elegant solution to this problem, side-stepping all of the JS talking to Python talking to C++, you know. I really don't know much about the internals of TensorFlow, but I believe the C++ parts of TensorFlow could be compiled to wasm (check the MVP features supported, but Unreal Engine 4 was compiled to wasm's predecessor and ran successfully in FireFox). Once the C/C++ API is compiled to wasm, you just need to ensure the necessary API is exposed. wasm will run in Node.js, all major browsers, and even outside of any of those, since it is meant to be an extremely portable bytecode. This seems like the best path forward to me.

Related resources/discussion:

\* https://groups.google.com/forum/#!topic/v8-reviews/DjiUKahI6ak

\* https://github.com/tomasreimers/tensorflow-emscripten

-------------------------------------------------------------------------

2017-11-28T19:28:32Z moeiscool

+1... we no longer live in the medieval portion of the information age. Please support node.js.

-------------------------------------------------------------------------

2017-12-12T22:30:34Z AyalaSaenzJorge

Why would anyone need another Javascript library? Why would anybody use a JS library to train NNs? Javascropt is a bad desigbed language.

-------------------------------------------------------------------------

2017-12-12T22:46:25Z somombo

@AyalaSaenzJorge lol (since you're trolling why dont I have at it) ... How about we LOVE "badly" designed languages? Javascript happens to be to most prevalent language currently in existance.. more code (on earth) is written in javascript than ANY other high level language.. And that's a FACT and it's never going away sorry lol

For those of you more serious than this troll Checkout the https://deeplearnjs.org ... It is influenced by tensorflow and backed by Google ... Maybe rather than starting from scratch we may consider porting that to Node.js instead

-------------------------------------------------------------------------

2017-12-12T22:54:28Z humphd

@somombo yes, it looks really interesting. RE: tensorflow + deeplearnjs, see esp:

\* https://deeplearnjs.org/demos/mnist/mnist.html

\* https://github.com/PAIR-code/deeplearnjs/issues/238

\* https://github.com/PAIR-code/deeplearnjs/issues/407

etc

-------------------------------------------------------------------------

2017-12-12T23:43:11Z thefill

@AyalaSaenzJorge this is a place for informative comments, not an opinionated-firestarters.

@somombo please see my comment from 26 Aug where I explain why deeplearnjs is irrelevant to this debate.

-------------------------------------------------------------------------

2017-12-13T01:10:51Z AyalaSaenzJorge

Ok sorry for the comment.

El dic. 12, 2017 6:46 PM, "Filip Dabrowski" <notifications@github.com>

escribi√≥:

> @AyalaSaenzJorge <https://github.com/ayalasaenzjorge> this is a place for

> informative comments, not an opinionated-firestarters.

> @somombo <https://github.com/somombo> please see my comment from 26 Aug

> where I explain why deeplearnjs is irrelevant to this debate.

>

> ‚Äî

> You are receiving this because you were mentioned.

> Reply to this email directly, view it on GitHub

> <https://github.com/tensorflow/tensorflow/issues/37#issuecomment-351232483>,

> or mute the thread

> <https://github.com/notifications/unsubscribe-auth/AeZvKtru1EQjtRnoz6ZR36Kh-DTfVNozks5s\_xBVgaJpZM4Ge5NV>

> .

>

-------------------------------------------------------------------------

2017-12-19T03:45:49Z cpple

+1

-------------------------------------------------------------------------

2017-12-19T04:16:38Z lastmjs

@cpple Remember not to add +1s, they cause noise and have been replaced by reactions. Try giving the first comment a thumbs up

-------------------------------------------------------------------------

2017-12-31T09:23:11Z nikhilk

Just want to share an update -- revamped https://github.com/nikhilk/node-tensorflow with plan to have that support using TensorFlow graphs (and later, saved models) for prediction/inference in node.js. Thought I'd share since a number of folks have expressed interest on this issue.

-------------------------------------------------------------------------

2018-01-04T15:01:16Z dfoody

I've created a fork of headless-gl that works with deeplearnjs (which in turn works with tensorflow) - this allows models to be run natively on the GPU from node.js (note that it's only been tested on OSX so far).

You can find the install directions and a basic sample at https://github.com/dfoody/headless-gl

And, of course https://deeplearnjs.org for more details.

General directions to install on OSX:

```

brew install pkg-config cairo pango libpng jpeg giflib

npm install deeplearn-gl

npm install deeplearn

```

And a quick sample to show how it's used together with deeplearnjs:

```

var gl = require('deeplearn-gl')

var dl = require('deeplearn');

var math = new dl.NDArrayMath('webgl');

var size = 3000;

var start = Date.now();

var a = dl.Array2D.ones([size, size]);

var b = dl.Array2D.ones([size, size]);

var result = math.matMul(a, b);

var promise = result.data().then(function(data) {

console.log("done in ", (Date.now()-start)/1000, "seconds with value", data[0]);

}).catch(console.log);

```

-------------------------------------------------------------------------

2018-01-04T18:57:10Z thefill

@dfoody thank you for sharing this with the community but the statement "which in turn works with tensorflow" is incorrect. Deeplearn.js just provide (quote from their website) "execution model mirroring the TensorFlow API" - that's all it has common with TF project, I'm afraid;-(

Also please see my comment from 26 Aug where I explain why deeplearnjs is irrelevant to this debate.

@nikhilk amazing, keep on going! I will keep an eye on your project for sure ;-D

-------------------------------------------------------------------------

2018-02-26T02:03:45Z LinBoLen

+1

-------------------------------------------------------------------------

2018-03-01T19:57:36Z 7immer

[propelml.org](http://propelml.org) - Looks interesting. I've not used it but its GPU enabled and runs in both the browser and on node

-------------------------------------------------------------------------

2018-03-06T22:32:05Z thefill

@7ammer propelml.org looks rather promising. Thanks for sharing this with us ;-)

-------------------------------------------------------------------------

2018-03-24T08:42:16Z torianBr

Because NodeJS it's fast! ;D

-------------------------------------------------------------------------

2018-03-24T23:45:04Z jart

If an ambitious member of the community wants the glory of solving this problem, and having it merged into the TensorFlow contrib codebase, here are some tips on how I would do it. Please note I'm not going to do this.

You can add Node to [workspace.bzl](https://github.com/tensorflow/tensorflow/blob/master/tensorflow/workspace.bzl) just like TensorBoard did in [js.bzl](https://github.com/tensorflow/tensorboard/blob/99a7437/third\_party/js.bzl#L25).

Please note TensorFlow can not depend on [rules\_nodejs](https://github.com/bazelbuild/rules\_nodejs).

```py

load("@io\_bazel\_rules\_closure//closure:defs.bzl", "filegroup\_external")

filegroup\_external(

name = "org\_nodejs",

# MIT with portions licensed:

# - MIT

# - Old MIT

# - 2-Clause-BSD

# - 3-Clause-BSD

# - ISC

# - Unicode

# - zlib

# - Artistic 2.0

licenses = ["notice"],

sha256\_urls\_extract\_macos = {

"910395e1e98fb351c62b5702a9deef22aaecf05d6df1d7edc283337542207f3f": [

"https://mirror.bazel.build/nodejs.org/dist/v6.9.1/node-v6.9.1-darwin-x64.tar.xz",

"http://nodejs.org/dist/v6.9.1/node-v6.9.1-darwin-x64.tar.xz",

],

},

sha256\_urls\_windows = {

"1914bfb950be8d576ce9e49c8a0e51c9f2402560fe3c19093e69bc1306a56e9e": [

"https://mirror.bazel.build/raw.githubusercontent.com/nodejs/node/v6.9.1/LICENSE",

"https://raw.githubusercontent.com/nodejs/node/v6.9.1/LICENSE",

],

"513923b0490ebb7466a56483a62595814ed9d036d6f35476debb0cd606bec526": [

"https://mirror.bazel.build/nodejs.org/dist/v6.9.1/win-x64/node.exe",

"http://nodejs.org/dist/v6.9.1/win-x64/node.exe",

],

"3951aefa4afd6fb836ab06468b1fc2a69fa75bd66ec2f5a0e08c4e32547681e3": [

"https://mirror.bazel.build/nodejs.org/dist/v6.9.1/win-x64/node.lib",

"http://nodejs.org/dist/v6.9.1/win-x64/node.lib",

],

},

sha256\_urls\_extract = {

"d4eb161e4715e11bbef816a6c577974271e2bddae9cf008744627676ff00036a": [

"https://mirror.bazel.build/nodejs.org/dist/v6.9.1/node-v6.9.1-linux-x64.tar.xz",

"http://nodejs.org/dist/v6.9.1/node-v6.9.1-linux-x64.tar.xz",

],

},

strip\_prefix = {

"node-v6.9.1-darwin-x64.tar.xz": "node-v6.9.1-darwin-x64",

"node-v6.9.1-linux-x64.tar.xz": "node-v6.9.1-linux-x64",

},

executable = [

"node",

"node.exe",

],

default\_visibility = ["//tensorflow/contrib/node:\_\_subpackages\_\_"],

)

```

Now let's say you want you have a Node program, e.g. [tsc.js](https://github.com/tensorflow/tensorboard/blob/99a7437/third\_party/js.bzl#L73), which you want to turn into something you can `bazel run //tensorflow/contrib/node:generate`. One quick way you could do this in Bazel is by defining a macro in `tensorflow/contrib/node/defs.bzl`:

```py

def node\_binary(name, srcs, data=None, visibility=None, testonly=None, \*\*kwargs):

native.sh\_binary(

name = name,

srcs = [name + ".sh"],

data = srcs + data + ["@org\_nodejs"],

testonly = testonly,

visibility = visibility,

\*\*kwargs

)

native.genrule(

name = name + "\_sh",

srcs = [srcs[0]],

outs = [name + ".sh"],

cmd = "cat >$@ <<'EOF'\n" +

"#!/bin/bash\n" +

"NODE=external/org\_nodejs/bin/node\n" +

"if [[ -e external/org\_nodejs/node.exe ]]; then\n" +

" NODE=external/org\_nodejs/node.exe\n" +

"fi\n" +

"exec $${NODE} $(location " + srcs[0] + ") \"$$@\"\n" +

"EOF",

executable = True,

testonly = testonly,

visibility = ["//visibility:private"],

)

```

Now for the fun part. I would write a single .js file (even if it had to be 30,000 lines long like [tex.web](http://tug.org/texlive/devsrc/Build/source/texk/web2c/tex.web)) with zero dependencies other than the Node standard library. The inputs for this program would be [ops.pbtxt](https://github.com/tensorflow/tensorflow/blob/master/tensorflow/core/ops/ops.pbtxt) and all the other pbtxt files in [api\_def/base\_api](https://github.com/tensorflow/tensorflow/tree/master/tensorflow/core/api\_def/base\_api). The output to this program would be exactly one gigantic C++ file that talks to [TensorFlow C API](https://github.com/tensorflow/tensorflow/blob/master/tensorflow/c/c\_api.h) and [Node C++ Addon API](https://nodejs.org/api/addons.html) based on [this example](https://github.com/nodejs/node-addon-examples/tree/master/1\_hello\_world/node\_0.12).

```py

load("//tensorflow/contrib/node:defs.bzl", "node\_binary")

load("@domain\_registry//java/google/registry/builddefs:zip\_file.bzl", "zip\_file")

node\_binary(

name = "generate",

srcs = ["generate.js"],

data = [

"//tensorflow/core:ops/ops.pbtxt",

"//tensorflow/core/api\_def:base\_api\_def",

],

)

genrule(

name = "api",

srcs = [

"//tensorflow/core:ops/ops.pbtxt",

"//tensorflow/core/api\_def:base\_api\_def",

],

cmd = "$(location :generate) $(location api.cc) $(SRCS)",

outs = ["api.cc"],

tools = [":generate"],

)

zip\_file(

name = "tfnode",

srcs = [

"package.json",

"README.md",

"api.cc",

"binding.gyp",

"tfnode.js",

],

mappings = {"tensorflow/contrib/node": "package"},

)

```

Then you `bazel build //tensorflow/contrib/node:tfnode.zip` and bam you've got your NodeJS project all bundled and ready for distribution to places like NPM.

If I wrote this (which I won't) it would be a barebones direct mapping of the TensorFlow API definitions. Then I would encourage our friends in the community to veneer the library. There's a diversity of visions out there on friendly modern high-level idiomatic JS and ML APIs, each catering to different use cases. However they could all share this binding in common.

Please note there are examples of where we already generate language bindings. See [tensorflow/go/genop/main.go](https://github.com/tensorflow/tensorflow/blob/master/tensorflow/go/genop/main.go) and [tensorflow/go/op/generate.go](https://github.com/tensorflow/tensorflow/blob/master/tensorflow/go/op/generate.go) for inspiration.

-------------------------------------------------------------------------

2018-03-30T17:16:20Z lastmjs

Looks like the TensorFlow team is making this a top priority now: https://js.tensorflow.org/faq/

-------------------------------------------------------------------------

2018-03-31T07:22:51Z lastmjs

We might want to move this discussion to here: https://github.com/tensorflow/tfjs/issues/36

Progress on Node.js bindings to the C API will be tracked at that issue.

-------------------------------------------------------------------------

2018-04-26T21:52:41Z nkreeger

As update to this issue - we have open-sourced the Node.js binding for TFJS: https://github.com/tensorflow/tfjs-node

We are working hard at getting a proper NPM build and will release it soon!

-------------------------------------------------------------------------

2018-04-26T22:35:51Z martinwicke

I will close this issue. Please track tensorflow/tfjs and tensorflow/tfjs-node for further updates.

-------------------------------------------------------------------------

2018-05-08T17:37:13Z nuchi

Related and possibly of interest: I managed to get TF running in the browser via Webassembly. See https://humantoanimal.com for a demo; I will be providing more details in the future.

-------------------------------------------------------------------------

2018-05-08T17:58:47Z lastmjs

@nuchi, so did you compile the necessary TensorFlow code from the C API to WebAssembly? Or are you using TensorFlow.js?

-------------------------------------------------------------------------

2018-05-08T18:34:55Z nuchi

@lastmjs I explain in more detail in the link I provided. Short version: I added Webassembly as an XLA compilation target. I did not use Tensorflow.js in any way.

-------------------------------------------------------------------------

2018-05-09T03:59:24Z zixia

@nuchi Great work! and I know another WebAssemble research on TensorFlow at here:

https://medium.com/@tomasreimers/compiling-tensorflow-for-the-browser-f3387b8e1e1c

-------------------------------------------------------------------------

2018-06-07T20:09:15Z rchipka

Glad to see that there's official progress on this. I'd love to have fast, parallel GPU compute power at my fingertips with the ease and composability of JS.

I started working on a [NodeJS binding for TensorFlow](https://github.com/rchipka/node-tensorflow) a while ago, but a haven't had much free time to devote to it lately.

The concept is similar to @jart's suggested approach.

I had three goals in mind for the project:

#### 1. Don't require building or installing tensorflow

Instead, it should [download and use](https://github.com/rchipka/node-tensorflow/blob/master/scripts/install.js) the pre-built, multi-platform python binaries and download any needed source files on the fly.

#### 2. Don't require a complete C++ or JS reproduction or abstraction of the API

Instead, it should provide a complete 1-to-1 interface with the C API, [providing convenient JS abstractions](https://github.com/rchipka/node-tensorflow/tree/master/lib) as much as possible.

#### 3. Don't maintain the C API [bindings](https://github.com/tensorflow/tfjs-node/blob/master/binding/tfjs\_binding.cc) [by](https://github.com/tensorflow/tfjs-node/blob/master/binding/tfe\_utils.cc) [hand](https://github.com/tensorflow/tfjs-node/blob/master/binding/tfjs\_backend.cc)

Instead, it should [use a swig script](https://github.com/rchipka/node-tensorflow/blob/master/src/tensorflow.i) to map the core data structures between Tensorflow/stdc++/V8/node and the rest will follow.

---

I got pretty far along with this, but last I remember a was having issues with TF\_Session related segfaults.

Right now it's just collecting dust, so if someone wants to jump in and help with this I'd gladly accept PRs.

-------------------------------------------------------------------------

TF\_Session related segfaults.

Right now it's just collecting dust, so if someone wants to jump in and help with this I'd gladly accept PRs.

-------------------------------------------------------------------------