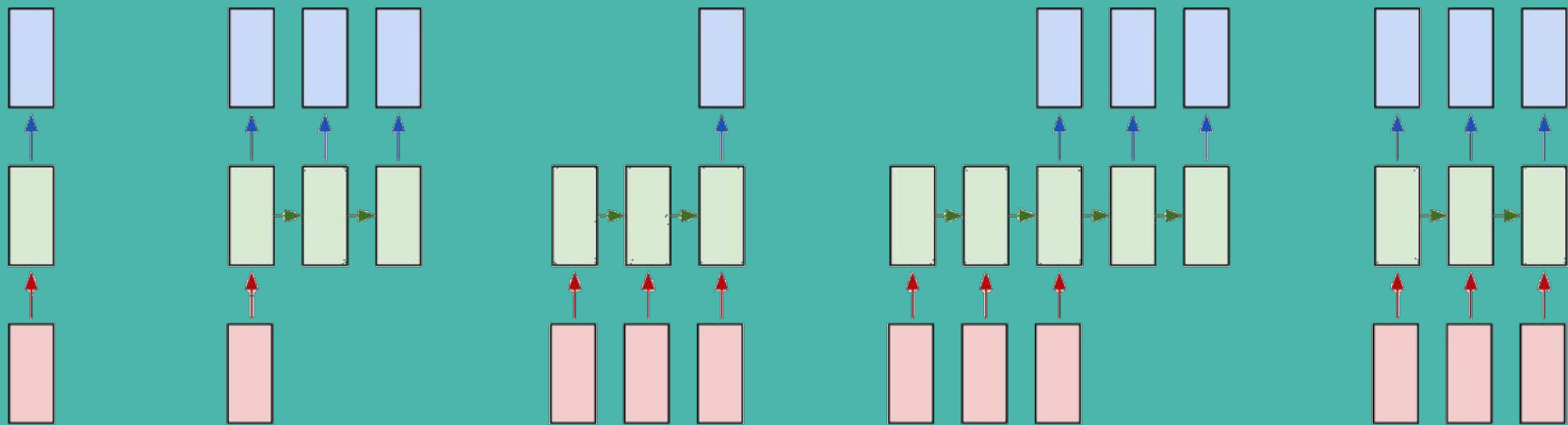
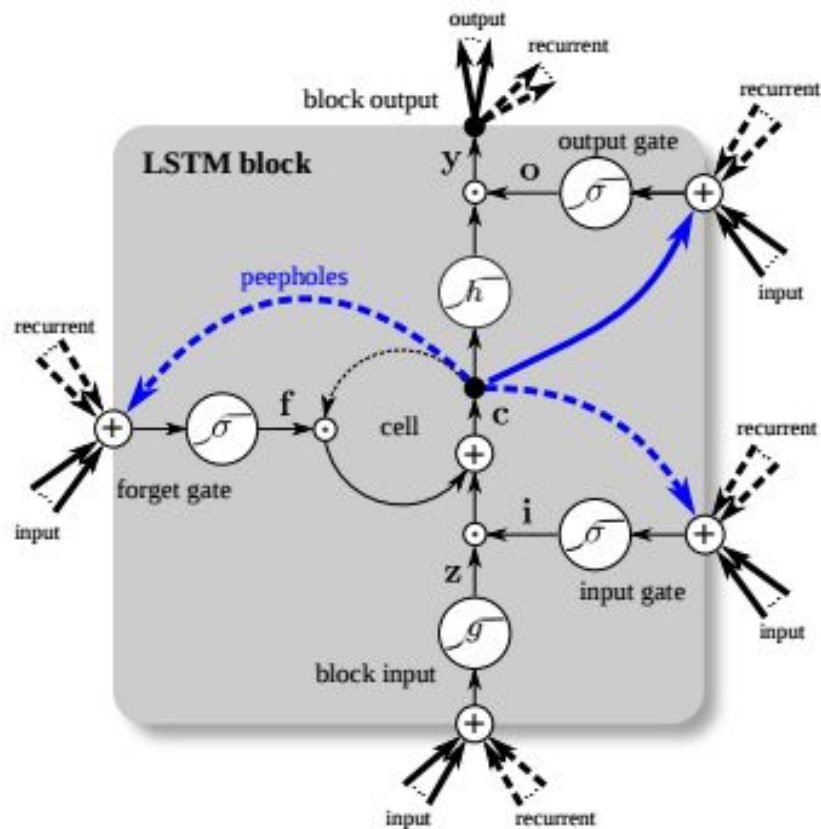


Recurrent Neural Networks



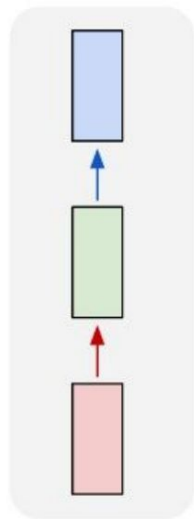


Legend

- unweighted connection
- weighted connection
- connection with time-lag
- branching point
- ⊙ multiplication
- ⊕ sum over all inputs
- σ gate activation function (always sigmoid)
- g input activation function (usually tanh)
- h output activation function (usually tanh)

RNN designs/architectures

one to one



Vanilla

one to many

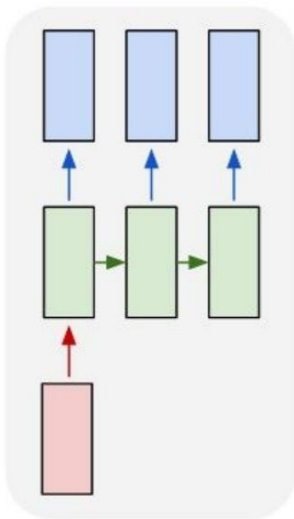
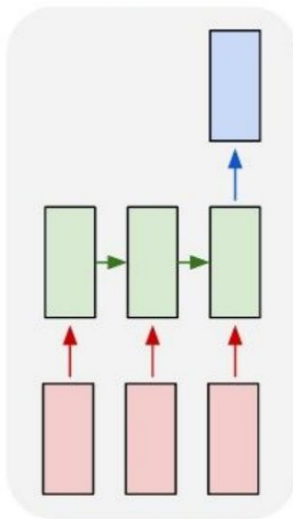


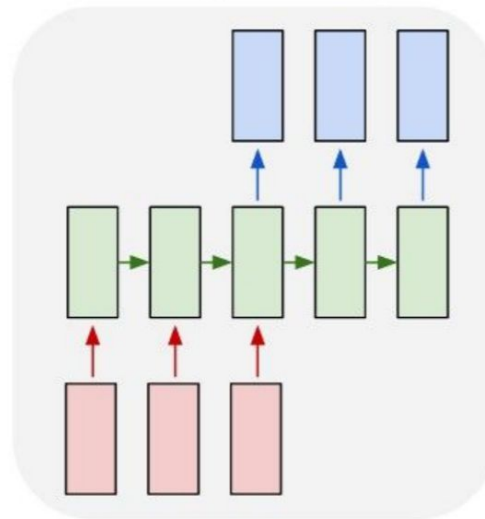
Image Captioning
Sequence of words

many to one



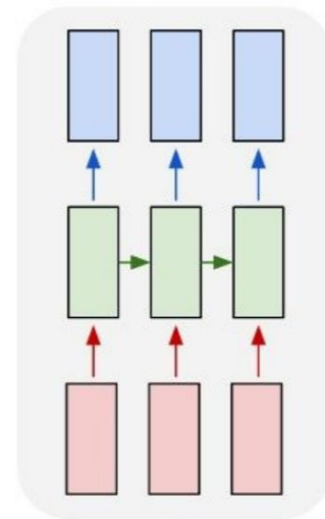
Sentiment Classification
Words -> Sentiment

many to many



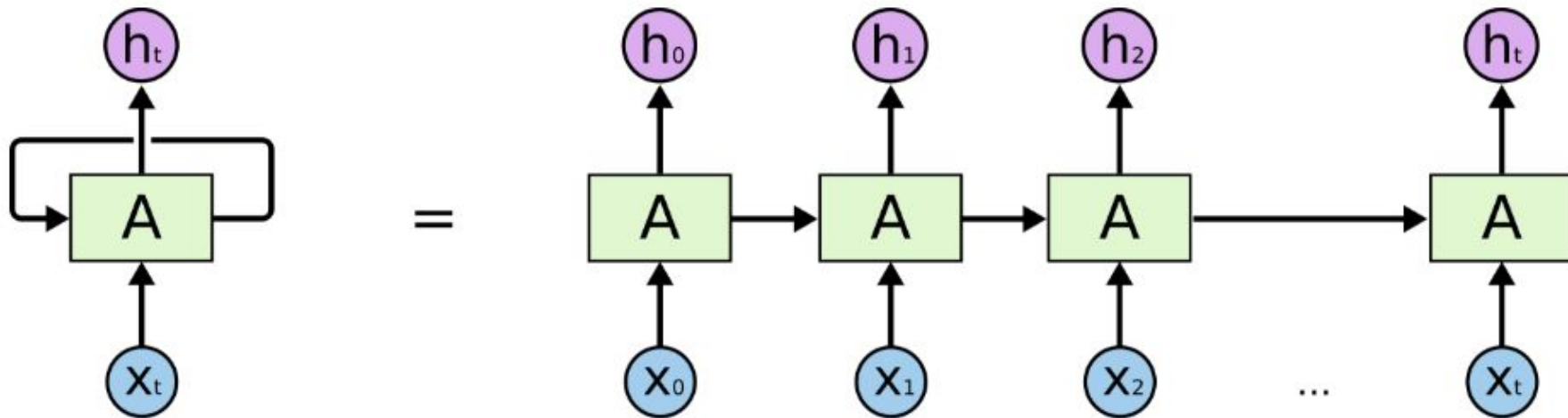
Translation
seq of words -> seq of words

many to many



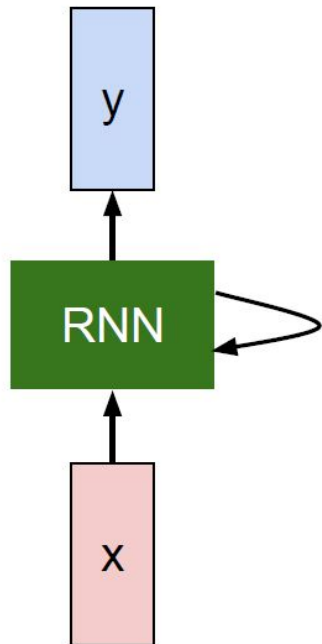
Video classification
frame level

Unrolling a recurrent neural network



(Vanilla) Recurrent Neural Network

The state consists of a single “hidden” vector h :



$$h_t = f_W(h_{t-1}, x_t)$$



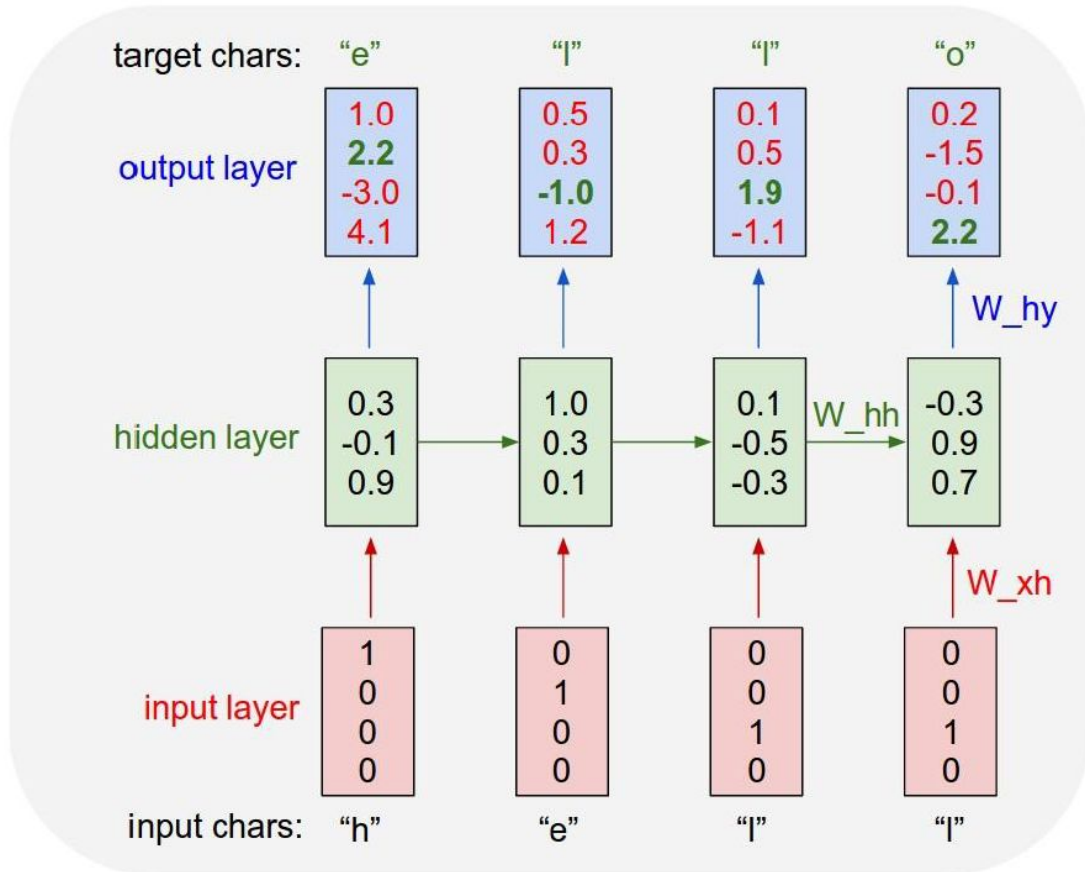
$$h_t = \tanh(W_{hh}h_{t-1} + W_{xh}x_t)$$

$$y_t = W_{hy}h_t$$

Character-level language model example

Vocabulary:
[h,e,l,o]

Example training sequence:
“hello”



* http://cs231n.stanford.edu/slides/winter1516_lecture10.pdf

“The unreasonable effectiveness of RNNs” -karpathy

PANDARUS:

Alas, I think he shall be come approached and the day
When little strain would be attain'd into being never fed,
And who is but a chain and subjects of his death,
I should not sleep.

Second Senator:

They are away this miseries, produced upon my soul,
Breaking and strongly should be buried, when I perish
The earth and thoughts of many states.

DUKE VINCENTIO:

Well, your wit is in the care of side and that.

Second Lord:

They would be ruled after this chamber, and
my fair nudes begun out of the fact, to be conveyed,
Whose noble souls I'll have the heart of the wars.

Clown:

Come, sir, I will make did behold your worship.

VIOLA:

I'll drink it.

Character level
RNN

Trained on
Shakespeare

VIOLA:

Why, Salisbury must find his flesh and thought
That which I am not apt, not a man and in fire,
To show the reining of the raven and the wars
To grace my hand reproach within, and not a fair are hand,
That Caesar and my goodly father's world;
When I was heaven of presence and our fleets,
We spare with hours, but cut thy council I am great,
Murdered and by thy master's ready there
My power to give thee but so much as hell:
Some service in the noble bondman here,
Would show him to her wine.

KING LEAR:

O, if you were a feeble sight, the courtesy of your law,
Your sight and several breath, will wear the gods
With his heads, and my hands are wonder'd at the deeds,
So drop upon your lordship's head, and your opinion
Shall be against your honour.

Trained on wikipedia

Naturalism and decision for the majority of Arab countries' capitalide was groundred by the Irish language by [[John Clair]], [[An Imperial Japanese Revolt]], associated with Guangzham's sovereignty. His generals were the powerful ruler of the Portugal in the [[Protestant Immineners]], which could be said to be directly in Cantonese Communication, which followed a ceremony and set inspired prison, training. The emperor travelled back to [[Antioch, Perth, October 25|21]] to note, the Kingdom of Costa Rica, unsuccessful fashioned in western [[Scotland]], near Italy t Copyright was the succession of indep was a famous German movement based on and sexual power post. Many governmen [[Civil Liberalization and Infantry R that is sympathetic to be to the [[Pu (PJS)[http://www.humah.yahoo.com/guar cfm/7754800786d17551963s89.htm Offici was swear to advance to the resources was starting to signing a major tripa

```
{ { cite journal | id=Cerling Nonforest Department|format=Newlymeslated|none } }  
'www.e-complete'.
```

```
'''See also''' : [[List of ethical consent processing]]
```

```
== See also ==  
*[[Iender dome of the ED]]  
*[[Anti-autism]]
```

```
=== [[Religion|Religion]] ===  
*[[French Writings]]  
*[[Maria]]  
*[[Revelation]]  
*[[Mount Agamul]]
```

```
== External links ==  
* [http://www.biblegateway.nih.gov/entrepre/ Website of the
```

```
==External links==  
* [http://www.romanology.com/ Constitution of the Netherlands and Hispanic Competition fo
```

```
<page>  
<title>Antichrist</title>  
<id>865</id>  
<revision>  
  <id>15900676</id>  
  <timestamp>2002-08-03T18:14:12Z</timestamp>  
  <contributor>  
    <username>Paris</username>  
    <id>23</id>  
  </contributor>  
  <minor />  
  <comment>Automated conversion</comment>  
  <text xml:space="preserve">#REDIRECT [[Christianity]]</text>  
</revision>  
</page>
```


Training on the entire linux source code

```
/*
 * Increment the size file of the new incorrect UI_FILTER group information
 * of the size generatively.
 */
static int indicate_policy(void)
{
    int error;
    if (fd == MARN_EPT) {
        /*
         * The kernel blank will coeld it to userspace.
         */
        if (ss->segment < mem_total)
            unblock_graph_and_set_blocked();
        else
            ret = 1;
        goto bail;
    }
    segaddr = in_SB(in.addr);
    selector = seg / 16;
    setup_works = true;
    for (i = 0; i < blocks; i++) {
        seq = buf[i++];
        bpf = bd->bd.next + i * search;
        if (fd) {
            current = blocked;
        }
    }
    rw->name = "Getjbbregs";
    bprm_self_clearl(&iv->version);
    regs->new = blocks[(BPF_STATS << info->historidac) | PFMR_CLOBATHINC_SECONDS << 12;
    return segtable;
}
```

```
/*
 * Copyright (c) 2006-2010, Intel Mobile Communications. All rights reserved.
 *
 * This program is free software; you can redistribute it and/or modify it
 * under the terms of the GNU General Public License version 2 as published by
 * the Free Software Foundation.
 *
 * This program is distributed in the hope that it will be useful,
 * but WITHOUT ANY WARRANTY; without even the implied warranty of
 * MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE. See the
 * GNU General Public License for more details.
 *
 * You should have received a copy of the GNU General Public License
 * along with this program; if not, write to the Free Software Foundation,
 * Inc., 675 Mass Ave, Cambridge, MA 02139, USA.
 */

#include <linux/kexec.h>
#include <linux/errno.h>
#include <linux/io.h>
#include <linux/platform_device.h>
#include <linux/multi.h>
#include <linux/kevent.h>

#include <asm/io.h>
#include <asm/prom.h>
#include <asm/e820.h>
#include <asm/system_info.h>
#include <asm/setw.h>
#include <asm/pgproto.h>

#define REG_PG vespa_slot_addr_pack
#define PFH_NOCOMP AFSR(0, load)
#define STACK_DDR(type) (func)

#define SWAP_ALLOCATE(nr) (e)
#define emulate_sigs() arch_get_unaligned_child()
#define access_rw(TST) asm volatile("movd %!esp, %0, %3" : "r" (0)); \
    if (_type & DO_READ)

static void stat_PC_SEC __read_mostly offsetof(struct seq_argqueue, \
    pC[1]);

static void
os_prefix(unsigned long sys)
{
    #ifdef CONFIG_PREEMPT
        PUT_PARAM_RAID(2, sel) = get_state_state();
        set_pid_sum((unsigned long)state, current_state_str(),
            (unsigned long)-1->lr_full; low;
    }
}
```

Visualizing the RNN

Cell sensitive to position in line:

The sole importance of the crossing of the Berezina lies in the fact that it plainly and indubitably proved the fallacy of all the plans for cutting off the enemy's retreat and the soundness of the only possible line of action--the one Kutuzov and the general mass of the army demanded--namely, simply to follow the enemy up. The French crowd fled at a continually increasing speed and all its energy was directed to reaching its goal. It fled like a wounded animal and it was impossible to block its path. This was shown not so much by the arrangements it made for crossing as by what took place at the bridges. When the bridges broke down, unarmed soldiers, people from Moscow and women with children who were with the French transport, all--carried on by vis inertiae--pressed forward into boats and into the ice-covered water and did not, surrender.

Cell that turns on inside quotes:

"You mean to imply that I have nothing to eat out of.... On the contrary, I can supply you with everything even if you want to give dinner parties," warmly replied Chichagov, who tried by every word he spoke to prove his own rectitude and therefore imagined Kutuzov to be animated by the same desire.

Kutuzov, shrugging his shoulders, replied with his subtle penetrating smile: "I meant merely to say what I said."

Cell that robustly activates inside if statements:

```
static int __dequeue_signal(struct sigpending *pending, sigset_t *mask,
                           siginfo_t *info)
{
    int sig = next_signal(pending, mask);
    if (sig) {
        if (current->notifier) {
            if (sigismember(current->notifier_mask, sig)) {
                if (!(current->notifier)(current->notifier_data)) {
                    clear_thread_flag(TIF_SIGPENDING);
                    return 0;
                }
            }
        }
        collect_signal(sig, pending, info);
    }
    return sig;
}
```

Color indicates the degree of activation of a specific, single cell in the network over the timesteps (per character)

Visualizing the RNN

A large portion of cells are not easily interpretable. Here is a typical example:

```
/* Unpack a filter field's string representation from user-space
 * buffer. */
char *audit_unpack_string(void **bufp, size_t *remain, size_t len)
{
    char *str;
    if (!*bufp || (len == 0) || (len > *remain))
        return ERR_PTR(-EINVAL);
    /* Of the currently implemented string fields, PATH_MAX
     * defines the longest valid length.
     */
}
```

Cell that turns on inside comments and quotes:

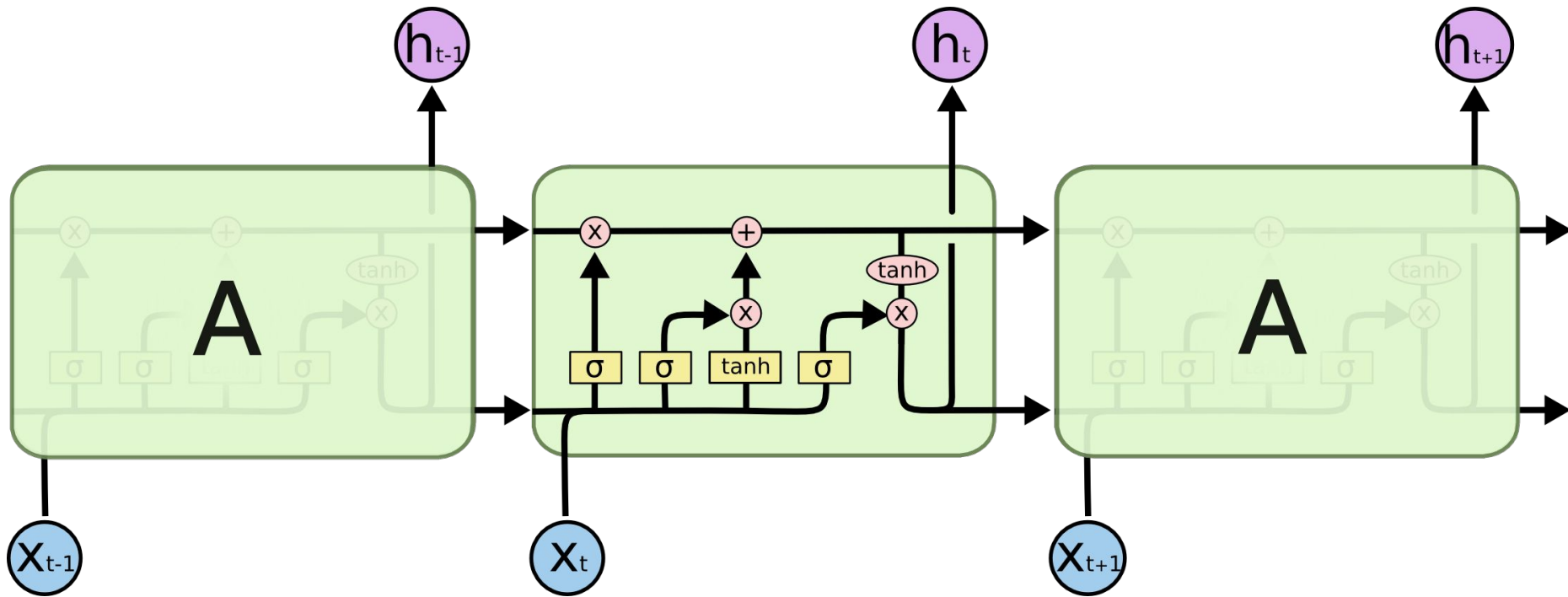
```
/* Duplicate LSM field information. The lsm_rule is opaque, so
 * re-initialized. */
static inline int audit_dup_lsm_field(struct audit_field *df,
    struct audit_field *sf)
{
    int ret = 0;
    char *lsm_str;
    /* our own copy of lsm_str */
    lsm_str = kstrdup(sf->lsm_str, GFP_KERNEL);
    if (unlikely(!lsm_str))
        return -ENOMEM;
    df->lsm_str = lsm_str;
    /* our own (refreshed) copy of lsm_rule */
    ret = security_audit_rule_init(df->type, df->op, df->lsm_str,
        (void *) &df->lsm_rule);
    /* Keep currently invalid fields around in case they
     * become valid after a policy reload. */
    if (ret == -EINVAL) {
        pr_warn("audit rule for LSM '%s' is invalid\n",
            df->lsm_str);
        ret = 0;
    }
    return ret;
}
```

Cell that is sensitive to the depth of an expression:

```
#ifdef CONFIG_AUDITSYSCALL
static inline int audit_match_class_bits(int class, u32 *mask)
{
    int i;
    if (classes[class]) {
        for (i = 0; i < AUDIT_BITMASK_SIZE; i++)
            if (mask[i] & classes[class][i])
                return 0;
    }
    return 1;
}
```

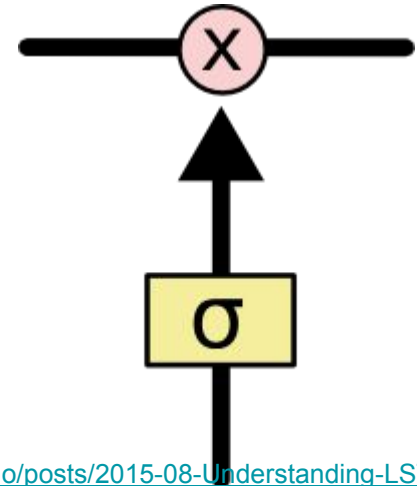
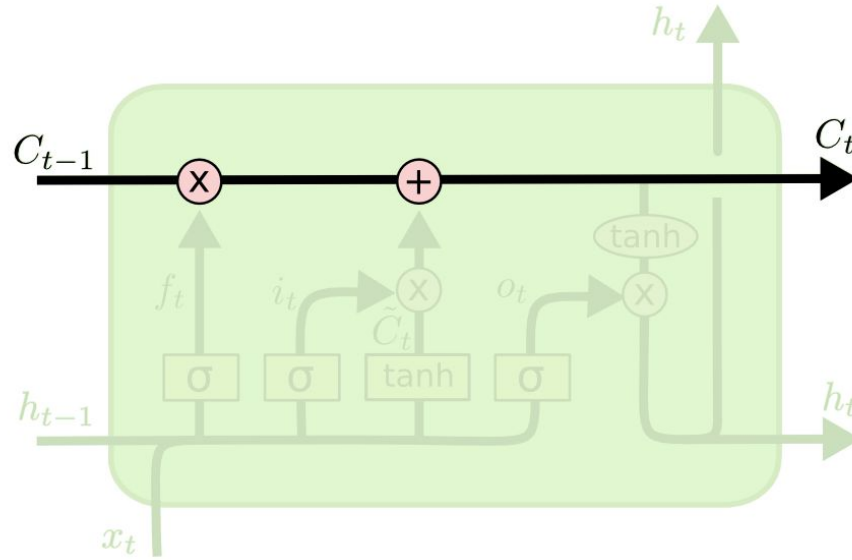
Color indicates the degree of activation of a specific, single cell in the network over the timesteps (per character)

LSTMs

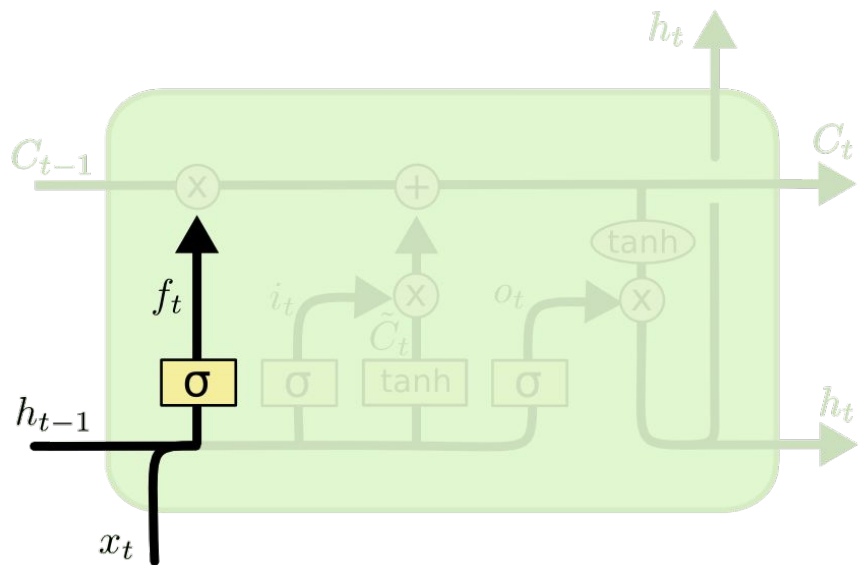


LSTM: Cell State

$$\begin{bmatrix} x_1 \\ x_2 \\ x_3 \\ x_4 \\ x_5 \\ \vdots \\ x_n \end{bmatrix}$$

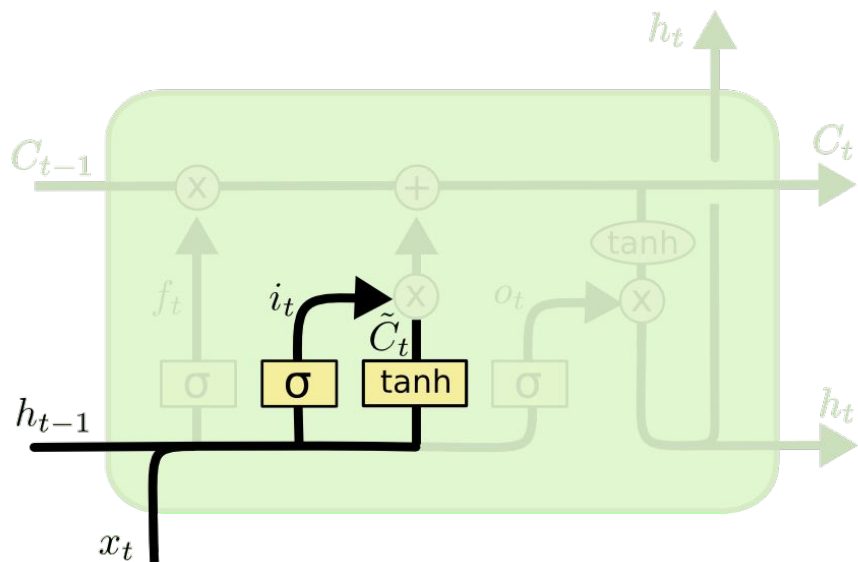


LSTM: Forget Gate



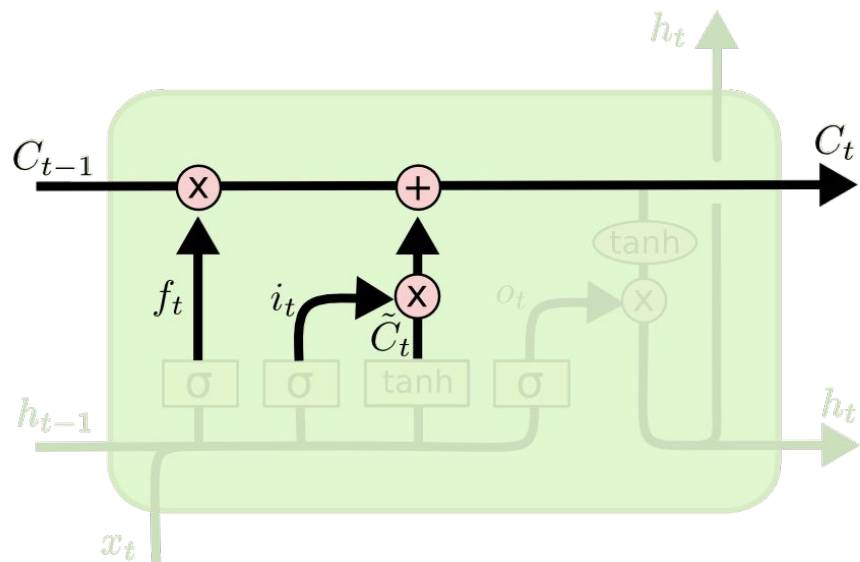
$$f_t = \sigma (W_f \cdot [h_{t-1}, x_t] + b_f)$$

LSTM: Cell State (aka Memory)



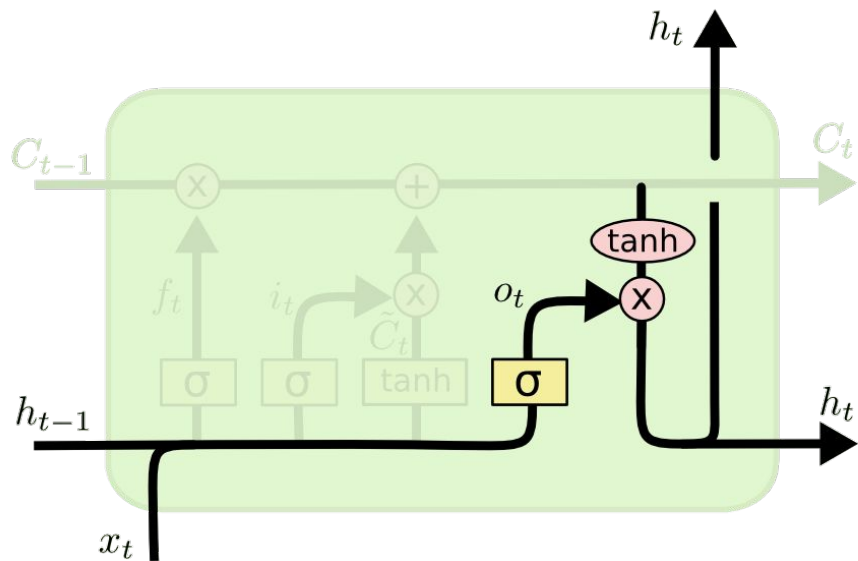
$$i_t = \sigma(W_i \cdot [h_{t-1}, x_t] + b_i)$$
$$\tilde{C}_t = \tanh(W_C \cdot [h_{t-1}, x_t] + b_C)$$

LSTM: Update Cell State (aka Memory)



$$C_t = f_t * C_{t-1} + i_t * \tilde{C}_t$$

LSTM: Output (“hidden” state & result)

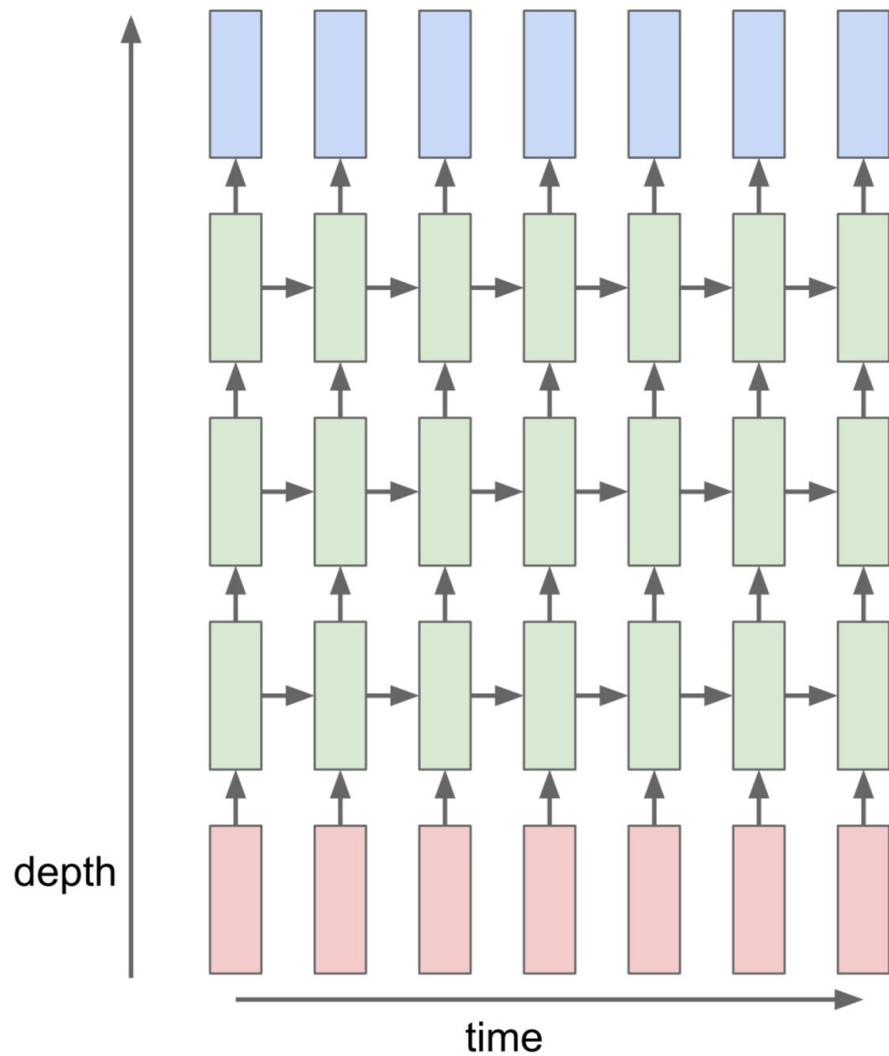


$$o_t = \sigma (W_o [h_{t-1}, x_t] + b_o)$$

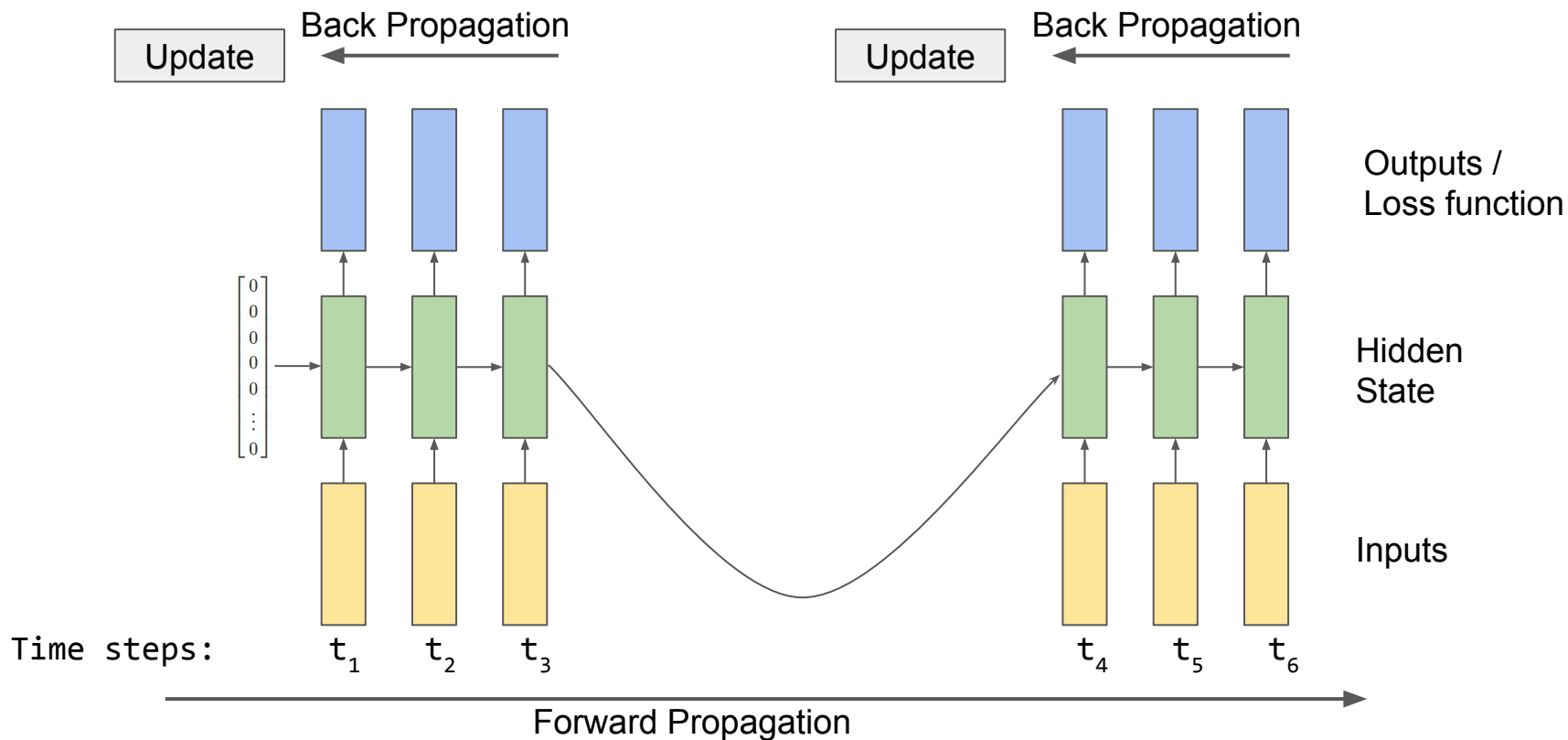
$$h_t = o_t * \tanh (C_t)$$

Deep RNNs

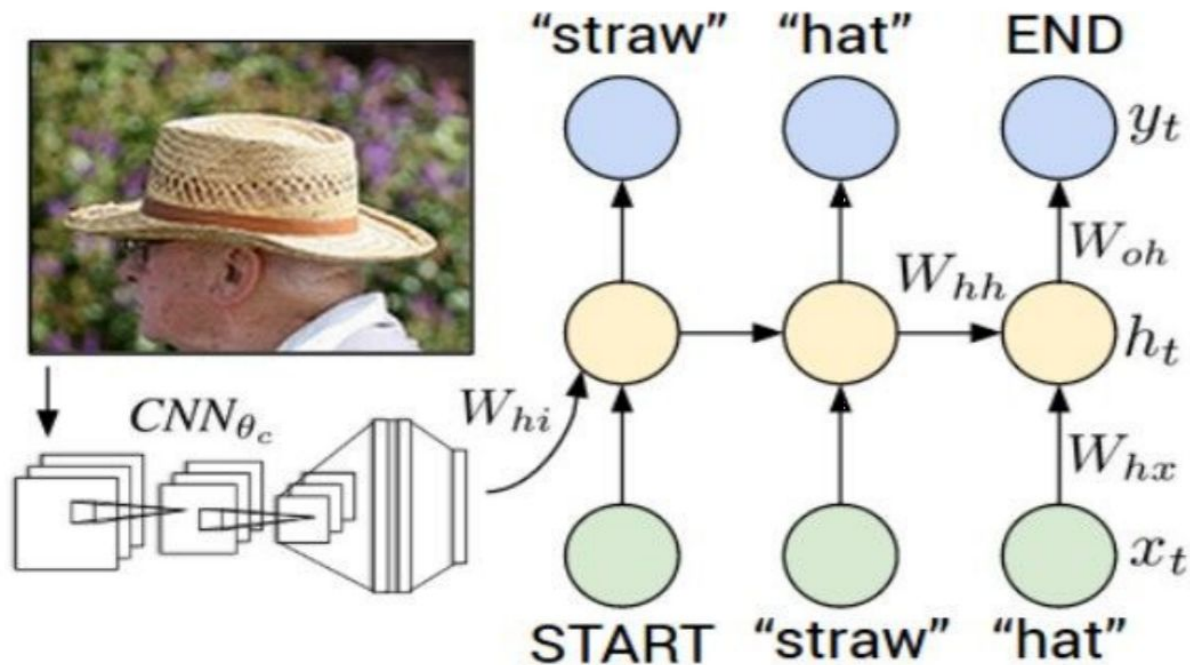
A 3 layer deep RNN is not uncommon, GRUs start to show their value now



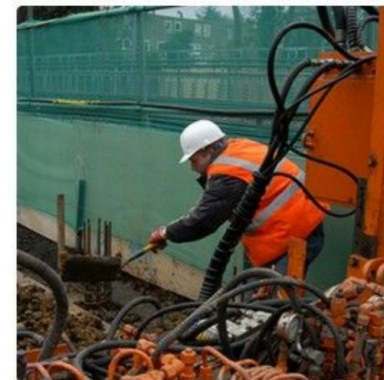
BPTT: Limit yourself to a batch of time steps



RNNs integrated with CNNs

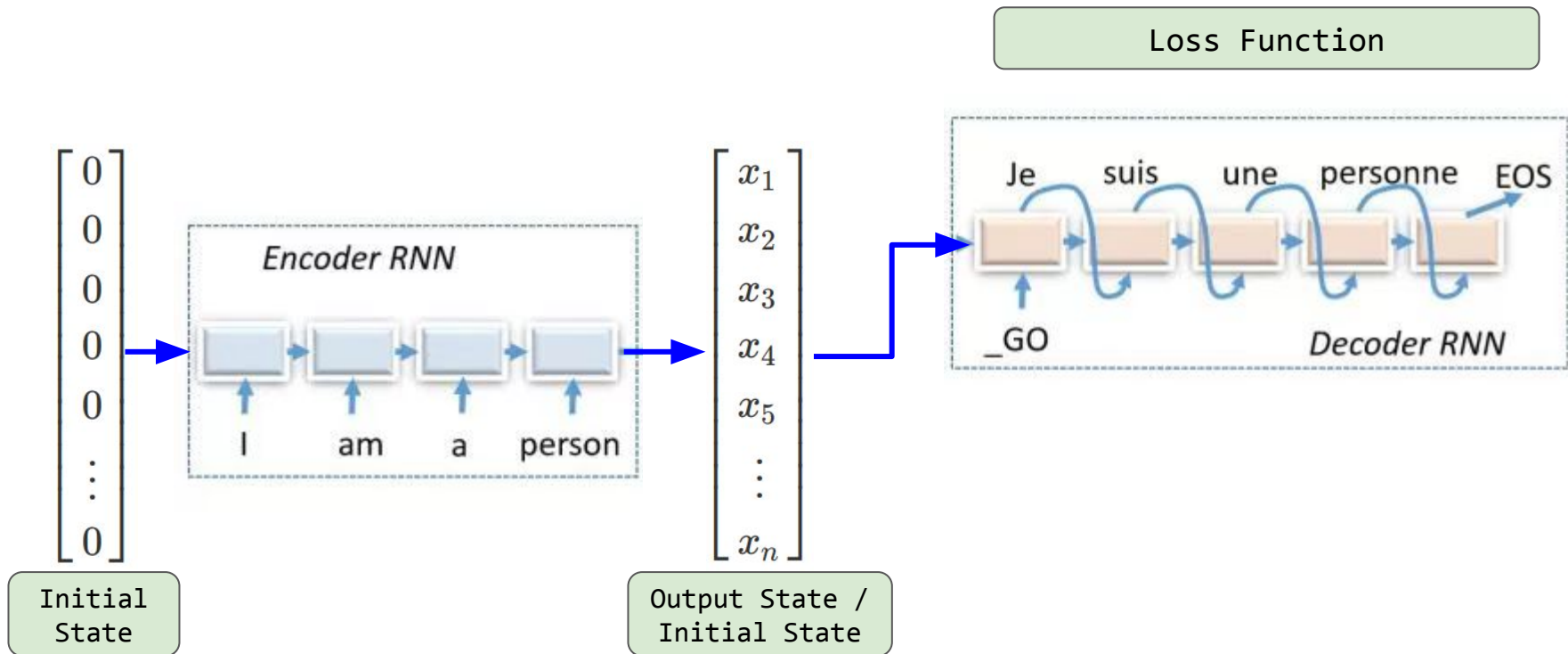


"man in black shirt is playing guitar."



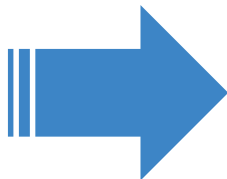
"construction worker in orange safety vest is working on road."

Encoder / Decoder Networks

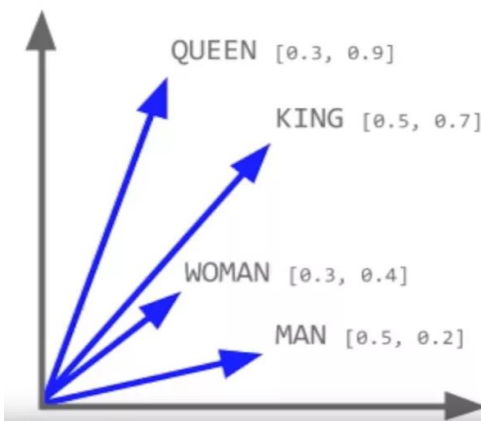


Word Embeddings

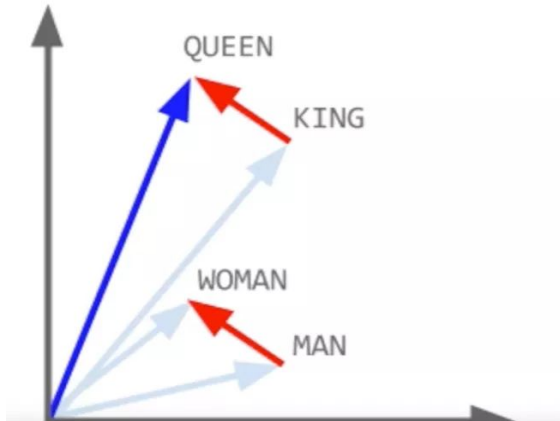
0 aardvark
 0 aardwolf
 0 aargau
 0 aarhus
 0 aarnet
 0 aaron
 0 aaronic
 1 aaronical
 0 aaronite
 0 aarp
 0 aarti
 : <omitting 228,116
 : entries for brevity>
 0 zyrian
 0 zyryan
 0 zyryenian
 0 zyzzyva
 0 zyzzyvas



Load up the word vectors



So king + man - woman = queen!



	King	Queen	Woman	Princess	...
Royalty	0.99	0.99	0.02	0.98	
Masculinity	0.99	0.05	0.01	0.02	
Femininity	0.05	0.93	0.999	0.96	
Age	0.7	0.6	0.5	0.1	
...					

Adding a word embedding layer

Diagram illustrating the calculation of a word embedding vector using a matrix multiplication of an embedding matrix and a one-hot vector.

The embedding matrix (left) has dimensions **Embedding Size** (rows) by **Dictionary Size** (columns). The one-hot vector (middle) has dimensions **Dictionary Size** (rows) by 1 (column).

The calculation is shown as:

$$\begin{bmatrix} 0.2 & -0.3 & -0.1 & 0.6 \\ 0.1 & 1.2 & 0.3 & -0.7 \\ 0.1 & 0.0 & -0.3 & 0.4 \end{bmatrix} \begin{bmatrix} 0 \\ 0 \\ 1 \\ 0 \end{bmatrix} = \begin{bmatrix} -0.1 \\ 0.3 \\ -0.3 \end{bmatrix}$$

The resulting vector (right) has dimensions **Embedding Size** (rows) by 1 (column).

Tensorflow

Installation, GPUs, and RNNs

Installing Tensorflow

Start with **Anaconda**, both python 2 and 3 are supported

All major python libraries are included: Numpy, Matplotlib, Jupyter Notebook, etc.

Then:

```
pip install tensorflow  
or  
pip install tensorflow-gpu
```

University GPU resources:

- **citrisdance.soe.ucsc.edu**
 - 2 older K20 GPUs, 4GB RAM and 32 cores - storage is an issue
- **<https://patternlab.calit2.optiputer.net/>**
 - 2 fast M40 GPUs 12GB RAM, 40 cores, owned by UCSD, shared with UCSC, 60+TB of storage
- More coming thanks to a recent NSF grant!

Working with GPUs

Get the state of the GPU(s)

```
nvidia-smi
```

Specify GPU(s) to use:

Use GPU 0:

```
export CUDA_VISIBLE_DEVICES=0
```

Use GPU 0 and 1:

```
export CUDA_VISIBLE_DEVICES=0,1
```

CPU only:

```
export CUDA_VISIBLE_DEVICES=-1
```

```
[dfparksucscedu@patternlab ~]$ nvidia-smi
Sat Oct 28 16:45:12 2017

+-----+
| NVIDIA-SMI 367.55                  Driver Version: 367.55          |
+-----+-----+
| GPU  Name            Persistence-M| Bus-Id        Disp.A | Volatile Uncorr. ECC | |
| Fan  Temp            Perf         | Pwr:Usage/Cap |      Memory-Usage     | GPU-Util  Compute M. |
|====|=====|
|  0   Tesla M40  24GB      Off      | 0000:85:00.0    Off  |             0%      0 |
| N/A   31C           P0          |   59W / 250W   |  0MiB / 22939MiB     |             Default |
+-----+-----+
|  1   Tesla M40  24GB      Off      | 0000:8D:00.0    Off  |             98%      0 |
| N/A   29C           P0          |   58W / 250W   |  0MiB / 22939MiB     |             Default |
+-----+-----+
|                                     4.4 MB Video  Fri 27 Oct 2017 04:04:06 PM PDT |
+-----+-----+
| Processes:                                     GPU Memory |
|   GPU          PID    Type    Process name                               Usage      |
|=====|
| No running processes found                               |
+-----+-----+
[dfparksucscedu@patternlab ~]$
```

Demos

<https://github.com/aymericdamien/TensorFlow-Examples>

Tutorial index

0 - Prerequisite

- [Introduction to Machine Learning](#).
- [Introduction to MNIST Dataset](#).

1 - Introduction

- [Hello World \(notebook\) \(code\)](#). Very simple example to learn how to print "hello world" using TensorFlow.
- [Basic Operations \(notebook\) \(code\)](#). A simple example that cover TensorFlow basic operations.

2 - Basic Models

- [Linear Regression \(notebook\) \(code\)](#). Implement a Linear Regression with TensorFlow.
- [Logistic Regression \(notebook\) \(code\)](#). Implement a Logistic Regression with TensorFlow.
- [Nearest Neighbor \(notebook\) \(code\)](#). Implement Nearest Neighbor algorithm with TensorFlow.
- [K-Means \(notebook\) \(code\)](#). Build a K-Means classifier with TensorFlow.
- [Random Forest \(notebook\) \(code\)](#). Build a Random Forest classifier with TensorFlow.

3 - Neural Networks

Supervised

- [Simple Neural Network \(notebook\) \(code\)](#). Build a simple neural network (a.k.a Multi-layer Perceptron) to classify MNIST digits dataset. Raw TensorFlow implementation.
- [Simple Neural Network \(tf.layers/estimator api\) \(notebook\) \(code\)](#). Use TensorFlow 'layers' and 'estimator' API to build a simple neural network (a.k.a Multi-layer Perceptron) to classify MNIST digits dataset.
- [Convolutional Neural Network \(notebook\) \(code\)](#). Build a convolutional neural network to classify MNIST digits dataset. Raw TensorFlow implementation.
- [Convolutional Neural Network \(tf.layers/estimator api\) \(notebook\) \(code\)](#). Use TensorFlow 'layers' and 'estimator' API to build a convolutional neural network to classify MNIST digits dataset.
- [Recurrent Neural Network \(LSTM\) \(notebook\) \(code\)](#). Build a recurrent neural network (LSTM) to classify MNIST digits dataset.
- [Bi-directional Recurrent Neural Network \(LSTM\) \(notebook\) \(code\)](#). Build a bi-directional recurrent neural network (LSTM) to classify MNIST digits dataset.
- [Dynamic Recurrent Neural Network \(LSTM\) \(notebook\) \(code\)](#). Build a recurrent neural network (LSTM) that performs dynamic calculation to classify sequences of different length.

Unsupervised

- [Auto-Encoder \(notebook\) \(code\)](#). Build an auto-encoder to encode an image to a lower dimension and re-construct it.
- [Variational Auto-Encoder \(notebook\) \(code\)](#). Build a variational auto-encoder (VAE), to encode and generate images from noise.
- [GAN \(Generative Adversarial Networks\) \(notebook\) \(code\)](#). Build a Generative Adversarial Network (GAN) to generate images from noise.
- [DCGAN \(Deep Convolutional Generative Adversarial Networks\) \(notebook\) \(code\)](#). Build a Deep Convolutional Generative Adversarial Network (DCGAN) to generate images from noise.

4 - Utilities

- [Save and Restore a model \(notebook\) \(code\)](#). Save and Restore a model with TensorFlow.
- [Tensorboard - Graph and loss visualization \(notebook\) \(code\)](#). Use Tensorboard to visualize the computation Graph and plot the loss.
- [Tensorboard - Advanced visualization \(notebook\) \(code\)](#). Going deeper into Tensorboard; visualize the variables, gradients, and more...

5 - Data Management

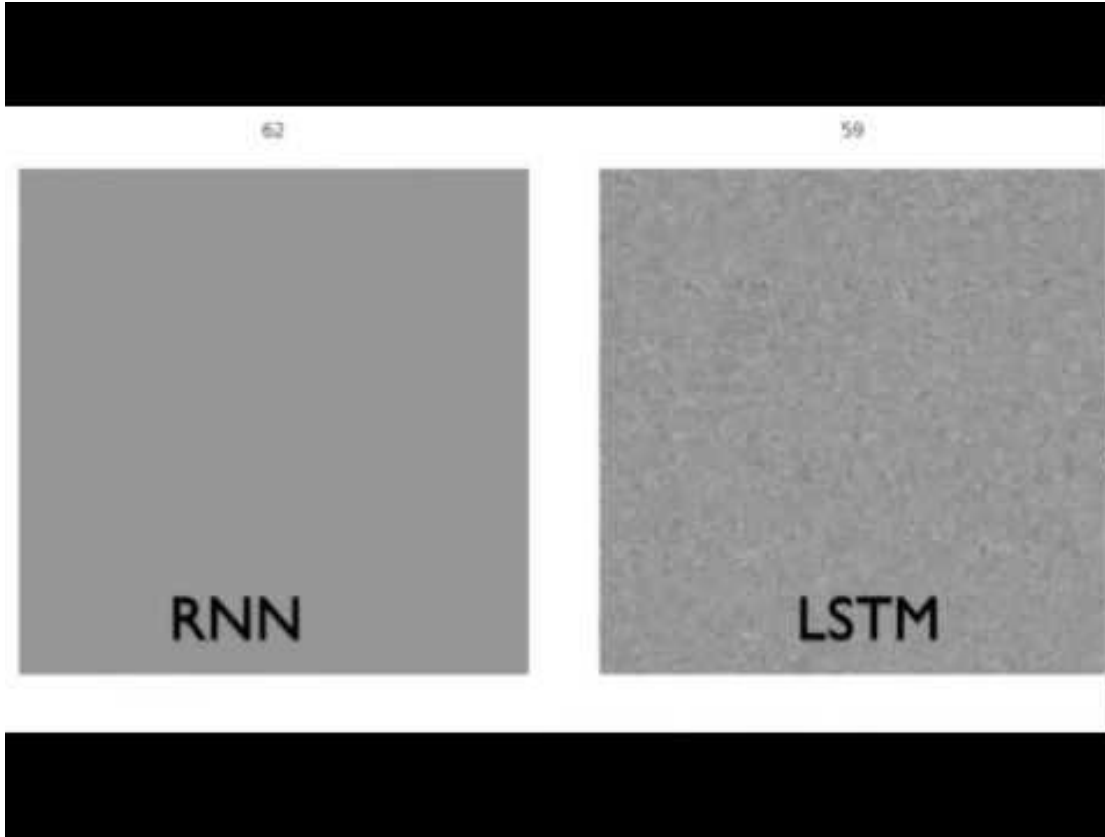
- [Build an image dataset \(notebook\) \(code\)](#). Build your own images dataset with TensorFlow data queues, from image folders or a dataset file.
- [TensorFlow Dataset API \(notebook\) \(code\)](#). Introducing TensorFlow Dataset API for optimizing the input data pipeline.

6 - Multi GPU

- [Basic Operations on multi-GPU \(notebook\) \(code\)](#). A simple example to introduce multi-GPU in TensorFlow.
- [Train a Neural Network on multi-GPU \(notebook\) \(code\)](#). A clear and simple TensorFlow implementation to train a convolutional neural network on multiple GPUs.

Addendum

Vanishing Gradient LSTM vs vanilla RNN

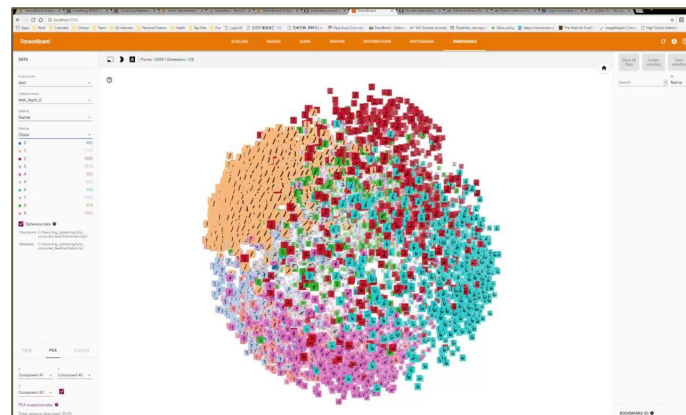
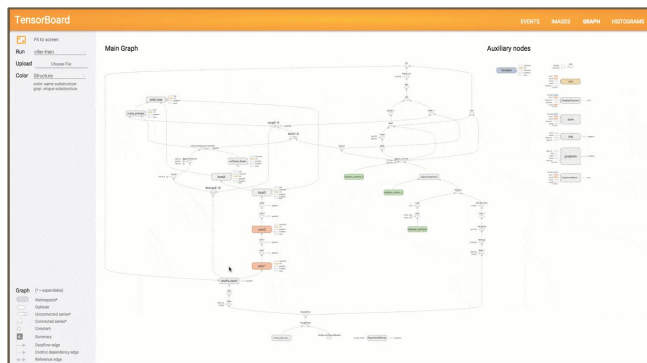
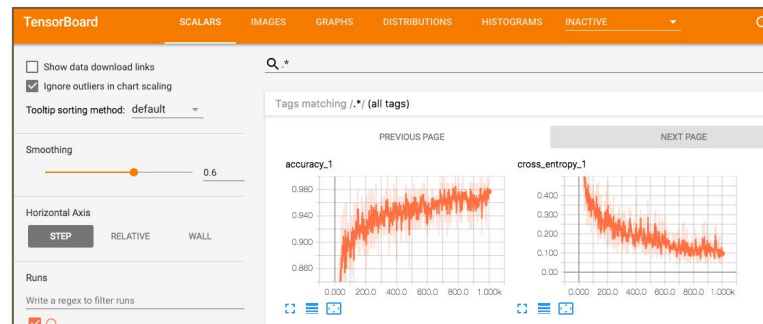


Tensorboard (optional)

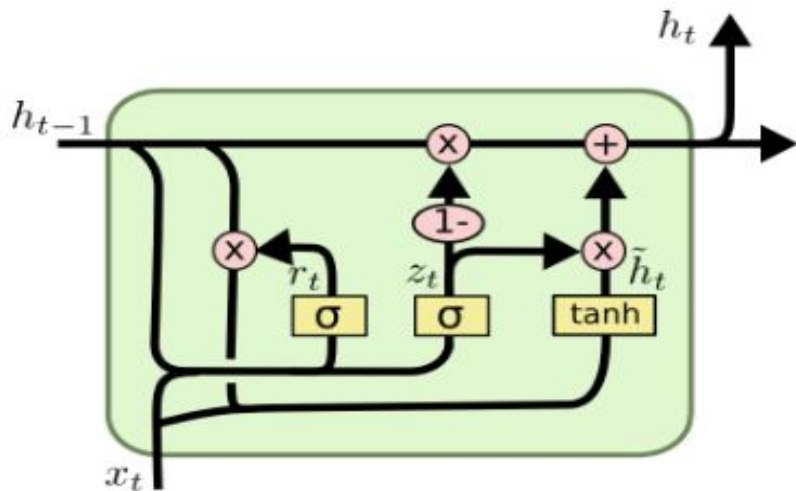
```
# In your build_graph() function define scalars, images, etc:  
tf.summary.image('image', image)  
tf.summary.scalar('accuracy', accuracy)  
summaries = tf.summary.merge_all()
```

```
# Create a file writer after opening your session  
summary_writer = tf.summary.FileWriter(log_dir, sess.graph)
```

```
# Compute & save summaries *periodically*  
results = sess.run([summaries, train_op], feed_dict={...})  
summary_writer.add_summary(results[0], step_num)
```



GRU: effectiveness of LSTMs w/ less computation



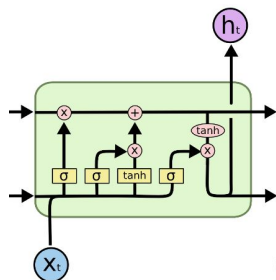
$$z_t = \sigma(W_z \cdot [h_{t-1}, x_t])$$

$$r_t = \sigma(W_r \cdot [h_{t-1}, x_t])$$

$$\tilde{h}_t = \tanh(W \cdot [r_t * h_{t-1}, x_t])$$

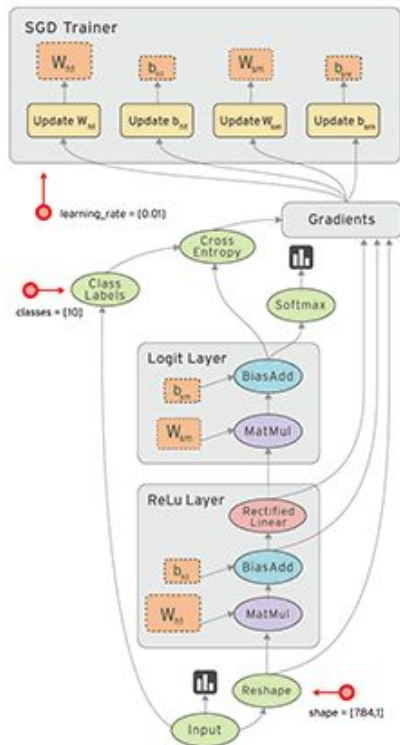
$$h_t = (1 - z_t) * h_{t-1} + z_t * \tilde{h}_t$$

LSTM:



Automatic Differentiation

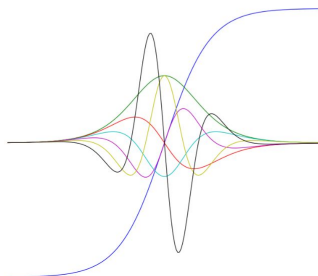
a.k.a.: How tensorflow and other tools do their magic



```
>>> import autograd.numpy as np # Thinly-wrapped numpy
>>> from autograd import grad # The only autograd function you may ever need
>>>
>>> def tanh(x): # Define a function
...     y = np.exp(-2.0 * x)
...     return (1.0 - y) / (1.0 + y)
...
>>> grad_tanh = grad(tanh) # Obtain its gradient function
>>> grad_tanh(1.0) # Evaluate the gradient at x = 1.0
0.41997434161402603
>>> (tanh(1.0001) - tanh(0.9999)) / 0.0002 # Compare to finite differences
0.41997434264973155
```

We can continue to differentiate as many times as we like, and use numpy's broadcasting of scalar-valued functions across many different input values:

```
>>> import matplotlib.pyplot as plt
>>> x = np.linspace(-7, 7, 200) # grad broadcasts across inputs
>>> plt.plot(x, tanh(x),
...         x, grad(tanh)(x), # first derivative
...         x, grad(grad(tanh))(x), # second derivative
...         x, grad(grad(grad(tanh)))(x), # third derivative
...         x, grad(grad(grad(grad(tanh)))(x), # fourth derivative
...         x, grad(grad(grad(grad(grad(tanh)))(x), # fifth derivative
...         x, grad(grad(grad(grad(grad(grad(tanh)))(x)) # sixth derivative
>>> plt.show()
```



<https://github.com/HIPS/autograd>