

Deep Learning: Day 4

chyld @ galvanize

Topics

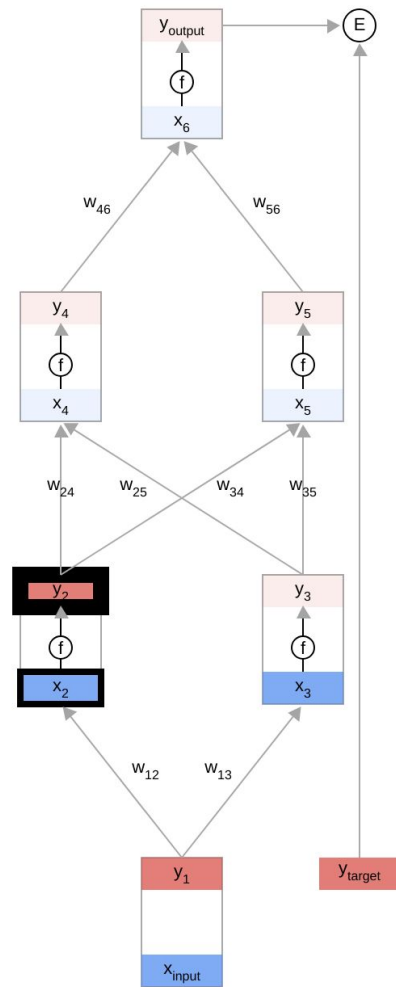
— — —

- Neural Network Demo
- Tensorboard

Neural Network Demo

— — —

<https://google-developers.appspot.com/machine-learning/crash-course/backprop-scroll/>



Tensorboard Setup

```
from time import time

from keras.models import Sequential
from keras.layers import Dense, Activation
from keras.callbacks import TensorBoard

model = Sequential()

model.add(Dense(10, input_shape=(784,)))
model.add(Activation('softmax'))

model.compile(optimizer='sgd', loss='categorical_crossentropy')

tensorboard = TensorBoard(log_dir="logs/{}".format(time()))

model.fit(x_train, y_train, verbose=1, callbacks=[tensorboard])
```

Tensorboard Run

```
from keras.callbacks import TensorBoard  
from time import time
```

```
tensorboard --logdir=logs
```

⬅ ➡ ↺ 🏠 ⓘ localhost:6006/

★ 🎲 📁 stk 📁 edu 📁 p1 📁 p2 📁 gal 📁 fin 📁

TensorBoard

SCALARS

GRAPHS

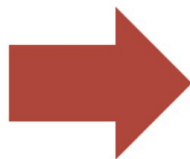
Word Embedding

Word embedding is the collective name for a set of language modeling and feature learning techniques in natural language processing (NLP) where words or phrases from the vocabulary are mapped to vectors of real numbers. Conceptually it involves a mathematical embedding from a space with one dimension per word to a continuous vector space with a much lower dimension.

Word Embedding

Vocabulary:

Man, woman, boy,
girl, prince,
princess, queen,
king, monarch

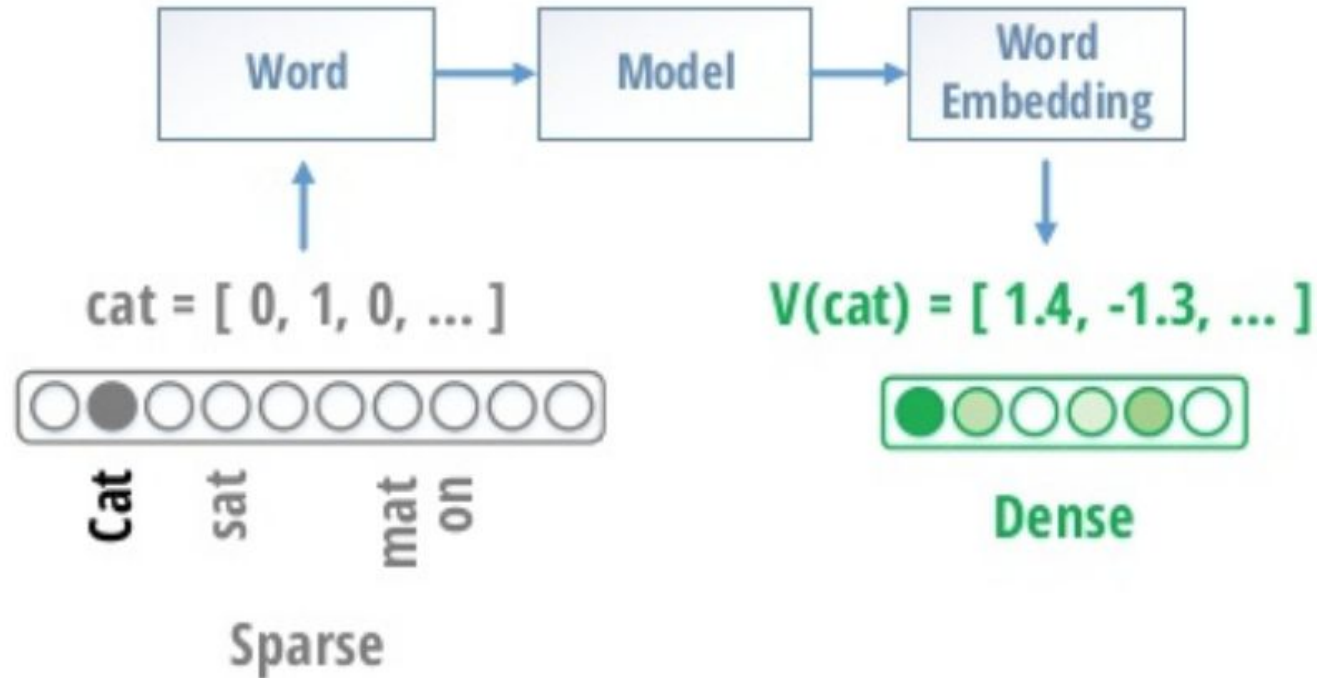


	1	2	3	4	5	6	7	8	9
man	1	0	0	0	0	0	0	0	0
woman	0	1	0	0	0	0	0	0	0
boy	0	0	1	0	0	0	0	0	0
girl	0	0	0	1	0	0	0	0	0
prince	0	0	0	0	1	0	0	0	0
princess	0	0	0	0	0	1	0	0	0
queen	0	0	0	0	0	0	1	0	0
king	0	0	0	0	0	0	0	1	0
monarch	0	0	0	0	0	0	0	0	1

Each word gets
a 1x9 vector
representation

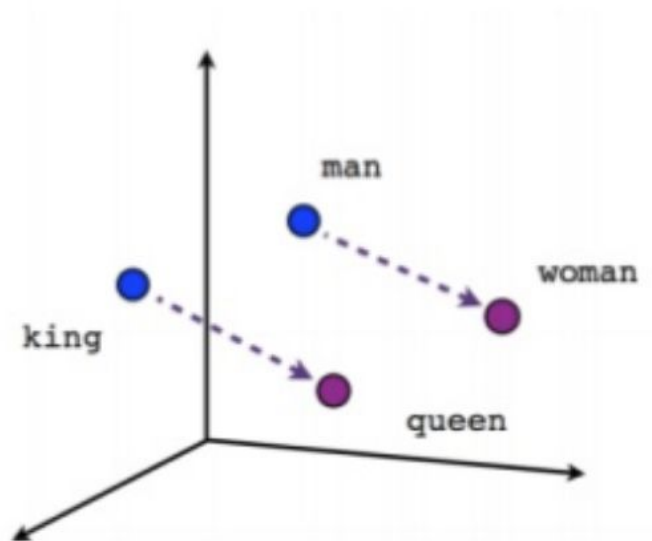
Word Embedding

— — —

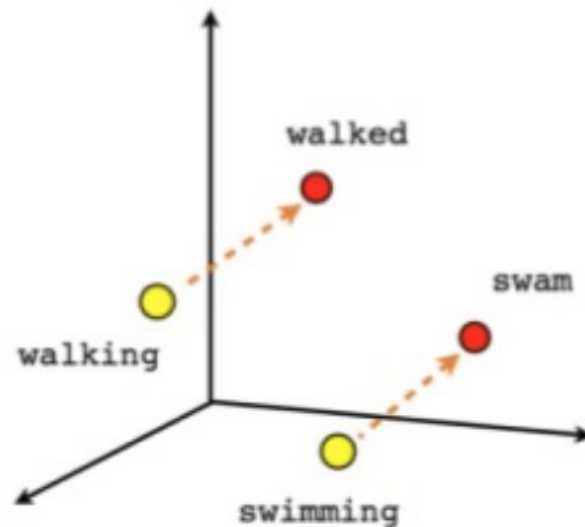


Word Embedding

— — —



Male-Female



Verb tense

Time Dependent Words

— — —

problem: we can't model long-term dependencies

“In **France**, I had a great time and I learnt some of the _____
language.”

We need information from the far past and future to accurately guess the correct word.

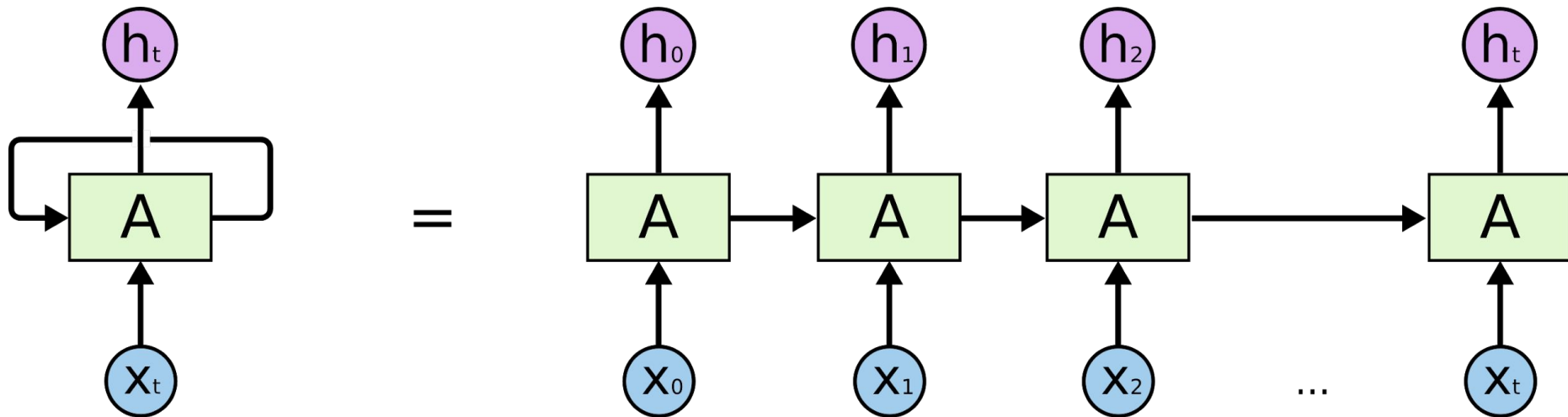
Use Recurrent Neural Networks

1. to maintain word order
2. to share parameters across the sequence
3. to keep track of long-term dependencies

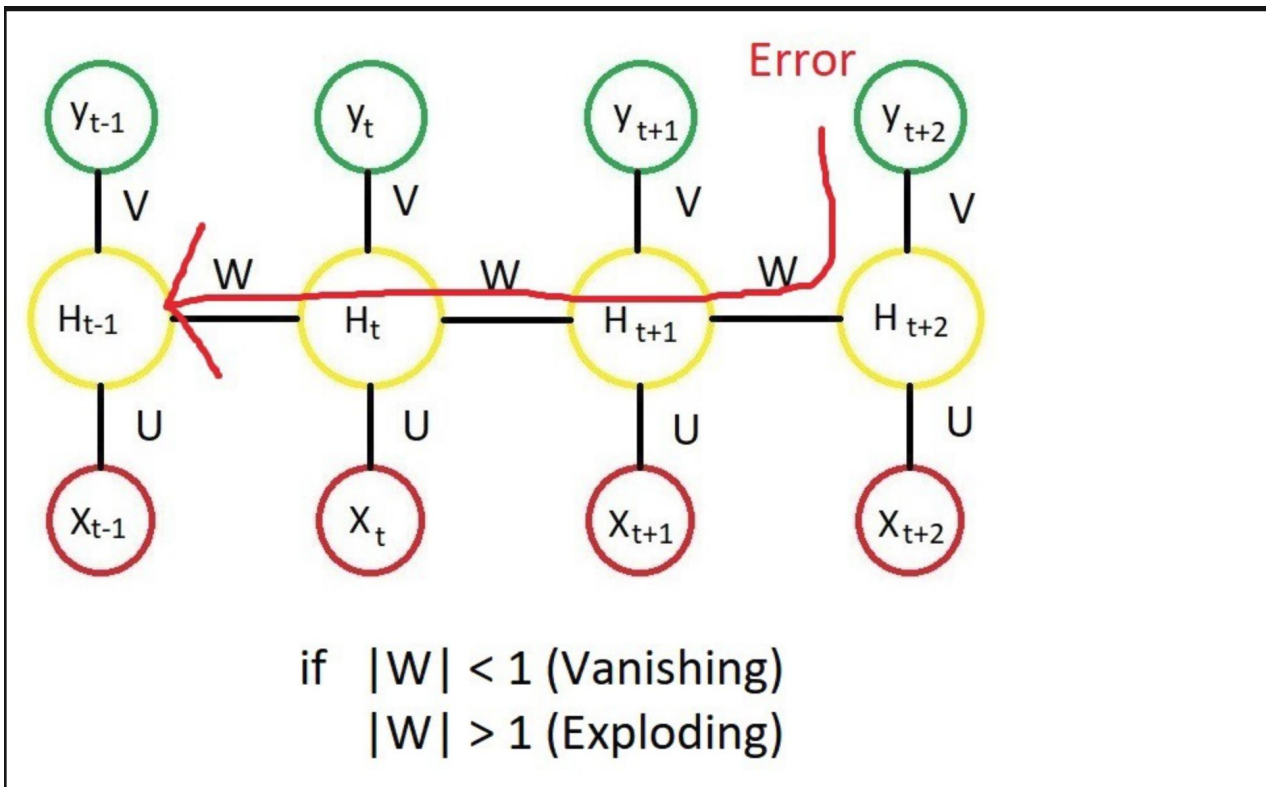
Recurrent Neural Network, RNN

A recurrent neural network (RNN) is a class of artificial neural network where connections between nodes form a directed graph along a sequence. This allows it to exhibit dynamic temporal behavior for a time sequence. Unlike feedforward neural networks, RNNs can use their internal state (memory) to process sequences of inputs.

Recurrent Neural Network, RNN



Vanishing Gradient



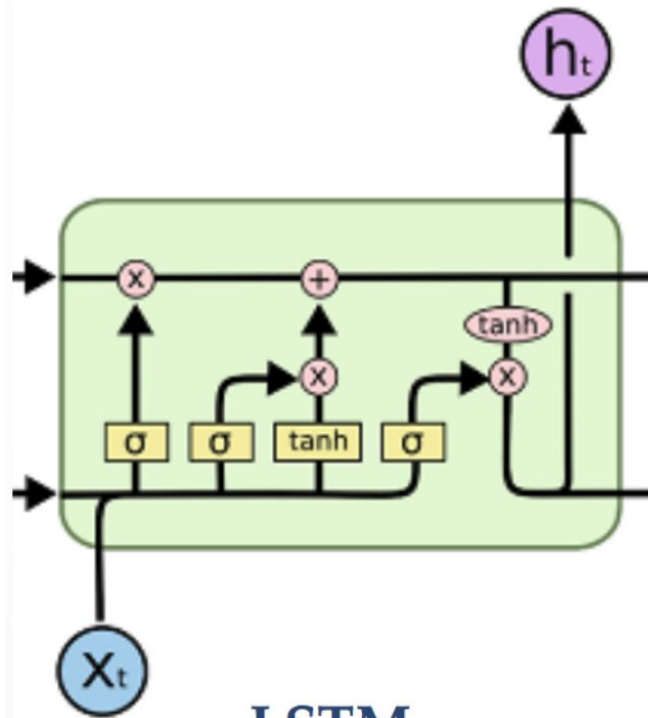
Long Short Term Memory, LSTM

Long short-term memory (LSTM) units (or blocks) are a building unit for layers of a recurrent neural network (RNN). A RNN composed of LSTM units is often called an LSTM network. A common LSTM unit is composed of a cell, an input gate, an output gate and a forget gate. The cell is responsible for "remembering" values over arbitrary time intervals; hence the word "memory" in LSTM.

Long Short Term Memory, LSTM

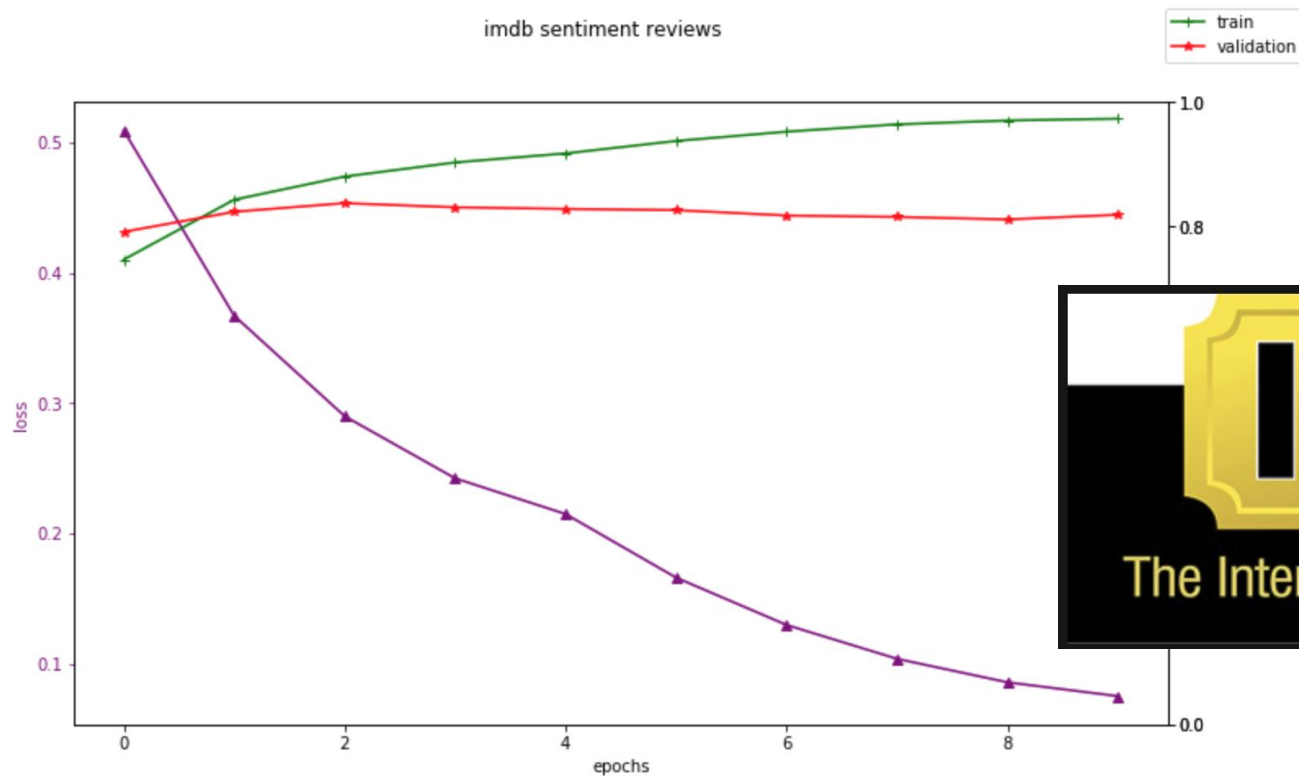
An LSTM is well-suited to classify, process and predict time series given time lags of unknown size and duration between important events. LSTMs were developed to deal with the exploding and vanishing gradient problem when training traditional RNNs. Relative insensitivity to gap length gives an advantage to LSTM over alternative RNNs.

LSTM Cell

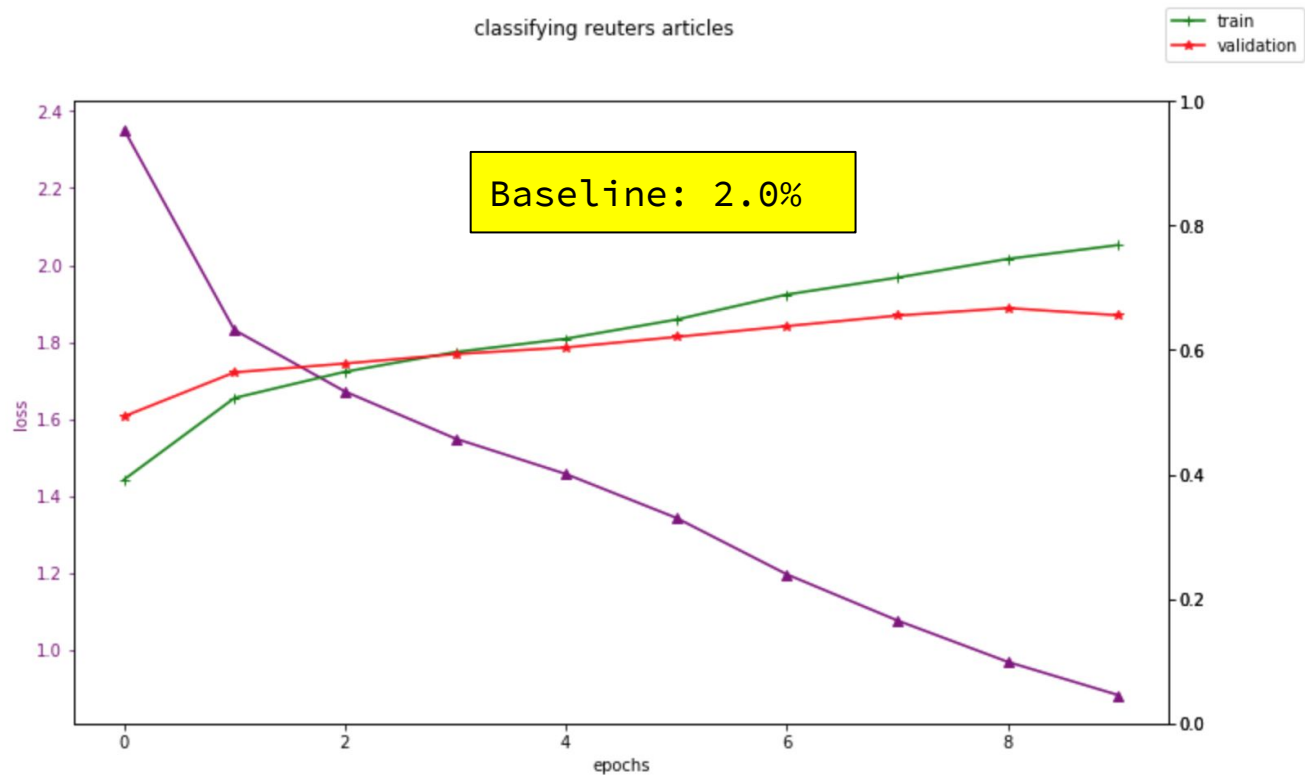


LSTM

IMDB Movie Review Sentiment Analysis



Reuters News Article Classification



Text Generation Model - Alice in Wonderland corpus

