DEEP LEARNING

(SOME SLIDES FROM HU LI)

NORTHWESTERN UNIVERSITY EECS 349

WHAT IS MACHINE LEARNING?

- Machine Learning: automatically learning functions from inputs to outputs, e.g.
  - Input Features = Does news release text include "X acquired Y" ?
Are X, Y in company database? … etc.
  - Output = estimate that Acquired(X, Y) is true
- Trained on **examples**, e.g. news releases labeled as an acquisition, or not

MACHINE LEARNING YESTERDAY AND TODAY

- Yesterday: Conventional Analytics
  - Emphasis on **Feature** Design
  - Still important today
- Today: Deep Learning
  - Emphasis on Raw Data, Scale, **Model** Design
  - Needs up to millions of examples (100s of each kind of output)
  - Especially applicable when features are hard to design
    - Image/speech recog, language modeling – hard for humans to explain how they do it

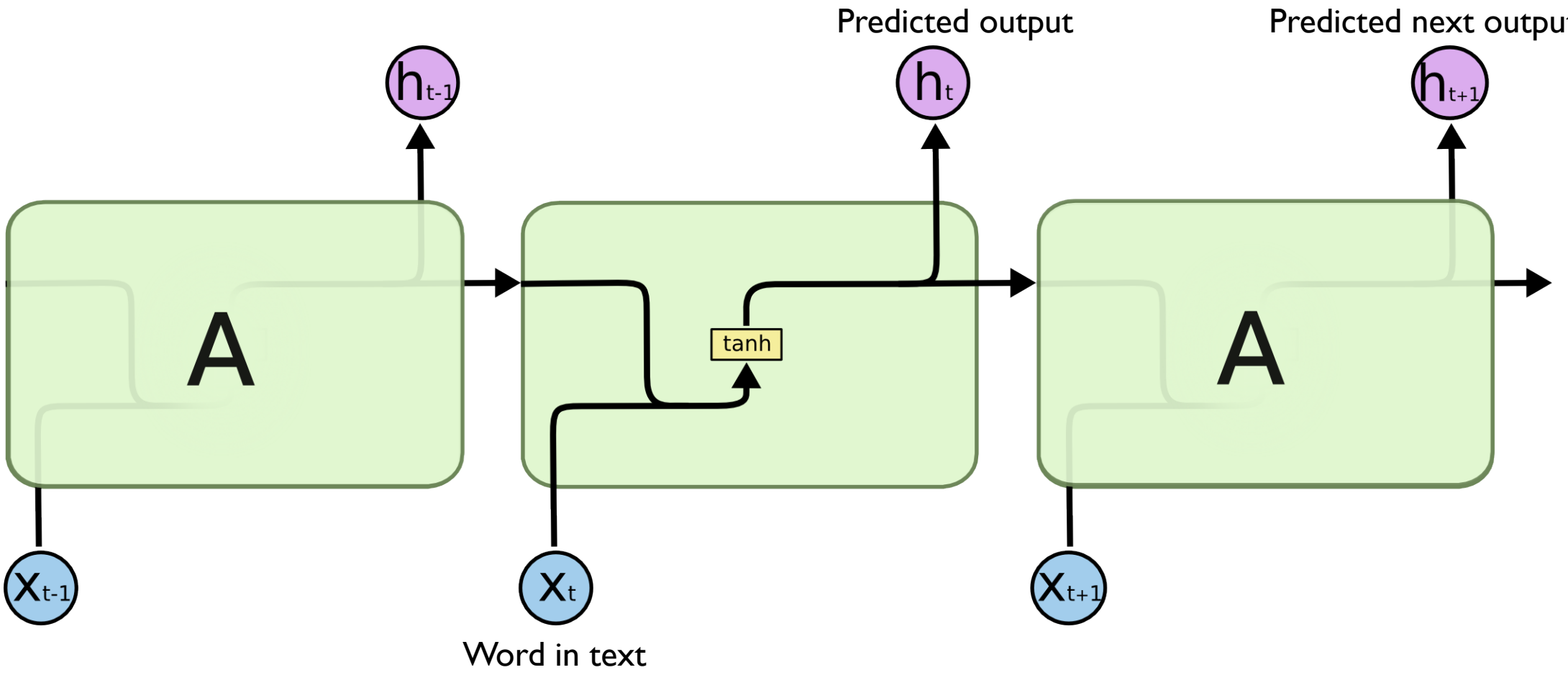
WHY DEEP LEARNING?

- Deep Learning is not a **new** learning technique
  - Neural nets date to the late 1940s
  - Have gone furiously in and out of vogue since then
- So why now? We have:
  - More data (ImageNet, Web-scale corpora, EMR, high-throughput bio, IoT, …)
  - More compute (GPU-based training, cloud)
  - A handful of new optimization tricks (e.g. dropout)
  - Democratizing software packages (Theano, Torch, TensorFlow, Caffe)
  - An interest in tasks for which features are hard to design

LANGUAGE MODELS: RNNS AND LSTMS

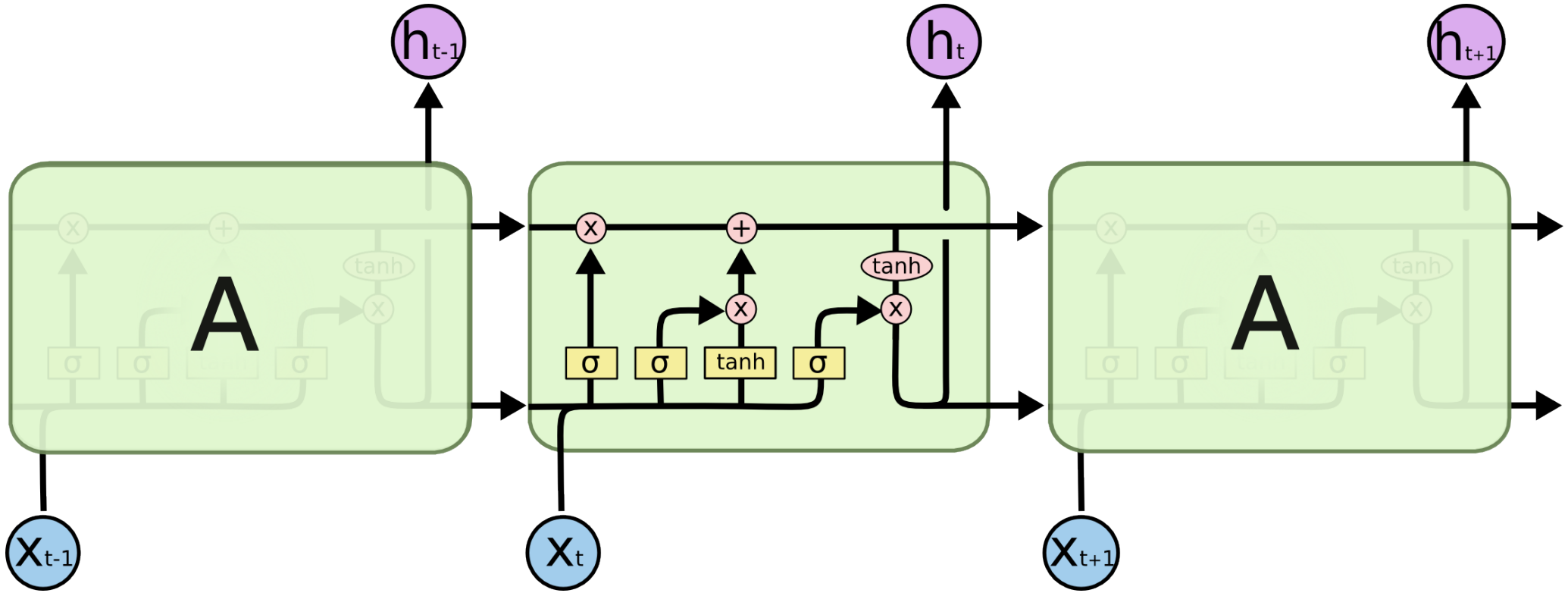
- Estimate string probabilities:
  - P("Iowan touts Ide") << P("I went outside")
  - Speech, handwriting recognition
- Or map string to label
  - "This movie is totally awesome" => positive sentiment (sentiment analysis)
- Or map string to string
  - Machine Translation
  - Summarization
  - Definition modeling

RNNS



http://colah.github.io/posts/2015-08-Understanding-LSTMs/

LSTMS



INTUITION BEHIND LSTMS

- Harness long-range dependencies
- "She's _____ so I hope that she will _____ ."

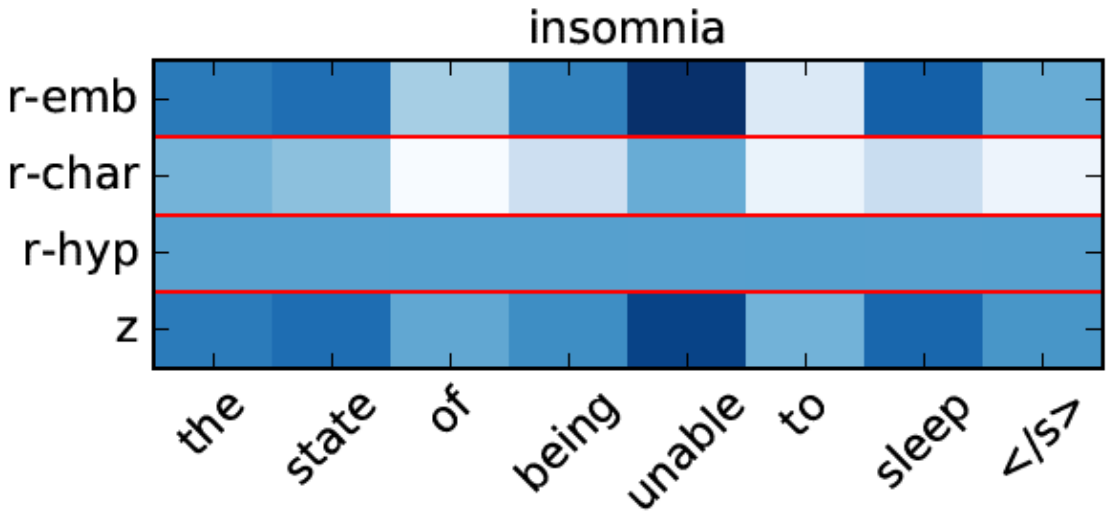
brilliant
sick

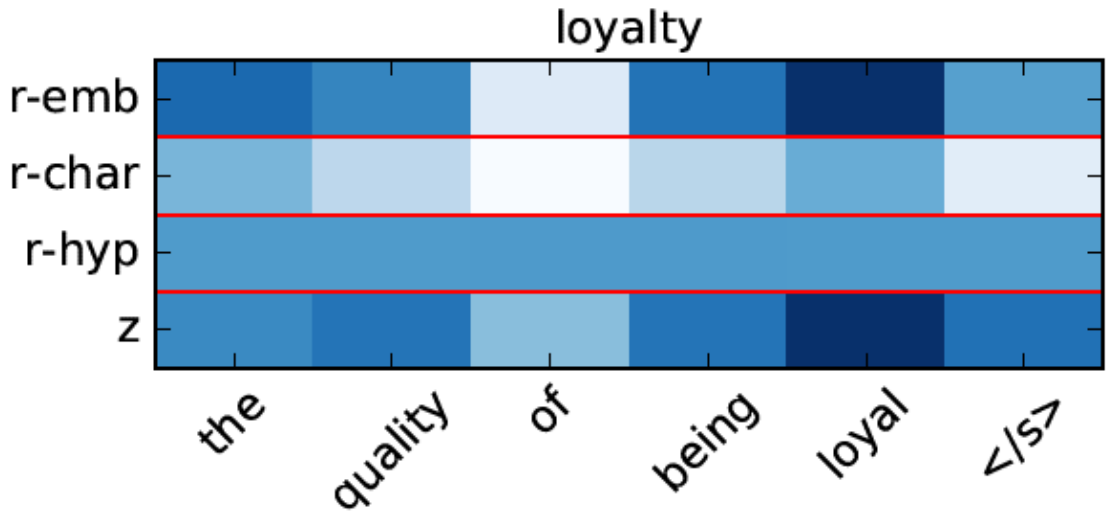
join us
not breathe on me

WHAT DEEP NEURAL NETS OF LANGUAGE DO

- Simultaneously, learn:
  - *embeddings* of words (vectors of numbers – e.g. 300 dimensions) and
  - A predictor of which words occur around other words, in terms of the embeddings.
- Embeddings have been shown to capture syntax and semantics
  - But, what *exactly* do they capture?

LSTM EXAMPLE





RESULTS

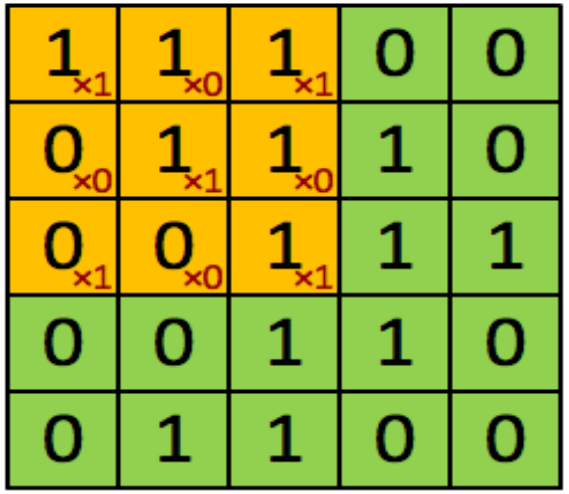
- Incredibly good language modeling results
  - Good Turing Smoothing: ~160 perplexity (lower is better)
  - Kneser-Ney smoothing: ~140 perplexity
  - Today's best LSTMs: ~50 perplexity [Salesforce, ca 2017]

Perplexity numbers on Penn TreeBank data set (approx.)

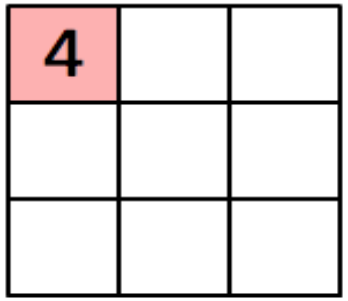
WHAT'S NEW WITH NEURAL NETS

- What model features are well-suited to deep neural networks?
  - Recurrence
  - Convolution
  - Multi-task

WHAT IS CONVOLUTION?



Image

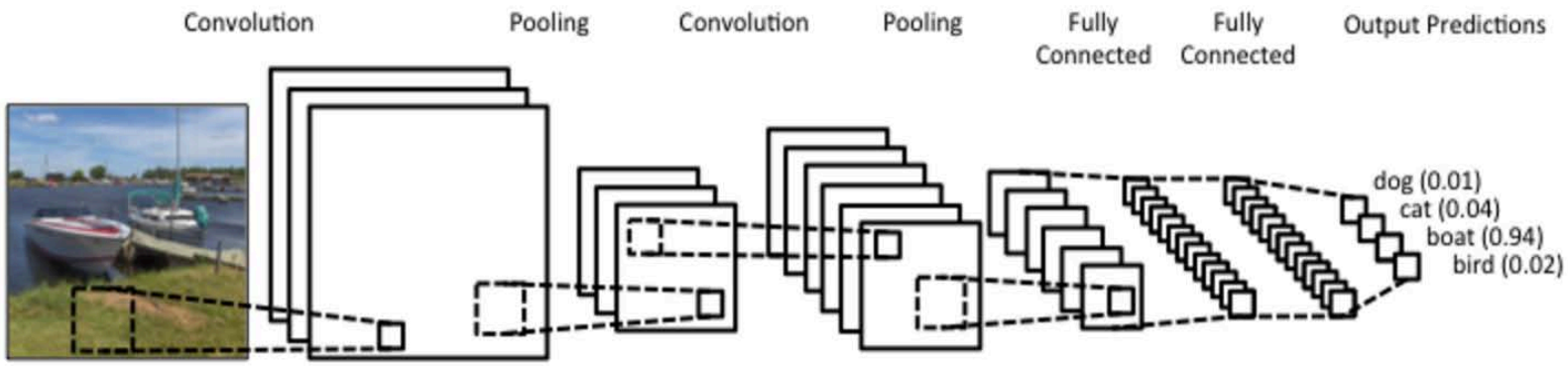


Convolved Feature

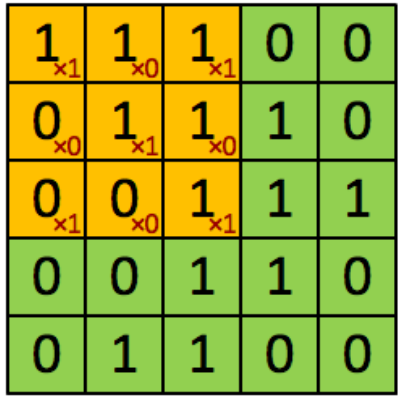
WHAT CAN CONVOLUTION DO?

0	0	0	0	0
0	1	1	1	0
0	1	1	1	0
0	1	1	1	0
0	0	0	0	0

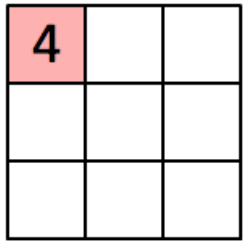
CONVOLUTIONAL NEURAL NETWORK



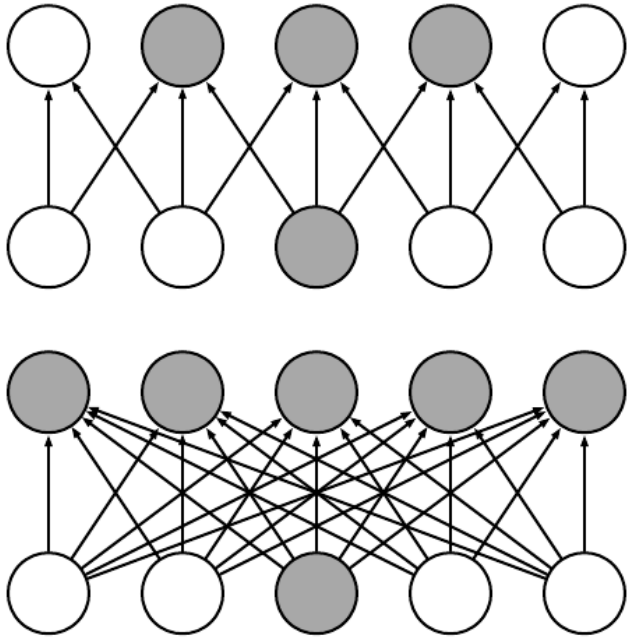
CONVOLUTIONAL NEURAL NETWORK



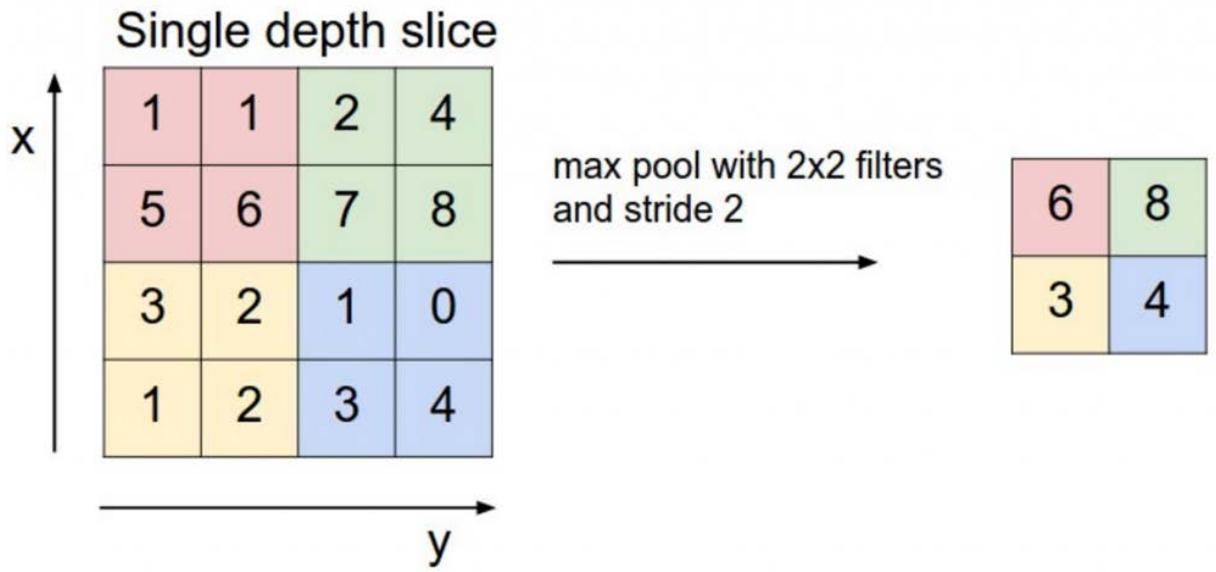
Image



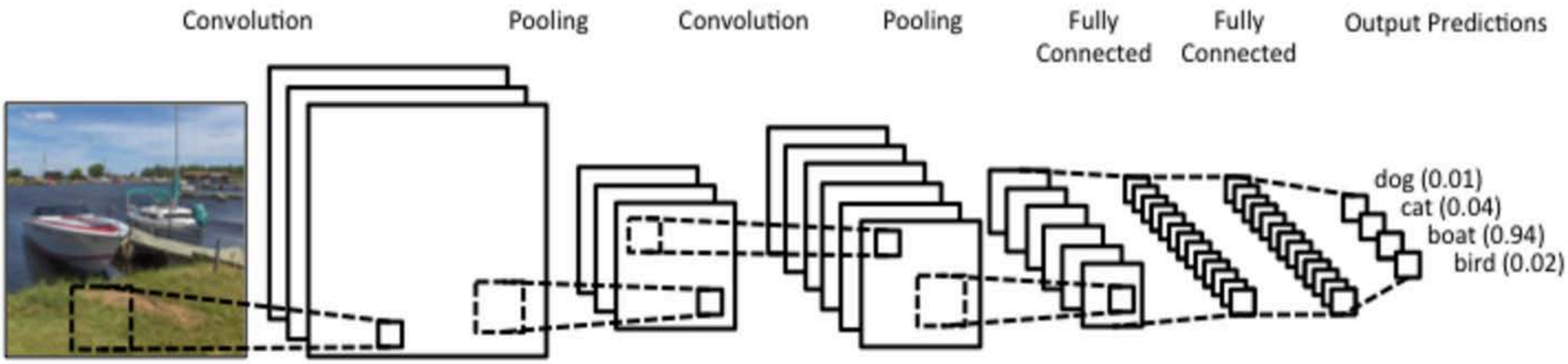
Convolved Feature



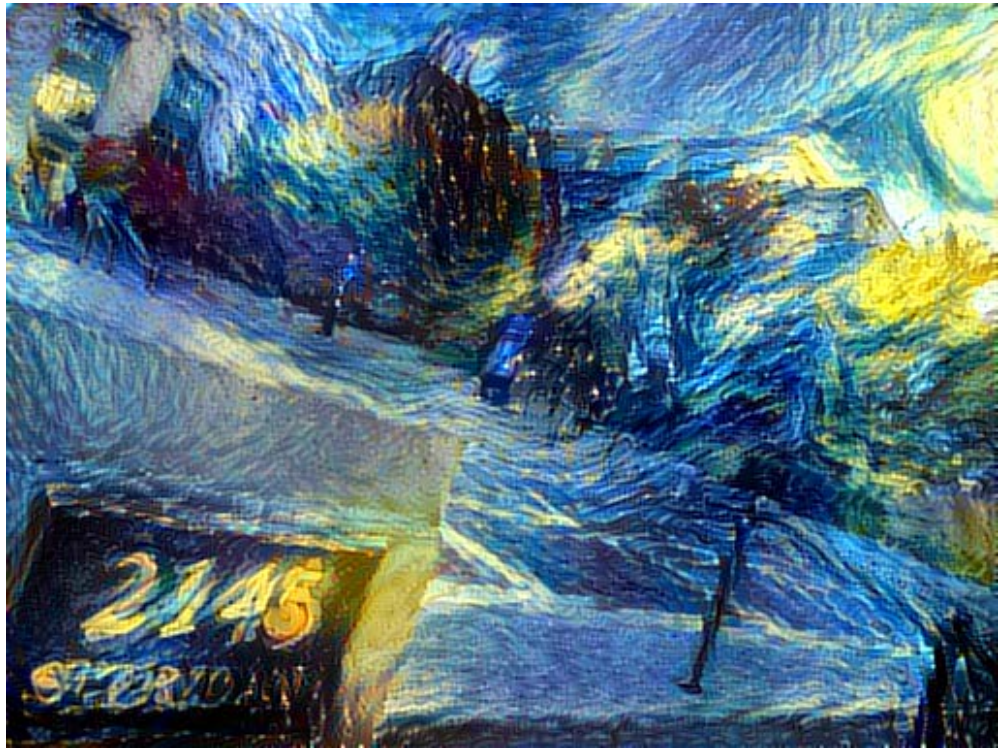
MAX POOLING


Single depth slice
x
y
1 1 2 4
5 6 7 8
3 2 1 0
1 2 3 4
max pool with 2x2 filters
and stride 2
6 8
3 4

CONVOLUTIONAL NEURAL NETWORK




2145
SHERIDAN


2145

MULTI-TASK

- Inputs: retail sales, gross metro product, population, %millennials, percentage of jobs in growth industries, existing stock, total inventory, etc. etc.
- Outputs: the same variables, but in the next time period

- Often, training a *single* network to perform *many* prediction tasks outperforms individual models trained on a single task

POINTERS

- Tensorflow, Keras, PyTorch
  - Libraries that make training neural nets not too hard
- Bengio's textbook:
  - http://www.deeplearningbook.org/
- Image demos:
  - http://cs.stanford.edu/people/karpathy/deepimagesent/
  - https://www.captionbot.ai/
- Word embedding demos:
  - http://thor.cs.northwestern.edu:24603/
  - https://rare-technologies.com/word2vec-tutorial/