An introduction to deep learning in CASP

Prof. David Jones
UCL Dept. of Computer Science &
The Francis Crick Institute
What is a Deep Neural Network?

Deep nets are defined as having at least 3 layers (2 hidden layers), but typically many more.

Training of very deep networks is difficult because of the VANISHING GRADIENT PROBLEM.
Key Developments

- 2006, Hinton’s Layer by Layer training of Deep Belief Nets
- 2011, Rectified Linear Units
- 2015, Batch Normalization
- 2016, Residual nets

- 2010, Acceleration by GPUs (CUDA/theano)

Turned out that this is all we needed to change to train deep nets!
Open Source Resources

Deep Learning Tutorials: 
deeplearning.net/tutorial/
deeplearning4j.org
Deep Learning

• Many connected nonlinear processing units (e.g. layers in a DNN)

• These units learn a hierarchy of representations that correspond to different levels of abstraction

“Deep learning has proven to be good at machine vision or text processing problems, but not a lot else.” Anon. reviewer

This is obviously not true, but there may be some wisdom in it.

Deep learning is effective at any problem where there is a strong element of hierarchical parsing to be done. This means that deep learning does not work particularly well on unstructured data.
Convolutional networks are popular because they are good for machine vision problems.

The basic idea is to take an input image and apply “filters” to produce new images that highlight specific features of the input image, e.g. edges.

In this second case, residue covariance matrices are the inputs, and want to model the probability of each residue pair being in contact.
Convolutional nets act on image-like inputs by applying small filters to co-located groups of pixels in the image.

The numbers in the filter are trainable weights.

After training, the outputs are also image-like, but now convey extra information e.g. presence of an edge.

Actually a “filter” is just a small neural network that is applied at every set of pixels (with the same set of weights).
Growing the receptive field by stacking convolutional layers

Deep Residual Networks

Residual neural networks avoid vanishing gradient effects in very deep networks by utilizing *skip connections* or *short-cuts* to jump over some layers.

Further improvements may also come from effective ensembling of different architectures during training.

Deep ResNets are now more or less the standard deep convolutional network models.
Fully Convolutional Deep Neural Nets

- FCNs are a powerful AI approach to locating objects in images.
- Successful examples are recognizing tumours in X-rays, pedestrians crossing the road, faces in photographs or even cats and dogs…
Covariance between MSA columns

We need to use an input feature set in which each element only has information from given pairs of columns in an MSA (no information from any other columns). We consider a simple covariance measure $S$:

$$S_{ij}^{ab} = \frac{1}{n} \sum_{k=1}^{n} (x_i^{ak} - \bar{x}_i^a)(x_j^{bk} - \bar{x}_j^b),$$

where:

$n$: number of rows in the MSA
$k$: row index in MSA
$i, j$: column indices in MSA
$a, b$: residue types (Ala, Gly etc., with gaps considered as a 21st type)
$x$: residue presence or absence (1 or 0) in specified MSA column and row
$\bar{x}$: relative frequency of residue in specified MSA column
DeepCOV: Analysing Residue Covariation using FCNs

- **Input layer**: 441 x m x m
- **2D Convolutional layer (128 1x1 filters)**: 128 x m x m
- **Feature max-pooling layer**: 64 x m x m
- **2D Convolutional layer (64 3x3 or 5x5 filters)**: 64 x m x m
- **2D Convolutional layer (64 3x3 or 5x5 filters)**: 64 x m x m
- **2D Convolutional layer (64 3x3 or 5x5 filters)**: 64 x m x m
- **2D Convolutional layer (64 3x3 or 5x5 filters)**: 64 x m x m
- **Convolutional Maxout layer**: Dimensionality reduction to 64 channels
- **Output layer**: 1 x m x m

**Input**: 21x21 Covariance feature channels

**8-10 Padded Convolutional layers**: 5x5x64 Filters
- **ReLU Activation**
- **BatchNorm**

**Output**: Contact Probability Map
Determining sequence locality of chaining effects using varying sized receptive fields

Alignment has $m$ columns
Contact map is $m \times m$ matrix

Features calculated in pairwise fashion on alignment columns will also be $m \times m$

The receptive field is a square set of the covariances that are used to get the contact prediction for residue pair $(i, j)$

This corresponds to 2 windows in the MSA around columns $i$ and $j$ (blue and orange)

Fully convolutional networks (FCNs) allow us to easily control the receptive field size
Performance on PSICOV150 test set using only covariance as input

Precision increases with receptive field size

Mean precision plateaus at receptive field of only around 15 residues

DeepCov is more precise than CCMpred and MetaPSICOV2
Performance on shallow alignments

232 alignments with fewer than 200 raw sequences (effective count < 159)

Retrained DeepCov model using different training set with no overlap to this test set (by ECOD classification)

Fewer sequences needed to get reasonably accurate predictions! Why?
Do we still need deep alignments?

- It appears that DL methods are able to derive contacts from relatively shallow alignments.
- Classical global covariation methods e.g. PSICOV, plmDCA worked well with deep alignments but fail on shallow alignments.
- This is not necessarily a new aspect of DL methods but more a failure mode of global methods e.g. SICE model in PSICOV:
  \[
  \sum_{i,j=1}^{d} S_{ij} \Theta_{ij} - \log \det \Theta + \rho \sum_{i,j=1}^{d} |\Theta_{ij}|
  \]
- Note that in global models we are forced to fill-in missing data e.g. by using shrinking and pseudocounts.
- DL models just learn to ignore the missing data!
Deep learning models are relatively insensitive to missing data...

https://clarifai.com/demo
Simple Recurrent Network (RNN)

LSTM RNN

RNN = CNN?
Sequence-to-sequence models

Useful for...

- Secondary structure prediction
  - Sequence profile $\rightarrow$ DSSP codes

- Torsion angle prediction
  - Sequence profile $\rightarrow$ main chain torsion angles

- Estimation of model accuracy
  - Sequence of 3-D coordinates $\rightarrow$ GDT score

- End-to-end differentiable folding models
  - Amino acid sequence $\rightarrow$ torsion angles
    - Torsion angles $\rightarrow$ 3-D coordinates
Summary

• Deep learning (deep residual networks) has improved covariation-based contact prediction
  • Problem fits the deep learning model of hierarchical feature extraction well, and deep learning is robust to missing data (shallow alignments)
  • By varying the network depth we can gain some insight into the most significant contributions to indirect correlation chaining
  • Can easily be adapted to predict distance distribution outputs, or to produce an EMA method
• Sequence-to-sequence models have proven effective in predicting 1-D protein features e.g. secondary structure & torsion angles
  • End-to-end differentiable models may be interesting to explore
Acknowledgements