Schomaker (2016) How deep is deep and what is next in computational intelligence? [keynote lecture]

Copyright 2016 L. Schomaker
Note: not all image material may have been cleared as regards copyright. l.r.b.schomaker@rug.nl

Recent advances in ML?
› Neural networks & Deep Learning
› Critical remarks
› Monk: massive shallow but convenient learning

Deep learning / Recent advances in ML
› ‘Google self-driving cars’
› Predicting internet user interests (‘cookies’)
› Twitter-based epidemiology (‘flu tweets’)
› Create a van Gogh or Munch version of a photograph
› Coloring of B/W movies
› Learning to play Atari Breakout, Pacman etc.
› AlphaGO: computer wins at playing GO
› Improved training (loss function, softmax, ReLU)
› With 1000 hidden layers (Susillo & Abbot, 2015)
› etc.

List of current successes in deep learning
History of NN's

- 1957 - 1st generation (Rosenblatt’s Perceptron)
- 1983 – 2nd generation (Werbos/Rumelhart)
- 1996 – NN – winter
- 2000 – 3rd generation: Deep Learning (Hinton/Lecun)
  - Computer vision
  - Speech/handwriting: sequence classification
    LSTM/BLSTM (Schmidhuber/Liwicki/Graves)
  - Remark: handwriting recognition played an important role. Early 2D convolutional nets by LeCun: IWFHR 1990, Cenparmi, Montreal

Brief history of NNs

- It is remarkable that after Minsky and Papert the rejection of the perceptron was so massive. After all, linear systems with only an input and an output layer still can do a lot and also were in use. Consider for instance Widrow & Hoff, telephone line echo cancellation using a linear system. But indeed, non-linear mappings are impossible and the fact that the output units are thresholded does not introduce a non-linearity in the forward mapping itself.

- Finally: non-linear mappings are possible. Rumelhart & McClelland came from psychology. They published two books, one blue, one brown, as a set, with a yellow third book for students. It had a diskette with C code. By 1996 there were many frustrations with MLP. There was not enough labeled data, computers were slow and generalization was not good. The SVM was developed at AT&T by Isabelle Guyon, her husband Bernard Boser and Vapnik, on the basis of the problems in training.
handwriting recognizers. The bosses at AT&T were not happy with the fact that NNs yielded different solutions from different randomisations.

Could have been a better slide, but we all know the drill: CNNs finally have their breakthrough. It must be admitted that for a long time Yan LeCun was the only one with very good results on CNNs. The community was also surprised with the guts of Hinton to publish in Nature about what many were already doing quite extensively: using autoencoders or diabolo MLP for dimensionality reduction because there is no need for labels. But Hinton added some very useful tricks that would ultimately allow for the deep learning revolution. Both researchers were essential, in any case. Hinton, G. E. and Salakhutdinov, R. R

Reducing the dimensionality of data with neural networks.
Slide 8

The ‘GO’ example

- Board game, black/white
- Enclose the opponent
- GO: $10^{70}$ possible states (chess: $10^{47}$ states)
- Google/DeepMind: Very limited game knowledge, bootstrap with a limited data set of expert games.
- NN1: Learn the value of any given board configuration
- Then NN2 learns the policy to choose the best move per board
- Computer won 15 March 2016 from human world champion Lee Sedol

AlphaGO: a real system entails much more than just one single deep net

Slide 9

The ‘GO’ example

- NN1: Learn the value of any given board pattern from human expert games
- Then train NN2 to detect the policy to choose the best move given a board state
- by playing 30 million times against ‘itself’

The point is that Deep Learning in itself is hardly interesting. Only by integrating multiple networks into a functional architecture for the operational stage, they will be useful, as in AlphaGO.

My favorite metaphor for an isolated NN in this respect is the Ferrari engine bolted to a workbench in your shed. Very powerful but utterly useless.

Slide 10

Example: finding DATE blocks in handwritten manuscripts
Zhenwei Shi (2016)

We are currently (evidently) also working on CNN and LSTM. The example here is challenging because of the low prior probability of finding a data in a sea of regions of interest that are not representing a DATE block.
Results for DATE detection

<table>
<thead>
<tr>
<th>Items</th>
<th>Recall</th>
<th>Precision</th>
<th>F1 score</th>
</tr>
</thead>
<tbody>
<tr>
<td>SVM-based Classifier</td>
<td>20.0 ± 7.0%</td>
<td>13.8 ± 6.4%</td>
<td>14.9 ± 7.4%</td>
</tr>
<tr>
<td>ConvNet Model</td>
<td>24.0 ± 5.8%</td>
<td>14.8 ± 7.5%</td>
<td>15.5 ± 7.5%</td>
</tr>
<tr>
<td>Deep-Positional Expectancy Model and SVM-based Classifier</td>
<td>40.0 ± 3.8%</td>
<td>36.7 ± 2.8%</td>
<td>36.7 ± 2.8%</td>
</tr>
<tr>
<td>Deep-Positional Expectancy Model and ShapeNet Model</td>
<td>41.5 ± 3.5%</td>
<td>44.1 ± 6.7%</td>
<td>44.1 ± 6.7%</td>
</tr>
</tbody>
</table>

Please note: the 'class' concept is challenging.

Results on generic DATE block detector. Please note that the prior is very small, 30% is not so bad. Here positional density was used to weigh likelihoods (positional expectancy model).

With a dimensionality of 50k, it really becomes useful to do dimensionality reduction. NNs are in many ways convenient, but it is difficult to convince biomedical researchers.

The goal of this study was writer identification on musical script. The staff lines do not represent relevant information. How to get rid of them? In the image or in feature space?
Composer/copyist identification is better if staff-line is removed by autoencoder in feature space, than by traditional image processing. 'Deep' wins.

Niituma, Masahiro and Schomaker, Lambert, van Oosten, Jean-Paul, Tomita, Yo and Bell, David (2016). Musicologist-driven writer identification in early music manuscripts, Multimedia Tools and Applications, 75(11), pp. 6463—6479

abstract="Recent renewed interest in computational writer identification has resulted in an increased number of publications. In relation to historical musicology its application has so far been limited. One of the obstacles seems to be that the clarity of the images from the scans available for computational analysis is often not sufficient. In this paper, the use of the Hinge feature is proposed to avoid segmentation and staff-line removal for effective feature extraction from low quality scans. The use of an auto encoder in Hinge feature space is suggested as an alternative to staff-line removal by image processing, and their performance is compared. The result of the experiment shows an accuracy of 87{\%} for the dataset containing 84 writers’ samples, and superiority of our segmentation and staff-line removal free approach. Practical analysis on Bach's autograph manuscript of the Well-Tempered Clavier II (Additional MS. 35021 in the British Library, London) is also presented and the extensive applicability of our approach is demonstrated."

http://dx.doi.org/10.1007/s11042-015-2583-8

Deep learning  ‘Deep Genomics’

- Example in our own lab with NN: tissue classification, via autoencoder, also classifying on residuals

Advantage: data sets can be much larger with NN than in standard numerical tools such as PCA/SVD. Franke et al (2012) showed that 80{\%} of singular values are ‘the boring RNA stuff’, i.e., concerning processes that are the same in all cells.

Rudolf Fehrmann, Lude Franke et al. found this. They have several publications, among which one on SSVD (sparse svd) in Nature.
Define ‘deep’!

› Is it the convolutional aspect?
› Is it the number of layers?
› Is it the dimensionality reduction?

Two notions of ‘deep’

 › Deep: is hidden, geometric, averaged over concrete instances, a subspace, etc., like PCA, correlation patterns, non-verbal

    Yes!

 › Deep: is an exact fact, hidden in a graph, an unexpected explanation, precise, explainable to humans?

    Not yet!
    As e.g., in causality inference ('deep cause')

Time to identify what cannot be done!

 › Deep Learning is no computational intelligence, yet
 › Intelligence by proxy; over the supervised labels
 › A smart human PhD is always necessary
 › No general intelligence: each experiment is a one-trick pony
 › Extensive, laboratory-based training
 › Generalisation to real, new data from new sensors, from new contexts is still difficult: k-fold evaluation is still a scam:
 › ’i.i.d.’ and sampled from one cleaned pool of data yields overly optimistic performance estimates
Intelligent learners should be able to know what they don’t know and propose experiments (i.e., samplings from the total data) to improve the current learning status.

This example can be read quickly by most human subjects. This did not involve a (brain-based HMM) training on these words including the random permutation probabilities. Rather, an opportunistic use is made by the human reader, of the fact that: The first and last letter are ok; and that the other letters only show their presence within the word while their position is irrelevant. All this is done on the fly, in the operational stage, by humans at least.

Experiments performed with Hans-Leo Teulings and students in my Nijmegen era (pre 2001).
koekstommel achterdeur democratie

Condition B

koekstommel achterdeur democratie

Condition A
Results featural context experiment

- Test words were handwritten (natural, shopping note)
- Same-writer flanking words (72% correct):
  - Better human word recognition
    than different-writer flanking words (54%)
- Conclusion: human readers exploit similarities at the letter and feature level, 'live' per trial

isogeny principle (Baird/Nagy)


Scaling Up Whole-Book Recognition Pingping Xiu & Henry S. Baird Computer Science & Engineering Dept Lehigh University 19 Memorial Drive West, Bethlehem, PA 18017 …
Published in: international conference on document analysis and recognition · 2009
Authors: Pingping Xiu · Henry S Baird

Functions that are needed

- One-shot learning ⇒ attribute classification
- Transfer learning ⇒ reusability of skills
- Improved unsupervised learning
- Systems that are adaptive 24/7 (always on)
- Active learning: knowing what you don’t know
  - Identifying information that would help to disambiguate
- ⇒ cognitive architecture
  as opposed to rigid pipeline processing

Timing: You really should be halfway now
Large-scale processing of handwritten historical documents: The Monk system

Monk e-Science web service addressing these questions:

- **What?** Word retrieval by 24/7 machine learning
  - Aim: Build a European Google for handwriting

- **When?** Medieval manuscript dating
  - Where? Geographical localization
  - Goal: uploading of charters from 1300-1550 on a server

- **Who?** Writer identification
  - On Monk server or on internet
  - Using GIWIS Windows tool

What? (is written): recognition and retrieval of text
Slide 30

When? (has it been written)

When? Qumran ERC project with Mladen Popovic
Dual-mode time axis estimation (PhD: Maruf Dhali)

In the MPS project (PhD student: Sheng He), we developed textural methods for
dating of acts. However, also individual
segmented characters show
paleographical developments in a
traditional manner.

Slide 31

Medieval Paleographic Scale (MPS)

In the MPS project (PhD student: Sheng He), we developed textural methods for
dating of acts. However, also individual
segmented characters show
paleographical developments in a
traditional manner.

Slide 32

Writer identification (1:N)
and verification (1:1)

1. Interactive,
fully manual
2. Automatic,
ROI based
3. OCR based

Who? (wrote it)

It is much less work to just define a region
of interest for writer identification (or
dating)
People who has worked on Monk or had an influence on its development.

Also humanities scholars: They are high
Current technology: neat text/known language

- Why is ‘OCR’, i.e., letter by letter transcription on handwriting so difficult?
- Machine print: per character, per font, 8000 training examples are needed, typically
- E. Barney-Smith: 200k instances of printed e vs e
- Address reading: reduced lexicon, zip codes etc., help
- In linguistic modeling: 20th century newspaper corpora do very little for 15th century acts
- Literary text, acts and charters each need their own knowledge models in order for OCR to work

For example: What would a TREC corpus do for medieval administrative text?
In any case, shape recognition needs to be strong if such additional sources cannot be used.

Handwriting recognition: eat this!
- Many languages, scripts
- Over historical periods
- Contractions of letters
- ‘Suggested’ sloppy letter shapes
- Individual writer styles
- Image problems
- Sliding window for character search usually problematic:

Exit OCR as: Exit the methods that assume identifiable individual characters in the input stream for all letters of the intended word.

Monk -

1. Don’t promise perfection
2. Don’t promise ‘transcription’
3. Don’t promise exhaustive coverage (as in databases)
4. Make use of human trainers, volunteers

Word retrieval / word spotting:
   * "a Google for handwritten documents"
Monk -

Design considerations

• Word based:
  • "a Google for handwritten documents"
  • The word is a reliable chunk of information with many shape features: redundancy

• Big Data: With sufficient data, there is always a reasonable response on a query

Monk's world model:

• Institutes
  • Collections
  • Books (i.e., documents)
  • Pages
    • Paragraphs
      • Lines
        • Word zones and characters
        • Pixels

Note that almost all documents are handwritten, but some machine printed text is also in Monk. This entails difficult material such as German fraktur, Arabic and printed hieroglyphs.
While binarisation is usually too destructive, Otsu-based contrast enhancement works well, especially if local Otsu (or other method) is used.
Fig. 4. Field drawing of a red-throated Barbet (Begalaima mystacophanos), Buitenzorg, Java, May 1827. Each field drawing contains place names, dates, scientific names and person names. The NWO Making Sense project.
Several years of experimentation (online 3300+ days)

- started in 2005
- Monk was switched on, to a largely autonomous mode, in 2009

For developing and optimizing two functions:
- Retrieval: return images for a given keyword
- Recognition: return the most likely word given an image

Contrary to expectation, a good classifier for recognition (in terms of recall and precision, e.g.) is not guaranteed to provide an a posteriori likelihood that is useful for intuitive ranking. For retrieval, other methods may be more applicable. The distance to a centroid is more informative in this respect than the distance to a separating boundary.
distracting images is strongly reduced, followed by nearest-centroid based instance ranking that retains an intuitive (low-edit distance) ranking. We show that in handwritten word-image retrieval, precision improvements of up to 35 percentage points can be achieved, yielding up to 100% top hit precision and 99% top-7 precision in datasets with 84,000 instances, while maintaining high recall performances. The method is conveniently implemented in a massive scale, continuously trainable retrieval engine, Monk.

Users look at lines of text or at hit lists, a word model is computed, a new ranking is computed, presented to the user(s), and so on. In the early stage of the Monk development, Blue Gene (12k cores) was used for doing a grid search on optimal MLP configurations. However, centroid search proved to be much more convenient such that the current computational requirements are less demanding.

Labeling induces changes in the knowledge state, and leads to retraining, using a queuing of jobs in HPC.

As another metaphor, the Monk system is like a blossoming tree presenting flowers (word zone candidates) to bees (human labelers). The large surface that is needed was provided by the IBM gpfs file system.
Lessons learned during Monk development - 1

- A shape feature which is powerful for Retrieval may not be strong in Recognition!
- Requirement B: hit list should provide nice, intuitive ranking in a satisfying ‘hit list’
- Requirement A: target word class should survive competition with the other word classes (needle from the hay stack)

Problems with products of probabilities - 2

Draft can be found on arXiv, I am still working on this.
Lessons learned during Monk development

- Ballpark principle
  - no label: kmean, Rätschen, neural gas etc.
  - one label: 1NN - first nearest neighbor
  - ~5 labels: NC - nearest centroid (mean)
  - >20 labels: SVM
  - >100 labels: MLP's
- But nearest centroid is by far the most satisfying in a real big data context

Movie of Shin converging to a stable probability landscape for ink.
Normalisation on the basis of center of gravity and standard deviation of the radius, times a factor such as 2.5 to cover the ink sufficiently. At N=1878 instances:

Movie for the word Amsterdam, much less samples but still converging to a stable mean. This principle yields an attractor, also in other feature spaces, of which the inertia prevents drift in an online (24/7) learning setup. At N=498:
The paper in PAMI 2008 gave us the confidence that whole-word feature approaches can be very powerful. The Serre/Poggio neural network was a bit complicated and Matlab based, so I developed a more technical feature method for words in this period, that was generic enough to handle a wide range of handwritten scripts. Not only in averaged images but also in averaged feature spaces, big data or more than a hundred examples per class yield very stable models.

It is advisable to use Bhattacharyya or Chi-square on feature vectors that represent probability. I am sorry if that reduces the shine of your more complicated method. Up to 10% improvements in performance with respect to Euclidean or Manhattan can be observed in a wide range of pattern recognition problems.
Each trace represents the harvest curve for a book.

Please note phase transitions: at some point during label harvesting, the visual word models become successful in attracting new unseen instances and provide a clean hit list that is easy to confirm by the human users:

Phase transitions in the training!

Woodblock-printed documents (higher curves) elicit more labels than handwriting, as can be expected. Still, this category is considered ‘difficult’ by many researchers in Chinese script recognition and by the users in the humanities.
This slides shows a **phase transition** due to easy labeling of a correct hit list with about 60 correctly recognized and top-ranked instances.
Catalogue of mixed printed and written Chinese characters.

Baron is the target word. Words can be individually confirmed or per visible hit list as a whole.
126 manuscripts
Train: 25 words/class
Test: 25 words/class
Lexica: 1373 words (avg)

Monk usage example: May 2013 Qumran scrolls:
Daniel Stoekl (Sorbonne), Mladen Popovics (Groningen)

Continuous learning in Big Data
Qumran scrolls: 2400 photographs
- Using Monk for character labeling
- With Daniel Stoekl and Mladen Popovics
- Using its 24/7 machine-learning cycle:
  - Label \(\rightarrow\) Train
  - Label some More \(\rightarrow\) Train
  - Easily label Many
- Thousands of characters ‘mined’ out of the Qumran collection of photographs in just two weeks, with very little effort in human labeling

Continuous learning in Big Data
Conclusions

- Deep learning is a powerful concept
- But it is not enough, for building autonomous and intelligent agents
- The challenge is to design systems that handle unseen problems
- Part of deep-learning success may be just caused by the amount of data; better comparative evaluation is needed
- Iterative recognition and ranking works great!

- Engineered neural-network architectures vs engineered features: we’re still not there, either way!
We consider hieratic script on papyrus the most difficult material in Monk at the moment. Hieroglyphs are easier to recognize, but for interpretation a stochastic grammar needs to be trained (2D).