Deep Neural Networks

Where Do We Stand in Handwriting Recognition?

(Part II)

Who am I?

Théodore Bluche <tb@a2ia.com>

PhD defended at Université Paris-Sud last year

Deep Neural Networks
for Large Vocabulary Handwritten Text Recognition

Now working as a Reasearch Engineer at **a2ia** in Paris

... automatic document processing (handwriting recognition and more...)
... part of the research team (6 people)
... implementation of new neural networks
... improving the speed and accuracy of production models
... build the models of tomorrow

What have we seen so far...

→ Good **deep neural networks** as optical models of HWR

→ Good results with CTC and RNN (i.e. **predicting chars directly**, no HMM = no need to tune char length models)

→ Good results with sliding windows of **pixels** (= limited need for feature extraction)

BUT ...

- *... careful preprocessing*
- *... sliding window = early 2D → 1D conversion*
- *... assumption that text lines are available / segmented*

Spoiler!

Before I started my thesis, Graves et al. came up with a system

- made of deep nets
- trained with CTC (character sequence prediction)
- accepting pixel inputs
- without sliding window
- without preprocessing
- winning all international evaluations

(My colleagues at A2iA were all playing with …)

Multi-Dimensional Long Short-Term Memory Recurrent Neural Networks

End-to-End Handwriting Recognition

This is **attractive** :

→ you can just **throw your raw data in the training program** and wait for the result

That makes the creation **of models for new data / languages easier**

... that's why MDLSTM-RNNs are now in our products (a2ia website)

... but there are *still drawbacks, problems and challenges*

(e.g. still need to find the text lines, not as easy to segment characters as HMMs, ...)

Outline of this talk

- **End-to-End HWR -- from pixels to text**
  - Multidimensional Recurrent Neural Networks
  - A few results and tips
  - Limitations

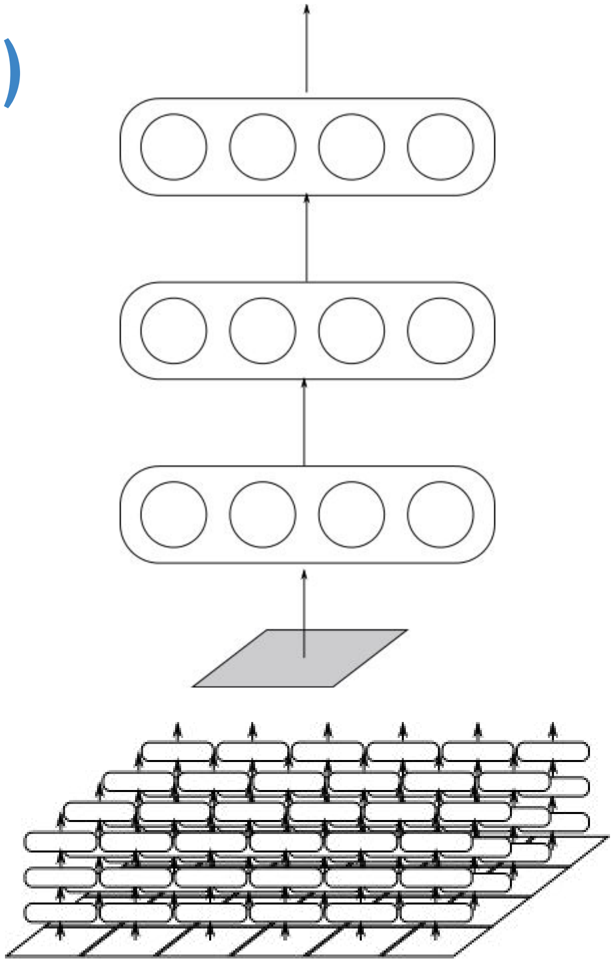
- **Beyond textlines -- segmentation-free recognition of handwritten paragraphs**
  - Attention-based models
  - A few results
  - Limitations

- **Future challenges ...**
- **Open discussion**

Outline of this talk

- **End-to-End HWR -- from pixels to text**
  - Multidimensional Recurrent Neural Networks
  - A few results and tips
  - Limitations
- **Beyond textlines -- segmentation-free recognition of handwritten paragraphs**
  - Attention-based models
  - A few results
  - Limitations
- **Future challenges …**
- **Open discussion**

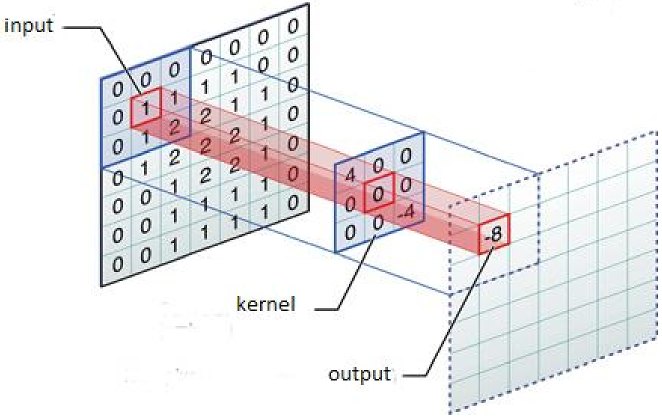
Neural Networks for Images (pixel level)



→ Instead of a feature vector, the **input is only one pixel value** (or a vector of 3 RGB values for color images)

→ The network is ***replicated*** at each position in the image

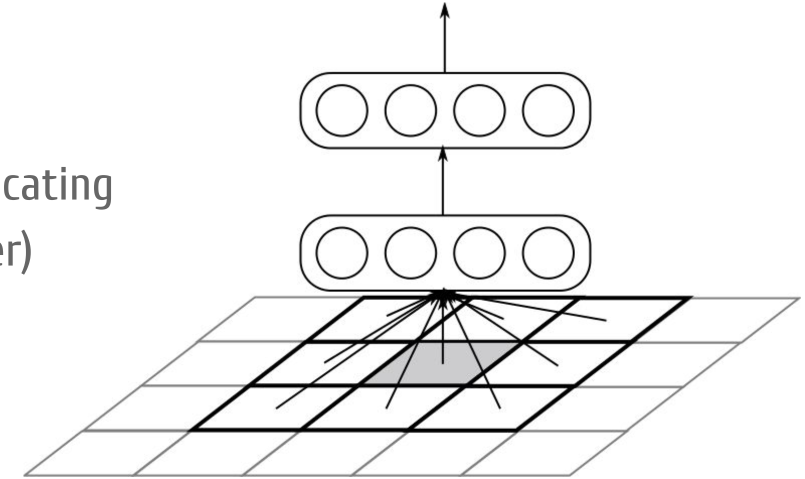
Convolutional Neural Network



→ We can include spatial (structured) context :

instead of giving 1 pixel value at the current position, we give the values of all pixels in a given neighborhood

→ Replicated at all positions = **convolution**, with kernel defined by the weights



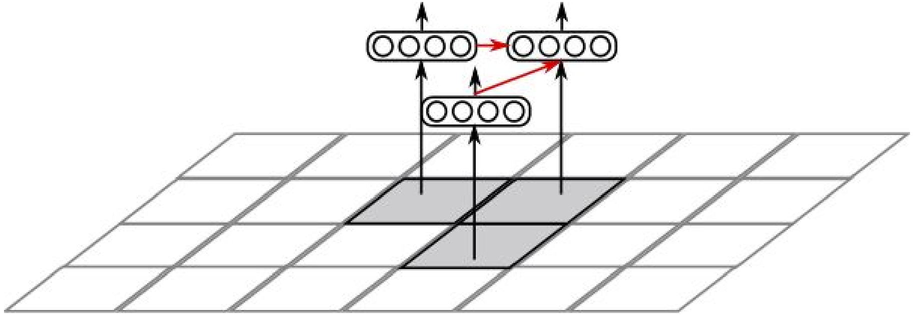
→ You can reduce the size of the feature maps by replicating the net every *N* positions (output will be *N* times smaller)

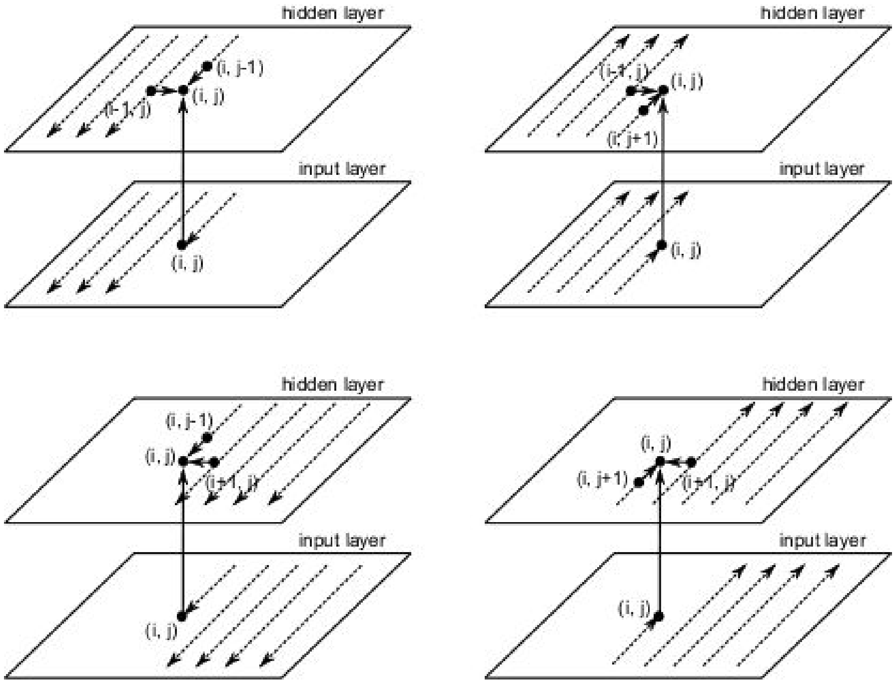
→ (nb. the sliding window of pixels = first layer was a convolution)

Multi-Dimensional Recurrent Neural Networks

the input at a given position includes the outputs of the same layer at neighbors

→ in MDLSTM cells, 2 forget gates, 2 inner states merged



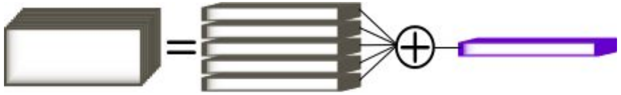


Multidimensional RNN

→ MD Recurrent + Convolutional layers

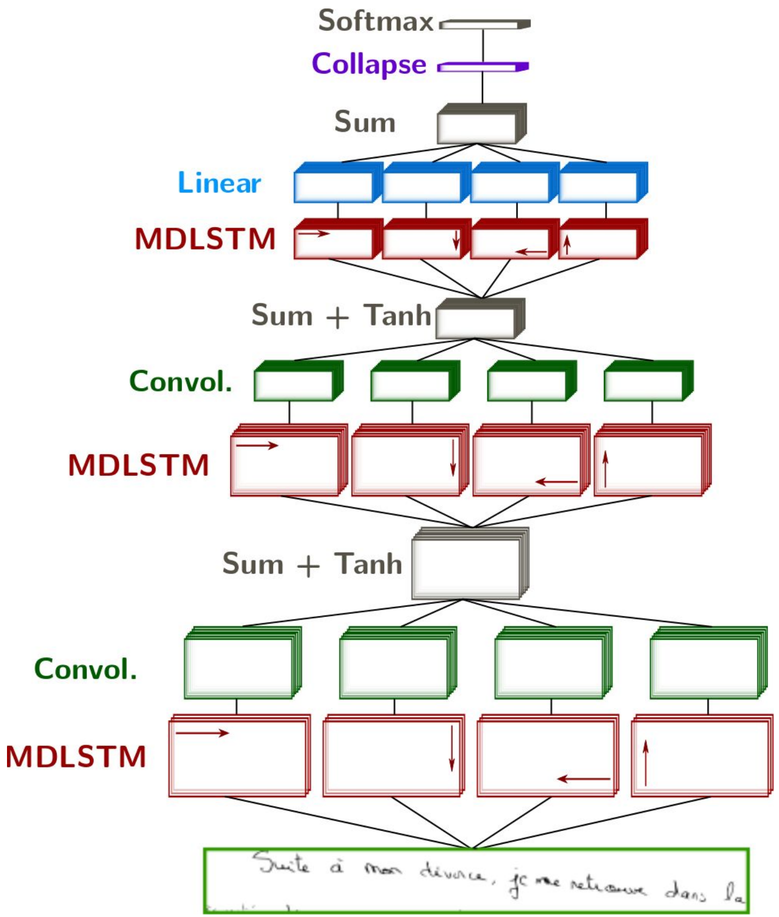
→ applied directly to the pixel of the raw text line image

→ A special Collapse layer on top to get sequential representation

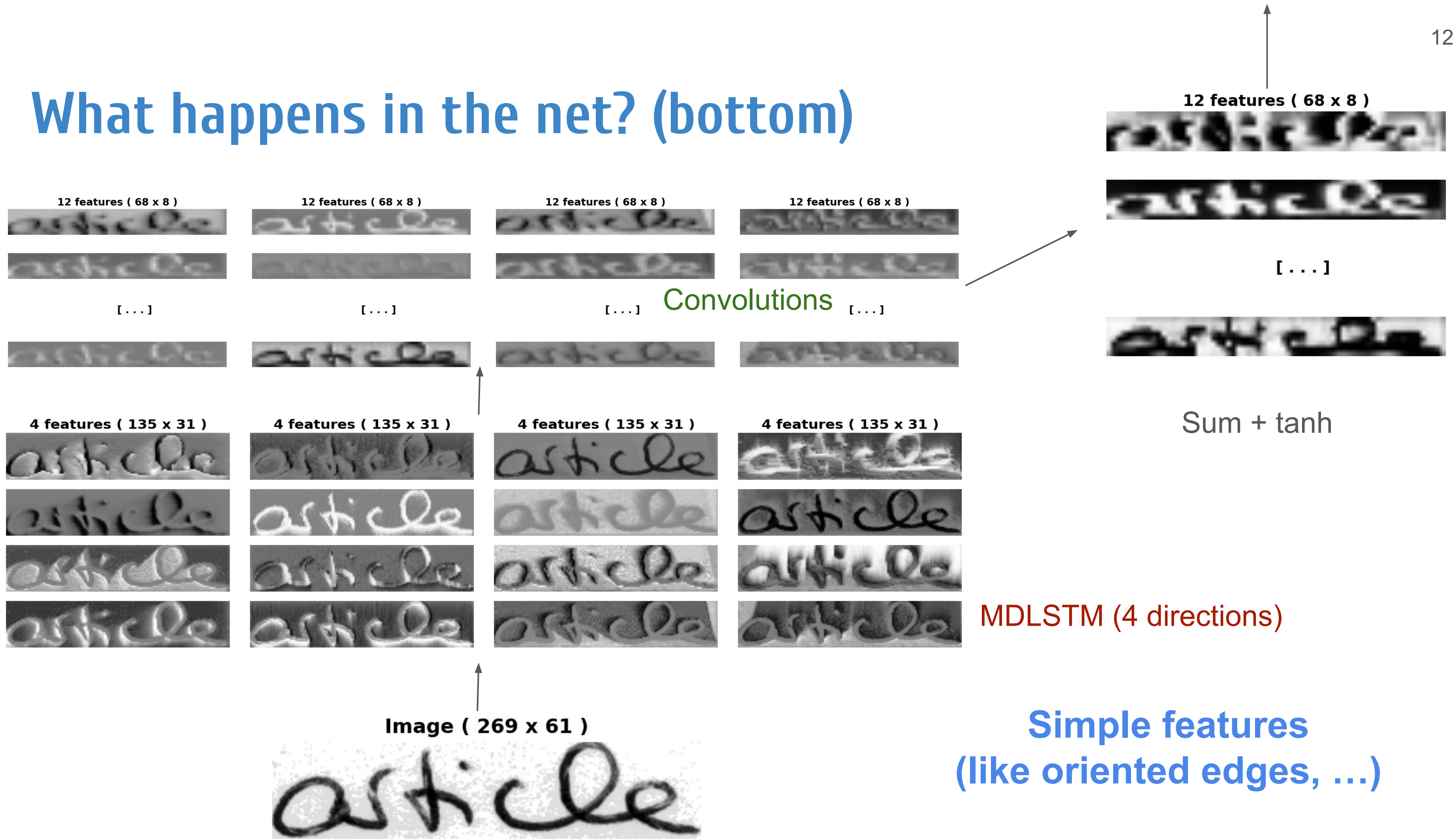


→ Trained with CTC to output character sequences

Current State-of-the-art!

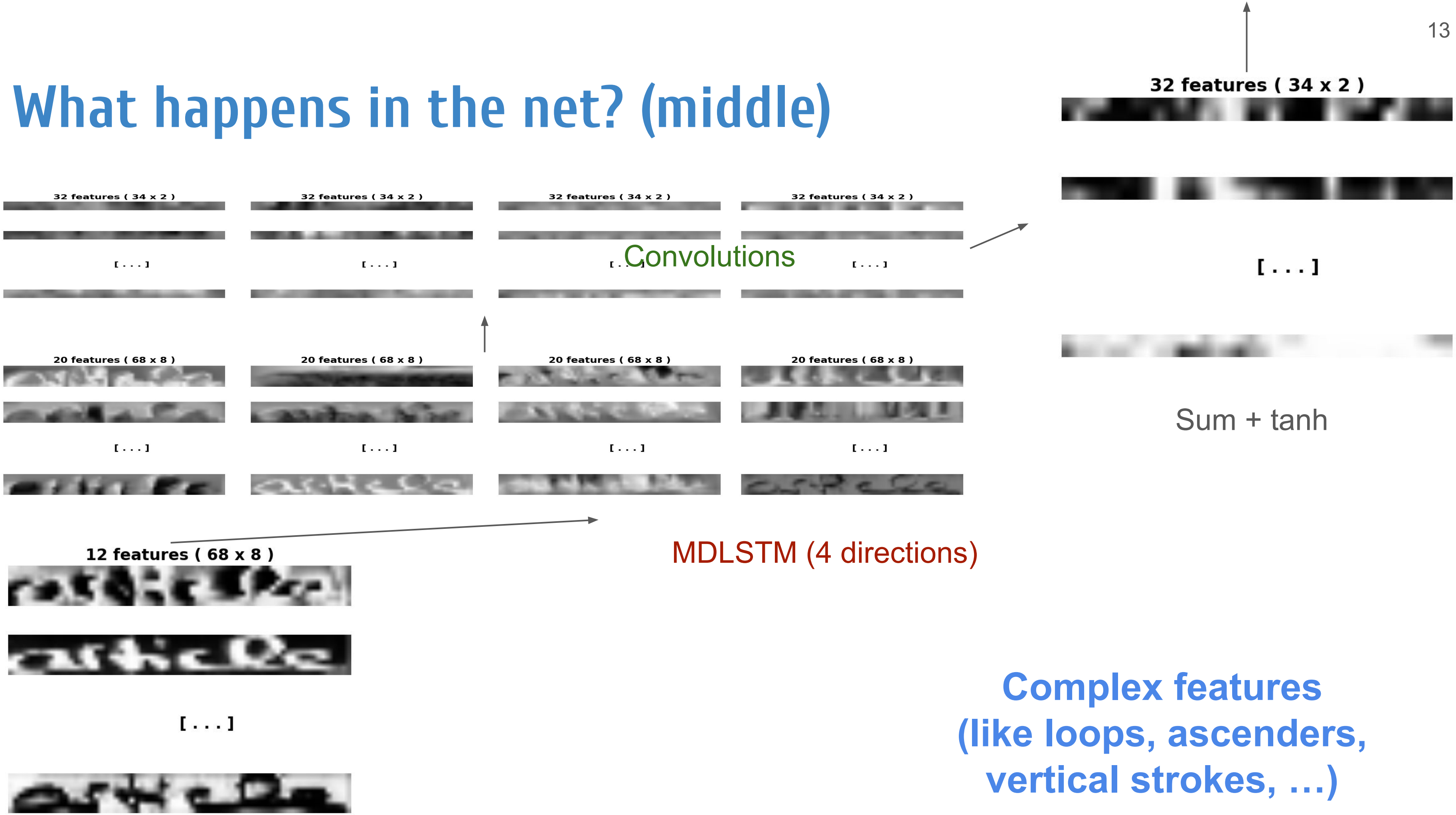


What happens in the net? (bottom)



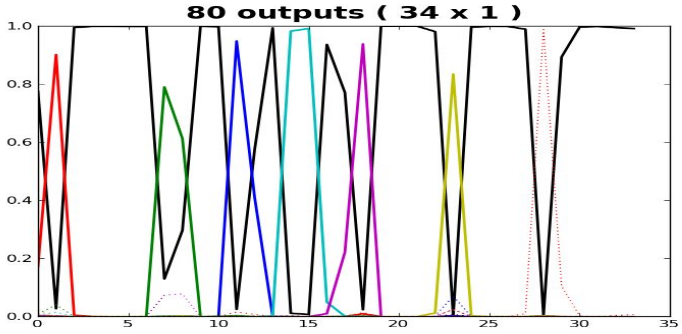
Simple features (like oriented edges, …)

What happens in the net? (middle)



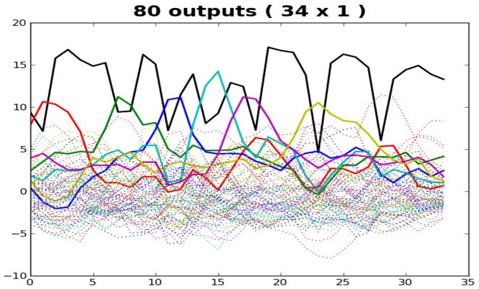
Complex features (like loops, ascenders, vertical strokes, …)

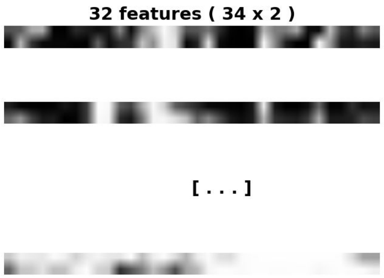
What happens in the net? (top)



Softmax

Collapse





MDLSTM (4 directions)

More abstract features (combination of features, closer to character level…)

Some results ...

Database	Rimes	IAM	Bentham
Best feature system (Part I)	12.6	13.2	10.2
Best pixels system (Part I)	12.4	13.3	11.5
MDLSTM - RNNs	12.3	13.6	8.6

WER (%)

Won all latest HWR competitions!

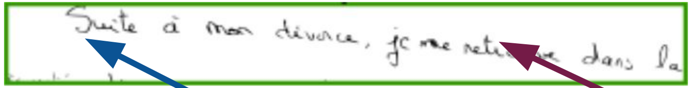
- OpenHaRT 2013 (Arabic)
- Maurdor 2013 (French, English, Arabic)
- ICDAR 2014, ICDAR 2015 (Old English)

Tips & Tricks

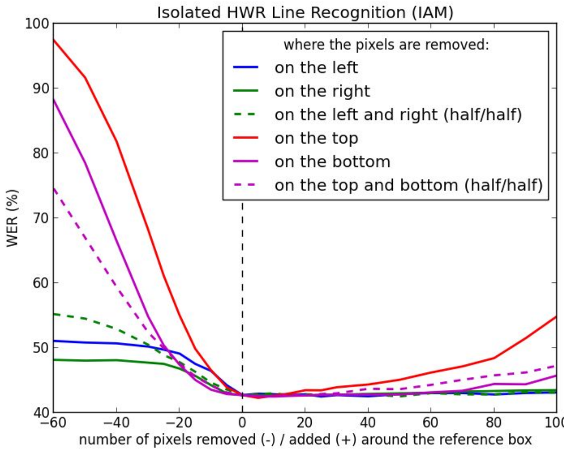
- Graves' architecture work very well
  - 2x2 tiling, 4x4 MDLSTM, 12 Conv. 2x4/2x4, 4x20 MDLSTM, 32 Conv. 2x4/2x4, 4x50 MDLSTM, Linear, Collapse
  - Learning rate = 0.001
  - !! weight initialization is important, GRADIENT CLIPPING in gates is crucial
  - Every modification we tried except dropout made results worse!
- Reimplement RNNlib
  - multithread the 4 directions of LSTM
  - use block operations as much as possible
  - !! the double `for` loop is costly, especially in the first layers
- For CTC with textlines (long sequences) → curriculum learning (Louradour et al. 2014)
- Start with an pre-trained RNN (e.g. train on IAM, finetune on your Db = works well even with less data or different languages)
- Regularize! (e.g. with dropout), because MDLSTMs overfit

Limitations

Machine learning on raw data = data(set)- and cost-dependent!



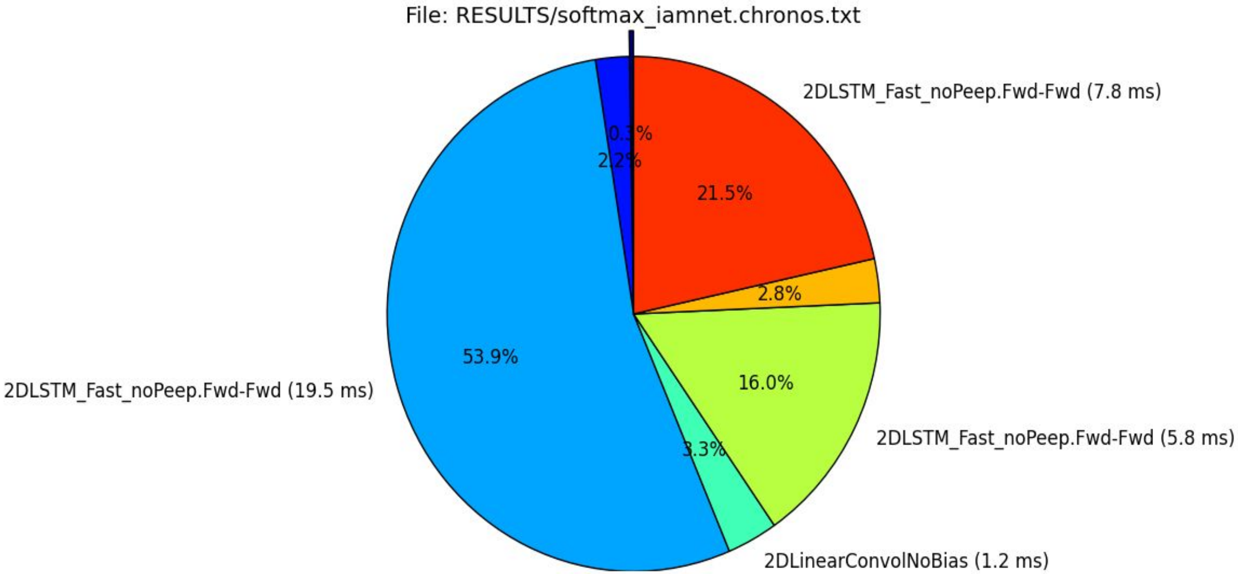
In the first MDLSTM layer, you don't prevent ***this pixel*** …



… to have an impact on the feature computed at ***this position***

The learnt features won't be local!

Limitations

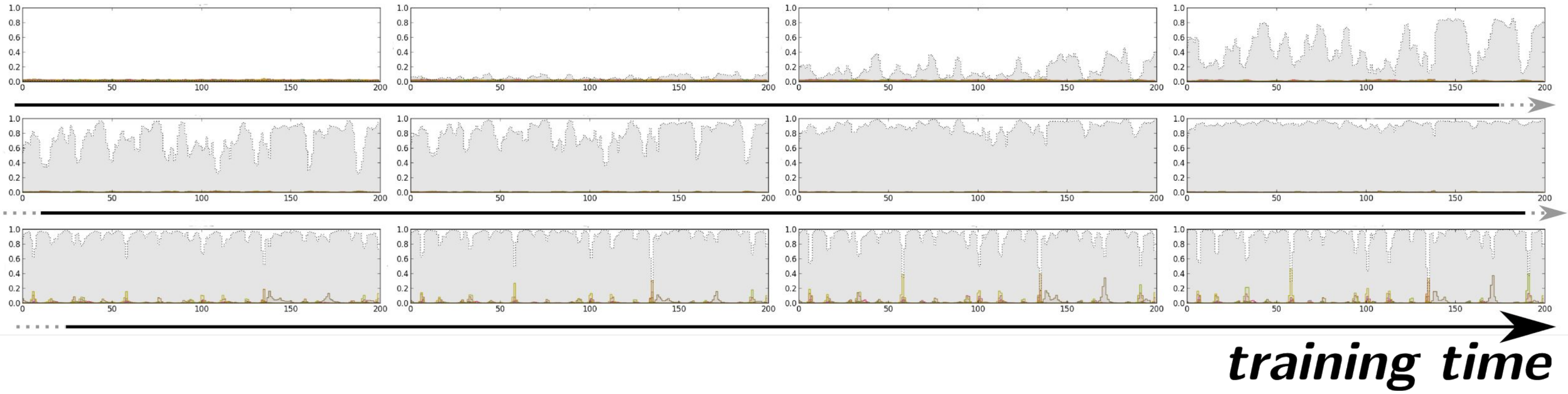


The **first LSTM takes more than half the computation time** to only extract low-level features!

→ position-wise computation on high-resolution images

Limitations

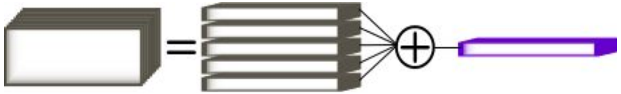
With CTC training, you **cannot retrieve the character positions**, and character predictions will be localized (peaks).



cf. http://www.tbluche.com/ctc_and_blank.html

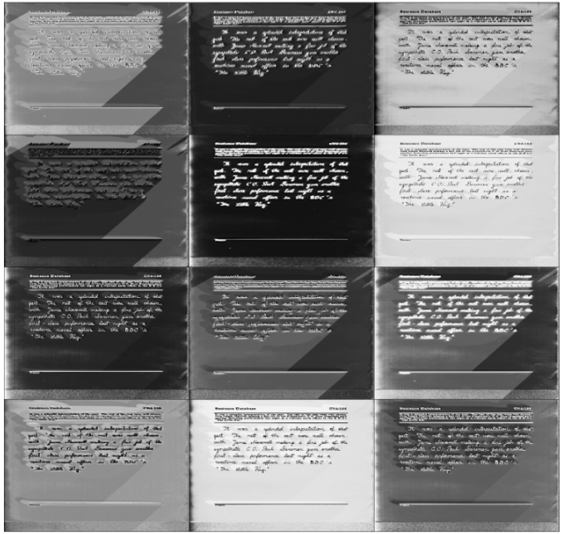
Limitations

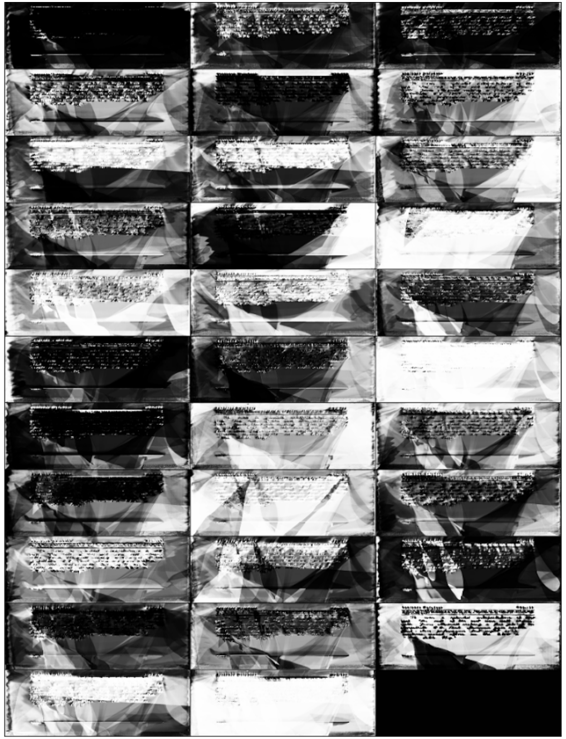
The Collapse layer :



- prevents the recognition of multiple lines
- gives the same importance to all positions across the vertical axis
- propagates the same gradient at all positions
- hence prevents using the intermediate representation as features for images representing more than one line (that and the MDLSTM not local enough)

Example - Post-LSTMs feature maps on paragraphs





Outline of this talk

- End-to-End HWR -- from pixels to text
  - Multidimensional Recurrent Neural Networks
  - A few results and tips
  - Limitations
- Beyond textlines -- segmentation-free recognition of handwritten paragraphs
  - Attention-based models
  - A few results
  - Limitations

NEW

- Future challenges …
- Open discussion

From line reco. with MDLSTM-RNN + Collapse and CTC ...

- line-per-line
- fixed reading order
- many predictions with fixed step size and map to character sequences
- sentitive to line segmentation

e.g. CER (%) on different line segmentations with MDLSTM-RNNs + CTC on IAM

Resolution (DPI)	Line segmentation			
	GroundTruth	Projection	Shredding	Energy
90	18.8	24.7	19.8	20.8
150	10.3	17.2	11.1	11.8
300	6.6	13.8	7.5	7.9

… to paragraph reco. char-by-char

General idea:

- process the whole image **without line information**
- make only **one prediction per character**
- at each timestep, predict the current character and **where to look next**

→ Attention-based Neural Networks

An attention neural network

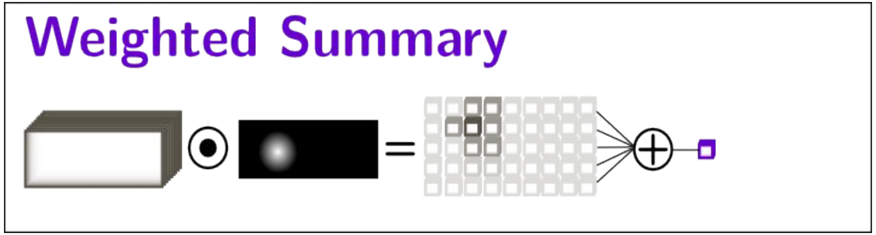
At each timestep *t* ...

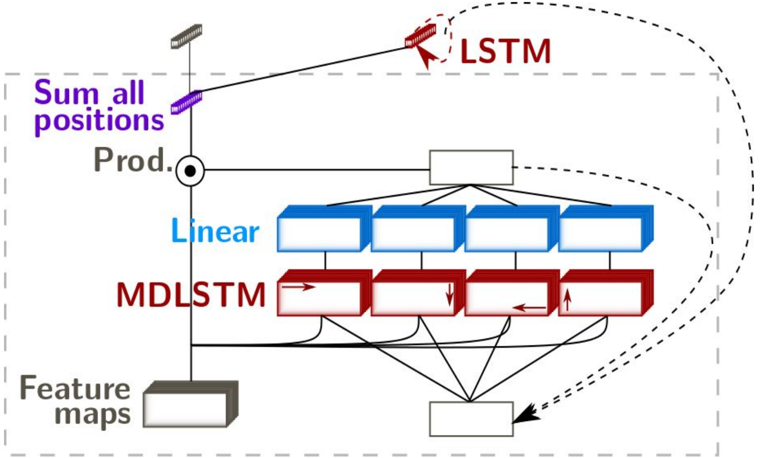
- An **attention network** predicts a probability distribution over positions in the feature maps
- The attention probabilities are used to compute a weighted sum of the feature vectors

$$\mathbf{a}^t = \{a^t_{ij}\} = AttNet(\mathbf{f}, \mathbf{a}^{t-1}, s_{t-1})$$

$$g_t = \sum_{i,j} a^t_{ij} f_{ij}$$

→ The **Attention Neural Network** predicts *where to look next*



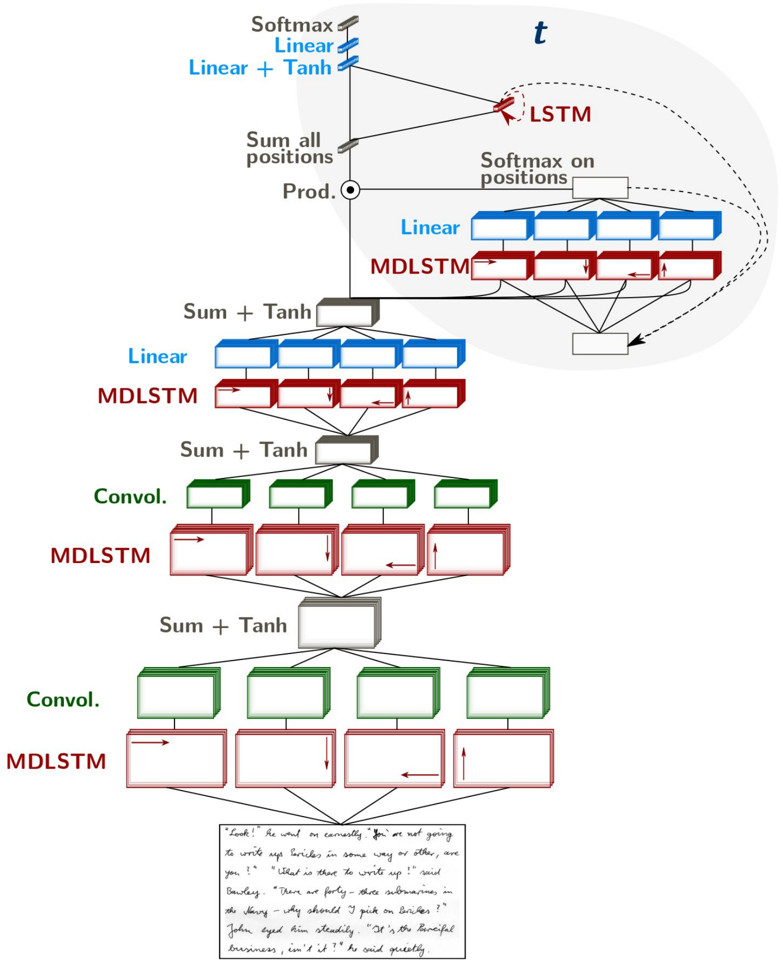


Attention Neural Network

The network is made of ...

- An **encoder** of the image into high level features
- An **attention network** iteratively computing weights for these features
- A **decoder** predicting characters from the sum

→ The attention net + decoder is applied *N* times
→ The whole net predicts characters + a special <EOS> token when it is done reading



Training

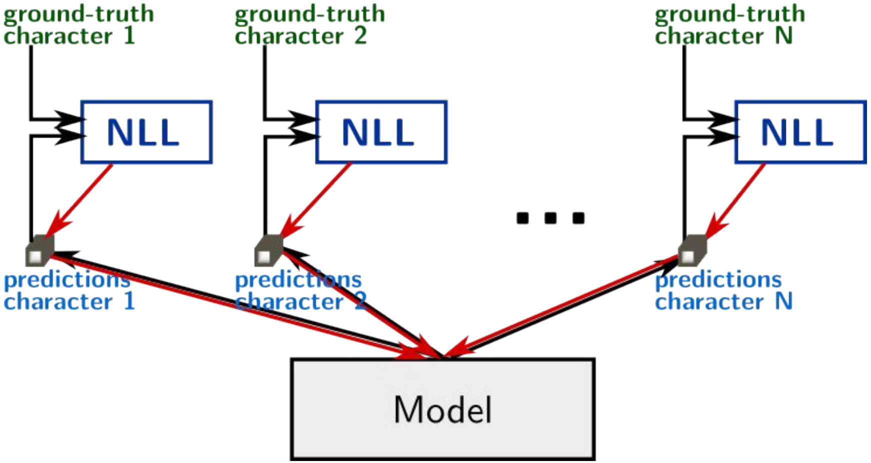
The net predicts one character at a time

$\longrightarrow$ **no need for CTC**

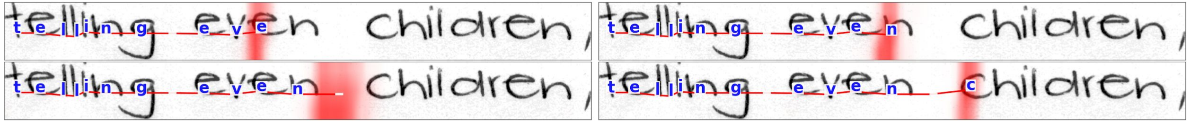
Loss :

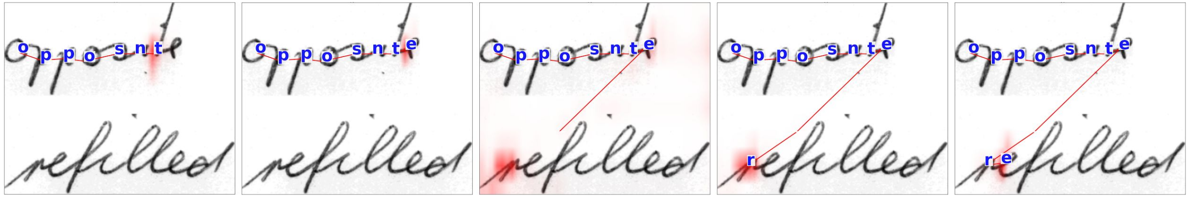
$$-\sum_{n=1..N} \log p(c_n|\mathbf{x})$$

i.e. forces the network to predict the first char at t=1, then the second one, etc...

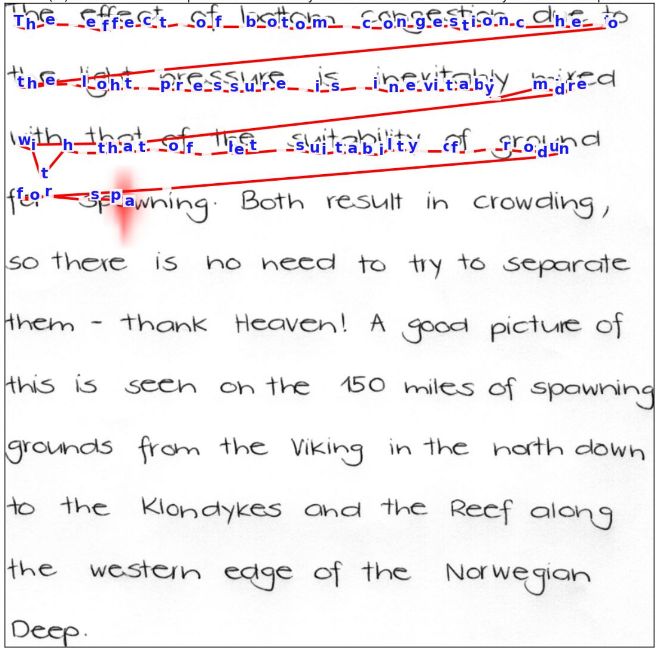


Attention Neural Network – Illustration





(...)tion che to the loht pressure is inevitably mired with that of the suitabilty of round for spa



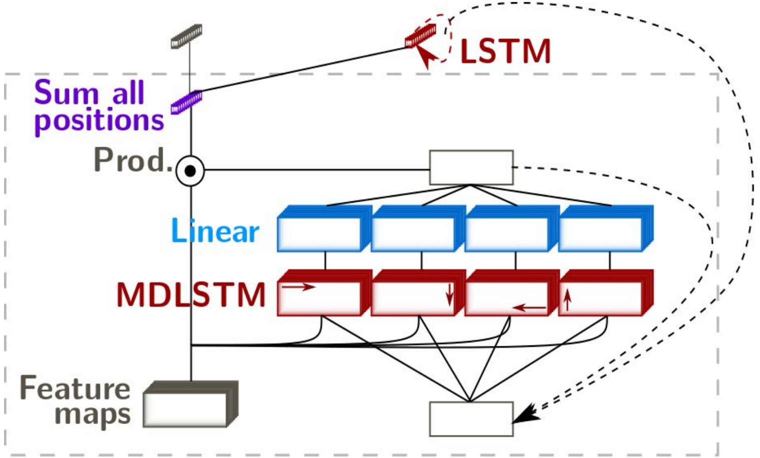
Results and Limitations

Resolution (DPI)	Line segmentation				Attention-based (*this work*)
	GroundTruth	Projection	Shredding	Energy	
90	18.8	24.7	19.8	20.8	-
150	10.3	17.2	11.1	11.8	16.2
300	6.6	13.8	7.5	7.9	-

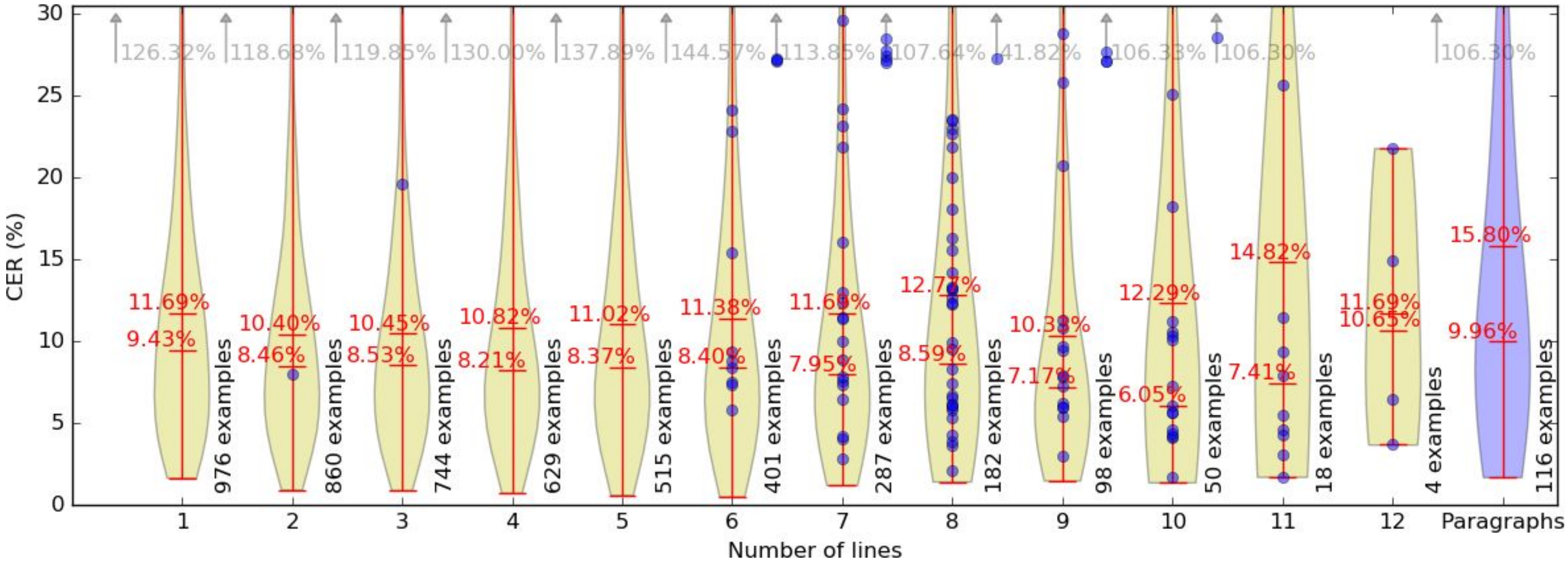
Table 1: Multi-word recognition results (CER%).

Model	Inputs	CER (%)
MDLSTM + CTC	Full Lines	6.6
Attention-based	1 word	12.6
	2 words	9.4
	3 words	8.2
	4 words	7.8
	Full Lines	7.0

- Need a good curriculum
  (1 line ⟶ 2 lines ⟶ Paragraphs)
- Attention net + decoder applied
  ~500x / paragraph
  ⟶ **time/memory inefficient**
- no language model (more difficult to integrate)



Detailed results



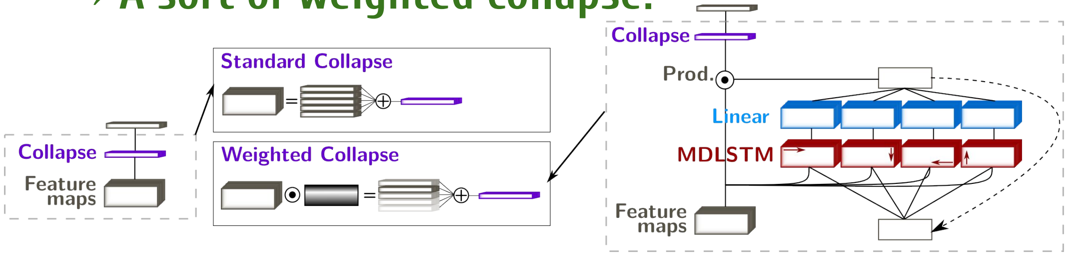
Aggregated error rates are penalized by the attention sometimes reading the same line multiple times... (> 100% error rate)

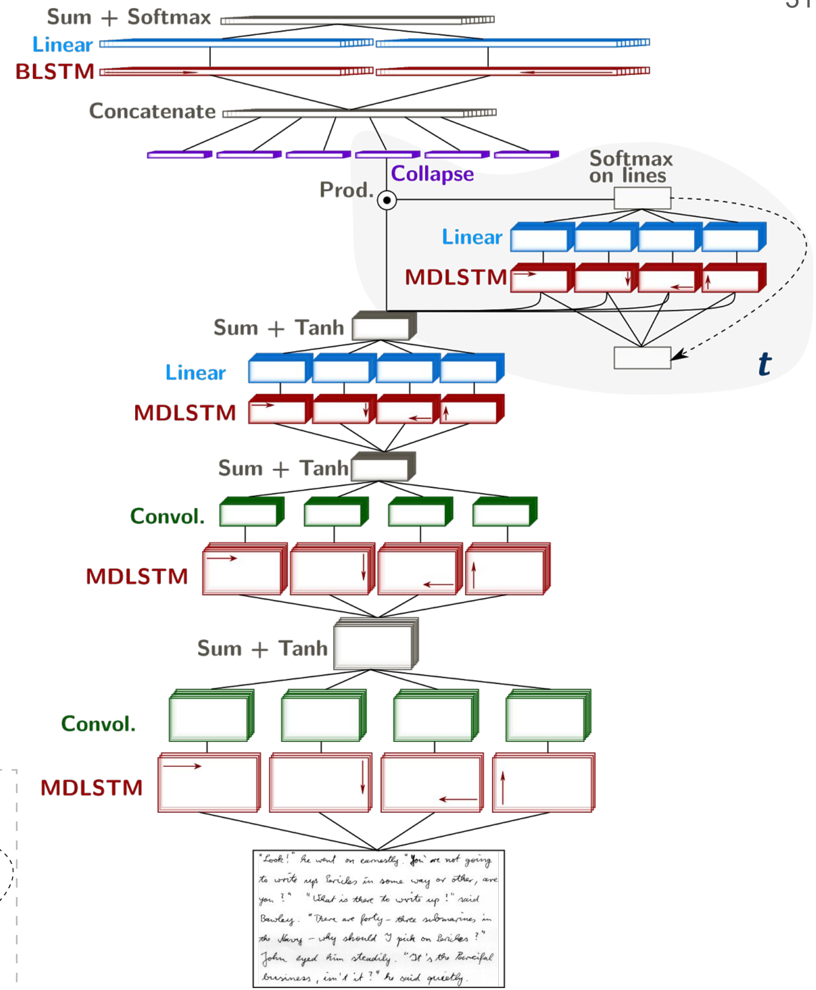
Attention-Based Collapse

To be more efficient :

- The **attention is now put on lines** and not on characters
- = softmax on each column (not on the whole map)
- sum column/column
- back to CTC

→ **A sort of weighted collapse!**

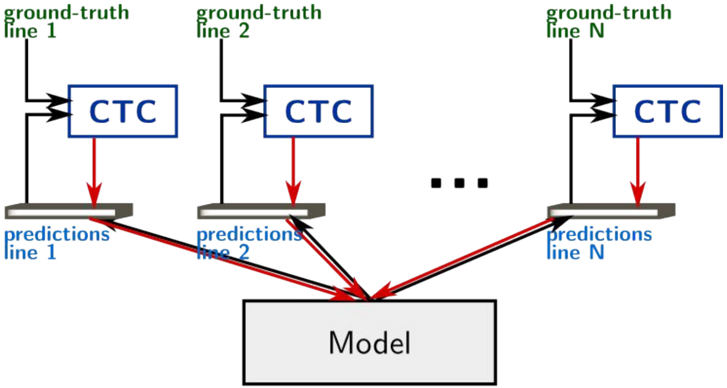




Training

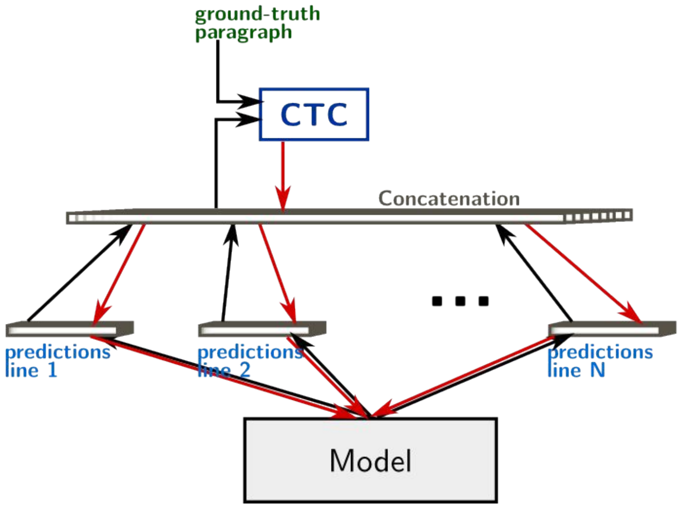
→ Case 1 : we know the line breaks

We can apply the CTC restricted to each line for each timestep



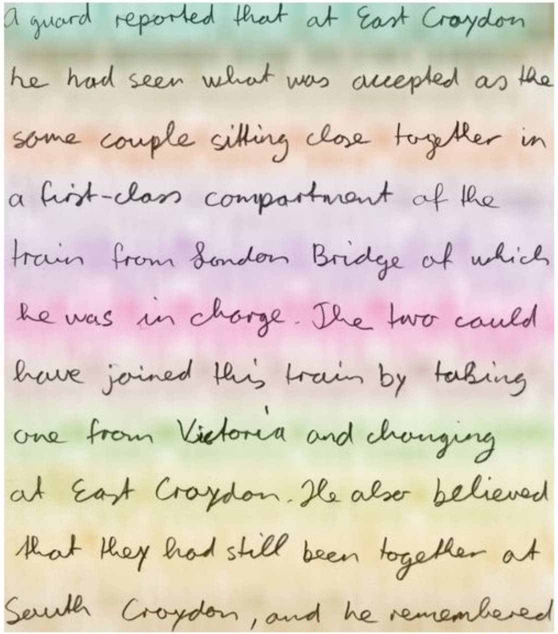
→ Case 2 : we only have the paragraph annotation

We can apply the CTC to the complete reco with the whole paragraph trancript

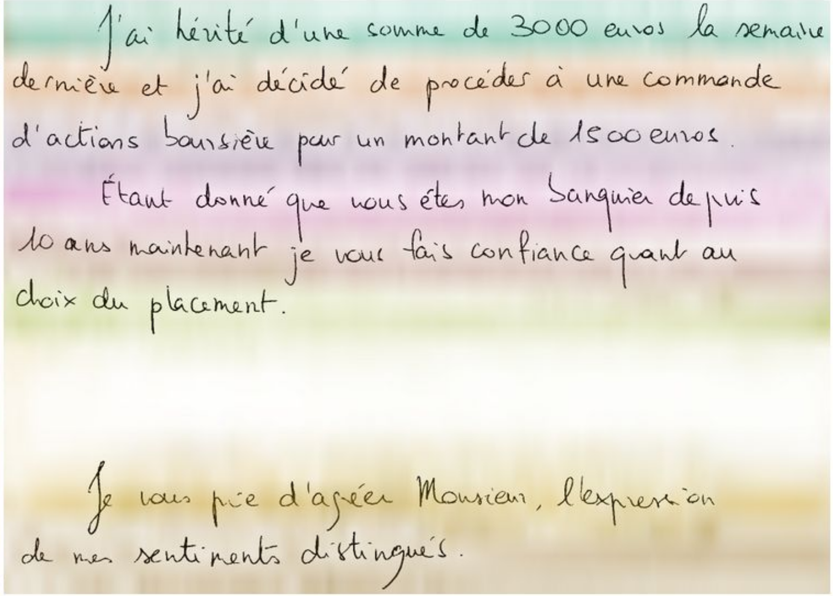


nb. : in many availabe corpora (e.g. in DH), that is the case!

Qualitative Results



A guard reported that at East Craydon he had seen what was ouepted as the some couple sitting close togther in a first-class compartment of the train from London Bridge of which he was in charge . The two could have joined this train by taking one from Vectorin and changing at East Craydon . He also believed that they had still been together at South Croydon , and he remembered



J'ai hérité d'une somme de 3000 euros la semaine dernière et j'ai décidé de procéder à une commande d'actions boursière pour un montant de 1500 euros. Etant donné que vous êtes mon banquier depuis 10 ans maintenant je vous fais confiance quant au choix du placement. Je vous prie d'agréer Monsieur, l'expression de mes sentiments distingués.

Quantitative Results

Database	Resolution	Line segmentation: GroundTruth	Projection	Shredding	Energy	This work
IAM	150 dpi	8.4	15.5	9.3	10.2	6.8
	300 dpi	6.6	13.8	7.5	7.9	4.9
Rimes	150 dpi	4.8	6.3	5.9	8.2	2.8
	300 dpi	3.6	5.0	4.5	6.6	2.5

Table 3: Final results on Rimes and IAM databases

		Rimes WER%	Rimes CER%	IAM WER%	IAM CER%
150 dpi	no language model	13.6	3.2	29.5	10.1
	with language model			16.6	6.5
300 dpi	no language model	12.6	**2.9**	24.6	7.9
	with language model			16.4	5.5
	Bluche, 2015 [5]	**11.2**	3.5	**10.9**	**4.4**
	Doetsch et al., 2014 [14]	12.9	4.3	12.2	4.7
	Kozielski et al. 2013 [26]	13.7	4.6	13.3	5.1
	Pham et al., 2014 [33]	12.3	3.3	13.6	5.1
	Messina & Kermorvant, 2014 [30]	13.3	-	19.1	-

Outline of this talk

- ➔ End-to-End HWR -- from pixels to text
  - Multidimensional Recurrent Neural Networks
  - A few results and tips
  - Limitations
- ➔ Beyond textlines -- segmentation-free recognition of handwritten paragraphs
  - Attention-based models
  - A few results
  - Limitations
- ➔ Future challenges ...
- ➔ Open discussion

Future Challenges

→ **Full page recognition**

- Reading order not easy to define
- Localized lines : should put attention on zones, between point (char. attention) and all width (line attention)
- Mixed languages, write-types in real-world documents

→ **Faster models**

- e.g. back to features to replace the first LSTM

→ Other challenges : efficient & robust DLA, challenging languages, ...

Danke! Danke!

Danke! Danke! Danke!

Thanks for your attention

Danke! Danke!

Théodore Bluche

tb@a2ia.com

(do not hesitate to reach me if you have questions)

Danke! Danke! Danke!

A few refs...

Graves, A., Fernández, S., Gomez, F., & Schmidhuber, J. (2006). **Connectionist temporal classification**: labelling unsegmented sequence data with recurrent neural networks. In *Proceedings of the 23rd international conference on Machine learning* (pp. 369-376).
((CTC -- briefly explained in first part))

Graves, A., & Schmidhuber, J. (2009). Offline handwriting recognition with **multidimensional recurrent neural networks**. In *Advances in neural information processing systems* (pp. 545-552).
((MDLSTM-RNN -- the state-of-the-art, still, 7 years later))

Bluche, T. (2015). ***Deep Neural Networks*** *for Large Vocabulary* ***Handwritten Text Recognition*** (Doctoral dissertation, Université Paris Sud-Paris XI).
((my thesis -- many refs / results inside))

Bluche, T., Louradour, J., & Messina, R. (2016). Scan, Attend and Read: End-to-End **Handwritten Paragraph Recognition** with MDLSTM **Attention**. *arXiv preprint arXiv:1604.03286.*
((Attention-based neural nets))

... ...