Deep Learning for Vision (DLV)

Handwritten Text Recognition

Denis Coquenet

2024-2025

Goals of this course

Knowledge

- How CTC works?
- Advantages/drawbacks of CTC and Attention paradigms
- Differences between line-level and end-to-end approaches for text recognition

Skills and know-how

- Compute Levenshtein distance, CER and WER between two sequences of characters/words
- Apply the decoding process of the CTC: from probability lattice to final string prediction
- Propose an approach to handle an image-to-sequence task

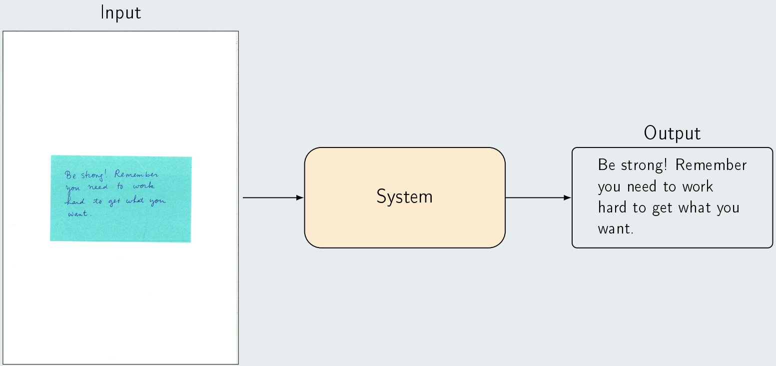
Table of contents

1 HTR task
- What? Why?
- Evaluation

2 Line-level approach

3 End-to-end approach

An image-to-sequence problem



Input: an image $\boldsymbol{X} \in \mathbb{R}^{H \times W \times C}$

Output: a sequence of characters $\boldsymbol{y}$ (with $\boldsymbol{y}_i \in \mathcal{A}$, an alphabet)

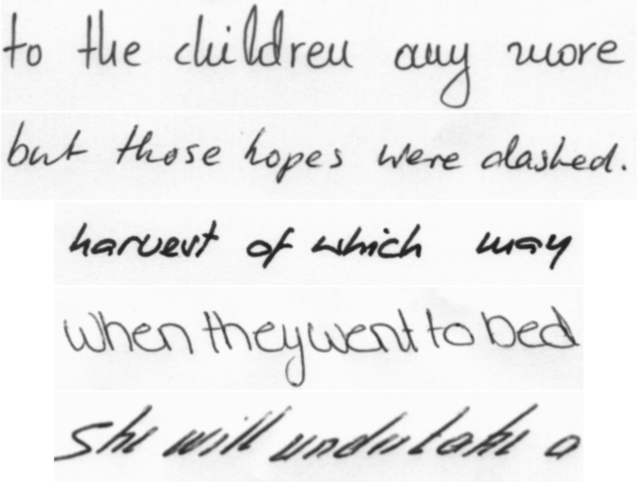
HTR task

Why?

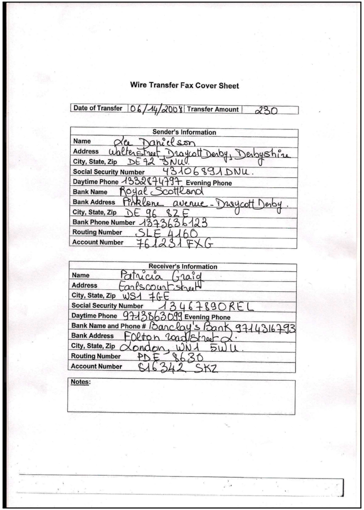
- Transcription of historical documents
- Industrial document processing: bank checks, forms, invoices
- Real-time document translation
- Exam correction

Challenges

- Writing style variety
- Heterogeneous layouts / background
- No a priori reading order, number of characters to recognize



- Spacing, character shapes, slant, color, stroke width



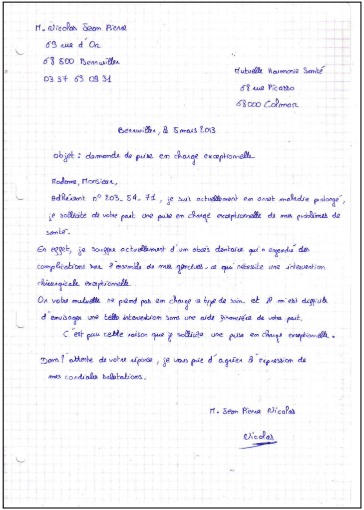
Wire Transfer Fax Cover Sheet

Date of Transfer	06/14/2008	Transfer Amount	230

Sender's Information	
Name	Lee Danielson
Address	Walter Street Draycott Derby, Derbyshire
City, State, Zip	DE 72 3NU.
Social Security Number	43106891DNU.
Daytime Phone	1332874797 Evening Phone
Bank Name	Royal Scottland
Bank Address	Pinklane avenue - Draycott Derby.
City, State, Zip	DE 96 82E
Bank Phone Number	187363E123
Routing Number	SLE 4160
Account Number	761231 FXG

Receiver's Information	
Name	Patricia Graig
Address	Earlscount street
City, State, Zip	WS1 7GE
Social Security Number	1346789OREL
Daytime Phone	9713863099 Evening Phone
Bank Name and Phone #	Barclay's Bank 9714316793
Bank Address	Folton road street 2.
City, State, Zip	London, WN1 5WU.
Routing Number	PDE 8630
Account Number	616342 SKZ

Notes:



M. Nicolas Jean Pierre
69 rue d'Or
68 500 Bennwiller
03 37 63 09 31

Mutuelle Harmonie Santé
68 rue Picasso
68000 Colmar

Bennwiller, le 8 mars 2013

objet : demande de prise en charge exceptionnelle

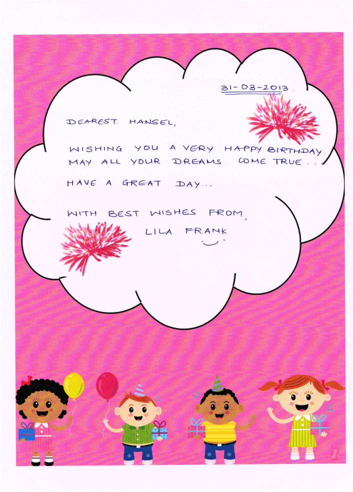
Madame, Monsieur,
Adhérent n° 203_54_71, je suis actuellement en arrêt maladie prolongé, je sollicite de votre part une prise en charge exceptionnelle de mes problèmes de santé.
En effet, je souffre actuellement d'un abcès dentaire qui a engendré des complications sur l'ensemble de mes gencives, ce qui nécessite une intervention chirurgicale exceptionnelle.
Or votre mutuelle ne prend pas en charge ce type de soin et il m'est difficile d'envisager une telle intervention sans une aide financière de votre part.
C'est pour cette raison que je sollicite une prise en charge exceptionnelle.

Dans l'attente de votre réponse, je vous prie d'agréer l'expression de mes cordiales salutations.

M. Jean Pierre Nicolas

Nicolas

➤ The reading order is conditioned by the layout



- Non-textual items, slanted lines

Metrics

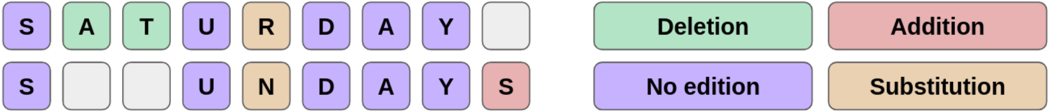
Character Error Rate (CER) and Word Error Rate (WER)
= edit distance between sequences of characters (or words)

$$\text{CER} = \frac{I + D + S}{N}$$

I: number of insertions
D: number of deletions
S: number of substitutions
N: number of characters in the ground truth sequence

Example

Ground truth: "SUNDAYS"
Prediction: "SATURDAY"



Metrics

$$\text{CER} = \frac{I + D + S}{N} = \frac{1 + 2 + 1}{7} \simeq 57.14\%$$

Levenshtein distance (= edit distance)

The Levenshtein distance d_{lev} between two sequences of tokens $\boldsymbol{s}_A$ and $\boldsymbol{s}_B$ is defined as:

$$d_{\text{lev}}(\boldsymbol{s}_A, \boldsymbol{s}_B) = \begin{cases} \max(|\boldsymbol{s}_A|, |\boldsymbol{s}_B|) & \text{if } \min(|\boldsymbol{s}_A|, |\boldsymbol{s}_B|) = 0 \\ d_{\text{lev}}(\boldsymbol{s}_{A_{[1:]}}, \boldsymbol{s}_{B_{[1:]}}) & \text{if } \boldsymbol{s}_{A_0} = \boldsymbol{s}_{B_0} \\ 1 + \min \begin{cases} d_{\text{lev}}(\boldsymbol{s}_{A_{[1:]}}, \boldsymbol{s}_B) & \textit{del.} \\ d_{\text{lev}}(\boldsymbol{s}_A, \boldsymbol{s}_{B_{[1:]}}) & \textit{ins.} \\ d_{\text{lev}}(\boldsymbol{s}_{A_{[1:]}}, \boldsymbol{s}_{B_{[1:]}}) & \textit{sub.} \end{cases} & \text{otherwise} \end{cases}$$

- Implementation with dynamic programming

		S	A	T	U	R	D	A	Y
	0	1	2	3	4	5	6	7	8
S	1	?							
U	2								
N	3								
D	4								
A	5								
Y	6								
S	7								

Matrix D

$$D_{i,j} = \min \begin{cases} 1 + D_{i-1,j} & \textit{ins.} \\ 1 + D_{i,j-1} & \textit{del.} \\ D_{i-1,j-1} + \begin{cases} 0 & \text{if } \boldsymbol{s}_{A_i} = \boldsymbol{s}_{B_j} \\ 1 & \text{otherwise} \end{cases} \end{cases}$$

Here:

$$D_{1,1} = \min \begin{cases} 1 + D_{0,1} \\ 1 + D_{1,0} \\ D_{0,0} + 0 \end{cases} = \min \begin{cases} 1 + 1 \\ 1 + 1 \\ 0 \end{cases} = 0$$

		S	A	T	U	R	D	A	Y
	0	1	2	3	4	5	6	7	8
S	1	0	1	2	3	4	5	6	7
U	2	1	1	2	2	3	4	5	6
N	3	2	2	2	3	3	4	5	6
D	4	3	3	3	3	4	3	4	5
A	5	4	3	4	4	4	4	3	4
Y	6	5	4	4	5	5	5	4	3
S	7	6	5	5	5	6	6	5	4

Determining path: from bottom-right to top-left

➤ Select a minimum between adjacent values

Reading path: from top-left to bottom-right

Diagonal cell: keep if same value, substitution otherwise

Right cell: removal

Bottom cell: addition

To go from SATURDAY to SUNDAYS:

Keep S, Substitue "A" by "U", Remove "T", Remove "U", Substitue "R" by "N", Keep "D", Keep "A", Keep "Y", Add "S"

Compute the WER for the following sequences using the dynamic programming algorithm:
Ground truth: "The dog is brown"
Prediction: "The brown dog"

Ground truth: s_A=['The','dog', 'is', 'brown']
Prediction: s_B=['The', 'brown', 'dog']

		The	dog	is	brown
	0	1	2	3	4
The	1	0	1	2	3
brown	2	1	1	2	2
dog	3	2	1	2	3

(1) From GT to prediction

		The	brown	dog
	0	1	2	3
The	1	0	1	2
dog	2	1	1	1
is	3	2	2	2
brown	4	3	2	3

(2) From prediction to GT

$$\text{WER} = \frac{d_{\text{lev}}(\boldsymbol{s}_A, \boldsymbol{s}_B)}{|\boldsymbol{s}_A|} = \frac{3}{4} = 75\%$$

- ➤ Interpretation (1): Keep 'The', Add 'brown', Keep 'dog', Remove 'is', Remove 'brown'
- ➤ Interpretation (2): Keep 'The', Remove 'brown', Keep 'dog', Add 'is', Add 'brown'

➤ Two main approaches

A sequential paradigm at line level

The recognition process is split into three steps that are performed sequentially: segmentation, ordering and recognition

➤ Mature approach

An end-to-end paradigm

The recognition of a whole document is performed in a single step

➤ Proposed in 2023

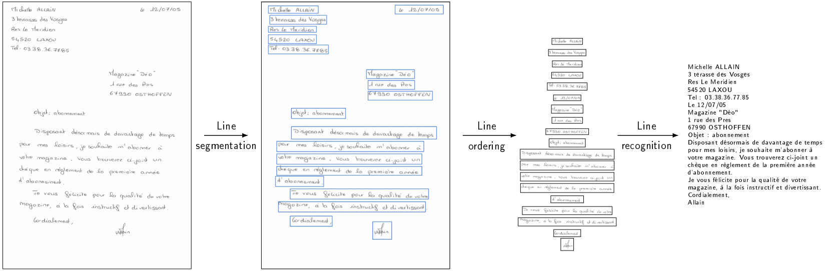
Table of contents

1 HTR task

2 Line-level approach
- Line segmentation
- Line ordering
- Line recognition
- Connectionist Temporal Classification
- Attention-based recognition

3 End-to-end approach

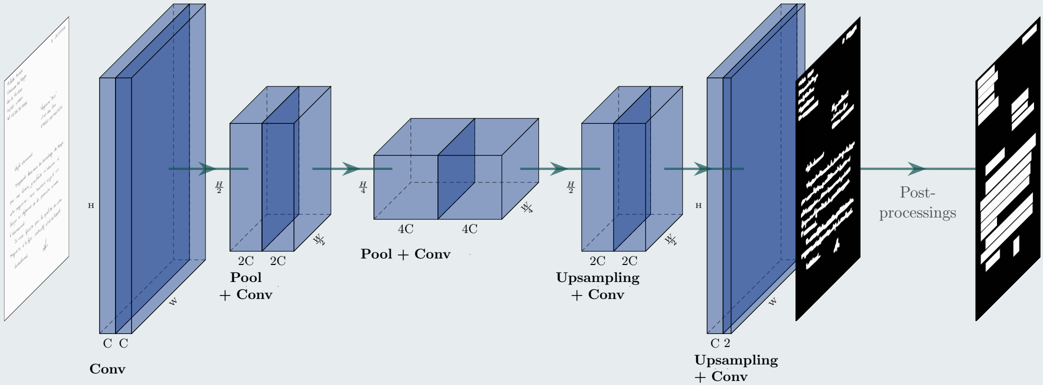
- Segmentation
- Ordering
- Recognition



Exercise

How would you solve the segmentation task? Which kind of model? Which loss?

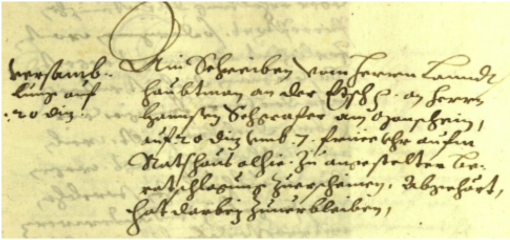
Text line segmentation architecture (FCN) [1, 2]



Could also be solved with an object detection approach [3]

A rule-based approach

Intuition: order bounding boxes from top to bottom and from left to right for most Latin languages.



Expected reading order by column.

Expected reading order by row.

➤ Must be adapted given the layout/dataset → human effort

Goal

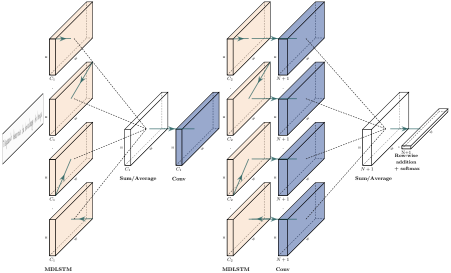
Input: 2D image $\boldsymbol{X} \in \mathbb{R}^{H \times W \times C}$
Output: 1D sequence of characters $\boldsymbol{y}$ of length L_y

➤ How to go from 2D input to 1D output?
➤ How to predict an ordered output whose length does not depend on that of the input?

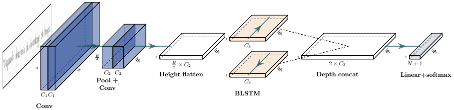
Before $\sim$ 2005

- Character segmentation
- Character classification

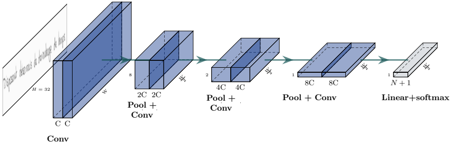
➤ Requires segmentation network (costly annotations) + ordering



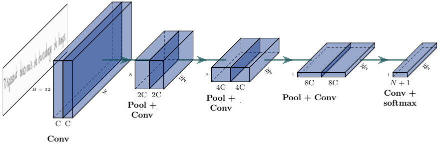
CNN + Multi-Dimensional LSTM [4, 5]



CNN + Bidirectional LSTM [6, 7]



CNN [8, 9]

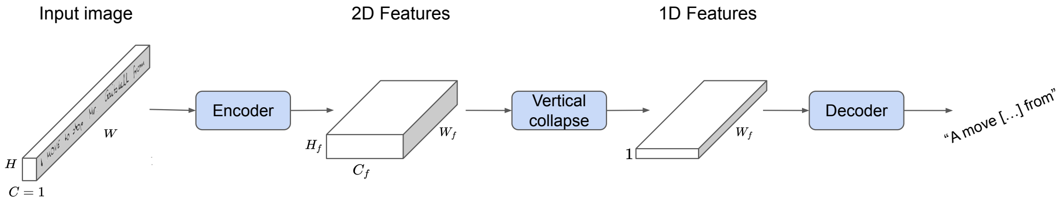


FCN [10, 11]

➤ Many architectures...

... but a common approach

- Extraction of 2D feature maps
- Collapse of the verticale axis (pooling/convolution with vertical kernel)
- Decoding with CTC



Goal

Handle the alignment between two 1D sequences of different length:

- 1D sequence of probability vectors (prediction $\boldsymbol{p} \in \mathbb{R}^{W_f \times |\mathcal{A}|}$)
- 1D sequence of characters (ground truth $\boldsymbol{y} \in \mathcal{A}^{L_y}$)

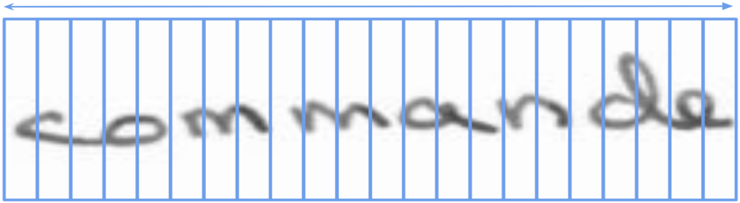
Input side:

➤ A character can be written over a variable number of pixels

Output side:

➤ No a priori knowledge about L_y

Frame-by-frame classification + post-processing

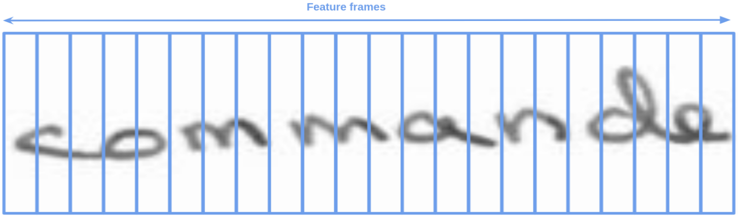


Alphabet (charset)																						
A	0.01	0	0.1	0.01	0.01	0	0.01	0	0	0	0.01	0.01	**0.95**	**0.99**	**0.25**	0	0	0.01	0.01	0	0	0
B	0	0	0	0.01	0.01	0	0.01	0.01	0	0.01	0	0.01	0.01	0	0	0.01	0.01	0	0.01	0	0.01	0
C	**0.9**	**0.95**	**0.2**	0	0.01	0	0.01	0.01	0.01	0.01	0	0.01	0	0	0.01	0.01	0	0.01	0	0.01	0	0
...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...
Z	0.01	0.01	0.01	0	0	0.01	0	0.01	0	0	0.01	0	0	0	0.01	0.01	0	0.01	0	0.01	0	0.01
argmax	C	C	C	O	O	M	M	M	M	M	M	M	A	A	A	N	N	D	D	D	E	E
rm dup.	C			O		M							A			N		D			E	

Predictions (probability lattice)

➤ Final prediction: "comande" $\neq$ "commande"

- Introduction of a new token: CTC blank token $\varnothing$ $(\mathcal{A}^* = \mathcal{A} \cup \{\varnothing\})$



A	0.01	0	0.1	0.01	0.01	0	0.01	0	0	0	0.01	0.01	**0.95**	**0.99**	0.1	0	0	0.01	0.01	0	0	0
B	0	0	0	0.01	0.01	0	0.01	0.01	0	0.01	0	0.01	0.01	0	0	0.01	0.01	0	0.01	0	0.01	0
C	**0.9**	**0.95**	0.01	0	0.01	0	0.01	0.01	0.01	0.01	0	0.01	0	0	0.01	0.01	0	0.01	0	0.01	0	0
...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...
Z	0.01	0.01	0.01	0	0	0.01	0	0.01	0	0	0.01	0	0	0	0.01	0.01	0	0.01	0	0.01	0	0.01
Ø	0.01	0.01	**0.9**	0.01	0.01	**0.9**	0	0	**0.9**	0	0.01	**0.9**	0.01	0.01	**0.6**	0	0.01	0	0.01	0	0	0.01

Alphabet (charset)

argmax

Predictions (probability lattice)

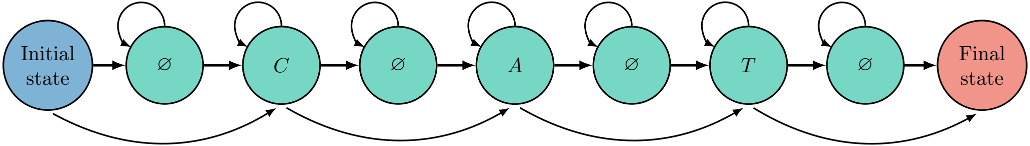
	C	C	Ø	O	O	Ø	M	M	Ø	M	M	Ø	A	A	Ø	N	N	D	D	D	E	E
rm dup.	C		Ø	O		Ø	M		Ø	M		Ø	A		Ø	N		D			E	
rm ø	C			O			M			M			A			N		D			E	

- How to train a model to generate a correct probability lattice?

What is a correct prediction sequence?

Let $\beta : \mathcal{A}^{*^L} \mapsto \mathcal{A}^{\leq L}$ be the mapping function which first remove all the successive duplicated predictions, and then remove all the blank tokens $\varnothing$.

For example, for the ground truth "CAT":
$\beta(\mathrm{CAAAT}) = \beta(\mathrm{CAT}) = \beta(\mathrm{C}\varnothing\mathrm{AAT}) = \mathrm{CAT}$, but $\beta(\mathrm{CCA}\varnothing\mathrm{AT}) = \mathrm{CAAT}$



Automaton describing a correct prediction

➤ Training must maximize the prediction of any prediction sequence (also known as path $\boldsymbol{\pi}$) leading to $\boldsymbol{y}$

Equivalent to minimizing $-\ln$

$$\mathcal{L}_{\text{CTC}}(\boldsymbol{p}, \boldsymbol{y}) = -\ln p(\boldsymbol{y}|\boldsymbol{p})$$

with $\boldsymbol{p} = f_\theta(\boldsymbol{X})$

Probability of $\boldsymbol{y}$

$$p(\boldsymbol{y}|\boldsymbol{p}) = \sum_{\boldsymbol{\pi} \in \mathcal{B}^{-1}(\boldsymbol{y})} p(\boldsymbol{\pi}|\boldsymbol{p})$$

$=$ all paths that lead to $\boldsymbol{y}$ through β

Probability of a specific path $\boldsymbol{\pi}$

$$p(\boldsymbol{\pi}|\boldsymbol{p}) = \prod_{t=1}^{W_f} \boldsymbol{p}^t_{\boldsymbol{\pi}^t}, \forall \boldsymbol{\pi} \in \mathcal{A}^{*^{W_f}}$$

where $\boldsymbol{p}^t_{\boldsymbol{\pi}^t}$ is the probability of observing label $\boldsymbol{\pi}^t$ at position t in the input sequence $\boldsymbol{p}$

	p^1	p^2	p^3	p^4	p^5	p^6	p^7	p^8	p^9	p^{10}
C	0.1	**0.9**	**0.8**	0	0.1	0	0.1	0.2	0	0.1
A	0.1	0	0.1	0.2	**0.7**	0.1	0.1	0.2	0.1	0.1
T	0.1	0.05	0.75	0.1	0.1	0.2	0.2	**0.5**	**0.9**	**0.8**
∅	**0.7**	0.05	0.25	**0.7**	0.1	**0.7**	**0.6**	0.1	0	0

$\boldsymbol{\pi} =$ CCAAAAATTT

$p(\boldsymbol{\pi}|\boldsymbol{p}) = 0.1 \times 0.9 \times 0.1 \times 0.2 \times 0.7 \times 0.1 \times 0.1 \times 0.5 \times 0.9 \times 0.8$

	p^1	p^2	p^3	p^4	p^5	p^6	p^7	p^8	p^9	p^{10}
C	0.1	**0.9**	**0.8**	0	0.1	0	0.1	0.2	0	0.1
A	0.1	0	0.1	0.2	**0.7**	0.1	0.1	0.2	0.1	0.1
T	0.1	0.05	0.75	0.1	0.1	0.2	0.2	**0.5**	**0.9**	**0.8**
∅	**0.7**	0.05	0.25	**0.7**	0.1	**0.7**	**0.6**	0.1	0	0

$\boldsymbol{\pi} =$ ∅∅∅CAAAT∅∅

$p(\boldsymbol{\pi}|\boldsymbol{p}) = 0.7 \times 0.05 \times 0.25 \times 0 \times 0.7 \times 0.1 \times 0.1 \times 0.5 \times 0 \times 0$

➤ Computed with dynamic programming

Best path decoding (greedy search)

The best path is computed by keeping the character with maximum probability at each step

$$\boldsymbol{\pi}^{*t} = \arg\max \boldsymbol{p}^t$$

	p^1	p^2	p^3	p^4	p^5	p^6	p^7	p^8	p^9	p^{10}
C	0.1	**0.9**	**0.8**	0	0.1	0	0.1	0.2	0	0.1
A	0.1	0	0.1	0.2	**0.7**	0.1	0.1	0.2	0.1	0.1
T	0.1	0.05	0.75	0.1	0.1	0.2	0.2	**0.5**	**0.9**	**0.8**
∅	**0.7**	0.05	0.25	**0.7**	0.1	**0.7**	**0.6**	0.1	0	0

$$\boldsymbol{\pi}^* = \varnothing\text{CC}\varnothing\text{A}\varnothing\text{TTT}$$
$$p(\boldsymbol{\pi}^*|\boldsymbol{p}) = 0.7 \times 0.9 \times 0.8 \times 0.7 \times 0.7 \times 0.7 \times 0.6 \times 0.5 \times 0.9 \times 0.8$$

➤ Very fast decoding approach (all steps are processed independently, in parallel)

Given an alphabet $\mathcal{A}^* = \{A, C, T, \varnothing\}$ and the following probability lattice:

- Deduce the best path chosen with best path decoding approach. Compute its probability.
- What are the paths that lead to the prediction "C" after CTC decoding? Compute the associated probability p("C").
- Conclude.

	p^1	p^2
C	0.3	0.35
A	0.25	0.4
T	0.2	0.1
$\varnothing$	0.25	0.15

Correction

Prediction	Paths	Probability
Null sequence	p(∅∅)	3.75 %
C	p(CC) + p(∅C) + p(C∅)	**23.75%**
A	p(AA) + p(∅A) + p(A∅)	23.75%
T	p(TT) + p(∅T) + p(T∅)	7.5%
AC	p(AC)	8.75%
AT	p(AT)	2.5%
CA	p(CA)	**12%**
CT	p(CT)	3%
TA	p(TA)	8%
TC	p(TC)	7%

Best path decoding: CA, with p("CA") = 12% < p("C") = 23.75%

➤ Best path decoding is not optimal

Best-path decoding

Local estimation: not optimal

Computation of all possible paths

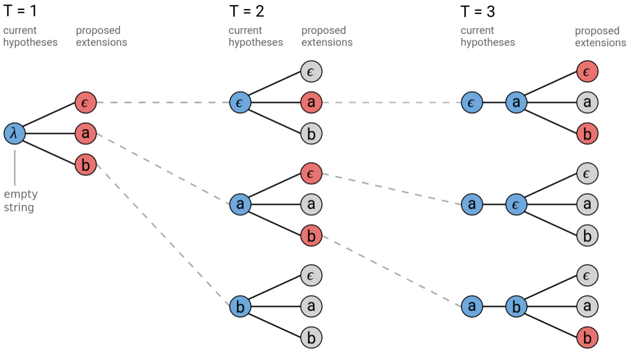
Number of paths: $|A^*|^{W_f}$
with $|A^*| \approx 10^2$ and $W_f \approx 10^2$
➤ Intractable

Trade-off: beam-search decoding

Iterative process which extends only the best partial candidates

CTC: prediction

- Beam search decoding

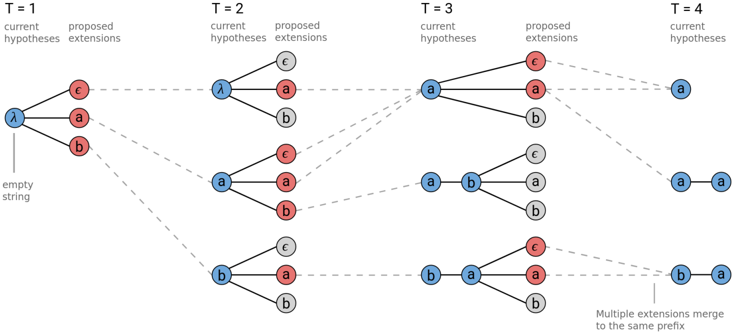


A standard beam search algorithm with an alphabet of $\{\epsilon, a, b\}$ and a beam size of three.

ϵa and $a\epsilon$ correspond to the same prediction after CTC decoding
- We should merge their probabilities

CTC: prediction

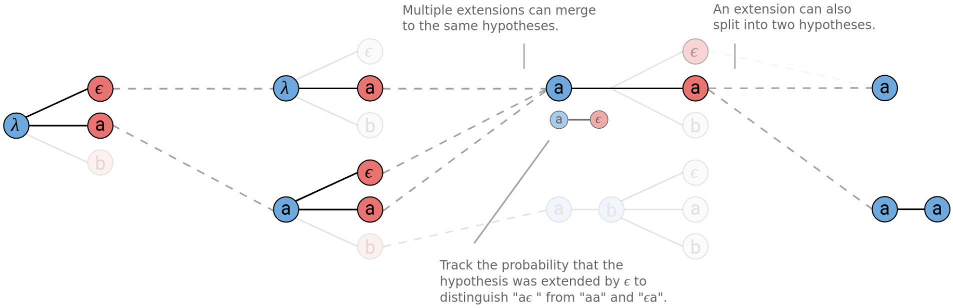
- Beam search decoding, merging equivalent prefixes



The CTC beam search algorithm with an output alphabet $\{\epsilon, a, b\}$ and a beam size of three.

$a\epsilon a$ and aaa do not correspond to the same prediction after CTC decoding
- We should split their probabilities

- Beam search decoding, merging equivalent prefixes, with two probabilities (ending with CTC blank or not)



An iterative decoding process

- Predict the characters one after the other
  - Begin with a specific start-of-transcription token: $\hat{y}^0 = $ <sot>
  - Stop with a specific end-of-transcription token: $\hat{y}^{L_y+1} = $ <eot>

At iteration t:

Input:

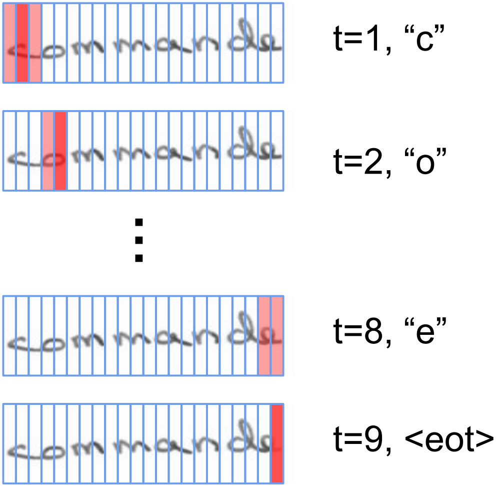
- The image features $\boldsymbol{f} \in \mathbb{R}^{1 \times W_f \times C}$
- The predicted tokens $\hat{\boldsymbol{y}}^{0:t-1} = [\hat{\boldsymbol{y}}^0, \hat{\boldsymbol{y}}^1, ..., \hat{\boldsymbol{y}}^{t-1}]$

Compute:

- The attention weights $\boldsymbol{\alpha}^t \in [0,1]^{W_f}$ $(\sum_{i=1}^{W_f} \boldsymbol{\alpha}_i^t = 1)$
- The character representation $\boldsymbol{c}^t = \sum_{i=1}^{W_f} \boldsymbol{\alpha}_i^t \cdot \boldsymbol{f}_i$
- The character probabilities $\boldsymbol{p}^t = \text{softmax}(\boldsymbol{c}_t)$

Output:

- The predicted token $\hat{\boldsymbol{y}}^t = \arg\max(\boldsymbol{p}^t)$



- Transformer decoder
- No direct left-to-right constraint
  ➤ Reading order learned through text supervision
- Stops only when predicting the <eot> token
  ➤ In practice, set a maximum number of iterations to avoid infinite loop

Recognition stage with attention (from $\sim$ 2020) [13, 14]

Training

$$\mathcal{L}_{\text{attention}} = \sum_{t=1}^{L_y+1} \mathcal{L}_{\text{CE}}(\boldsymbol{y}^t, \boldsymbol{p}^t)$$

➤ Requires to predict all the characters: can be long!

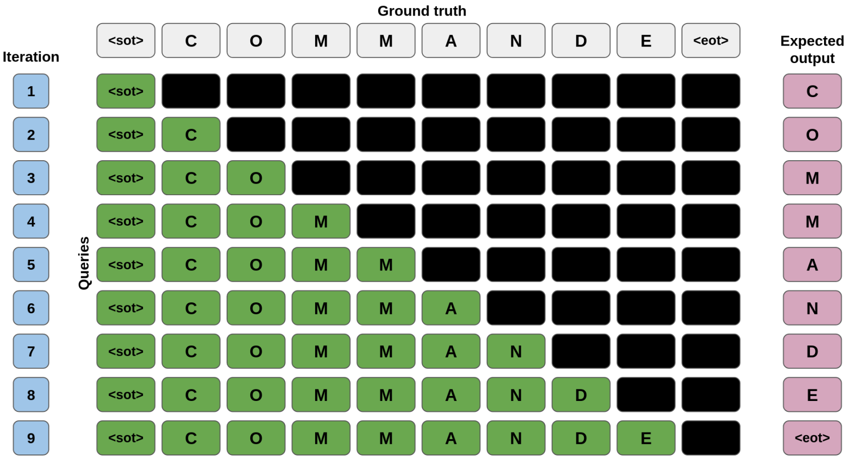
Teacher forcing

Speeding up training by parallelizing the decoding process using the ground truth $\boldsymbol{y}^{[0:t-1]}$ instead of the prediction $\hat{\boldsymbol{y}}^{[0:t-1]}$

➤ Only possible at training time!

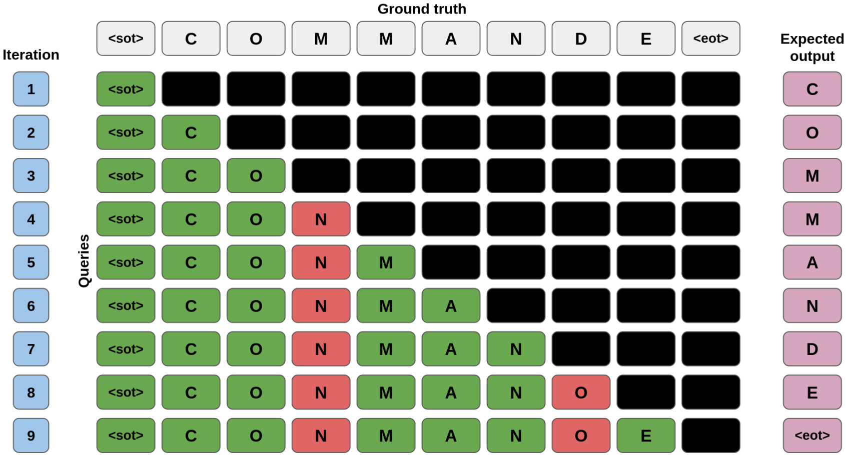
Teacher forcing

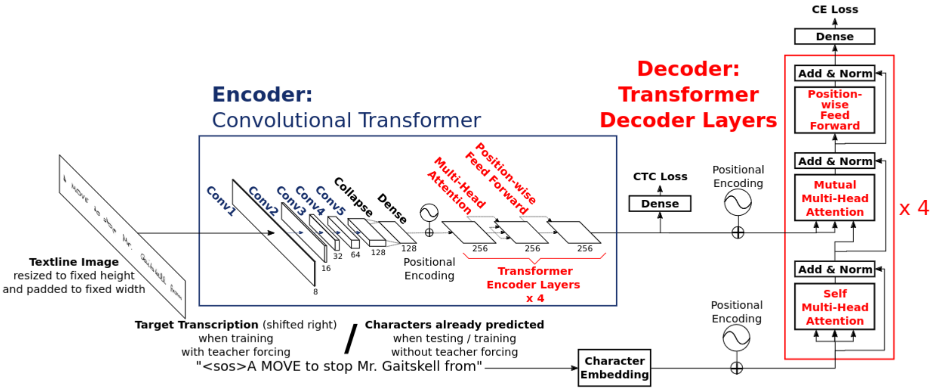
➤ Use a masking strategy



➤ Generalization issue: only trained with "perfect" queries

- Inject errors in queries





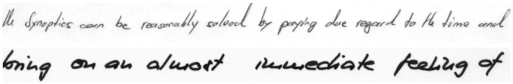
$$\mathcal{L} = \lambda \mathcal{L}_{\text{CTC}} + (1 - \lambda)\mathcal{L}_{\text{attention}}$$

- $\lambda = 0.5$

IAM dataset

Training	Validation	Test
6,482	976	2,915

(+10,000 synthetic samples per epoch)



Real samples

Synthetic samples

	IAM		IAM + synthetic	
	CER (%)	WER (%)	CER (%)	WER (%)
CTC	6.14	23.26	5.66	21.62
Attention	10.26	26.36	6.76	19.62
CTC + attention	**5.70**	**18.86**	**4.76**	**16.31**

Conclusion

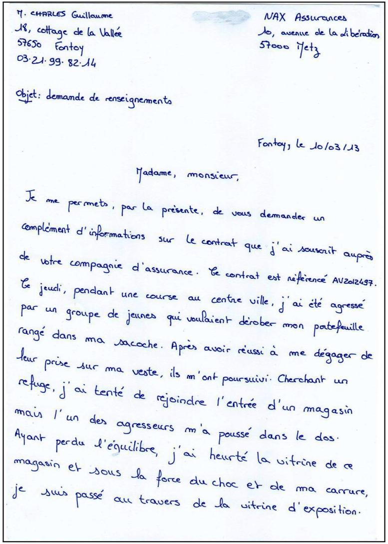
The line-level paradigm: a mature approach... with some limitations

- Three steps treated independently
- A complex pipeline, hard to maintain
- Cumulative errors between steps
- Additional segmentation annotations
- Rule-based reading order

➤ Towards end-to-end document recognition

Table of contents

1. HTR task
2. Line-level approach
3. End-to-end approach
   - Challenges
   - Specific metrics
   - DAN



M. CHARLES Guillaume
18, cottage de la Vallée
57650 Fontoy
03.21.99.82.14

NAX Assurances
10, avenue de la Libération
57000 Metz

Objet: demande de renseignements

Fontoy, le 10/03/13

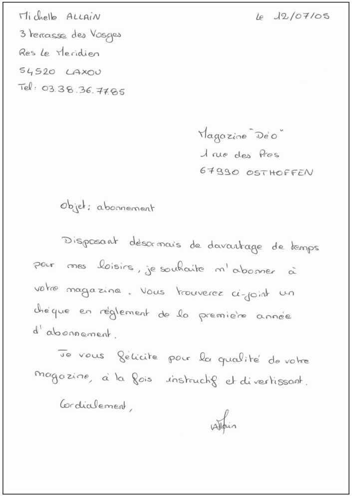
Madame, monsieur,

Je me permets, par la présente, de vous demander un complément d'informations sur le contrat que j'ai souscrit auprès de votre compagnie d'assurance. Ce contrat est référencé AV2012497. Ce jeudi, pendant une course au centre ville, j'ai été agressé par un groupe de jeunes qui voulaient dérober mon potefeuille rangé dans ma sacoche. Après avoir réussi à me dégager de leur prise sur ma veste, ils m'ont poursuivi. Cherchant un refuge, j'ai tenté de rejoindre l'entrée d'un magasin mais l'un des agresseurs m'a poussé dans le dos. Ayant perdu l'équilibre, j'ai heurté la vitrine de ce magasin et sous la force du choc et de ma carrure, je suis passé au travers de la vitrine d'exposition.

Challenges from paragraph to document

- Layout-dependent reading order
- Larger input images and output sequences
  - GPU constraints
  - More complex attention

Goal: joint recognition of both text and layout from whole documents


Michelle ALLAIN le 12/07/05
3 terasse des Vosges
Res Le Meridien
54520 LAXOU
Tel: 03.38.36.77.85

Magazine "Déo"
1 rue des Pres
67990 OSTHOFFEN

objet: abonnement

Disposant désormais de davantage de temps pour mes loisirs, je souhaite m'abonner à votre magazine. Vous trouverez ci-joint un chèque en règlement de la première année d'abonnement.

Je vous félicite pour la qualité de votre magazine, à la fois instructif et divertissant.

Cordialement,

Allain

Handwritten Document

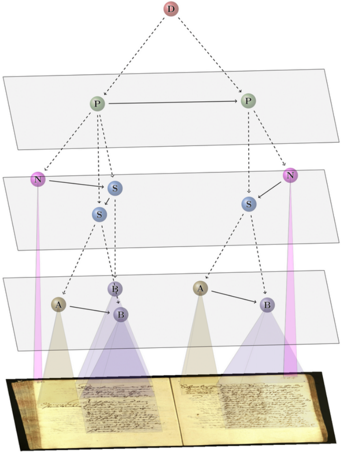
→

Recognition

Michelle ALLAIN
3 terasse des Vosges
Res Le Meridien
54520 LAXOU
Tel : 03.38.36.77.85
Le 12/07/05
Magazine "Déo"
1 rue des Pres
67990 OSTHOFFEN
Objet : abonnement
Disposant désormais de davantage de temps
pour mes loisirs, je souhaite m'abonner à
votre magazine. Vous trouverez ci-joint un
chèque en règlement de la première année
d'abonnement.
Je vous félicite pour la qualité de votre
magazine, à la fois instructif et divertissant.
Cordialement,
Allain

- Sender Coordinates
- Recipient Coordinates
- Place & Date
- Object
- Body
- Signature

How to encode both text and layout ?




```xml
<document>
  <page>
    <page_number>
      204
    </page_number>
    <section>
      <body>
        Schgrafer, [...] gehalt.
      </body>
    </section>
    <section>
      <annotation>
        Genneral [...] Raitūng
      </annotation>
      <body>
        Aūf den: [...] werden,
      </body>
    </section>
  </page>
  <page>
    <page_number>
      204
    </page_number>
    <section>
      <annotation>
        Schmalz. [...] bet:
      </annotation>
      <body>
        Verer [...] dar¬
      </body>
    </section>
  </page>
</document>
```

- XML paradigm

How to evaluate the performance ?

Evaluate the text recognition

- CER / WER

➤ Normalized edit distance between sequences of characters / words

Prediction: "<A><B>HTR</B>2<B>HDR</B></A>"
Metric computed on: "HTR2HDR"

How to evaluate the performance ?

Evaluate the text recognition

- CER / WER

Evaluate the layout recognition

- LOER (Layout Ordering Error Rate)

➤ Normalized edit distance between graphs

Prediction: "<A><B>HTR</B>2<B>HDR</B></A>"
Metric computed on: "<A><B></B><B></B></A>"

Evaluate the text recognition

- CER / WER

Evaluate the layout recognition

- LOER (Layout Ordering Error Rate)

⚠ **Not sufficient:**

Ground truth: "<A><B>HTR</B>2<B>HDR</B></A>"
Prediction: "<A><B></B><B></B></A>HTR2HDR"

LOER = 0%   CER = 0%

How to evaluate the performance ?

Evaluate the text recognition

- CER / WER

Evaluate the layout recognition

- LOER (Layout Ordering Error Rate)

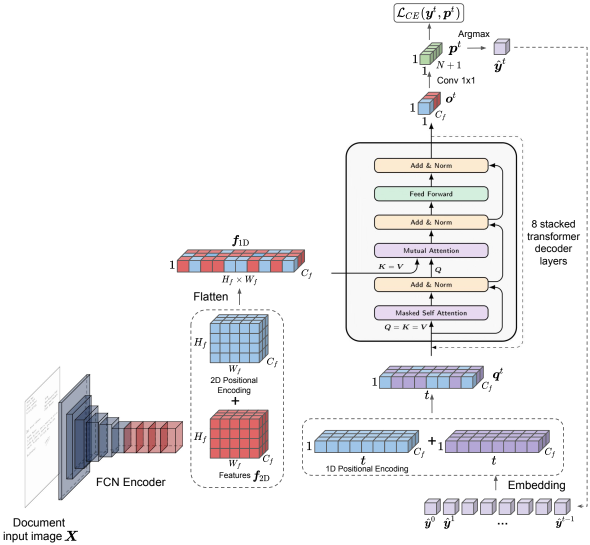
Evaluate text and layout recognition altogether

- $\mathrm{mAP}_{\mathrm{CER}}$

➤ Area under the precision / recall curve

Prediction: "<A><B>HTR</B>2<B>HDR</B></A>"
Metric computed on: "HTR2HDR", "HTR", "HDR"

Document Attention Network (2023) [15]



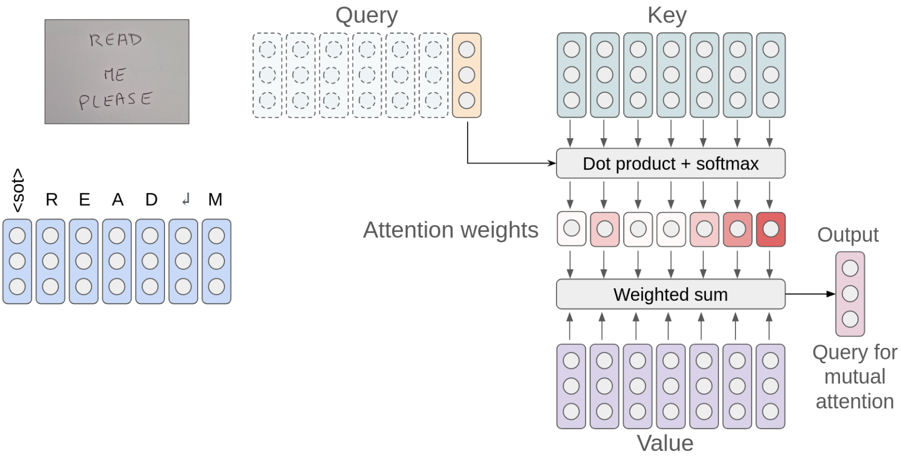
$$\mathcal{L} = \sum_{t=1}^{L_y+1} \mathcal{L}_{\mathsf{CE}}(\boldsymbol{y}^t, \boldsymbol{p}^t)$$

$$\boldsymbol{y}^t \in \mathcal{A}$$

$$\mathcal{A} = \mathcal{D}_{\mathsf{char}} \cup \mathcal{D}_{\mathsf{xml}} \cup \mathcal{D}_{\mathsf{eot}}$$

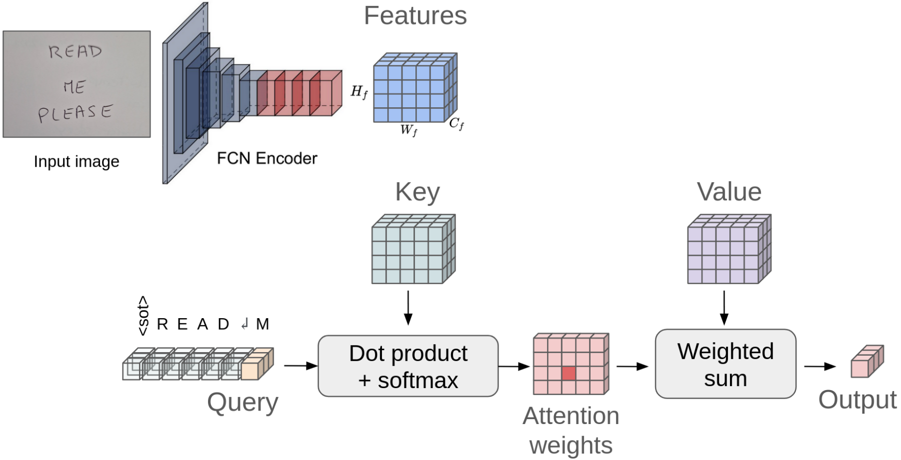
- Teacher forcing

DAN: self attention



- ➤ Query, Key and Value from same source (decoder input)

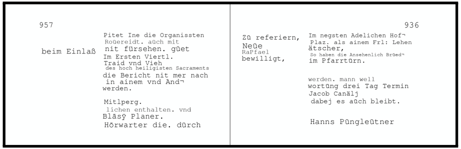
DAN: mutual attention



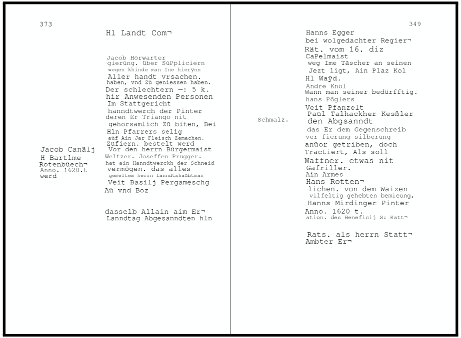
- Query from decoder, Key and Value from encoder (image features)

- Pre-training encoder on synthetic text line images (with CTC loss)
- Curriculum learning with synthetic documents:

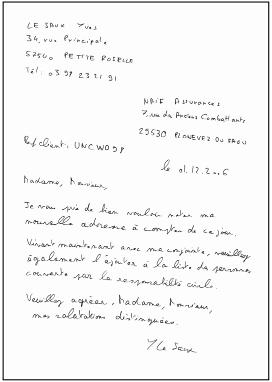
(a) $l = 3$.



(b) $l = 15$.



(c) $l = l_{\max} = 30$ (end of curriculum stage, no crop).


LE SAUX Yves
34, rue Principale
57540 PETITE ROSELLE
Tél : 03 97 23 21 91

NAIF Assurances
7, rue des Anciens Combattants
29530 PLONEVEZ DU FAOU

Ref client : UNCW09P

le 01.12.2006

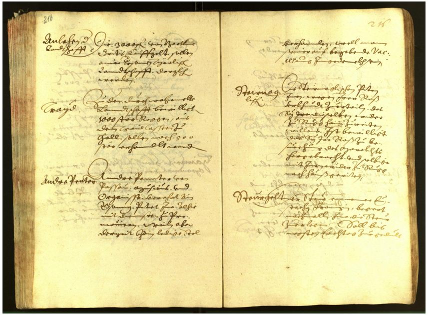
Madame, Monsieur,

Je vous prie de bien vouloir noter ma nouvelle adresse à compter de ce jour.

Vivant maintenant avec ma conjointe, veuillez également l'ajouter à la liste des personnes couverte par la responsabilité civile.

Veuillez agréer, Madame, Monsieur, mes salutations distinguées.

Y Le Saux



Dataset	Level	Training	Validation	Test	# char tokens	# layout tokens
RIMES 2009	Page	1,050	100	100	108	14
READ 2016	Page	350	50	50	89	10
	Double page	169	24	24		

DAN results on the RIMES dataset

➤ Metrics do not take into account the segmentation step

Dataset	Approach	CER (%) ↓	WER (%) ↓	LOER (%) ↓	$\mathrm{mAP}_{\mathrm{CER}}$ (%) ↑
RIMES 2011	**Line level**				
	[16] FCN	3.04	8.32	✗	✗
	[7] CNN+BLSTM[a]	**2.3**	9.6	✗	✗
	[15] DAN (FCN+transformer)[c]	2.63	**6.78**	✗	✗
	Paragraph level				
	[17] SPAN (FCN)	4.17	15.61	✗	✗
	[18] CNN+MDLSTM[b]	2.9	12.6	✗	✗
	[16] VAN (FCN+LSTM)[b]	1.91	6.72	✗	✗
	[15] DAN (FCN+transformer)[c]	**1.82**	**5.03**	✗	✗
RIMES 2009	**Paragraph level**				
	[15] DAN (FCN+transformer)[c]	5.46	13.04	✗	✗
	Page level				
	[15] DAN (FCN+transformer)[c]	4.54	11.85	3.82	93.74

[a] This work uses a slightly different split (10,203 for training, 1,130 for validation and 778 for test).

[b] with line-level attention.

[c] with character-level attention.

DAN results on the READ 2016 dataset

➤ Metrics do not take into account the segmentation step

Approach	CER (%) ↓	WER (%) ↓	LOER (%) ↓	$\mathrm{mAP}_{\mathrm{CER}}$ (%) ↑
Line level				
[19] CNN+BLSTM[a]	4.66	✗	✗	✗
[20] CNN+RNN	5.1	21.1	✗	✗
[16] VAN (FCN+LSTM)[b]	**4.10**	**16.29**	✗	✗
[15] DAN (FCN+transformer)[a]	**4.10**	17.64	✗	✗
Paragraph level				
[17] SPAN (FCN)	6.20	25.69	✗	✗
[16] VAN (FCN+LSTM)[b]	3.59	13.94	✗	✗
[15] DAN (FCN+transformer)[a]	**3.22**	**13.63**	✗	✗
Single-page level				
[15] DAN (FCN+transformer)[a]	3.53	13.33	5.94	92.57
Double-page level				
[15] DAN (FCN+transformer)[a]	3.69	14.20	4.60	93.92

[a] with character-level attention.

[b] with line-level attention.

DAN demonstration

https://youtu.be/HrrUsQfW66E

DAN conclusion

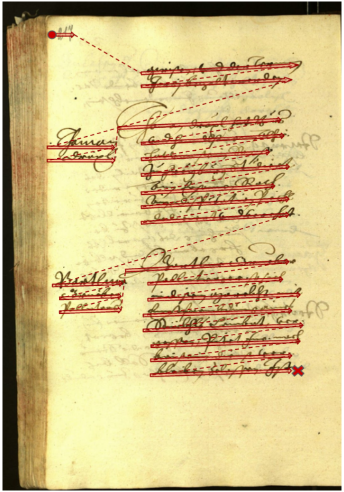
- A unique end-to-end process
- Structured output sequence
- No need for any physical segmentation annotation
- Can follow the slant of the lines (character-level attention)

Line-level / paragraph-level limitations

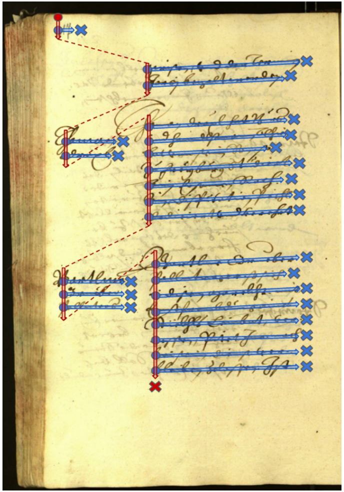
- ~~Three steps treated independently~~
- ~~A complex pipeline, hard to maintain~~
- ~~Cumulative errors between steps~~
- ~~Additional segmentation annotations~~
- ~~Rule-based reading order~~

Drawback: prediction times grow with the character sequence ($\sim$ 1 second / 100 characters)

Faster DAN: parallelizing text line recognition [21]

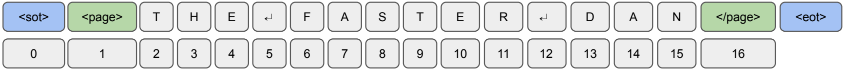


(a) DAN

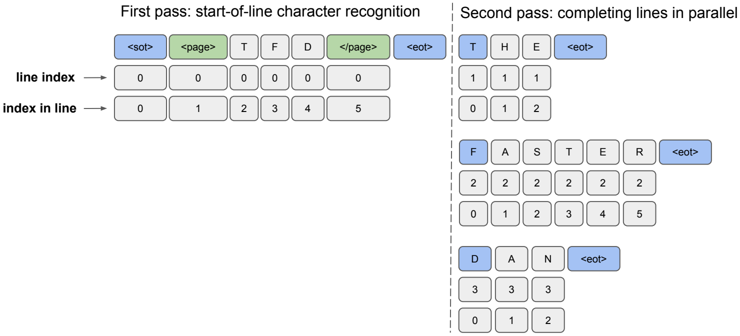


(b) Faster DAN

Faster DAN - Positional encoding

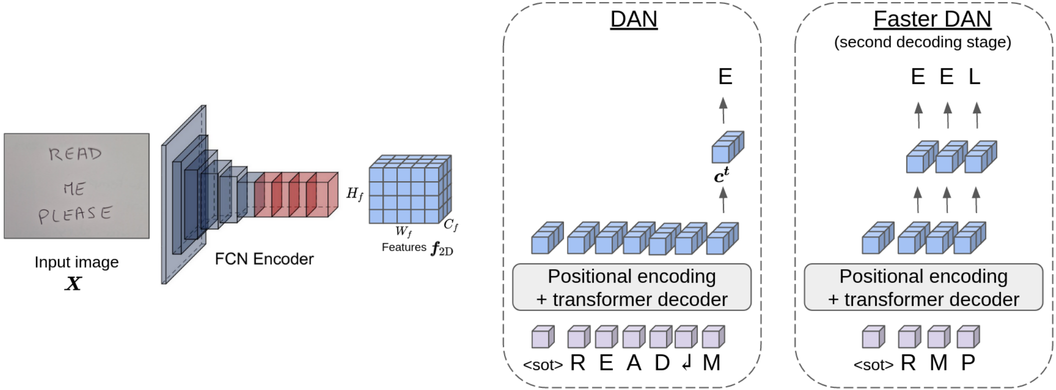


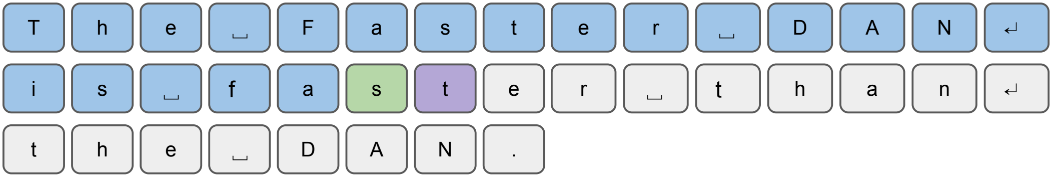
(a) DAN single-pass prediction process



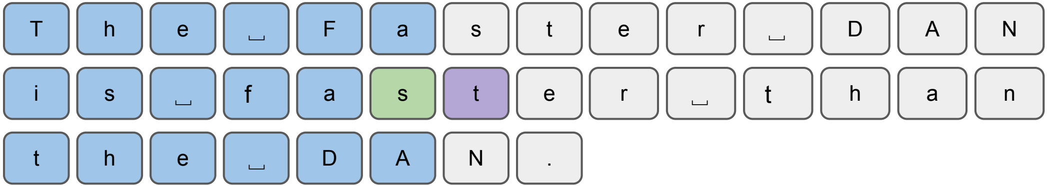
(b) Faster DAN two-pass prediction process

Faster DAN - Multi-target queries





(a) Context used by the DAN



(b) Context used by the Faster DAN

Results

Architecture	READ 2016 (single-page)				READ 2016 (double-page)			
	CER ↓	WER ↓	LOER ↓	$\mathrm{mAP}_{\mathrm{CER}}$ ↑	CER ↓	WER ↓	LOER ↓	$\mathrm{mAP}_{\mathrm{CER}}$ ↑
DAN [15]	**3.43**	**13.05**	5.17	93.32	**3.70**	**14.15**	4.98	93.09
Faster DAN [21]	3.95	14.06	**3.82**	**94.20**	3.88	14.97	**3.08**	**94.54**

Architecture	RIMES 2009			
	CER ↓	WER ↓	LOER ↓	$\mathrm{mAP}_{\mathrm{CER}}$ ↑
DAN [15]	**4.54**	**11.85**	**3.82**	**93.74**
Faster DAN [21]	6.38	13.69	4.48	91.00

Prediction times

	RIMES 2009	READ 2016	
		single-page	double-page
Dataset details (averaged for a document on the test set)			
width (px)	1,235	1,190	2,380
height (px)	1,751	1,755	1,755
# chars	578	528	1,062
# lines	18	23	47
# chars / line	31	22	22
# layout tokens	11	15	30
Prediction times (in seconds)			
DAN [15]	5.6	4.6	8.5
Faster DAN [21]	**1.4**	**0.9**	**1.9**
Speed factor	x4	x5.1	x4.5

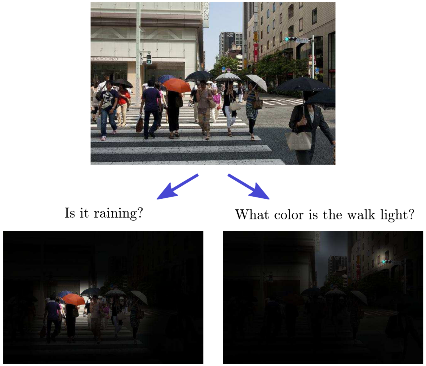
Faster DAN demonstration

https://youtu.be/_pBsO2W8XRE

A dog is standing on a hardwood floor.

A group of people sitting on a boat in the water.

Image captioning [22]



Visual Question-Answering [23]

What's next for HDR?

Still some limitations:

- Models are layout-specific
- Models are language-specific
- Models only recognize raw text items (what about equations, tables, images?)
- Prediction are still "slow"

➤ Next time: practical session!

[1] Guillaume Renton, Yann Soullard, Clément Chatelain, Sébastien Adam, Christopher Kermorvant, and Thierry Paquet. "Fully convolutional network with dilated convolutions for handwritten text line segmentation". In: *International Journal on Document Analysis and Recognition (IJDAR)* 21.3 (2018), pp. 177–186.

[2] Mélodie Boillet, Christopher Kermorvant, and Thierry Paquet. "Robust text line detection in historical documents: learning and Evaluation methods". In: *International Journal on Document Analysis and Recognition (IJDAR)* (2022).

[3] Manuel Carbonell, Alicia Fornés, Mauricio Villegas, and Josep Lladós. "A neural model for text localization, transcription and named entity recognition in full pages". In: *Pattern Recognition Letters* 136 (2020), pp. 219–227.

[4] Alex Graves and Jürgen Schmidhuber. "Offline Handwriting Recognition with Multidimensional Recurrent Neural Networks". In: *Advances in Neural Information Processing Systems 21 (NIPS)*. 2008, pp. 545–552.

[5] Paul Voigtlaender, Patrick Doetsch, and Hermann Ney. "Handwriting Recognition with Large Multidimensional Long Short-Term Memory Recurrent Neural Networks". In: *International Conference on Frontiers in Handwriting Recognition (ICFHR)*. 2016, pp. 228–233.

[6] Curtis Wigington, Seth Stewart, Brian L. Davis, Bill Barrett, Brian L. Price, and Scott Cohen. "Data Augmentation for Recognition of Handwritten Words and Lines Using a CNN-LSTM Network". In: *International Conference on Document Analysis and Recognition (ICDAR)*. 2017, pp. 639–645.

[7] Joan Puigcerver. "Are Multidimensional Recurrent Layers Really Necessary for Handwritten Text Recognition?" In: *14th IAPR International Conference on Document Analysis and Recognition (ICDAR)*. 2017, pp. 67–72.

[8] Raymond W. Ptucha, Felipe Petroski Such, Suhas Pillai, Frank Brockler, Vatsala Singh, and Paul Hutkowski. "Intelligent character recognition using fully convolutional neural networks". In: *Pattern Recognition* 88 (2019), pp. 604–613.

[9] Denis Coquenet, Yann Soullard, Clément Chatelain, and Thierry Paquet. "Have convolutions already made recurrence obsolete for unconstrained handwritten text recognition ?" In: *Workshop on Machine Learning (WML@ICDAR)*. 2019, pp. 65–70.

[10] Mohamed Yousef, Khaled F. Hussain, and Usama S. Mohammed. "Accurate, data-efficient, unconstrained text recognition with convolutional neural networks". In: *Pattern Recognition* 108 (2020), p. 107482.

[11] Denis Coquenet, Clément Chatelain, and Thierry Paquet. "Recurrence-free unconstrained handwritten text recognition using gated fully convolutional network". In: *17th International Conference on Frontiers in Handwriting Recognition (ICFHR)*. 2020, pp. 19–24.

[12] Alex Graves, Santiago Fernández, Faustino J. Gomez, and Jürgen Schmidhuber. "Connectionist temporal classification: labelling unsegmented sequence data with recurrent neural networks". In: *International Conference on Machine Learning (ICML)*. Vol. 148. 2006, pp. 369–376.

[13] Killian Barrere, Yann Soullard, Aurélie Lemaitre, and Bertrand Coüasnon. "A Light Transformer-Based Architecture for Handwritten Text Recognition". In: *Document Analysis Systems - 15th IAPR International Workshop*. Vol. 13237. 2022, pp. 275–290.

[14] Christoph Wick, Jochen Zöllner, and Tobias Grüning. "Transformer for Handwritten Text Recognition Using Bidirectional Post-decoding". In: *16th International Conference on Document Analysis and Recognition (ICDAR)*. Vol. 12823. 2021, pp. 112–126.

[15] Denis Coquenet, Clément Chatelain, and Thierry Paquet. "DAN: a Segmentation-free Document Attention Network for Handwritten Document Recognition". In: *IEEE Transactions on Pattern Analysis and Machine Intelligence (TPAMI)* 45.7 (2023), pp. 8227–8243,

[16] Denis Coquenet, Clément Chatelain, and Thierry Paquet. "End-to-end Handwritten Paragraph Text Recognition Using a Vertical Attention Network". In: *IEEE Transactions on Pattern Analysis and Machine Intelligence (TPAMI)* 45.1 (2023), pp. 508–524.

[17] Denis Coquenet, Clément Chatelain, and Thierry Paquet. "SPAN: a Simple Predict & Align Network for Handwritten Paragraph Recognition". In: *International Conference on Document Analysis and Recognition (ICDAR)*. Vol. 12823. 2021, pp. 70–84.

[18] Théodore Bluche. "Joint Line Segmentation and Transcription for End-to-End Handwritten Paragraph Recognition". In: *Advances in Neural Information Processing Systems 29 (NIPS)*. 2016, pp. 838–846.

References IV

[19] Johannes Michael, Roger Labahn, Tobias Grüning, and Jochen Zöllner. "Evaluating Sequence-to-Sequence Models for Handwritten Text Recognition". In: *International Conference on Document Analysis and Recognition (ICDAR)*. 2019, pp. 1286–1293.

[20] Joan-Andreu Sánchez, Verónica Romero, Alejandro H. Toselli, and Enrique Vidal. "ICFHR2016 Competition on Handwritten Text Recognition on the READ Dataset". In: *15th International Conference on Frontiers in Handwriting Recognition (ICFHR)*. 2016, pp. 630–635.

[21] Denis Coquenet, Clément Chatelain, and Thierry Paquet. "Faster DAN: Multi-target Queries with Document Positional Encoding for End-to-end Handwritten Document Recognition". In: *International Conference on Document Analysis and Recognition (ICDAR)*. Vol. 14190. Lecture Notes in Computer Science. 2023, pp. 182–199.

[22] Kelvin Xu, Jimmy Ba, Ryan Kiros, Kyunghyun Cho, Aaron C. Courville, Ruslan Salakhutdinov, Richard S. Zemel, and Yoshua Bengio. "Show, Attend and Tell: Neural Image Caption Generation with Visual Attention". In: *Proceedings of the 32nd International Conference on Machine Learning*. Vol. 37. 2015, pp. 2048–2057.

[23] Kevin J. Shih, Saurabh Singh, and Derek Hoiem. "Where to Look: Focus Regions for Visual Question Answering". In: *2016 IEEE Conference on Computer Vision and Pattern Recognition*. 2016, pp. 4613–4621.