Graph Transformer Networks

Léon Bottou

Joint work with
Yann Le Cun
Yoshua Bengio
Patrick Haffner

AT&T Labs – Research

Middletown, NJ

SUMMARY

Graph Transformer Network

Overview

Example: Word Reader

Step by step example
Scores vs. Probabilities
Training

Example: Check Reader

Graph Composition
GTN Building Blocks
Real World test.

SDNN

Replicated Convolutional Network
In–Seg vs. Out–Seg
Animations

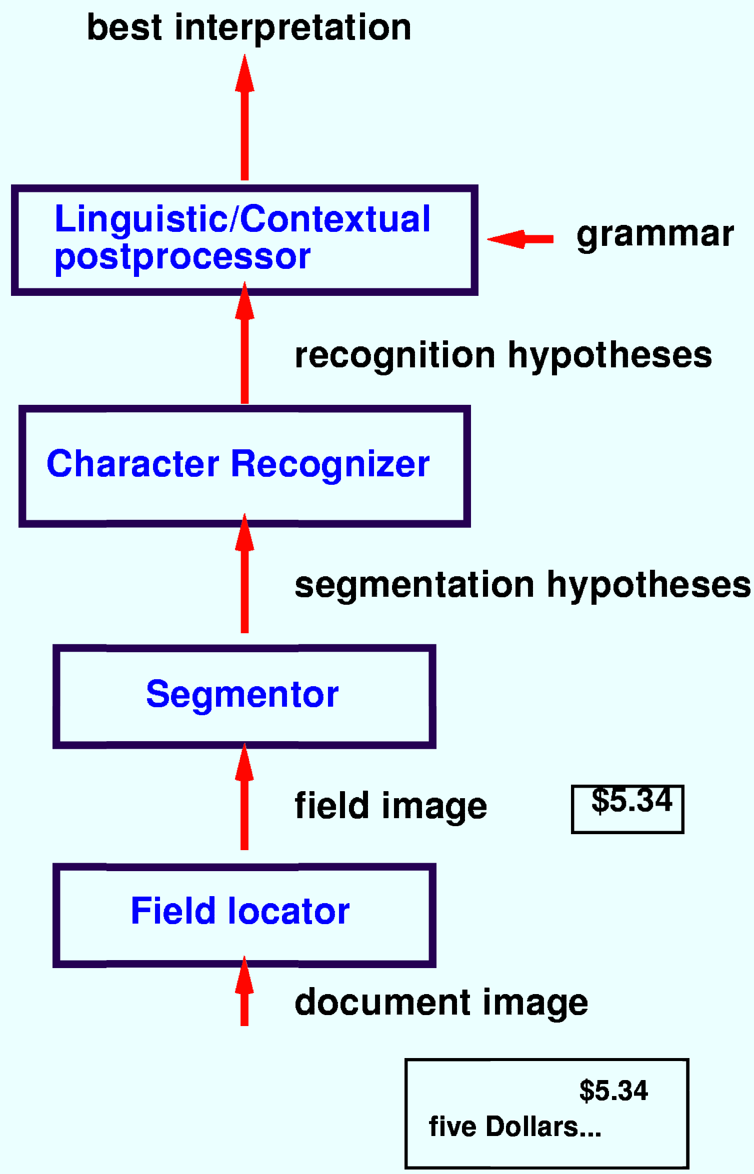
DOCUMENT RECOGNITION: THE TRADITIONAL WAY

Built by hand and manually adjusted.

Hand crafted features. Classifier trained on segmented characters.

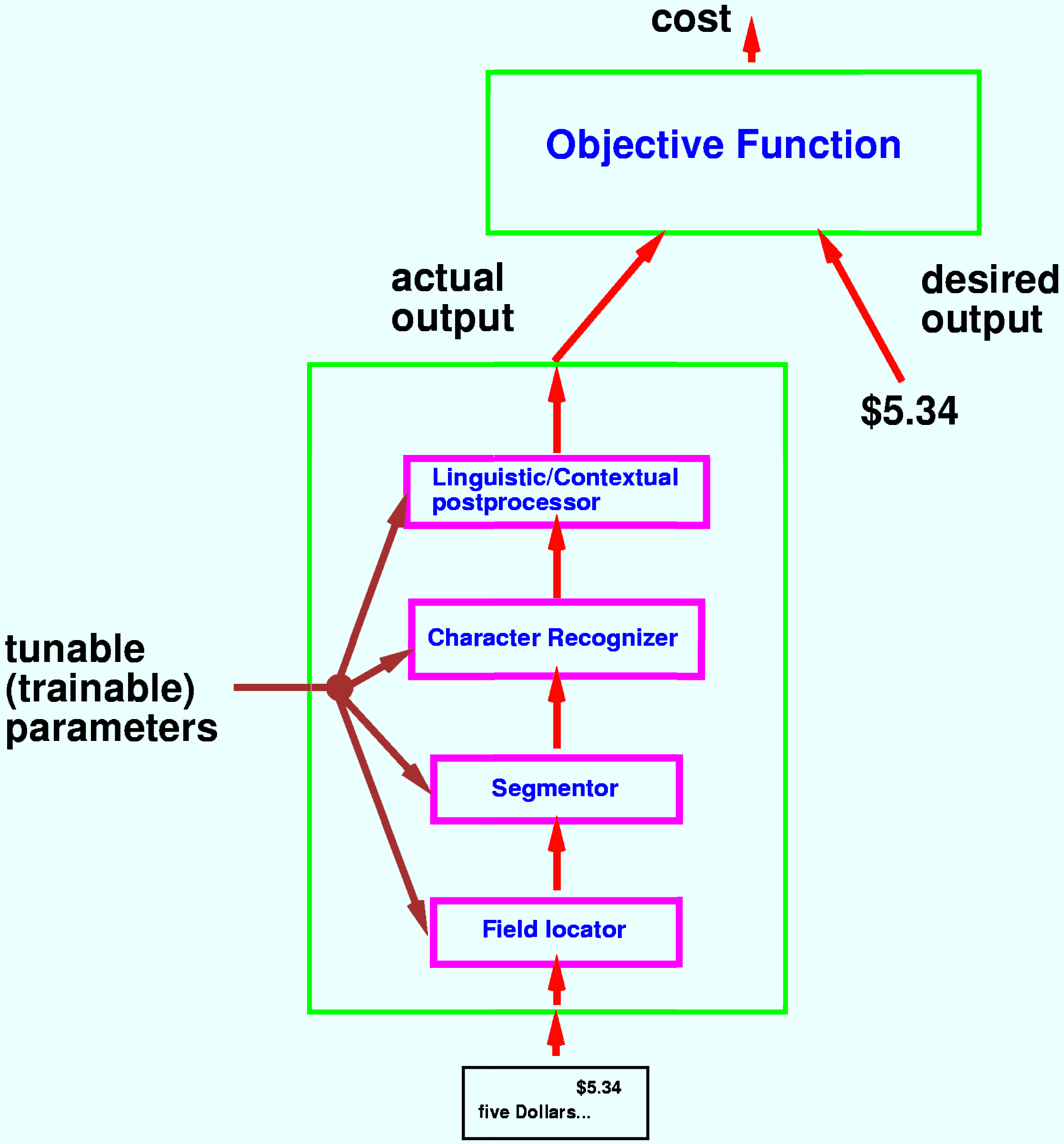
Built by hand and manually adjusted.

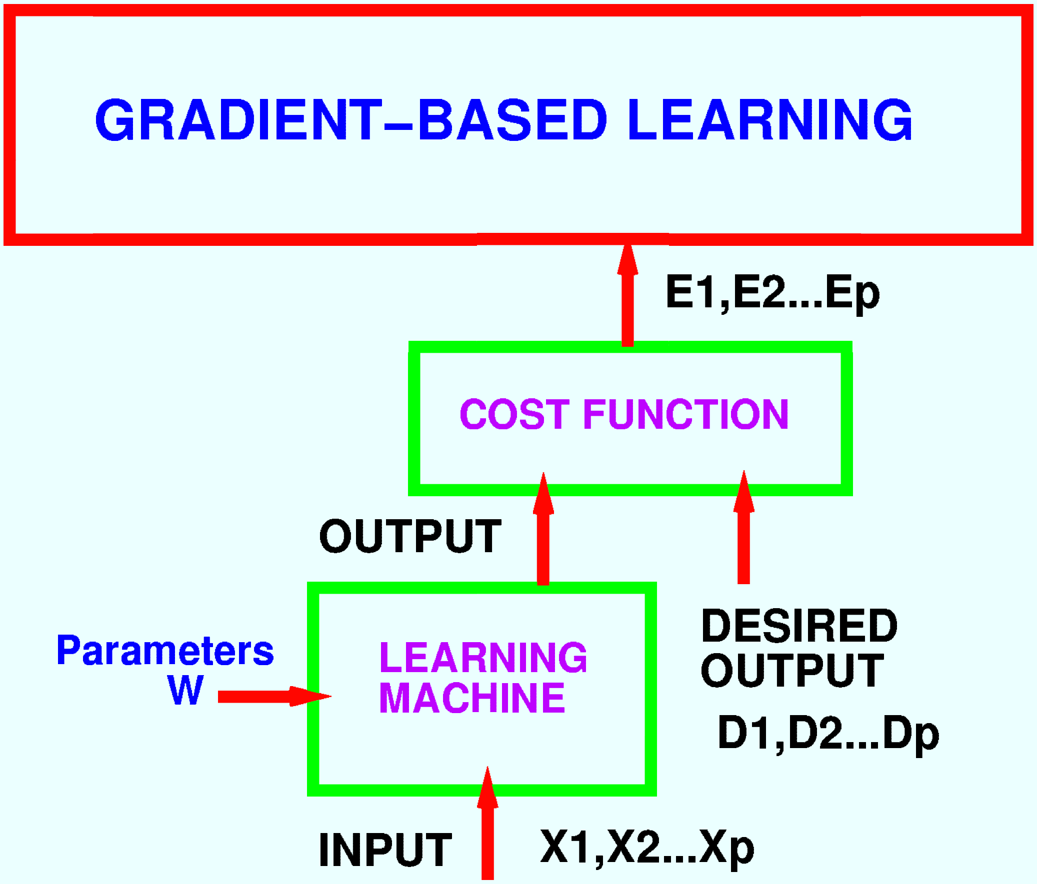
Built by hand and manually adjusted.



WHAT WE REALLY WANT

Train all the parameters in the system to optimize a global performance measure





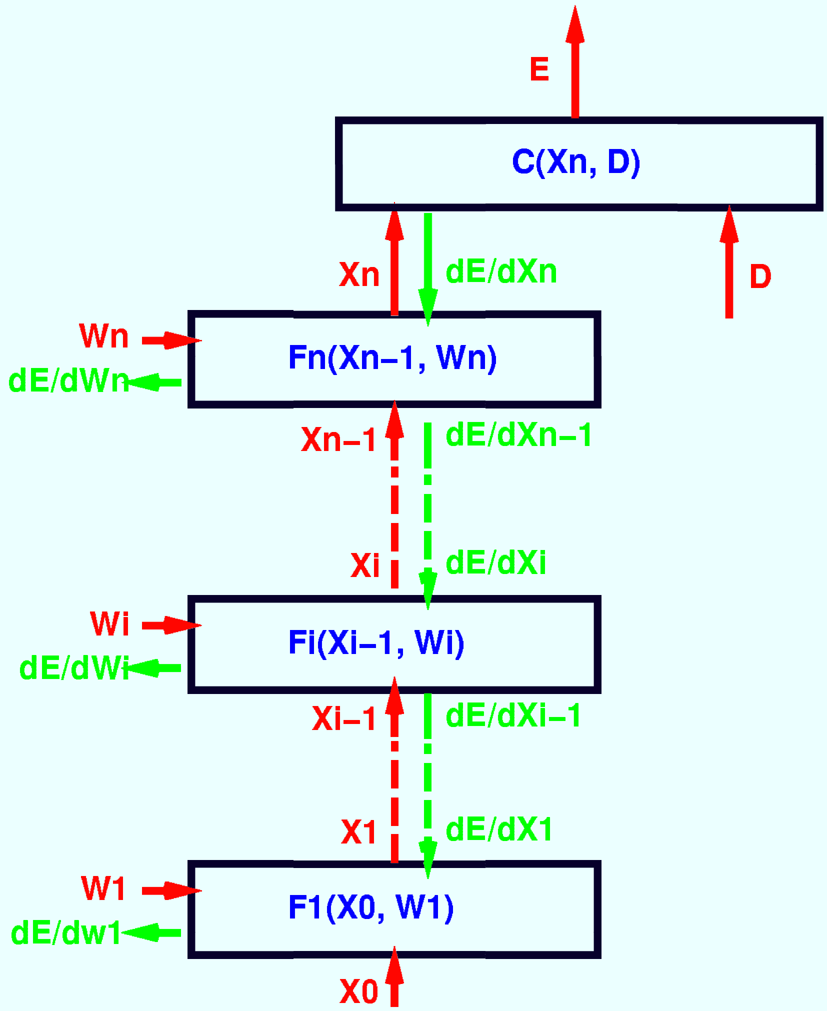
When the cost function and the learning machine module are differentiable with respect to the parameters, gradient−based methods can be used to minimize the cost function.

The learning machine can be as complex as desired, as long as it is composed of multiple differentiable "modules" [Layers of neurons and weights, RBF,...]

GRADIENT−BASED learning is the unifying concept behind many adaptive pattern recognition methods.

GRADIENT–BASED LEARNING IN MODULAR SYSTEMS

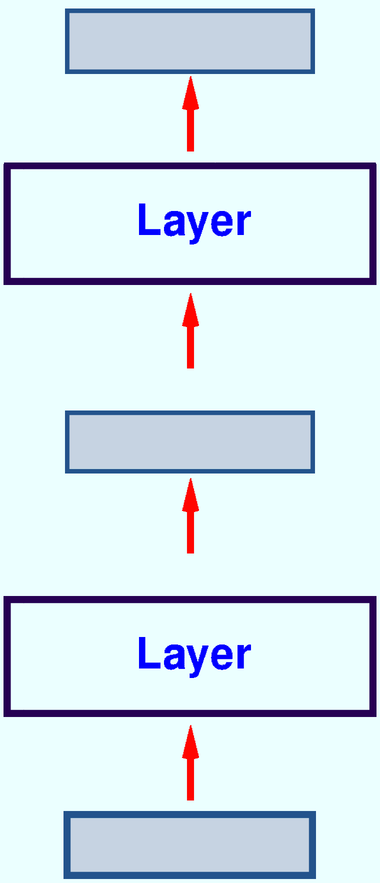
backpropagating gradient through differentiable modules [Bottou & Gallinari 1991]



$$\partial E/\partial Xi{-}1 = \partial Fi(Xi{-}1, Wi)/\partial Xi{-}1 \quad \partial E/\partial Xi$$

Jacobian of Fi

Multi Layer Network



State variables :

Fixed Size Vectors

Probabilistic Interpretation :

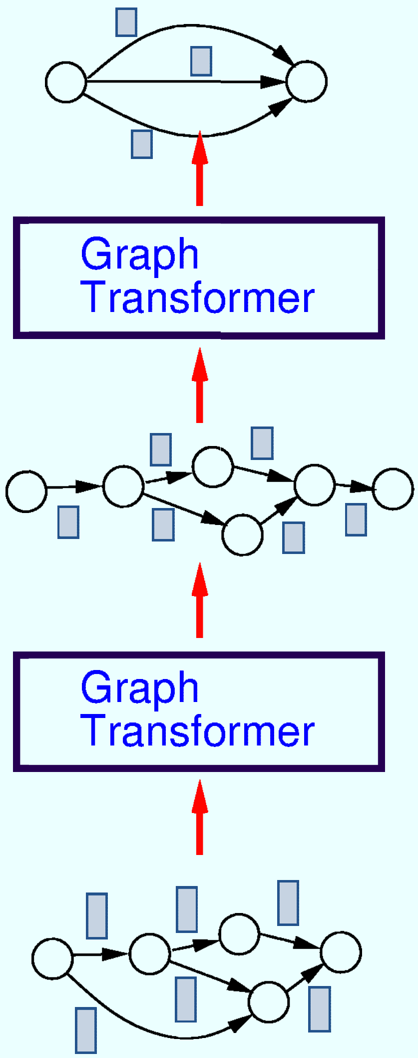
Simple Probability Distribution over Vectors
represented by its mean.

Training Procedure :

Optimise Mean Squared Error
Sometimes described
as Maximum Likelihood
with Gaussian Distributions.

Fixed Size Vectors
cannot represent sequential information
(e.g. speech recognition, structured images...)

Graph Transformer Network



State variables :

Weighted Graphs
with numerical information
attached to the arcs.

Probabilistic Interpretation :

Mixture Distribution
over Sequences
Arc weights are mixture coefficient

Graphs can represent alternative hypothesis.

Graphs can represent structured information.

WHY GRAPHS?

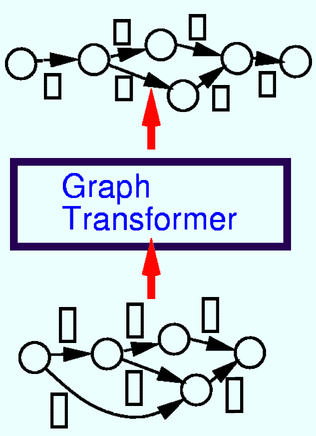
Graphs with numerical information on the arcs can represent:

– mixture distributions over sequences of symbols, vectors, or other objects (stochastic finite–state grammars).

– alternative interpretations of an input

– relationships between parts (or features) of an object

Question: Can we back–propagate gradients through graph transformer modules?



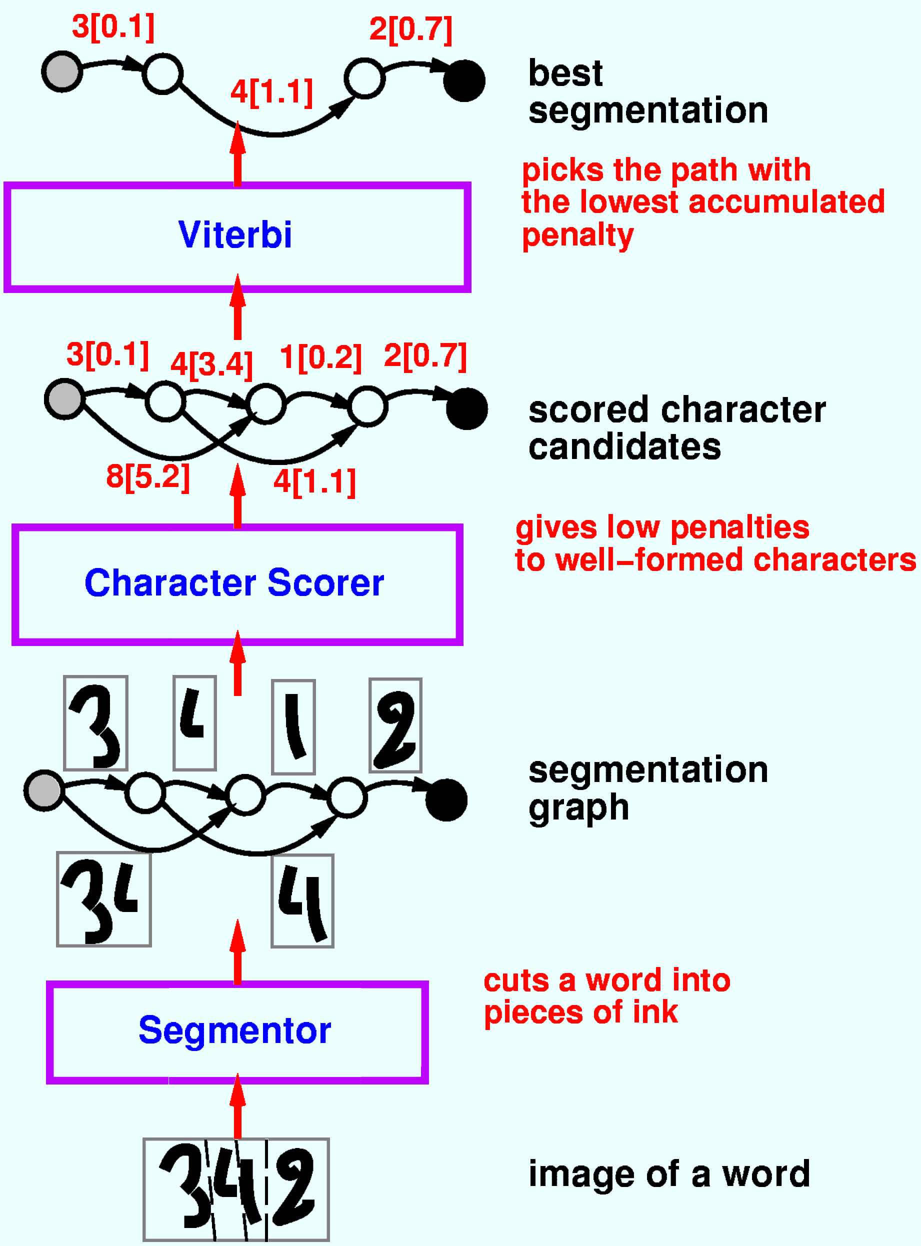
Graphs Transformation Models

- Extend Graphical Models (hmms, bayesian nets)
- Introduced in Speech and Language analysis [Pereira, Riley, Sprout, 94].

  Solid theoretical foundation.
- Our contribution: Global Discriminant Training of Graph Transformation models.

A SIMPLE EXAMPLE: WORD READER

A GTN that picks the best interpretation of a word by segmenting individual characters.



NORMALIZATION AND DISCRIMINATION

Generative (non–discriminant) training

Estimate P(x,y)

– define parametric model p(x,y,w)

$$\sum_{x,y} p(x,y,w) = 1 \qquad \text{(forall w)}$$

– maximize

$$\sum_{i} \log p(x_i,y_i,w)$$

The difference is the normalization

Discriminant training

Estimate P(y|x)

– define parametric model p(x,y,w)

$$\sum_{y} p(x,y,w) = 1 \qquad \text{(forall x,w)}$$

– maximize

$$\sum_{i} \log p(x_i,y_i,w)$$

PROBABILISTIC MODELS

Building models using probability functions

Generative example:
Hidden Markov Model

$$p(x,y,w) = p(x,y|w) = \sum_{s[t]:y} \prod_{t} p(\, s[t] \mid s[t-1] \,)\, p(\, x[t] \mid s[t] \,)$$

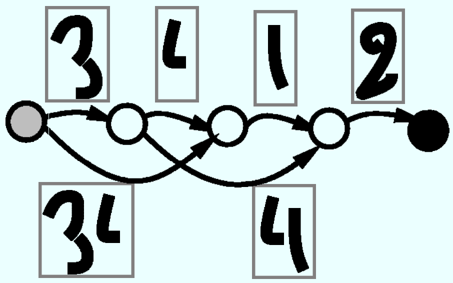
Probabilistic construction ensures normalization!

Discrimant Example
Discriminant Hidden Markov Model

$$p(x,y,w) = p(x,y|w) = \sum_{s[t]:y} \prod_{t} p(\, s[t] \mid x[t], s[t-1], \ldots \,)$$

Output of the local classifier must be normalized (softmax).

Ensures normalization. but is a BAD idea !

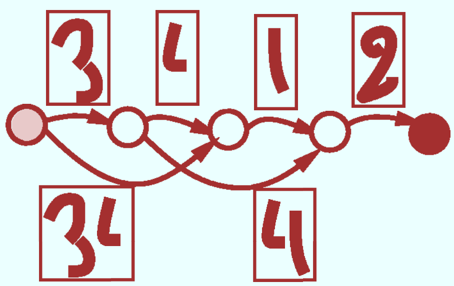


DENORMALIZED MODELS

Building models using "measures" (probabilities minus normalization)

– Use "penalties" instead of probabilities

– A "score" $e^{-\,\text{penalty}}$
is almost a probability but without normalization.
Additions, multiplications work like probabilities...

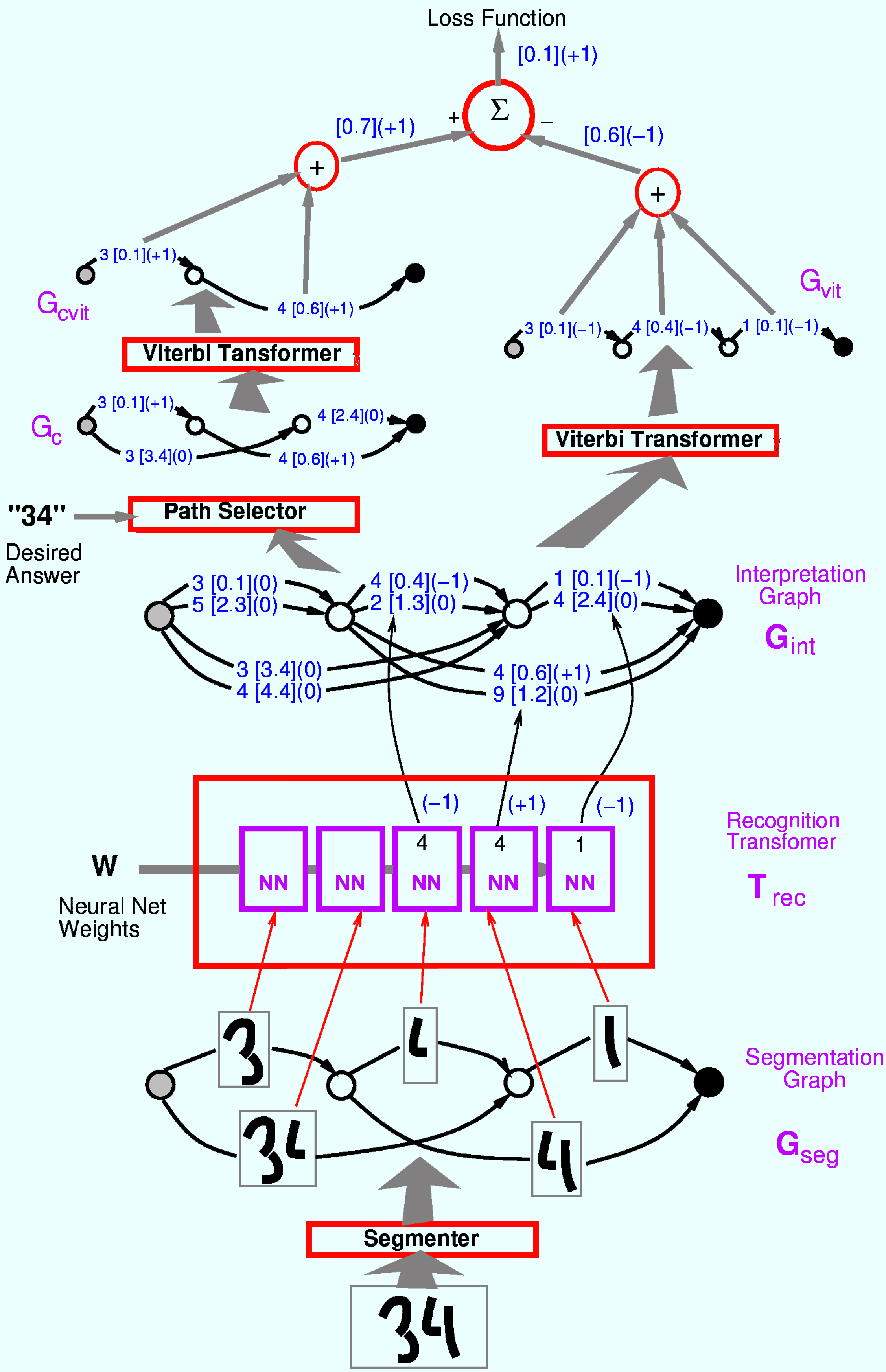


Score of a path = Product of the scores of its arcs.

Score of a subgraph = Sum (or Max) of the scores of its paths.

– Probabilities can be recovered by normalizing.
That is only necessary at the global level.

Train by maximizing : $\sum_i \log \frac{p(x_i, y_i, w)}{\sum_y p(x_i, y, w)}$



Loss Function
[0.1](+1)
[0.7](+1)
[0.6](−1)
3 [0.1](+1)
4 [0.6](+1)
Gcvit
Viterbi Tansformer
Gvit
3 [0.1](−1)
4 [0.4](−1)
1 [0.1](−1)
Gc
3 [0.1](+1)
4 [2.4](0)
3 [3.4](0)
4 [0.6](+1)
Viterbi Transformer
"34"
Path Selector
Desired Answer
3 [0.1](0)
5 [2.3](0)
4 [0.4](−1)
2 [1.3](0)
1 [0.1](−1)
4 [2.4](0)
Interpretation Graph
Gint
3 [3.4](0)
4 [4.4](0)
4 [0.6](+1)
9 [1.2](0)
(−1)
(+1)
(−1)
W
Neural Net Weights
NN
Recognition Transfomer
Trec
Segmentation Graph
Gseg
Segmenter

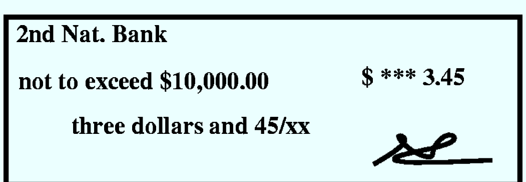
A Check Reader

[Bottou, LeCun, Burges, Nohl, Bengio, Haffner]

Reading the "Courtesy Amount" (numeric amount)

– Business Checks:

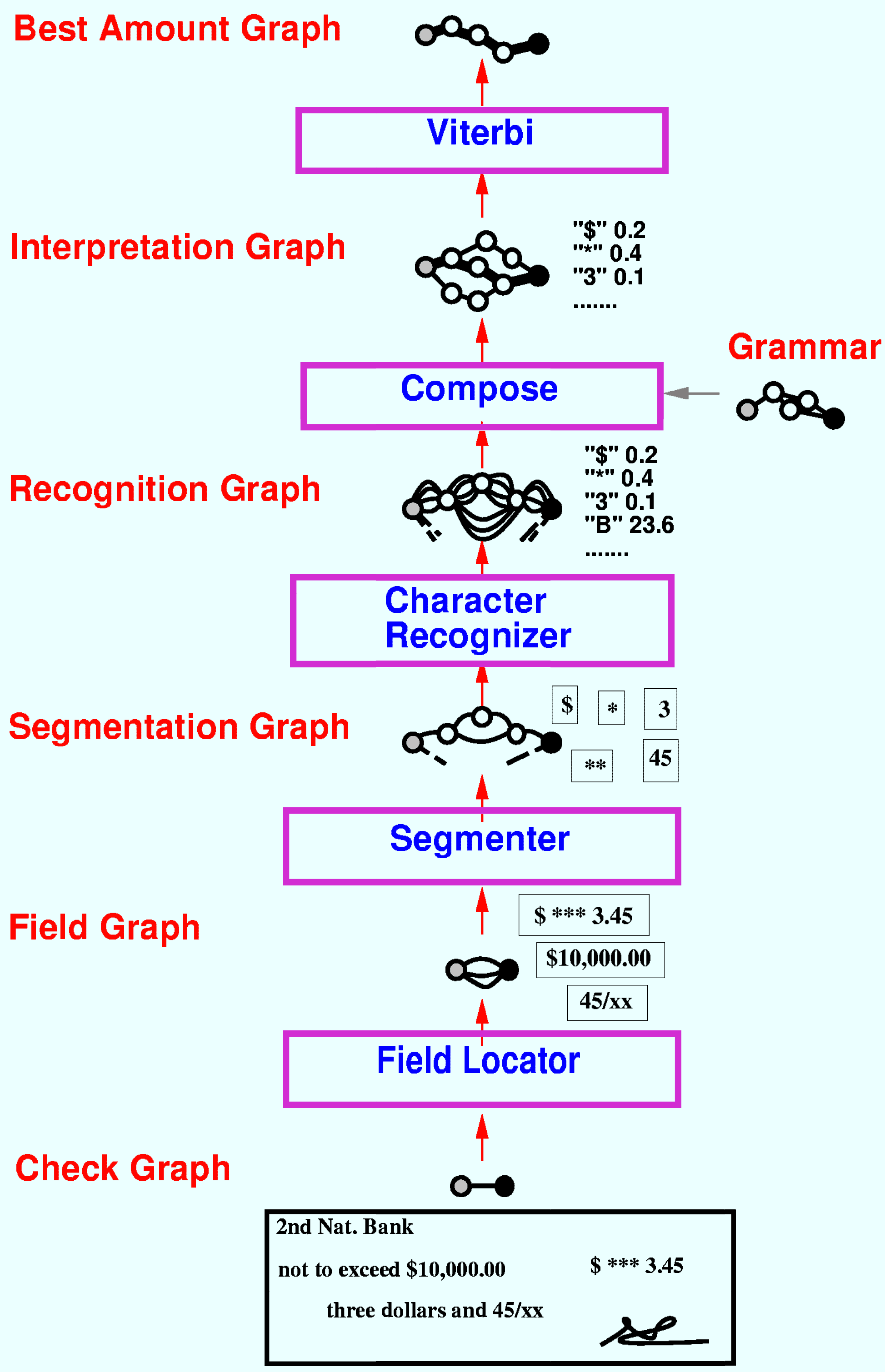
- – usually machine printed
- – layout not standardized
- – amount difficult to find
- – amount grammar not standardized
  ($****1*234*12*****)
- – not always easy to segment and read
  (dot matrix printers)



– Personal Checks:

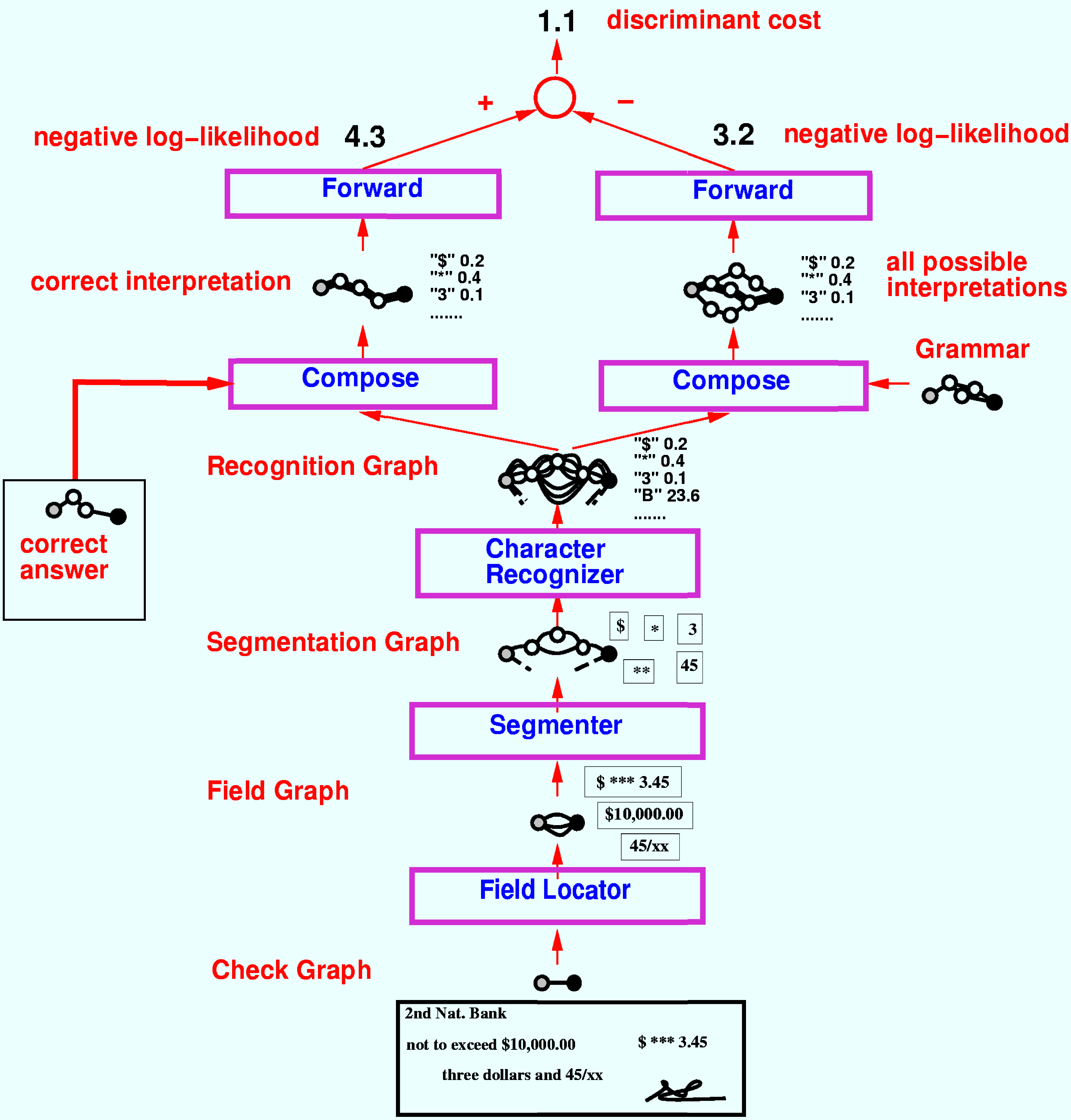
- – handwritten
- – layout more or less standardized
- – hard to segment
- – hard to read

Check Reader: Recognition Architecture



Check Reader: Training Architecture

gradient–based discriminant training



Graph Transformer Networks:
It works!

- Partial implentation
  based on previous work [Burges,Nohl, et al.]

  Graph Transformer Network runs
  – when the check is determined to be machine printed
  – using an uncleaned field image

- LeNet5 bootstraped on
  500,000 images of characters
  from various origins:

  – full printable ASCII set (95 classes)
  – machine printed and handwritten

- Accuracy [1995] : (correct / reject / error)

	old system (was state of the art)	new system (with graph transformers)
654 machine printed checks	68 / 31 / 1	82 / 17 / 1
realistic mixture of 1986 checks	45 / 54 / 1	50 / 49 / 1

- Integrated in NCRs check reading machines.
  Commercially deployed since June 1996.

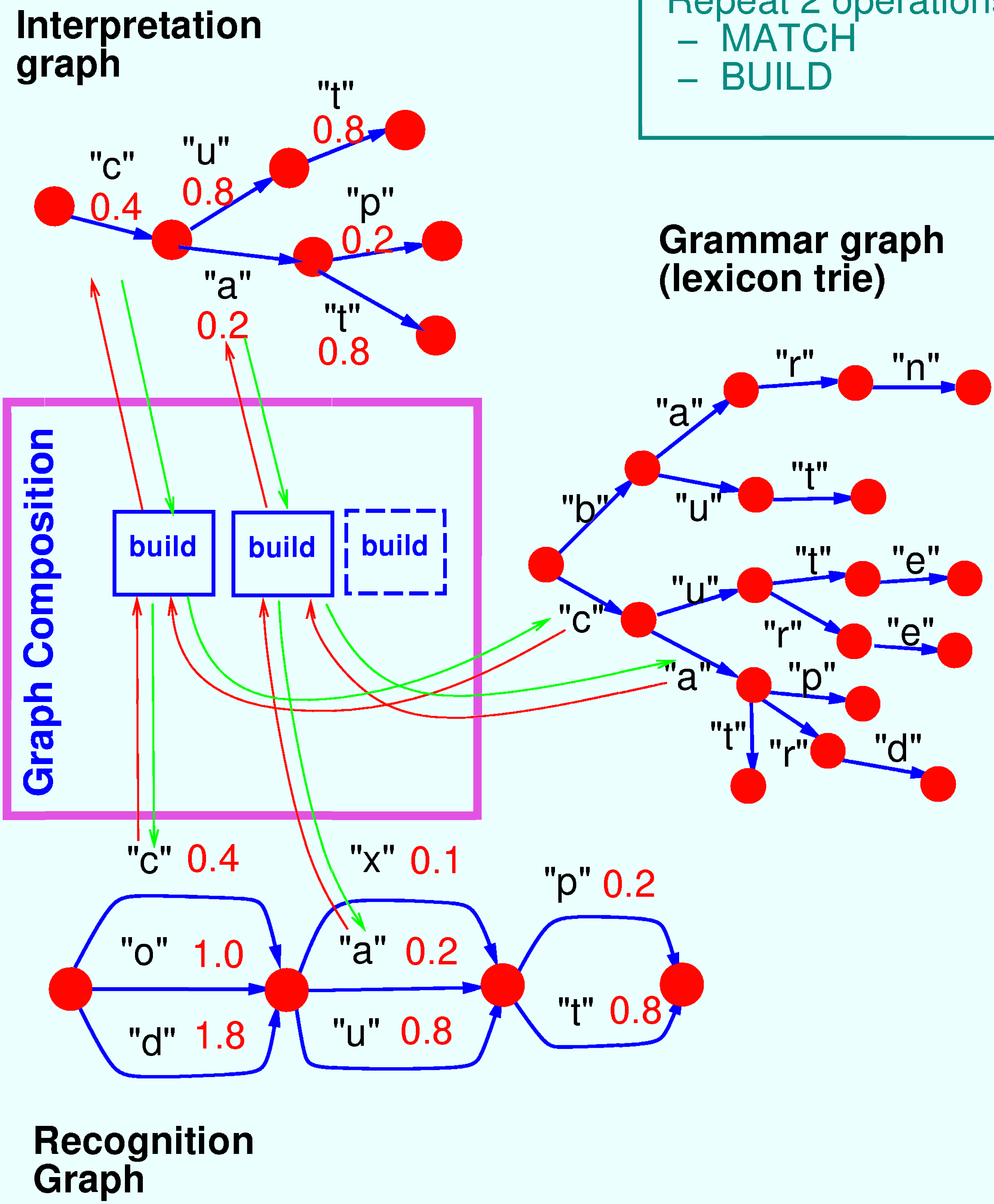
  Current estimates:
  – Processes 20,000,000 checks per day
  – Or 10% of all the checks in the U.S.

A CLASSIFICATION OF (USEFUL) TRANSFORMERS

	Composition Transformers	Pruning Transformers
– field location	✓	
– segmentation	✓	
– recognition	✓	
– grammatical constraints	✓	
– sequence normalization	✓	
– selection of correct paths	✓	
– viterbi		✓
– k–best paths		✓

Example: a lexicon

Repeat 2 operations
- MATCH
- BUILD



COMPOSITION TRANSFORMERS

Definition :

Perform a composition with a predefined "transducer" graph.

Remark :

The "transducer graph" is entirely defined by providing two functions: MATCH and BUILD.

[cf. slide about composition]

Examples:

- **Graph Expansion Operations**
  - field location
  - segmentation
  - recognition

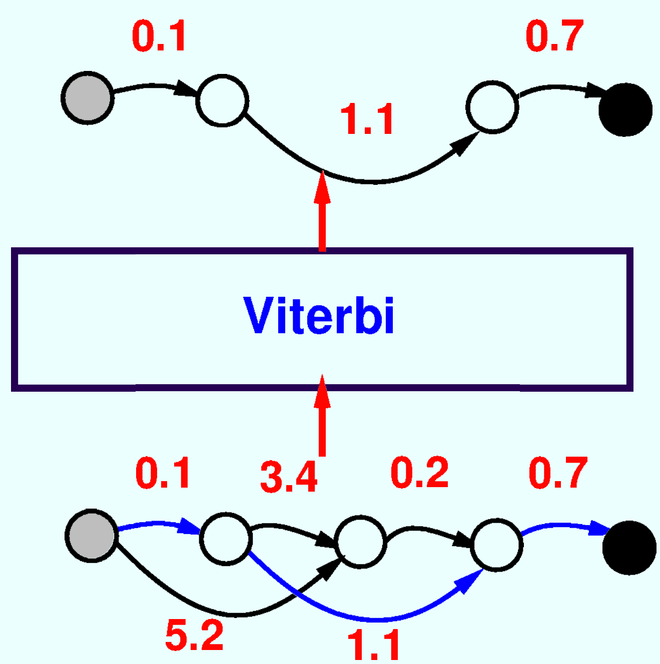
- **Graph Filtering and Rewriting**
  - grammatical constraints
  - amount normalization

Many Transformers
Same Code
(except functions
match & build)

PRUNING GRAPH TRANSFORMERS

Definition:

Remove selected arcs from a graph.



Examples:

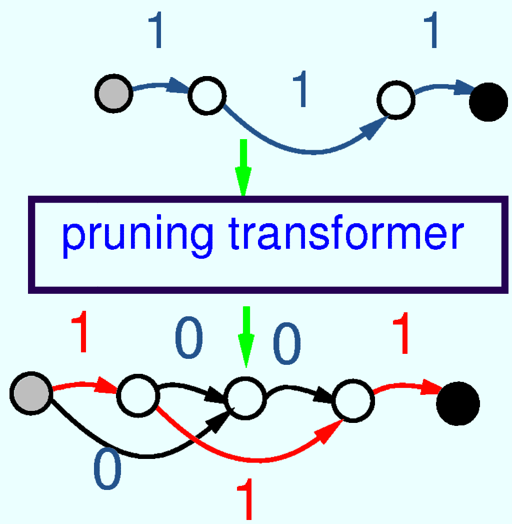
- Simple pruning algorithms
- Best path search algorithms
  - Viterbi, K Best Paths, Stack Decoding
  - Heuristic Search (A–star, Beam search).

BACK PROPAGATION THROUGH TRANSFORMERS

Pruning Transformers

– Set all gradients to zero

– Copy gradients from non pruned arcs



Composition Transformers

– Each arc of the output graph comes from an invocation of BUILD.

– BUILD must be a differentiable function.

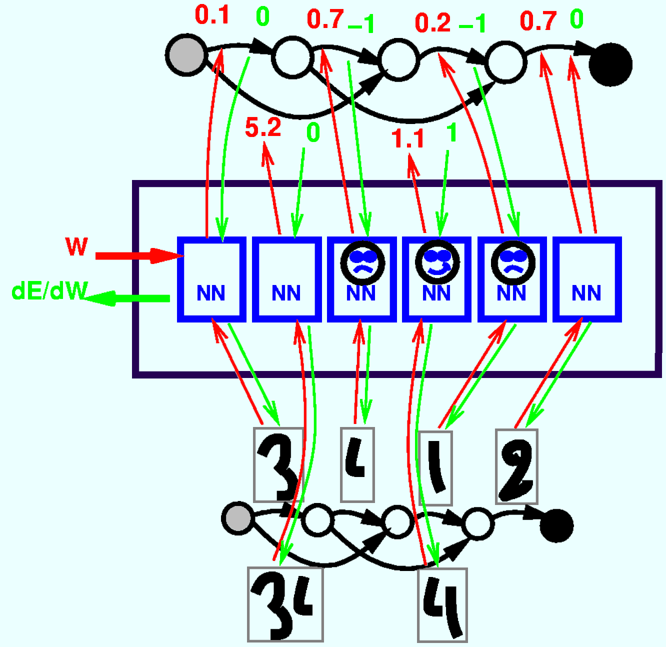
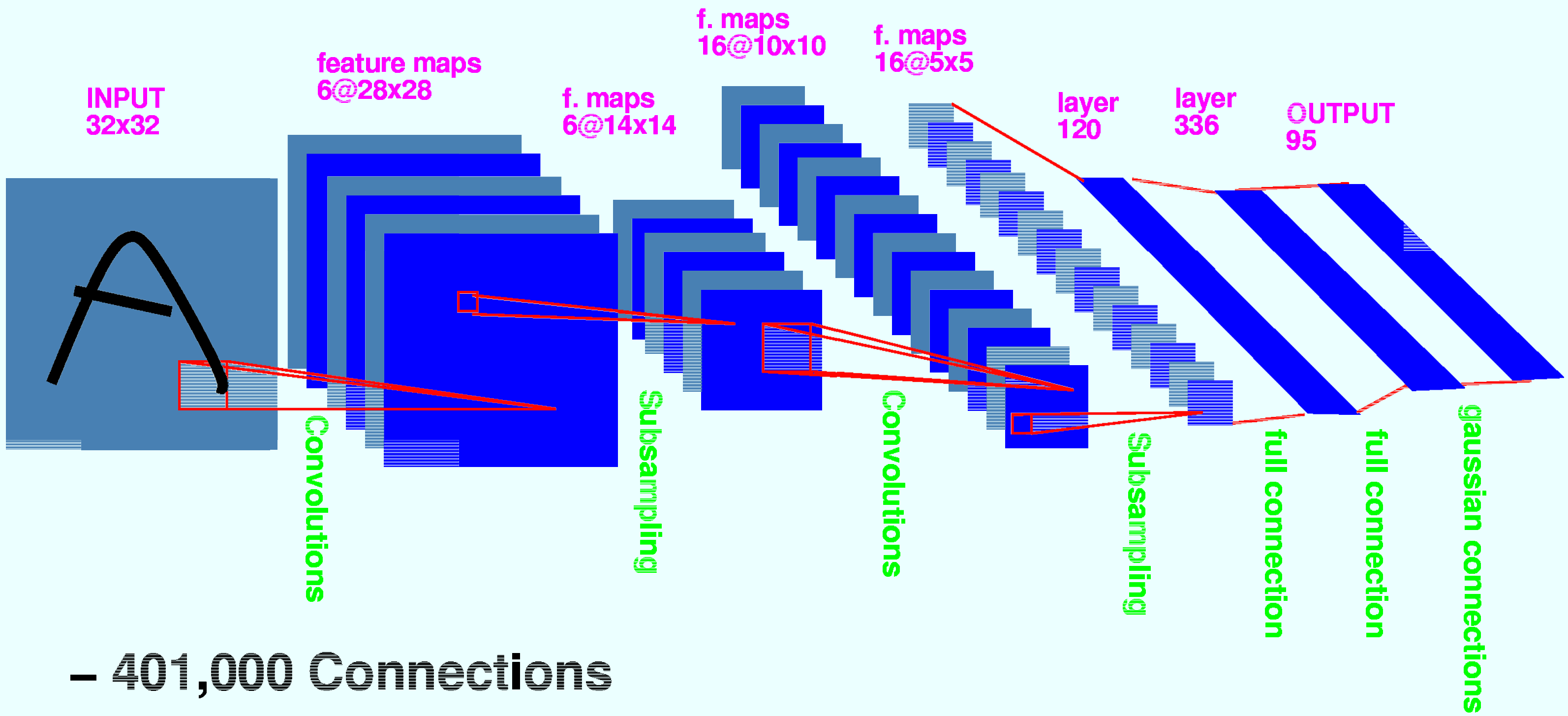


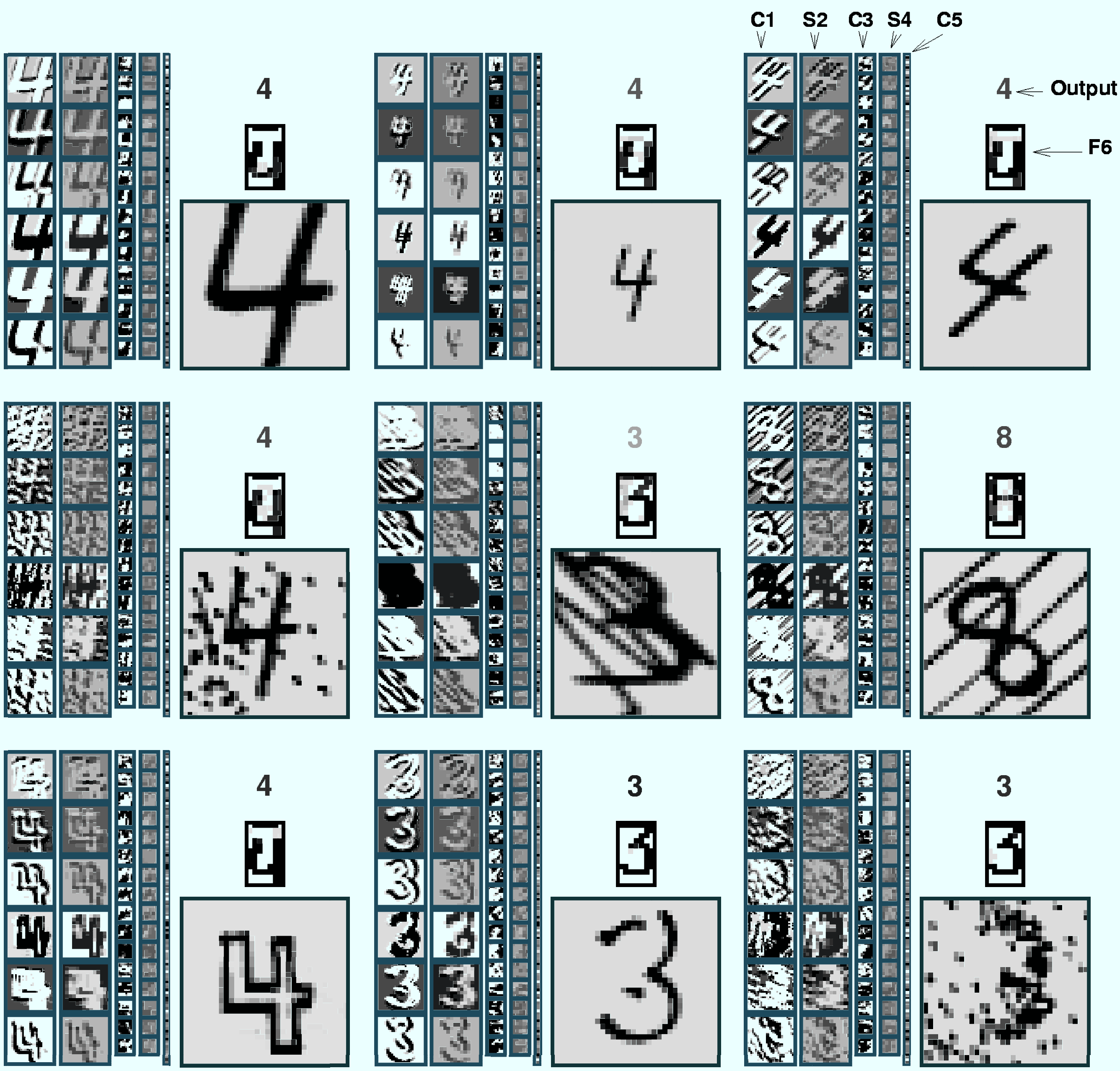
Image Recognition with "LeNet-5"



– 401,000 Connections

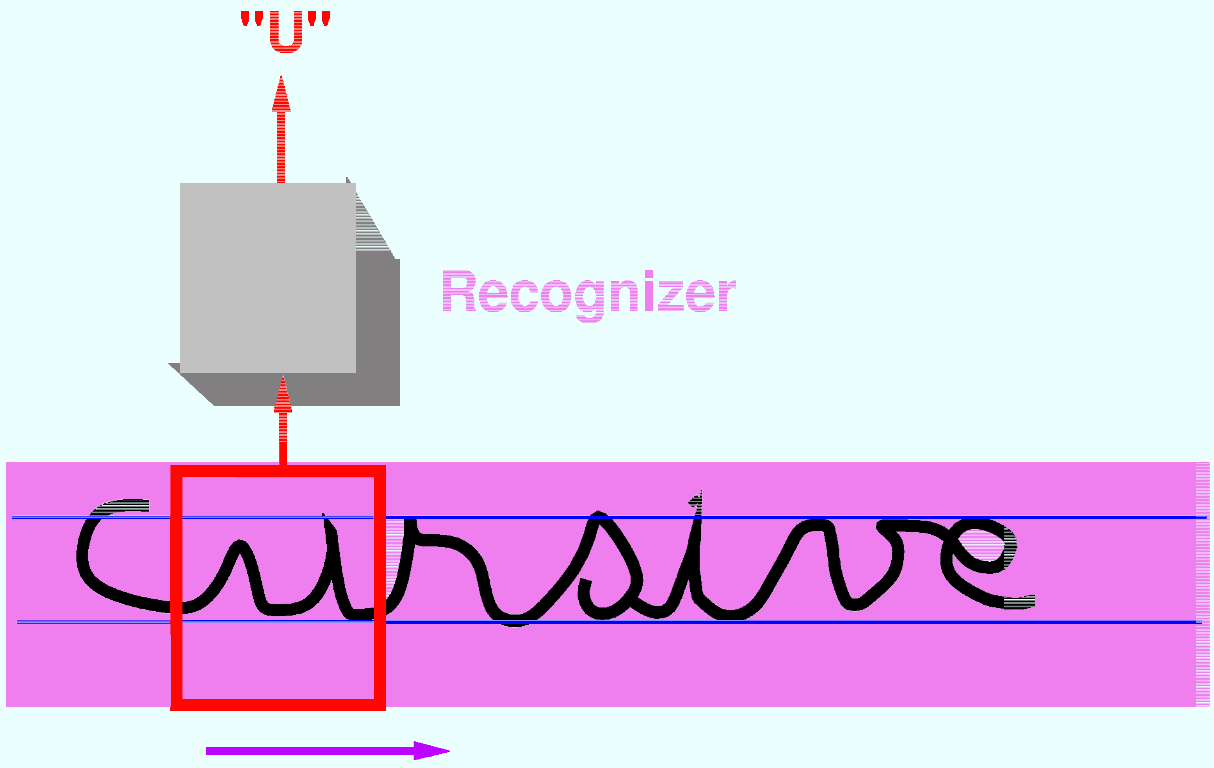
– 100,000 free parameters

– Trained with 500,000 character samples
(Full ASCII set, machine printed and handwritten)


C1
S2
C3
S4
C5
4
Output
F6
4
4
4
3
8
4
3
3

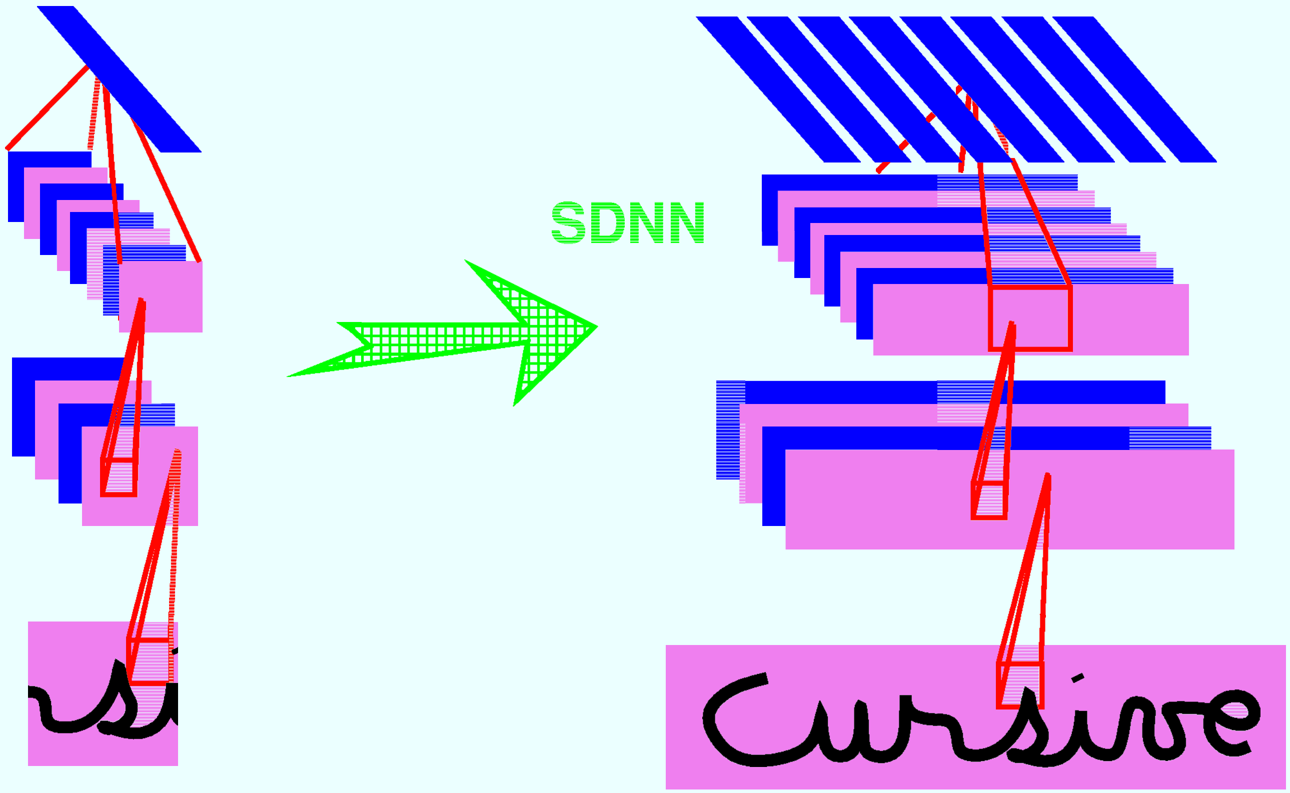
A simple (and very inefficient) way of avoiding segmentation: character spotting:

Scan the input with a recognizer for single objects



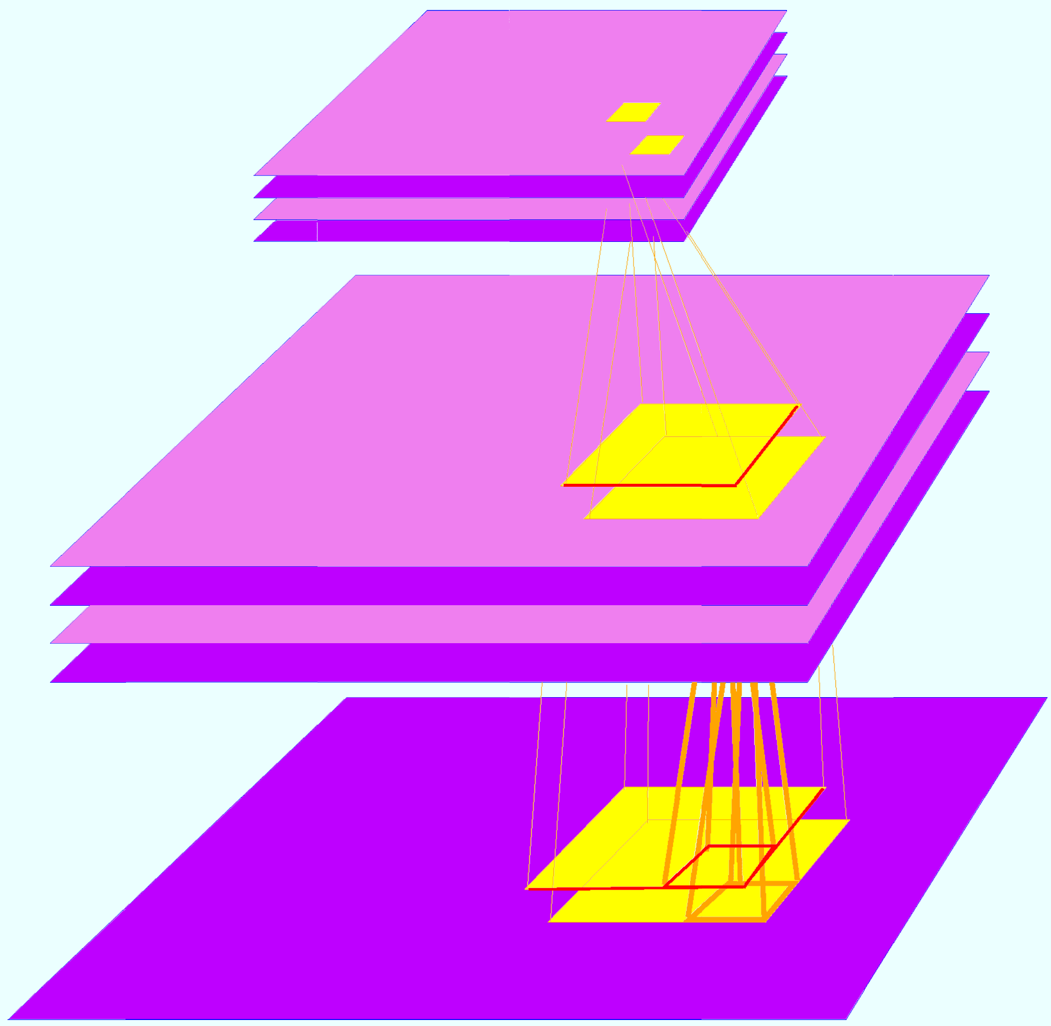
REPLICATED CONVOLUTIONAL NETWORKS FOR MULTIPLE OBJECTS RECOGNITION

(Space Displacement Neural Networks)



Single object convolutional networks are easily transformed into multi–object networks

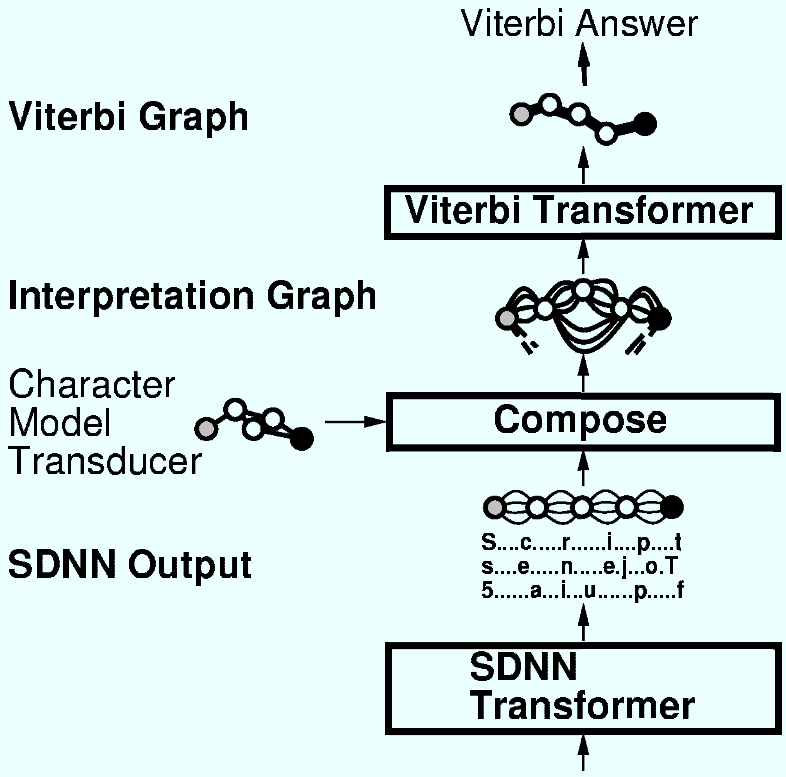
SPATIALLY REPLICATED
Convolutional Network
for Object Detection



WITH CONVOLUTIONAL NETS, SHIFT INVARIANCE BY REPLICATION IS VERY CHEAP BECAUSE MOST OF THE COMPUTATION IS SHARED BETWEEN NEIGHBORING INSTANCES.

SDNN HANDWRITING RECOGNIZER

OUTPUT INTERPRETATION WITH A WEIGTHED FINITE STATE MACHINE



SDNN HANDWRITING RECOGNIZER

REFERENCE

Le Cun, Bottou, Bengio, Haffner (1998):

Gradient Based Learning applied to
Document Recognition

Proceedings of the IEEE
86(11):2278–2324