Deep Learning and Weakly Supervised Learning
Negative Evidence Models

Séminaire annuel laboratoire CRIStAL, thématique "Image"

CRIStAL
Centre de Recherche en Informatique,
Signal et Automatique de Lille

Nicolas Thome - Conservatoire National des Arts et Métiers (Cnam)
CEDRIC Lab - Machine Learning Team (MSDMA)
5 Juillet 2018

Outline

1. Context: Big data & Deep Learning
2. Weakly Supervised Learning & Negative Evidence Models
3. Experiments
4. Conclusion

Context: Big Data

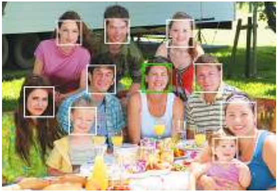
- Superabundance of data: images, videos, audio, text, user traces, *etc*

BBC: 2.4M videos

Social media,
e.g. Facebook: 1B each day

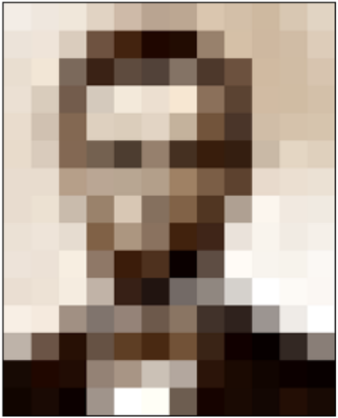
100M monitoring cameras

- Obvious need to access, search, or classify these data: **Recognition**
- Huge number of applications: mobile visual search, robotics, autonomous driving, augmented reality, medical imaging *etc*



Recognition of low-level signals: filling the semantic gap

- What we perceive *vs*
  What a computer sees



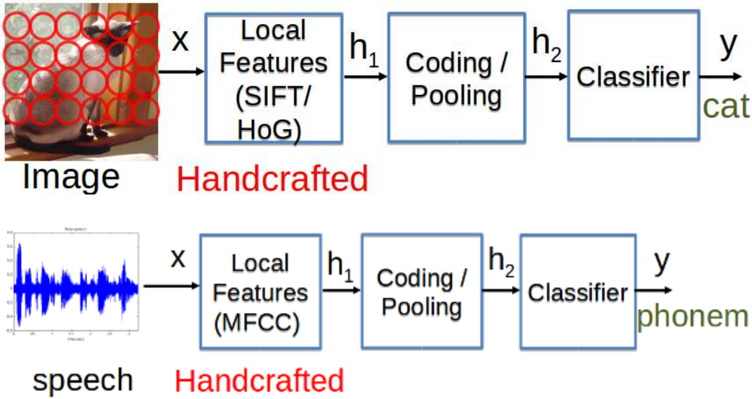
243	239	240	225	206	185	188	218	211	206	216	225
242	239	218	110	67	31	34	152	213	206	208	221
243	242	123	58	94	82	132	77	108	208	208	215
235	217	115	212	243	236	247	139	91	209	208	211
233	208	131	222	219	226	196	114	74	208	213	214
232	217	131	116	77	150	69	56	52	201	228	223
232	232	182	186	184	179	159	123	93	232	235	235
232	236	201	154	216	133	129	81	175	252	241	240
235	238	230	128	172	138	65	63	234	249	241	245
237	236	247	143	59	78	10	94	255	248	247	251
234	237	245	193	55	33	115	144	213	255	253	251
248	245	161	128	149	109	138	65	47	156	239	255
190	107	39	102	94	73	114	58	17	7	51	137
23	32	33	148	168	203	179	43	27	17	12	8
17	26	12	160	255	255	109	22	26	19	35	24

Recognition of low-level signals: input data variations



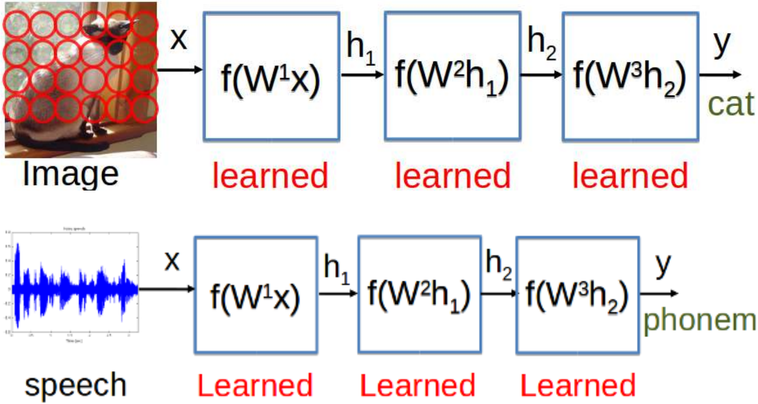

- Illumination variations
- View-point variations
- Deformable objects
- Intra-class variance

Deep Learning (DL) & Recognition of low-level signals



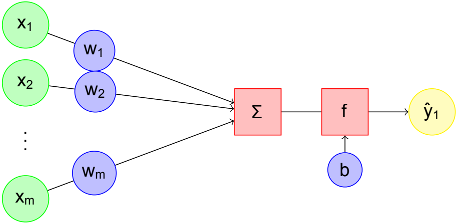
- Before DL: **handcrafted intermediate representations**
  - ⊖ Needs expertise in each field
  - ⊖ **Shallow archis**: low-level features

Deep Learning (DL) & Recognition of low-level signals



- **DL: learning intermediate representations**
  - $\oplus$ **Deep**: hierarchy, gradual learning
  - $\oplus$ Common learning methodology, no expertise

- **The formal Neuron**

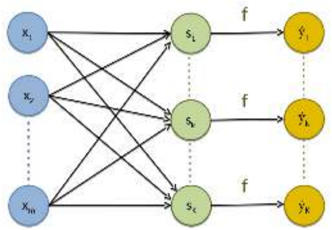


x_i: inputs
w_i, b: weights
f: activation function
y: output of the neuron

$$y = f(w^\top x + b)$$

Figure : The formal neuron – Credits: R. Herault

- **Neural Networks:** Stacking several formal neurons ⇒ **Perceptron**



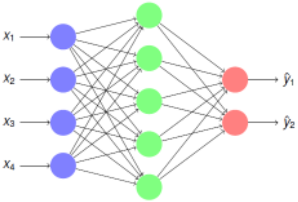
- **Soft-max Activation**:

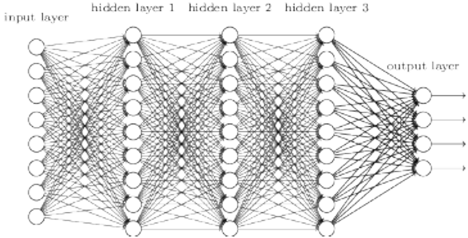
$$\hat{y_k} = f(s_k) = \frac{e^{s_k}}{\sum\limits_{k'=1}^{K} e^{s_{k'}}}$$

⇒ **Logistic Regression (LR) Model !**

Deep Neural Networks (DNN)

- Logistic Regression (LR): limited to linear decision boundaries
- **Multi-Layer Perceptron (MLP):** Stacking layers of neural networks
  - More complex and rich functions
  - Neural network with one single hidden layer ⇒ universal approximator [Cyb89]

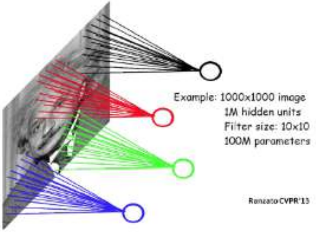


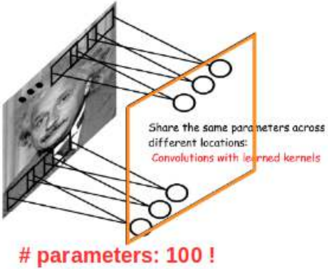


- **Basis of the "deep learning" field**
  - **Hidden layers: intermediate representations from data**
  - **Can be learned with Backpropagation algorithm [Lec85, RHW86]** (chain rule)

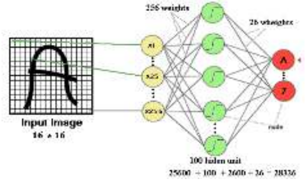
Convolutional Neural Networks (ConvNets)

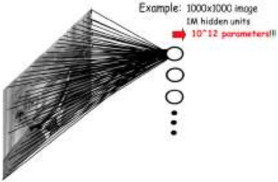
- **ConvNets:** sparse connectivity + shared weights





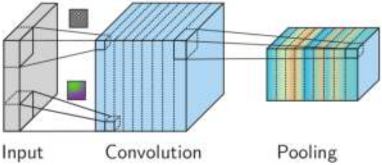
- Overcome parameter explosion for Fully Connected Networks on images
- Local feature extraction ($\neq$ FCN), equivariance



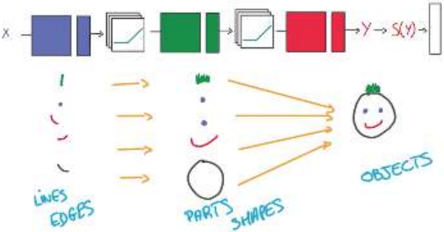


Convolutional Neural Networks (ConvNets)

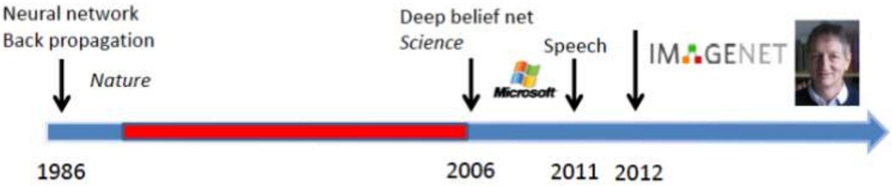
- Convolution on tensors, *i.e.* multidimensional arrays: T of size $W \times H \times D$
  - Convolution: $C[T] = T'$, T' tensor of size $W' \times H' \times K$
  - Each filter locally connected with shared weights (K number of filters)
- **An elementary block: Convolution + Non linearity (*e.g.* ReLU) + pooling**



- **Stacking several Blocks:** intuitive hierarchical information extraction

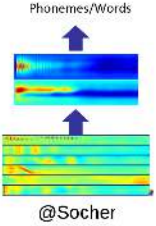


Deep Learning Succes since 2010



- 2011: Speech Recognition

Acoustic model	Recog \ WER	RT03S FSH	Hub5 SWB
Traditional features	1-pass −adapt	**27.4**	**23.6**
Deep Learning	1-pass −adapt	**18.5** (−33%)	**16.1** (−32%)



Deep Learning and ConvNet for Image Classification

- ImageNet ILSVRC Challenge (Stanford):
  - $1,200,000$ training images, $1,000$ classes, mono-label
  - Based on WordNet hierarchy (ontology)
  - Evaluation: top-5 error
- Up to 2012, leading approaches: BoW + SVM
- ILSVRC'12: the deep revolution $\Rightarrow$ outstanding success of ConvNets [KSH12]

Rank	Name	Error rate	Description
1	**U. Toronto**	0.15315	Deep learning
2	U. Tokyo	0.26172	Hand-crafted features and learning models. Bottleneck.
3	U. Oxford	0.26979	
4	Xerox/INRIA	0.27058	

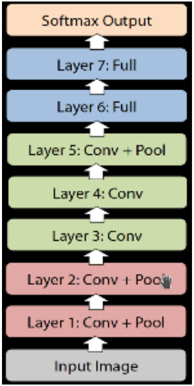
Deep ConvNet success at ILSVRC'12

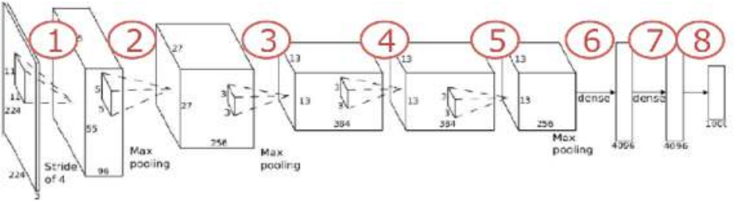
Two main practical reasons:

1. Huge number of labeled images (10^6 images)
   - Possible to train very large models without over-fitting
   - Larger models enables to learn rich (semantic) features hierarchies
2. GPU implementation for training
   - Relatively cheap and fast GPU
   - Training time reduced to 1-2 weeks (up to 50x speed up)

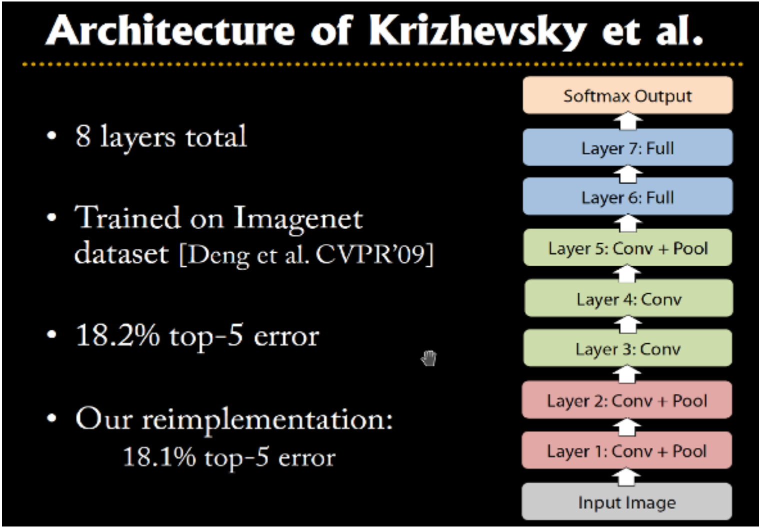
AlexNet [KSH12] in ILSVRC'12

- 60,000,000 parameters
- 650,000 neurons - 630,000,000 connections
- 5 convolutional layers, 3 Fully Connected (FC)
  - Convolution layer: Convolution + non linearity (ReLU) + pooling
  - Full= FC + non linearity - Final FC: 4096-dim
- Trained on 2 GPUs for a week



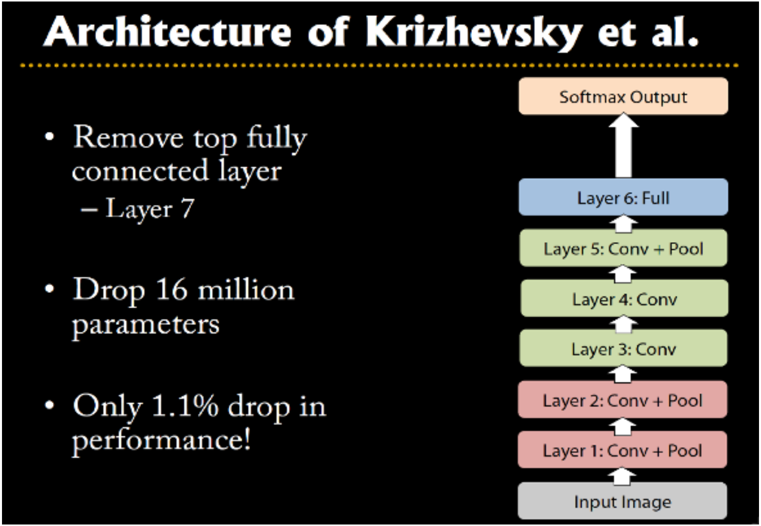


AlexNet [KSH12] in ILSVRC'12



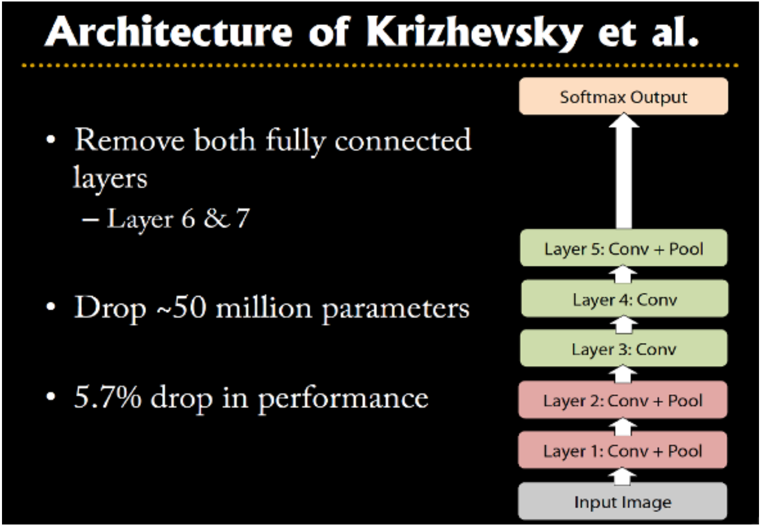
Credit: R. Fergus

AlexNet [KSH12] in ILSVRC'12



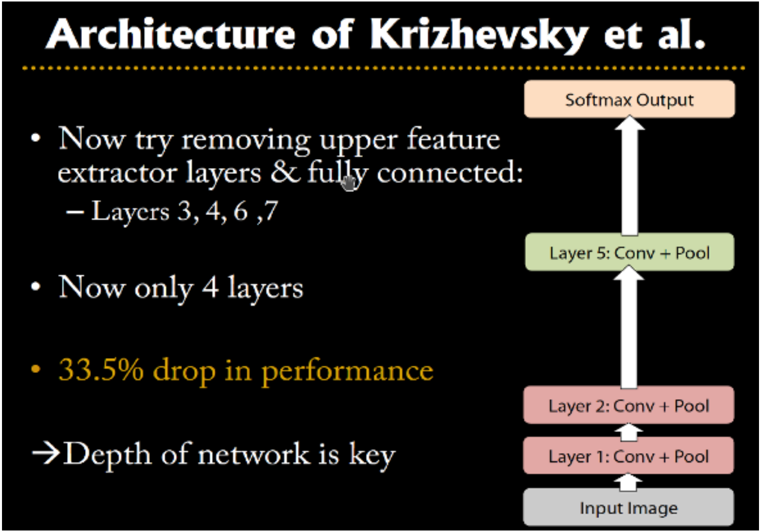
Credit: R. Fergus

AlexNet [KSH12] in ILSVRC'12



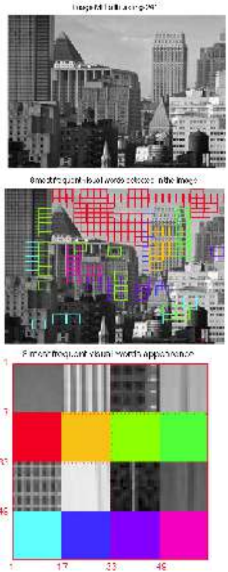
Credit: R. Fergus

AlexNet [KSH12] in ILSVRC'12



Credit: R. Fergus

Deep Learning in 2012: Representation Learning



Deep: more semantic features

Outline

ConvNet and invariance

- Standard ConvNets: limited invariance capacity (small shifts)
- ImageNet: single centered object $\neq$ other datasets (VOC, MS COCO)

$\Rightarrow$ **How to use deep architectures on complex scenes?**

ImageNet

VOC 2007

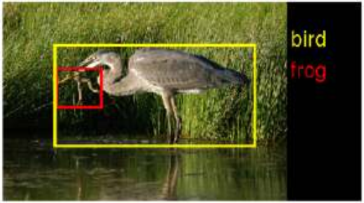
MS COCO

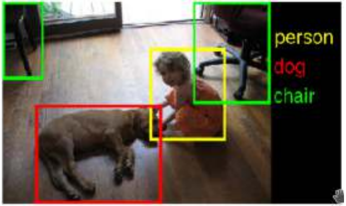
How to use deep architectures on complex scenes?

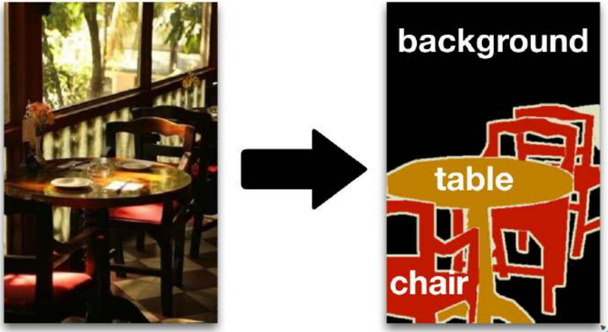
- Learning localized representation

How to use deep architectures on complex scenes?

- Using full (precise) annotation, *e.g.* BB or segmentation masks

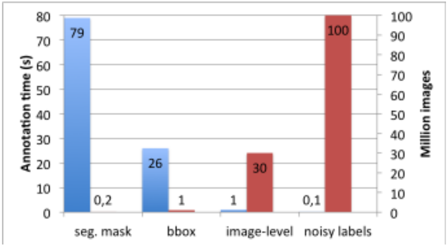


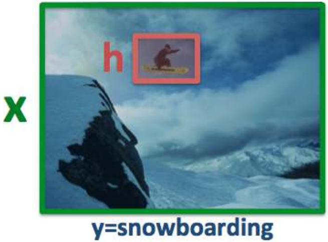




How to use deep architectures on complex scenes?

- Using full (precise) annotation, *e.g.* BB or segmentation masks
- **BUT:** full annotations expensive [BRFFF16]
  $\Rightarrow$ **training with weak supervision**



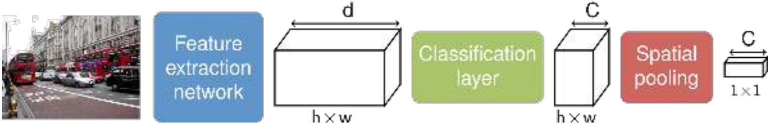


Variable	Notation	Space	Train	Test
Input	$\mathbf{x}$	$\mathcal{X}$	observed	observed
Output	$\mathbf{y}$	$\mathcal{Y}$	observed	unobserved
Latent	$\mathbf{h}$	$\mathcal{H}$	unobserved	unobserved

Outline

Deep Architecture for Weakly Supervised Learning

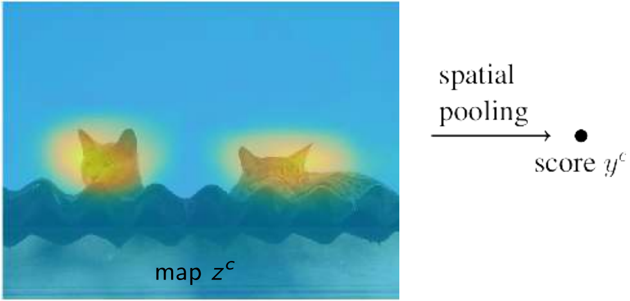
- Adapt deep architecture: **Pooling function** ⇒ global label from local predictions



- $h \times w \times C$ tensor: Class Activation Maps (CAM)

map z^c

How to pool?



Max [Oquab, CVPR15]

$$y^c = \max_{i,j} z_{ij}^c$$

Use 1 region

Average (GAP) [Zhou, CVPR16]

$$y^c = \frac{1}{N} \sum_{i,j} z_{ij}^c$$

Use all regions

Average pooling limitation

- Classifying with all regions
- Not efficient for small objects: lots of "noisy" regions

Max pooling

$$y^c = \max_{i,j} z_{ij}^c \quad (1)$$

- Classifying only with the max scoring region

- Loss of contextual information

Max pooling limitation

Max pooling

$$y^c = \max_{i,j} z^c_{ij} \tag{1}$$

- Classifying only with the max scoring region


- Loss of contextual information

max+min pooling

- **Contribution:** `max+min` **pooling function**

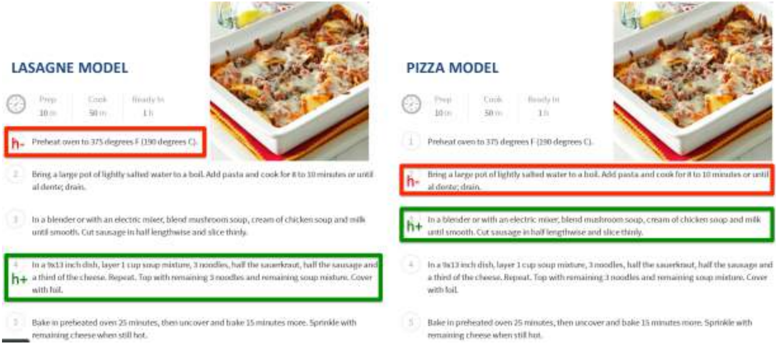
$$y^c = \max_{i,j} z_{ij}^c + \min_{i,j} z_{ij}^c \qquad (2)$$

- $\mathbf{h}^+$: presence of the class → high $\mathbf{h}^+$
- $\mathbf{h}^-$: localized evidence of the absence of class: **negative evidence**

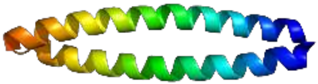
street image **x** | s(**street**) = 2 | s(**highway**) = 0.7

max+min pooling

- **Negative evidence** : OK pour **h** $\Leftrightarrow$ localization **x** (MIL) :
  - Text

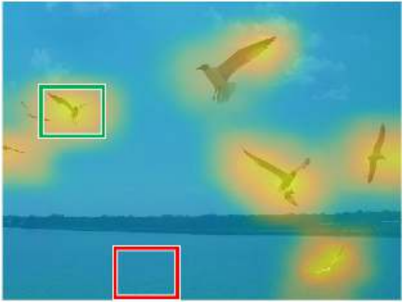


  - Molecule, *e.g.* **x** DNA, **h** DNA region, **y** chemical property
    - $\mathbf{h}^-$ inhibition region in DNA for the chemical property

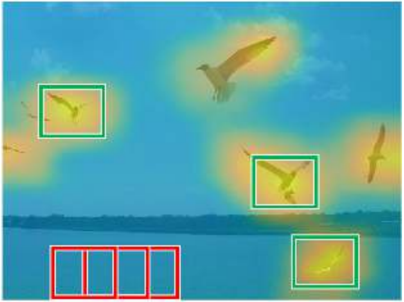


WELDON pooling

- Extension of `max+min` pooling
- Using several regions, more robust region selection



k=1



k=3

$$y^c = s^{top}_{k^+}(z^c) + s^{low}_{k^-}(z^c) \quad (3)$$

$$s^{top}_{k^+}(z^c) = \frac{1}{k^+}\sum_{i=1}^{k^+} i\text{-th-max}(z^c) \qquad s^{low}_{k^-}(z^c) = \frac{1}{k^-}\sum_{i=1}^{k^-} i\text{-th-min}(z^c)$$

WILDCAT pooling

- `max+min` pooling:
  - Both types of region are important
  - Complementary information
  - Not the same importance
- Pooling function

$$y^c = s_{k^+}^{top}(z^c) + \alpha \cdot s_{k^-}^{low}(z^c) \quad (4)$$

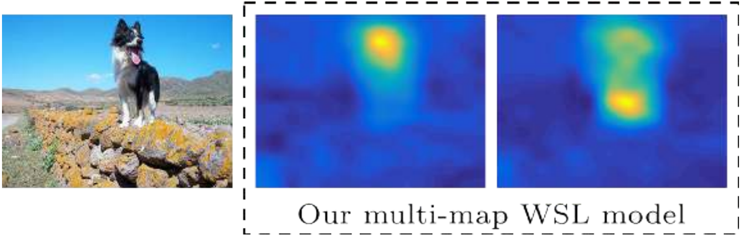
  - $\alpha \in [0, 1]$: trade off parameter

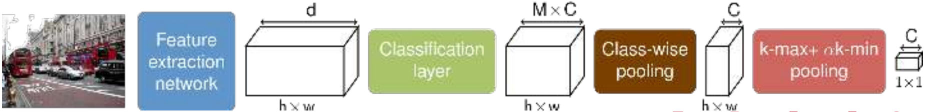
Pooling	k^+	k^-	α
`max`	1	0	0
GAP	n	0	0
`max+min`	1	1	1
WELDON	k	k	1

WILDCAT architecture

- WELDON: 1 model per class
  - Generalization to M models per class
  - Catch multiple class-related modalities

$$z_{ij}^{c} = \sum_{m=1}^{M} z_{ij}^{cm} \tag{5}$$

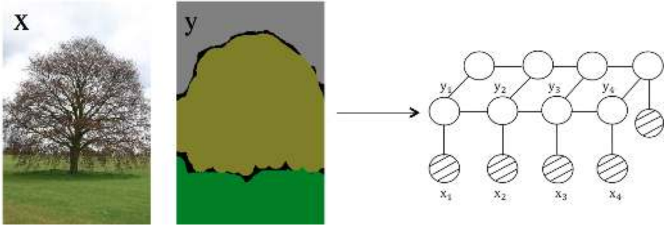




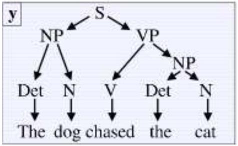
Outline

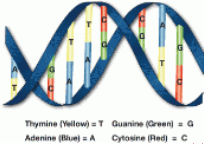
How to use deep architectures on complex scenes?

- **Structured Prediction:** use a structured loss on top of a deep ConvNet
- $\mathcal{X}$ arbitrary input space, $\mathcal{Y}$ discrete output space with **correlated variables** $\Rightarrow$ **probabilistic graphical models**
- Ex: semantic segmentation $\Rightarrow \mathcal{Y} = \{1, ..., k\}^D$



- Various applications: NLP (PoS tagging), sequences (*e.g.* ADN), *etc*





Structural SVM (SSVM) [TJHA05]

- $\Psi(\mathbf{x}, \mathbf{y}) \in \mathbb{R}^d$: relationship between input $\mathbf{x} \in \mathcal{X}$ and output $\mathbf{y} \in \mathcal{Y}$
- Scoring function linear in Ψ: $f_{\mathbf{w}}(\mathbf{x}, \mathbf{y}) = \langle w, \Psi(\mathbf{x}, \mathbf{y}) \rangle = s(\mathbf{y})$
- Prediction or **inference**: $\hat{y}(x, w) = \arg\max_{y \in \mathcal{Y}} s(\mathbf{y})$
  - Output space $\mathcal{Y}$ generally huge $\Rightarrow$ exhaustive maximization not tractable
  - Exploit structure (chain, tree), specific scoring functions (sub-modular), *etc*
- **Training:** a set of N labeled trained pairs $(\mathbf{x}_i, \mathbf{y}_i^*)$
  - Structured loss $\Delta(\hat{\mathbf{y}}_i, \mathbf{y}_i^*)$, $\hat{\mathbf{y}}_i(\mathbf{x}_i, \mathbf{w}) \Rightarrow$ *Prior* knowledge
  - Dependence of Δ wrt $\mathbf{w}$ complex (non-convex, non-smooth)
  - **Margin rescaling:** convex upper bound $\Delta(\hat{\mathbf{y}}_i, \mathbf{y}_i^*) \leq \ell(\mathbf{x}_i, \mathbf{y}_i^*, \mathbf{w})$

$$\ell(\mathbf{x}_i, \mathbf{y}_i^*, \mathbf{w}) = \max_{\mathbf{y} \in \mathcal{Y}} \left[\Delta(\mathbf{y}_i^*, \mathbf{y}) + s(\mathbf{y})\right] - s(\mathbf{y_i})$$

  - $\tilde{\mathbf{y}}_i = \underset{\mathbf{y} \in \mathcal{Y}}{arg\ max} \left[\Delta(\mathbf{y}_i^*, \mathbf{y}) + \langle \mathbf{w}, \Psi(\mathbf{x}_i, \mathbf{y}) \rangle\right]$ "Loss Augmented Inference" (LAI)
    - For computing $\frac{\partial \ell}{\partial \mathbf{w}} = \Psi(\mathbf{x_i}, \tilde{\mathbf{y}}_\mathbf{i}) - \Psi(\mathbf{x_i}, \mathbf{y_i^*})$: generally harder than inference

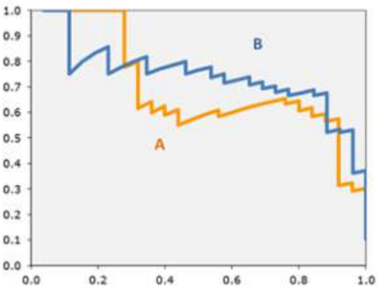
Structured Output Ranking

- Input $\mathbf{x} \in \mathcal{X}$ list of n examples: $\mathbf{x} = (d_1, \ldots d_n)$, $\phi(\boldsymbol{d_i}) \in \mathbb{R}^d$
- Structured output $\mathbf{y} \in \mathcal{Y}$: ranking of example, represented by matrix $\mathbf{y}$ s.t.

$$y_{ij} = \begin{cases} +1 & \text{if } d_i \prec_y d_j \ (d_i \text{ is before } d_j \text{ in the sorted list}) \\ -1 & \text{if } d_i \succ_y d_j \ (d_i \text{ is after } d_j \end{cases}$$

- Ranking feature map: $\Psi(\mathbf{x}, \mathbf{y}) = \frac{1}{N_+ \cdot N_-} \sum_{\boldsymbol{d_i} \in \oplus} \sum_{\boldsymbol{d_j} \in \ominus} y_{ij} \left[\phi(\boldsymbol{d_i}) - \phi(\boldsymbol{d_j})\right]$, $y^*_{ij} = 1 \ \forall (i,j)$
- **Inference** ($|\mathcal{Y}| \sim 2^{n^2/2}$): exact by sorting example wrt $\langle \mathbf{w}; \phi(\boldsymbol{d_i}) \rangle$ [YFRJ07]
- **Training:** LAI with Average Precision (AP) loss: $\Delta_{AP}(y_i, y) = 1 - AP(y)$

Precision-recall curves - examples

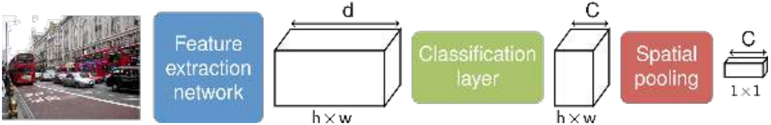


- AP: Precision $= \frac{TP}{|\hat{P}|}$ *vs* Recall $= \frac{TP}{N_+}$
- Δ_{AP}: no linear decomposition wrt examples $\neq$ AUC ROC (TPR *vs* FPR)
  - Optimal greedy algorithm in $O(N_+ N_-)$ [YFRJ07], speed-up in [MJK14]

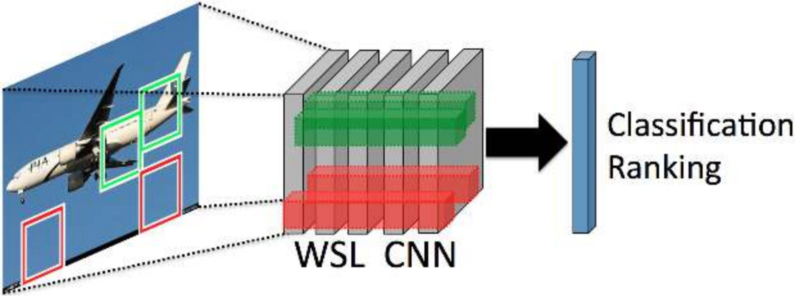
Structured prediction with latent variables

- **Latent Structural SVM (LSSVM)** [YJ09]
  - **Prediction:** $s(\mathbf{y}) = \max_{\mathbf{h}\in\mathcal{H}} \langle \mathbf{w}, \Psi(\mathbf{x}_i, \mathbf{y}, \mathbf{h})\rangle \Rightarrow \hat{\mathbf{y}} = \arg\max_{\mathbf{y}\in\mathcal{Y}} s(\mathbf{y})$
  - **LAI for training:** $\max_{(\mathbf{y},\mathbf{h})\in\mathcal{Y}\times\mathcal{H}} \left[\Delta(\mathbf{y}_i^*, \mathbf{y}) + \langle \mathbf{w}, \Psi(\mathbf{x}, \mathbf{y}, \mathbf{h})\rangle\right]$
    - **Structured AP ranking:** no exact solution LSSVM
      $\Rightarrow$ Approximate solution in [BMJK15]
- **Negative Evidence Models**
  - **MANTRA Prediction:** $s(\mathbf{y}) = \max_{\mathbf{h}\in\mathcal{H}} \langle \mathbf{w}, \Psi(\mathbf{x}_i, \mathbf{y}, \mathbf{h})\rangle + \min_{\mathbf{h}\in\mathcal{H}} \langle \mathbf{w}, \Psi(\mathbf{x}_i, \mathbf{y}, \mathbf{h})\rangle$
    - **WELDON:** k-`max`+k-`min`
  - **LAI for training:** $\max_{\mathbf{y}\in\mathcal{Y}} \left[\Delta(\mathbf{y}_i^*, \mathbf{y}) + s(\mathbf{y})\right]$
    - ▷ **Structured AP ranking: exact solution!**
    - ▷ **Symmetrization due to the (k-)`max`+(k-)`min` scoring**
    - ▷ **Decoupling optimization over y and h, ≠ [YJ09, BMJK15]**

WSL Ranking with Deep Negative Evidence Models



- $\Psi(\mathbf{x}_i, \mathbf{y}, \mathbf{h})$: feature representation for a given image region
- $s(\mathbf{y}) = \max_{\mathbf{h}\in\mathcal{H}} \langle \mathbf{w}, \Psi(\mathbf{x}_i, \mathbf{y}, \mathbf{h})\rangle + \min_{\mathbf{h}\in\mathcal{H}} \langle \mathbf{w}, \Psi(\mathbf{x}_i, \mathbf{y}, \mathbf{h})\rangle$: score for a given output
  - **WELDON:** k-max+k-min
- Learning $\Psi(\mathbf{x}_i, \mathbf{y}, \mathbf{h})$ with deep ConvNet and AP loss: end-to-end training!
  - Incorporating multiple positive & negative evidence



Outline

Experimental Setup

Dataset	#Train	#Test	#Classes	Evaluation
VOC 07	5,011	4,952	20	MAP
VOC 12	11,540	10,991	20	MAP
VOC 12 Action	2,296	2,292	10	MAP
MS COCO	82,783	40,504	80	MAP
MIT67	5,360	1,340	67	accuracy
CUB-200	5,994	5,794	200	accuracy
ILSVRC 2012	1,281,167	50,000	1000	accuracy

- Feature extraction network: ResNet-101 pretrained on ImageNet

Classification Results

Method	VOC 2007	VOC 2012	MS COCO
ResNet-101	89.8	89.2	72.5
Deep MIL	-	86.3	62.8
ProNet	-	89.3	70.9
SPLeaP	88.0	-	-
WILDCAT	**95.0**	**93.4**	**80.7**

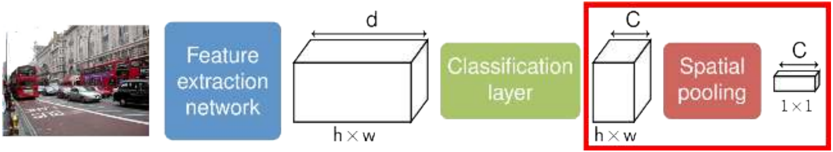
ImageNet	Top-5 error
ResNet-101 (1 crop)	6.21
ResNet-200 (10 crops)	4.93
ResNeXt-101 (1 crop)	4.4
Inception-ResNet-v2 (12 crops)	**4.1**
WILDCAT **(**$M = 1$**)**	4.23

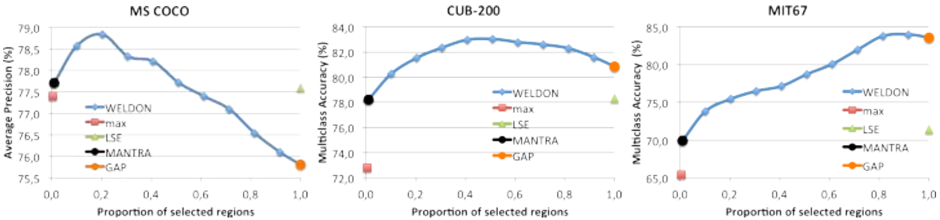
AP Ranking Results

Dataset	VOC07	VOCAct	MS COCO
`max` + classif. loss	86.8	71.8	77.4
`max` + AP loss (LAPSVM [BMJK15])	87.9	73.3	77.9
`max+min` + classif. loss	89.9	78.5	77.7
`max+min` + AP loss	**91.2**	**80.7**	**78.7**

- Optimizing the evaluation metric during training is important

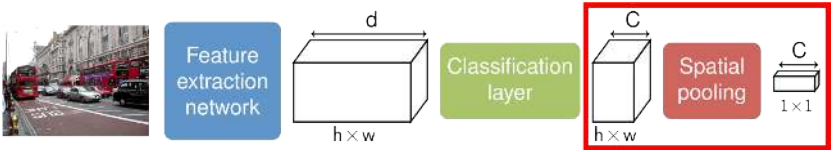
Pooling analysis

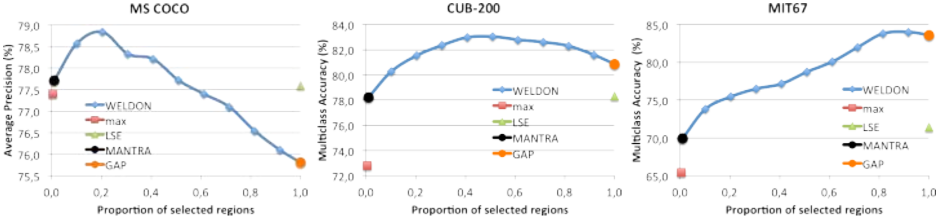




- max / LSSVM
- max+min / MANTRA
- k-max+k-min / WELDON
- average / GAP
- soft-max / LSE / HCRF

Pooling analysis



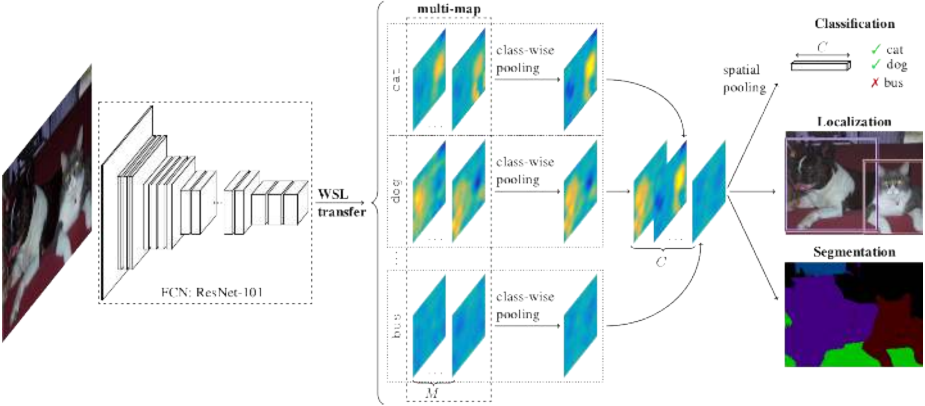


Unified pooling function

$$s_{\mathbf{w}}^{(\alpha,\beta_h^+,\beta_h^-)}(\mathbf{x},\mathbf{y}) = \frac{1}{2\beta_h^+}\log\left(\frac{1}{|\mathcal{H}|}\sum_{\mathbf{h}\in\mathcal{H}}\exp[\beta_h^+\langle\mathbf{w},\Psi(\mathbf{x}_i,\mathbf{y},\mathbf{h})\rangle]\right)$$

$$+\,\alpha\frac{1}{2\beta_h^-}\log\left(\frac{1}{|\mathcal{H}|}\sum_{\mathbf{h}\in\mathcal{H}}\exp[\beta_h^-\langle\mathbf{w},\Psi(\mathbf{x}_i,\mathbf{y},\mathbf{h})\rangle]\right)$$

Weakly Supervised Experiments

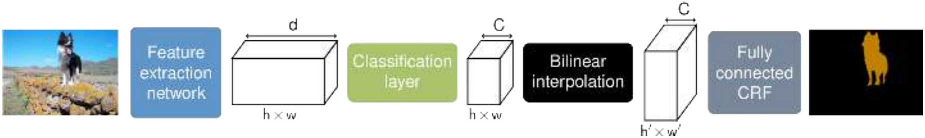


Weakly supervised localization

Method	VOC 2012	MS COCO
Deep MIL [Oquab, CVPR15]	74.5	41.2
ProNet [Sun, CVPR16]	77.7	46.4
WSLocalization [Bency, ECCV16]	79.7	49.2
WILDCAT	**82.9**	**53.4**

- Pointwise metric [Oquab, CVPR15]

- Test architecture



Method	Mean IoU
MIL-FCN	24.9
MIL-Base+ILP+SP-sppxl	36.6
EM-Adapt + FC-CRF	33.8
CCNN + FC-CRF	35.3
WILDCAT + FC-CRF	**43.7**

Weakly supervised segmentation

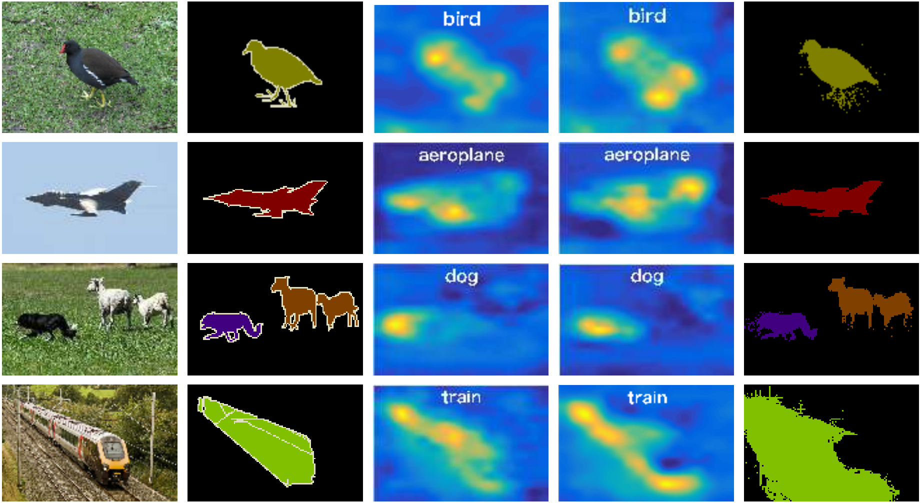


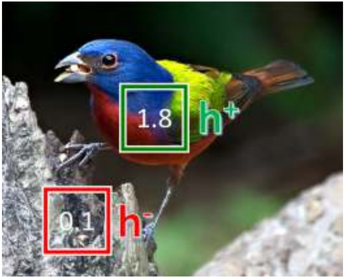
image | ground truth | heatmap1 | heatmap2 | prediction

Outline

Negative Evidence Models: Conclusion

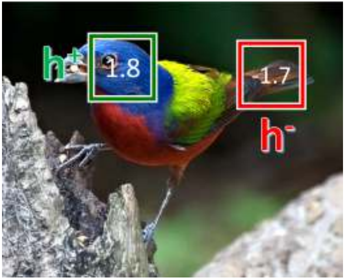
- Local evidence of class absence
- State-of-the-art for many image classification datasets
- Applicable for weakly supervised localization & segmentation
- Application on different type of data: image, text, molecule
- **Structured output prediction:** AP ranking

true class



painted bunting

wrong class



indigo bunting

Resources

[1] Thibaut Durand, Nicolas Thome, and Matthieu Cord
MANTRA: Minimum Maximum Latent Structural SVM for Image Classification and Ranking.
In *IEEE International Conference on Computer Vision (ICCV)*, 2015.

[2] Thibaut Durand, Nicolas Thome, and Matthieu Cord
WELDON: Weakly Supervised Learning of Deep ConvNets.
In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2016.

[3] Thibaut Durand*, Taylor Mordan*, Nicolas Thome, and Matthieu Cord
WILDCAT: Weakly Supervised Learning of Deep ConvNets for Image Classification, Pointwise Localization and Segmentation.
In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2017.

[4] Thibaut Durand, Nicolas Thome, and Matthieu Cord
Exploiting Negative Evidence for Deep Latent Structured Models.
In *IEEE Transactions on Pattern Analysis and Machine Intelligence (T-PAMI)*, 2018.

Code available on GitHub:

- MANTRA: `https://github.com/durandtibo/mantra-python`
- WELDON: `https://github.com/durandtibo/wsl.resnet.torch`
- WILDCAT: `https://github.com/durandtibo/wildcat.pytorch`

Thank you for your attention !

Thibaut Durand

Nicolas Thome

Matthieu Cord

- Cnam Paris - CEDRIC Lab / MSDMA Team
- Sorbonne Université Associate member - LIP6 Lab / MLIA Team (P. Gallinari)

Questions ?

References I

[BMJK15] Aseem Behl, Pritish Mohapatra, C. V. Jawahar, and M. Pawan Kumar, *Optimizing average precision using weakly supervised data*, IEEE Trans. Pattern Anal. Mach. Intell. 37 (2015), no. 12, 2545–2557.

[BRFFF16] Bearman, Russakovsky, Ferrari, and Fei-Fei, *What's the Point: Semantic Segmentation with Point Supervision*, ECCV (2016).

[Cyb89] George Cybenko, *Approximation by superpositions of a sigmoidal function*, Mathematics of control, signals and systems 2 (1989), no. 4, 303–314.

[KSH12] Alex Krizhevsky, Ilya Sutskever, and Geoffrey E Hinton, *Imagenet classification with deep convolutional neural networks*, Advances in neural information processing systems, 2012, pp. 1097–1105.

[Lec85] Yann Lecun, *Une procedure d'apprentissage pour reseau a seuil asymmetrique (a learning scheme for asymmetric threshold networks)*, pp. 599–604, 1985.

[MJK14] Pritish Mohapatra, C.V. Jawahar, and M. Pawan Kumar, *Efficient optimization for average precision svm*, NIPS, 2014.

[RHW86] D.E. Rumelhart, G.E. Hinton, and R.J. Williams, *Learning representations by back-propagating errors*, Nature 323 (1986), 533–536.

[TJHA05] Ioannis Tsochantaridis, Thorsten Joachims, Thomas Hofmann, and Yasemin Altun, *Large margin methods for structured and interdependent output variables*, Journal of Machine Learning Research, 2005, pp. 1453–1484.

[YFRJ07] Yisong Yue, Thomas Finley, Filip Radlinski, and Thorsten Joachims, *A support vector method for optimizing average precision*, Proceedings of the 30th annual international ACM SIGIR conference on Research and development in information retrieval, ACM, 2007, pp. 271–278.

[YJ09] Chun-Nam Yu and T. Joachims, *Learning structural svms with latent variables*, ICML, 2009.