ADIOS: Architectures Deep In Output Space

Moustapha Cissé, Maruan Al-Shedivat, Samy Bengio
Facebook AI Research, Carnegie Mellon University, Google Brain

1. Introduction

Multi-label classification is a generalization of binary classification where the task consists in predicting *sets of labels*. With the availability of ever larger datasets, as expected, deep learning approaches are now yielding state-of-the-art performance for this class of problems. Unfortunately, they usually do not take into account the often unknown but nevertheless rich relationships between labels. We propose:

- ▷ Partitioning of the labels into a Markov blanket chain.
- ▷ An architecture deep in the output space and aware of the statistical relationships between the labels.

2. Problem Formulation & Previous Work

- ▷ Multilabel Classification: Learn a classifier $\mathbf{h} \in \mathcal{H}$ to predict a subset of relevant labels ($\mathbf{y} \in 2^{\mathcal{Y}}$) given $X \in \mathcal{X}$, $\mathbf{h} : \mathcal{X} \to 2^{\mathcal{Y}}$.
- ▷ Multitasking NN (Nam et al., 2014) outperforms most previous approaches (Structured SVMs, Chaining, etc.).

$$\mathcal{L}_\theta^{CE}(x, \mathbf{y}) = -\sum_{i=1}^{m} y_i \cdot \log(h(x)) + (1 - y_i) \cdot \log(1 - h(x))$$

Labels assumed independent: dependencies are only captured implicitly, through a joint learning process.

3. Dependence Between Labels

Labels carry rich semantic structure and dependencies:
entailment, exclusion, positive/negative correlation, etc.

$\mathbf{y} = \{TomBrady, Football\}$

$\mathbf{y} = \{Obama, WhiteHouse\}$

Leverage the dependencies between the labels to:

1. Reduce complexity: Label Embedding Trees (Bengio et al., 2010)
2. Improve predictive performance: HEX Model (Deng et al., 2014).

4. Labels vs. Features

What if we use some of the labels as features to predict other labels? How this compares when a model (MLP) is trained on the original features:

Dataset	$L.C.$	G_2 size	maF1 on G_2		P@5 on G_2	
			X	G_1	X	G_1
Delicious	19	500	21.54	**38.70**	**55.61**	47.44
MediaMill	4.5	50	11.49	**22.57**	**45.48**	17.36
NUS-WIDE	2.4	40	**11.93**	10.64	12.83	**25.36**

IDEA: ▷ Find a good partitioning $\mathcal{Y} = (G_1, G_2)$ of the labels.
▷ Learn (jointly) to predict G_1 from X and G_2 from (G_1, X).

5. Markov Blanket Chained Partitioning

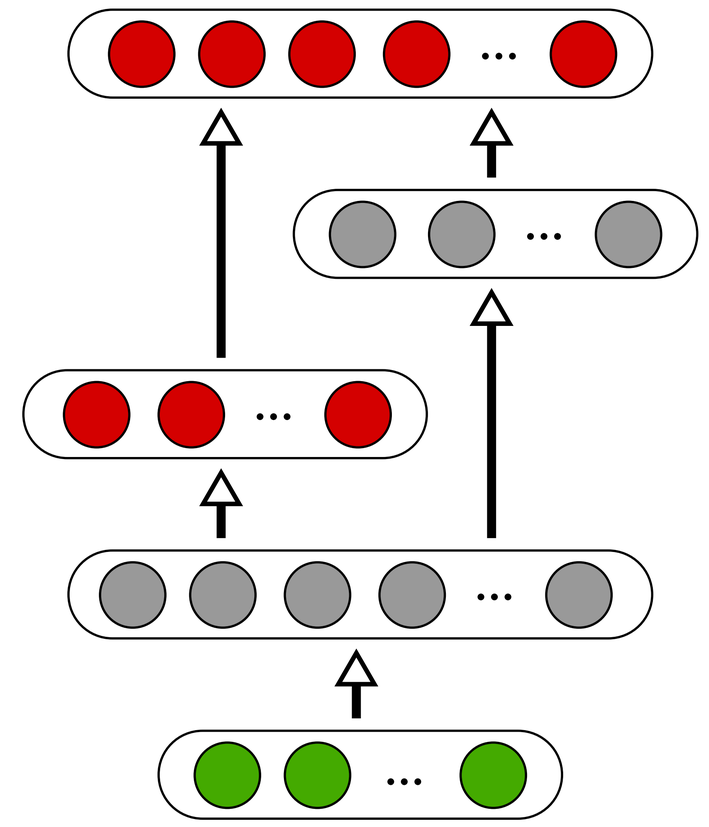
Find a partition $\mathcal{Y} = (G_1, G_2)$ $(G_1 \cap G_2 = \emptyset)$ such that labels in G_1 are predictive of G_2: reduction of uncertainty in G_2 given G_1
$\Rightarrow$ Information Gain as the criterion:

$$\underset{G_1 \subset \mathcal{Y}, |G_1| \leq k}{\arg\max} \; I(G_2; G_1) = H(G_2) - H(G_2|G_1)$$

- ▷ Theory says this is generally NP-hard.
- ▷ If labels in G_2 are independent given G_1 $I(\cdot)$ is *submodular & monotone*
  $\Rightarrow$ near optimal solution by greedy algorithm (Krause & Guestrin, 2012).

6. ADIOS Architectures

We propose the following architectures:



7. Results

The following are performances of the baselines and ADIOS models on the large BioASQ dataset:

Model	maF1	miF1	P@1	P@5	P@10
BR	0.03	14.69	13.74	13.49	6.78
PLST	0.03	14.69	13.80	13.48	6.77
MLCS	0.03	14.73	11.98	13.95	7.01
FastXML	0.19	3.80	19.55	13.68	9.20
MLP	14.86	42.40	**67.46**	41.32	27.52
ADIOS$_{\text{RND}}$	15.91	43.11	66.97	40.96	27.46
ADIOS$_{\text{MBC}}$	**16.14**	**43.49**	67.28	**41.77**	**27.98**

For the results on three other datasets, please refer to our paper.

8. Small Data Scenario

ADIOS improves performance in small data scenario: