Deep Learning with a Rethinking Structure for Multi-label Classification

Yao-Yuan Yang and Yi-An Lin and
Hong-Min Chu and Hsuan-Tien Lin

Department of Computer Science & Information Engineering
National Taiwan University

November 14, 2018

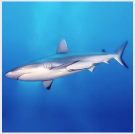
Which animal?

dog

rabbit

cat

guinea pig



shark

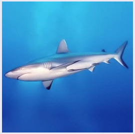
Which animals?

dog
1

rabbit
1

cat
1

guinea pig
1



shark
0

Multi-label Classification (MLC)

Problem definition

- train set $\mathcal{D} = \{(\mathbf{x}^{(n)}, \mathbf{y}^{(n)})\}_{n=1}^{N}$
- label vector $\mathbf{y} \in \{0,1\}^K$ as a binary vector
  - $\mathbf{y}[k] = 1$ if and only if the k-th label is relevant to $\mathbf{x}$
- learn a f that maps $\mathbf{x}$ to $\mathbf{y}$
- test data $(\mathbf{x}, \mathbf{y})$, prediction $\hat{\mathbf{y}} = f(\mathbf{x})$
- goal is to make $\hat{\mathbf{y}}$ close to ground truth $\mathbf{y}$

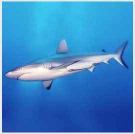
Evaluation

- cost function $C(\mathbf{y}, \hat{\mathbf{y}})$: the cost of predicting $\mathbf{y}$ as $\hat{\mathbf{y}}$
- F1 score, Accuracy score, Hamming loss, Rank loss

Baseline 1: Binary Relevance (BR)

Binary Relevance (BR)

- train a dog, rabbit, cat, guinea pig, shark classifier independently

$f_1(\mathbf{x})$	$f_2(\mathbf{x})$	$f_3(\mathbf{x})$	$f_4(\mathbf{x})$	$f_5(\mathbf{x})$
				
dog	rabbit	cat	guinea pig	shark

Label Correlation

- If we predict guinea pig as 1, we will also have the information about the shark should not be here
- BR ignores the label correlation information

- Learn each classifiers dependently (sequence prediction)
- Memory between labels

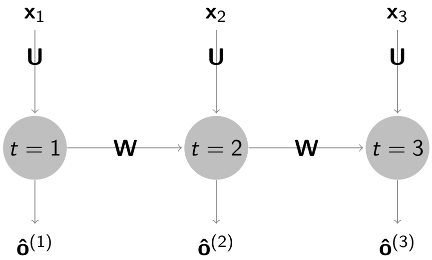
	BR	CC
dog	$f_1(\mathbf{x})$	$f_1(\mathbf{x})$
rabbit	$f_2(\mathbf{x})$	$f_2([f_1(\mathbf{x}); \mathbf{x}])$
cat	$f_3(\mathbf{x})$	$f_3([f_1(\mathbf{x}); f_2(\mathbf{x}); \mathbf{x}])$
guinea pig	$f_4(\mathbf{x})$	$f_4([f_1(\mathbf{x}); f_2(\mathbf{x}); f_3(\mathbf{x}); \mathbf{x}])$
shark	$f_5(\mathbf{x})$	$f_5([f_1(\mathbf{x}); f_2(\mathbf{x}); f_3(\mathbf{x}); f_4(\mathbf{x}); \mathbf{x}])$

Extension with neural network

- Use recurrent neural network to extend CC
- Able to attach other deep learning architecture for specific data feature extraction [WYM+16], [CCYW17]

Recurrent Neural Network (RNN)

- Solves sequence prediction problem
- Maps $\{\mathbf{x}^{(1)} \dots \mathbf{x}^{(T)}\}$ to $\{\mathbf{o}^{(1)} \dots \mathbf{o}^{(T)}\}$ $(\mathbf{x} \in \mathcal{R}^d, \mathbf{o} \in \mathcal{R}^K)$
- Applied in MLC, $T = K$ (# labels), $\mathbf{o}^{(i)} = \mathbf{y}[i]$, $\mathbf{x}^{(1)} \dots \mathbf{x}^{(T)} = \mathbf{x}$
- Feature transformation $\mathbf{U}$, Memory transformation $\mathbf{W}$



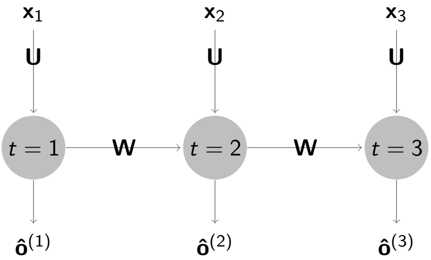
Revisit

- Problem with CC: label ordering
- Different ways of searching for optimal ordering [LT15]
- Imbalance information received for each classifier in the chain

Instead of treating the prediction of labels as a sequence, treat the process of gradually understand the problem as a sequence

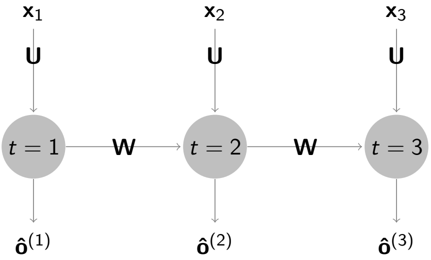
RethinkNet

- The output $\mathbf{o}^{(i)} \in \{0,1\}^K$ instead of $\mathbf{o}^{(i)} \in \{0,1\}$



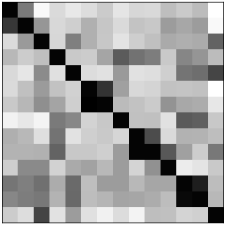
RethinkNet (modeling label correlation)

- RethinkNet uses memory to model label correlation
- $\mathbf{W} \in R^{K\times K}$ and $\mathbf{U} \in R^{K\times d}$
- $\hat{\mathbf{y}}^{(t)} = \hat{\mathbf{o}}^{(t)} = \sigma(\mathbf{U}\mathbf{x} + \mathbf{W}\hat{\mathbf{o}}^{(t-1)})$.
- feature term: $\mathbf{U}\mathbf{x}$, memory term: $\mathbf{W}\hat{\mathbf{o}}^{(t-1)}$
- $\hat{\mathbf{o}}^{(t)}[j] = \sum_{i=1}^{K} \hat{\mathbf{o}}^{(t-1)}[i] * \mathbf{W}[i,j]$
- $\mathbf{W}[i,j]$ represents the correlation between i-th and j-th label

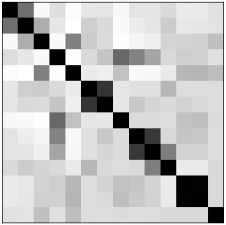


RethinkNet

- We plot the memory transform **W** and the correlation coefficient of the yeast data set.



memory transform



correlation coefficient

1. person, cup, fork, bowl, chair, table
2. person, bottle, cup, fork, knife, spoon, bowl, chair, table
3. person, cup, fork, knife, spoon, bowl, chair, table

Experiment

- Compare with 11 MLC data sets
- RethinkNet versus the other algorithms based on t-test at 95% confidence level (#win/#tie/#loss)

	PCC	CFT	CC-DP	CC	CC-RNN	BR
hamming (↓)	6/1/4	3/4/4	5/2/1	6/1/4	8/3/0	3/6/2
rank loss (↓)	5/1/5	5/2/4	7/1/0	10/1/0	10/1/0	10/1/0
f1 (↑)	6/2/3	5/4/2	5/2/1	8/3/0	10/1/0	9/2/0
acc (↑)	7/1/3	5/4/2	5/1/2	7/4/0	9/2/0	9/2/0
total	**24**/5/15	**18**/14/12	**22**/6/4	**31**/9/4	**37**/7/0	**31**/11/2

Experiment (MSCOCO)

- Compare with other deep learning MLC models on large scale image data (MSCOCO)
- Pre-trained Resnet-50 for feature extraction
- Baseline is logistic regression with Resnet's feature
- Baseline model is not able to be fine-tuned with the pre-trained model

	baseline	CNN-RNN	Att-RNN	RethinkNet
hamming (↓)	0.0279	0.0267	0.0270	**0.0234**
rank loss (↓)	60.4092	56.6088	43.5248	**35.2552**
f1 (↑)	0.5374	0.5759	0.6309	**0.6622**
acc (↑)	0.4469	0.4912	0.5248	**0.5724**

Conclusion

- Taken a memory view of the original chain based algorithms
- Developed a novel multi-label classification algorithm (RethinkNet)
  - Instead of treating labels as a sequence, treat MLC problem as a sequence
  - Models the of label correlation without label ordering problem
- Empirical study demonstrates on both classic data sets as well as image data sets with CNN to extract feature, we were able to outperform existing algorithms

Thank you for listening. Any question?

References

- Shang-Fu Chen, Yi-Chen Chen, Chih-Kuan Yeh, and Yu-Chiang Frank Wang, *Order-free RNN with visual attention for multi-label classification*, arXiv preprint arXiv:1707.05495 (2017).
- Weiwei Liu and Ivor Tsang, *On the optimality of classifier chain for multi-label classification*, NIPS, 2015.
- Jiang Wang, Yi Yang, Junhua Mao, Zhiheng Huang, Chang Huang, and Wei Xu, *Cnn-rnn: A unified framework for multi-label image classification*, CVPR, 2016.