

# A Deep Reinforced Sequence-to-Set Model for Multi-Label Classification

Pengcheng Yang<sup>1,2</sup>, Fuli Luo<sup>2</sup>, Shuming Ma<sup>2</sup>, Junyang Lin<sup>2</sup>, Xu Sun<sup>1,2</sup>

<sup>1</sup>Deep Learning Lab, Beijing Institute of Big Data Research, Peking University

<sup>2</sup>MOE Key Lab of Computational Linguistics, School of EECS, Peking University  
{yang\_pc, luofuli, shumingma, linjunyang, xusun}@pku.edu.cn

## Abstract

Multi-label classification (MLC) aims to predict a set of labels for a given instance. Based on a pre-defined label order, the sequence-to-sequence (Seq2Seq) model trained via maximum likelihood estimation method has been successfully applied to the MLC task and shows powerful ability to capture high-order correlations between labels. However, the output labels are essentially an unordered set rather than an ordered sequence. This inconsistency tends to result in some intractable problems, e.g., sensitivity to the label order. To remedy this, we propose a simple but effective sequence-to-set model. The proposed model is trained via reinforcement learning, where reward feedback is designed to be independent of the label order. In this way, we can reduce the dependence of the model on the label order, as well as capture high-order correlations between labels. Extensive experiments show that our approach can substantially outperform competitive baselines, as well as effectively reduce the sensitivity to the label order.<sup>1</sup>

## 1 Introduction

Multi-label classification (MLC) aims to assign multiple labels to each sample. It can be applied in many real-world scenarios, such as text categorization (Schapire and Singer, 2000) and information retrieval (Gopal and Yang, 2010). Due to the complex dependency between labels, a key challenge for the MLC task is how to effectively capture high-order correlations between labels (Zhang and Zhou, 2014).

When involving in capturing high-order correlations between labels, one line of research focuses on exploring the hierarchical structure of the label space (Prabhu and Varma, 2014; Jernite et al., 2017; Peng et al., 2018; Singh et al., 2018), while

<sup>1</sup>The code is available at <https://github.com/lancopku/Seq2Set>

another line strives to extend specific learning algorithms (Zhang and Zhou, 2006; Baker and Korhonen, 2017; Liu et al., 2017). However, most of these work tends to result in intractable computational costs (Chen et al., 2017).

Recently, based on a pre-defined label order, Nam et al. (2017); Yang et al. (2018) succeeded in applying the sequence-to-sequence (Seq2Seq) model to the MLC task, which shows its powerful ability to capture high-order label correlations and achieves excellent performance. However, the Seq2Seq model suffers from some thorny flaws on the MLC task. The output labels are essentially an unordered set with *swapping-invariance*<sup>2</sup>, rather than an ordered sequence. This inconsistency usually leads to some intractable problems, e.g., sensitivity to the label order. Previous work (Vinyals et al., 2016) has shown that the order has a great impact on the performance of the Seq2Seq model. Therefore, the performance of classifier is sensitive to the pre-defined label order. Besides, even if the model accurately predicts all true labels, it still may result in an unreasonable training loss due to the inconsistent order with the pre-defined label sequence<sup>3</sup>.

Therefore, in this work, we propose a simple but effective sequence-to-set model, which aims at alleviating the dependence of the model on the label order. Instead of maximizing the log-likelihood of pre-defined label sequences, we apply reinforcement learning (RL) (Sutton et al., 1999) to guide the model training. The designed reward not only comprehensively evaluates the quality of the output labels, but also satisfies *swapping-invariance* of the set, which leads to a reduction in the dependence of the model on the label order.

<sup>2</sup>*Swapping-invariance* means that swapping any two elements in the set will make no difference.

<sup>3</sup>For example, for the pre-defined label sequence [A, B, C], the training loss will be large if model generates [C, A, B].

The main contributions of this paper are summarized as follows:

- We propose a simple but effective sequence-to-set (**Seq2Set**) model based on reinforcement learning, which not only captures the correlations between labels, but also alleviates the dependence on the label order.
- Experimental results show that our Seq2Set model can outperform baselines by a large margin. Further analysis demonstrates that our approach can effectively reduce the sensitivity of the model to the label order.

## 2 Methodology

### 2.1 Overview

Here we define some necessary notations and describe the MLC task. Given a text sequence  $\mathbf{x}$  containing  $m$  words, the MLC task aims to assign a subset  $\mathbf{y}$  containing  $n$  labels in the total label set  $\mathcal{Y}$  to  $\mathbf{x}$ . From the perspective of sequence learning, once the order of output labels is pre-defined, the MLC task can be regarded as the generation of target label sequence  $\mathbf{y}$  conditioned on the source text sequence  $\mathbf{x}$ .

### 2.2 Neural Sequence-to-Set Model

Our proposed Seq2Set model consists of an encoder  $\mathcal{E}$  and a set decoder  $\mathcal{D}$ , which are introduced in detail as follows.

**Encoder  $\mathcal{E}$ :** We implement the encoder  $\mathcal{E}$  as a bidirectional LSTM. Given the input text  $\mathbf{x} = (x_1, \dots, x_m)$ , the encoder computes the hidden states of each word as follows:

$$\vec{h}_i = \overrightarrow{\text{LSTM}}(\vec{h}_{i-1}, e(x_i)) \quad (1)$$

$$\overleftarrow{h}_i = \overleftarrow{\text{LSTM}}(\overleftarrow{h}_{i+1}, e(x_i)) \quad (2)$$

where  $e(x_i)$  is the embedding of  $x_i$ . The final representation of the  $i$ -th word is  $h_i = [\vec{h}_i; \overleftarrow{h}_i]$ , where semicolon denotes vector concatenation.

**Set decoder  $\mathcal{D}$ :** Due to its powerful ability of LSTM to model sequence dependency, we also implement  $\mathcal{D}$  as a LSTM model to capture high-order correlations between labels. In particular, the hidden state  $s_t$  of the set decoder  $\mathcal{D}$  at time-step  $t$  is computed as:

$$s_t = \text{LSTM}(s_{t-1}, [e(y_{t-1}); c_t]) \quad (3)$$

where  $[e(y_{t-1}); c_t]$  denotes the concatenation of vectors  $e(y_{t-1})$  and  $c_t$ ,  $e(y_{t-1})$  is the embedding

of the label  $y_{t-1}$  generated at the last time-step, and  $c_t$  is the context vector obtained by the attention mechanism. Readers can refer to Bahdanau et al. (2015) for more details. Finally, the set decoder  $\mathcal{D}$  samples a label  $y_t$  from the output probability distribution, which is computed as follows:

$$o_t = \mathbf{W}_2 f(\mathbf{W}_1 s_t + \mathbf{U} c_t) \quad (4)$$

$$y_t \sim \text{softmax}(o_t + I_t) \quad (5)$$

where  $\mathbf{W}_1$ ,  $\mathbf{W}_2$ , and  $\mathbf{U}$  are trainable parameters,  $f$  is a nonlinear activation function, and  $I_t \in \mathbb{R}^{|\mathcal{Y}|}$  is the mask vector that prevents  $\mathcal{D}$  from generating repeated labels,

$$(I_t)_i = \begin{cases} -\infty & \text{if the } i\text{-th label has been predicted.} \\ 0 & \text{otherwise.} \end{cases}$$

### 2.3 Model Training

#### MLC as a RL Problem

In order to alleviate the dependence of the model on the label order, here we model the MLC task as a RL problem. Our set decoder  $\mathcal{D}$  can be viewed as an *agent*, whose *state* at time-step  $t$  is the current generated labels  $(y_1, \dots, y_{t-1})$ . A stochastic *policy* defined by the parameter of  $\mathcal{D}$  decides the *action*, which is the prediction of the next label. Once a complete label sequence  $\mathbf{y}$  is generated, the *agent*  $\mathcal{D}$  will observe a *reward*  $r$ . The training objective is to minimize negative expected reward, which is as follows:

$$\mathcal{L}(\theta) = -\mathbb{E}_{\mathbf{y} \sim p_\theta} [r(\mathbf{y})] \quad (6)$$

where  $\theta$  refers to the model parameter. In our model, we use the self-critical policy gradient algorithm (Rennie et al., 2017). For each training sample in the minibatch, the gradient of Eq.(6) can be approximated as:

$$\nabla_\theta \mathcal{L}(\theta) \approx -[r(\mathbf{y}^s) - r(\mathbf{y}^g)] \nabla_\theta \log(p_\theta(\mathbf{y}^s)) \quad (7)$$

where  $\mathbf{y}^s$  is the label sequence sampled from probability distribution  $p_\theta$  and  $\mathbf{y}^g$  is the label sequence generated with the greedy search algorithm.  $r(\mathbf{y}^g)$  in Eq.(7) is the baseline, which aims to reduce the variance of gradient estimate and enhance the consistency of the model training and testing to alleviate *exposure bias* (Ranzato et al., 2016).

#### Reward Design

The ideal reward is supposed to be a good measure of the quality of the generated labels. Besides,

Models	HL (-)	0/1 Loss (-)	F1 (+)	Precision (+)	Recall (+)
BR-LR (Boutell et al., 2004)	0.0083	0.393	0.858	0.919	0.804
PCC-LR (Read et al., 2011)	0.0079	0.325	0.864	0.901	0.827
FastXML (Prabhu and Varma, 2014)	0.0078	0.358	0.863	<b>0.956</b>	0.786
XML-CNN (Liu et al., 2017)	0.0086	0.390	0.853	0.914	0.799
CNN-RNN (Chen et al., 2017)	0.0085	0.378	0.856	0.889	0.825
Seq2Seq (Yang et al., 2018)	0.0076	0.332	0.871	0.906	0.838
<b>Seq2Set (Ours)</b>	<b>0.0073</b>	<b>0.314</b>	<b>0.879</b>	0.900	<b>0.858</b>

Table 1: Performance of different systems. “HL”, “0/1 Loss”, “F1”, “Precision”, and “Recall” denote hamming loss, subset zero-one loss, micro-F<sub>1</sub>, micro-precision, and micro-recall, respectively. “+” indicates higher is better and “-” is opposite. The best performance is highlighted in bold.

in order to free the model from the strict restriction of label order, it should also satisfy *swapping-invariance* of the output label set. Motivated by this, we design the reward  $r$  as the F<sub>1</sub> score calculated by comparing the generated labels  $\mathbf{y}$  with ground-truth labels  $\mathbf{y}^*$ .<sup>4</sup>

$$r(\mathbf{y}) = F_1(\mathbf{y}, \mathbf{y}^*) \quad (8)$$

We also tried other reward designs, such as hamming accuracy. Results show that reward based on F<sub>1</sub> score gives the best overall performance.

### 3 Experiments

#### 3.1 Datasets

We conduct experiments on the RCV1-V2 corpus (Lewis et al., 2004), which consists of a large number of manually categorized newswire stories. The total number of labels is 103. We adopt the same data-splitting in Yang et al. (2018).

#### 3.2 Settings

We tune hyper-parameters on the validation set based on the micro-F<sub>1</sub> score. The vocabulary size is 50,000 and the batch size is 64. we set the embedding size to 512. Both encoder and set decoder is a 2-layer LSTM with the hidden size 512, but the former is set to bidirectional. We pre-train the model for 20 epochs via MLE method. The optimizer is Adam (Kingma and Ba, 2015) with 10<sup>-3</sup> learning rate for pre-training and 10<sup>-5</sup> for RL. Besides, we use dropout (Srivastava et al., 2014) to avoid overfitting and clip the gradients (Pascanu et al., 2013) to the maximum norm of 8.

#### 3.3 Baselines

We compare our approach with the following competitive baselines:

- **BR-LR** (Boutell et al., 2004) amounts to independently training one binary classifier (logistic regression) for each label.
- **PCC-LR** (Read et al., 2011) transforms the MLC task into a chain of binary classification (logistic regression) problems.
- **FastXML** (Prabhu and Varma, 2014) learns a hierarchy of training instances and optimizes the objective at each node of the hierarchy.
- **XML-CNN** (Liu et al., 2017) uses a dynamic max pooling scheme and a hidden bottleneck layer for better representations of documents.
- **CNN-RNN** (Chen et al., 2017) presents an ensemble approach of CNN and RNN to capture both global and local textual semantics.
- **Seq2Seq** (Nam et al., 2017; Yang et al., 2018) adapts the Seq2Seq model to perform multi-label classification.

#### 3.4 Evaluation Metrics

The evaluation metrics include: subset zero-one loss calculating the fraction of misclassifications, hamming loss denoting the fraction of wrongly predicted labels to total labels, and micro-F<sub>1</sub> that is the weighted average of F<sub>1</sub> score of each class. Micro-precision and micro-recall are also reported for reference.

## 4 Results and Discussion

Here we conduct an in-depth analysis on the model and experimental results. For simplicity, we use BR to represent the baseline BR-LR.

### 4.1 Experimental Results

The comparison between our approach and all baselines is presented in Table 1, showing that

<sup>4</sup>When calculating F<sub>1</sub> score, we convert  $\mathbf{y}$  and  $\mathbf{y}^*$  into  $|\mathcal{Y}|$ -dimensional sparse vectors.

Models	HL (-)	0/1 Loss (-)	F1 (+)
BR	0.0083 ( $\downarrow$ 0.0%)	0.393 ( $\downarrow$ 0.0%)	0.858 ( $\downarrow$ 0.0%)
Seq2Seq	0.0083 ( $\downarrow$ 9.2%)	0.363 ( $\downarrow$ 9.3%)	0.859 ( $\downarrow$ 1.4%)
<b>Seq2Set</b>	<b>0.0075 (<math>\downarrow</math>2.7%)</b>	<b>0.318 (<math>\downarrow</math>1.2%)</b>	<b>0.876 (<math>\downarrow</math>0.3%)</b>

Table 2: Comparison on the label-shuffled RCV1-V2 dataset. “ $\downarrow$ ” indicates that the model is degraded.

the proposed Seq2Set model can outperform all baselines by a large margin in all evaluation metrics. Compared to BR which completely ignores the label correlations, our Seq2Set model achieves a reduction of 12.05% hamming-loss. It shows that modeling high-order label correlations can largely improve results. Compared to Seq2Seq that makes strict requirements on the label order, our Seq2Set model achieves a reduction of 3.95% hamming-loss on the RCV1-V2 dataset. This indicates that our approach can achieve substantial improvements by reducing the dependence of the model on the label order.

#### 4.2 Reducing Sensitivity to Label Order

To verify that our approach can reduce the sensitivity to the label order, we randomly shuffle the order of the label sequences. Table 2 presents the performance of various models on the label-shuffled RCV1-V2 dataset. Results show that for the shuffled label order, BR is not affected, but the performance of Seq2Seq declines drastically. The reason is that the decoder of Seq2Seq is essentially a conditional language model. It relies heavily on a reasonable label order to model the intrinsic association between labels, while labels in this case present an unordered state. However, our model’s performance on subset zero-one loss drops by only 1.2%<sup>5</sup>, while Seq2Seq drops by 9.3%. This shows that our Seq2Set model is more robust, which can resist disturbances in the label order. Our model is trained via reinforcement learning and reward feedback is independent of the label order, which reduces sensitivity to the label order.

#### 4.3 Improving Model Universality

The labels in the RCV1-V2 dataset exhibits a long-tail distribution. However, in real-scenarios, there are other common label distributions, e.g., uniform distribution (Lin et al., 2018a). Therefore, here we analyze the universality of the Seq2Set

<sup>5</sup>This weak decline can be attributed to the influence of the label order on the pre-training.

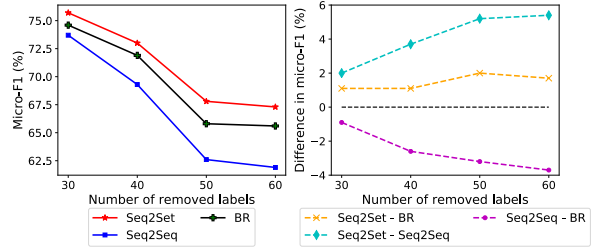


Figure 1: **Left:** Performance of different models. **Right:** The gap of performance of different models.

model, which means that it can achieve stable improvements in performance under different label distributions. In detail, we remove the most frequent  $k$  labels in turn on the RCV1-V2 dataset and perform the evaluation on the remaining labels. The larger the  $k$ , the more uniform the label distribution. Figure 1 shows changes in the performance of different systems.

First, as the number of removed high-frequency labels increases, the performance of all methods deteriorates. This is reasonable because predicting low-frequency labels is relatively difficult. However, compared to other methods, the performance of the Seq2Seq model is greatly degraded. We suspect this is because it’s difficult to define a reasonable order for uniformly distributed labels while Seq2Seq imposes strict requirements on the label order. This conflict may damage performance. However, as shown in Figure 1, as more labels are removed, the advantage of Seq2Set over Seq2Seq continues to grow. This illustrates that our Seq2Set model has excellent universality, which works for different label distributions. Our approach not only has the ability of Seq2Seq to capture label correlations, but also alleviates the strict requirements of Seq2Seq for label order via reinforcement learning. This avoids the problem of difficulty in predefining a reasonable label order on the uniform distribution, leading to excellent universality.

#### 4.4 Error Analysis

We find that all methods perform poorly when predicting low-frequency (LF) labels compared to high-frequency (HF) labels. This is reasonable because samples assigned LF labels are sparse, making it hard for the model to learn an effective pattern to make predictions. Figure 2 shows the results of different methods on HF labels and



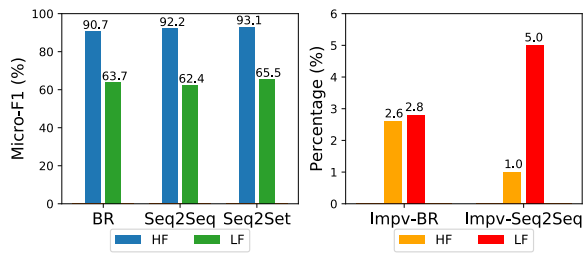


Figure 2: Performance of different systems on the HF labels and LF labels. “Impv-BR” and “Impv-Seq2Seq” denote the improvement of our model compared to BR-LR and Seq2Seq, respectively.

LF labels<sup>6</sup>. However, compared to other systems, our proposed Seq2Set model achieves better performance on both LF labels and HF labels. Besides, the relative improvements achieved by our approach are greater on LF labels than HF labels. In fact, the distribution of LF labels is relatively more uniform. As analyzed in Section 4.3, the label order problem is more serious in the uniform distribution. Our Seq2Set model can reduce the dependence on the label order via reinforcement learning, leading to larger improvements in performance on the LF labels.

## 5 Related Work

Multi-label classification (MLC) aims to assign multiple labels to each sample in the dataset. Early work on exploring the MLC task focuses on machine learning algorithms, mainly including problem transformation methods and algorithm adaptation methods. Problem transformation methods, such as BR (Boutell et al., 2004), LP (Tsoumakas and Katakis, 2006) and CC (Read et al., 2011), aim at mapping the MLC task into multiple single-label learning tasks. Algorithm adaptation methods strive to extend specific learning algorithms to handle multi-label data directly. The corresponding representative work includes ML-DT (Clare and King, 2001), Rank-SVM (Elisseeff and Weston, 2001), ML-KNN (Zhang and Zhou, 2007), and so on. In addition, some other methods, including ensemble method (Tsoumakas et al., 2011) and joint training (Li et al., 2015), can also be used for the MLC task. However, they can only be used to capture the first or second order label correlations (Chen et al., 2017), or are computationally intractable when high-order label correlations are considered.

<sup>6</sup>By frequency, the top 10% of labels are regarded as HF labels, and the last 10% of labels are regarded as LF labels.

Recent years, some neural network models have also been successfully used for the MLC task. For instance, the BP-MLL proposed by Zhang and Zhou (2006) applies a fully-connected network and the pairwise ranking loss to perform classification. Nam et al. (2013) further replace the pairwise ranking loss with cross-entropy loss function. Kurata et al. (2016) present an initialization method to model label correlations by leveraging neurons. Chen et al. (2017) present an ensemble approach of CNN and RNN so as to capture both global and local semantic information. Liu et al. (2017) use a dynamic max pooling scheme and a hidden bottleneck layer for better representations of documents. Graph convolution operations are employed by Peng et al. (2018) to capture non-consecutive and long-distance semantics. The two milestones are Nam et al. (2017) and Yang et al. (2018), both of which utilize the Seq2Seq model to capture the label correlations. Going a step further, Lin et al. (2018b) propose a semantic-unit-based dilated convolution model and Zhao et al. (2018) present a label-graph based neural network equipped with a soft training mechanism to capture label correlations. Most recently, Qin et al. (2019) present new training objectives propose based on set probability to effectively model the mathematical characteristics of the set.

## 6 Conclusion

In this work, we present a simple but effective sequence-to-set model based on reinforcement learning, which aims to reduce the stringent requirements of the sequence-to-sequence model for label order. The proposed model not only captures high-order correlations between labels, but also reduces the dependence on the order of output labels. Experimental results show that our Seq2Set model can outperform competitive baselines by a large margin. Further analysis demonstrates that our approach can effectively reduce the sensitivity to the label order.

## Acknowledgement

We thank the anonymous reviewers for their thoughtful comments. We also would like to thank Lei Li, Yi Zhang, and Xuancheng Ren for their insightful suggestions. Xu Sun is the contact author of this paper.

## References

- Dzmitry Bahdanau, Kyunghyun Cho, and Yoshua Bengio. 2015. Neural machine translation by jointly learning to align and translate. In *3rd International Conference on Learning Representations, Conference Track Proceedings*.
- Simon Baker and Anna Korhonen. 2017. Initializing neural networks for hierarchical multi-label text classification. pages 307–315.
- Matthew R. Boutell, Jiebo Luo, Xipeng Shen, and Christopher M. Brown. 2004. Learning multi-label scene classification. *Pattern Recognition*, 37(9):1757–1771.
- Guibin Chen, Deheng Ye, Zhenchang Xing, Jieshan Chen, and Erik Cambria. 2017. Ensemble application of convolutional and recurrent neural networks for multi-label text categorization. In *2017 International Joint Conference on Neural Networks*, pages 2377–2383.
- Amanda Clare and Ross D King. 2001. Knowledge discovery in multi-label phenotype data. In *European Conference on Principles of Data Mining and Knowledge Discovery*, pages 42–53. Springer.
- André Elisseeff and Jason Weston. 2001. A kernel method for multi-labelled classification. In *Advances in Neural Information Processing Systems 14 [Neural Information Processing Systems: Natural and Synthetic]*, pages 681–687.
- Siddharth Gopal and Yiming Yang. 2010. Multilabel classification with meta-level features. In *Proceeding of the 33rd International ACM SIGIR Conference on Research and Development in Information Retrieval*, pages 315–322.
- Yacine Jernite, Anna Choromanska, and David Sontag. 2017. Simultaneous learning of trees and representations for extreme classification and density estimation. In *Proceedings of the 34th International Conference on Machine Learning*, pages 1665–1674.
- Diederik P. Kingma and Jimmy Ba. 2015. Adam: A method for stochastic optimization. In *3rd International Conference on Learning Representations, Conference Track Proceedings*.
- Gakuto Kurata, Bing Xiang, and Bowen Zhou. 2016. Improved neural network-based multi-label classification with better initialization leveraging label co-occurrence. In *NAACL HLT 2016, The 2016 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 521–526.
- David D. Lewis, Yiming Yang, Tony G. Rose, and Fan Li. 2004. RCV1: A new benchmark collection for text categorization research. *Journal of Machine Learning Research*, 5:361–397.
- Li Li, Houfeng Wang, Xu Sun, Baobao Chang, Shi Zhao, and Lei Sha. 2015. Multi-label text categorization with joint learning predictions-as-features method. In *Proceedings of the 2015 Conference on Empirical Methods in Natural Language Processing*, pages 835–839.
- Junyang Lin, Qi Su, Pengcheng Yang, Shuming Ma, and Xu Sun. 2018a. Semantic-unit-based dilated convolution for multi-label text classification. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*, pages 4554–4564.
- Junyang Lin, Qi Su, Pengcheng Yang, Shuming Ma, and Xu Sun. 2018b. Semantic-unit-based dilated convolution for multi-label text classification. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*, pages 4554–4564.
- Jingzhou Liu, Wei-Cheng Chang, Yuexin Wu, and Yiming Yang. 2017. Deep learning for extreme multi-label text classification. In *Proceedings of the 40th International ACM SIGIR Conference on Research and Development in Information Retrieval*, pages 115–124.
- Jinseok Nam, Jungi Kim, Iryna Gurevych, and Johannes Fürnkranz. 2013. Large-scale multi-label text classification - revisiting neural networks. *arXiv preprint arXiv:1312.5419*.
- Jinseok Nam, Eneldo Loza Mencía, Hyunwoo J Kim, and Johannes Fürnkranz. 2017. Maximizing subset accuracy with recurrent neural networks in multi-label classification. In *Advances in Neural Information Processing Systems 30: Annual Conference on Neural Information Processing Systems 2017*, pages 5419–5429.
- Razvan Pascanu, Tomas Mikolov, and Yoshua Bengio. 2013. On the difficulty of training recurrent neural networks. In *Proceedings of the 30th International Conference on Machine Learning*, pages 1310–1318.
- Hao Peng, Jianxin Li, Yu He, Yaopeng Liu, Mengjiao Bao, Lihong Wang, Yangqiu Song, and Qiang Yang. 2018. Large-scale hierarchical text classification with recursively regularized deep graph-cnn. In *Proceedings of the 2018 World Wide Web Conference on World Wide Web*, pages 1063–1072.
- Yashoteja Prabhu and Manik Varma. 2014. Fastxml: A fast, accurate and stable tree-classifier for extreme multi-label learning. In *The 20th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, pages 263–272.
- Kechen Qin, Cheng Li, Virgil Pavlu, and Javed A Aslam. 2019. Adapting rnn sequence prediction model to multi-label set prediction. *arXiv preprint arXiv:1904.05829*.

- Marc’Aurelio Ranzato, Sumit Chopra, Michael Auli, and Wojciech Zaremba. 2016. Sequence level training with recurrent neural networks. In *4th International Conference on Learning Representations, Conference Track Proceedings*.
- Jesse Read, Bernhard Pfahringer, Geoff Holmes, and Eibe Frank. 2011. Classifier chains for multi-label classification. *Machine learning*, 85(3):333.
- Steven J. Rennie, Etienne Marcheret, Youssef Mroueh, Jarret Ross, and Vaibhava Goel. 2017. Self-critical sequence training for image captioning. In *2017 IEEE Conference on Computer Vision and Pattern Recognition*,, pages 1179–1195.
- Robert E Schapire and Yoram Singer. 2000. Boostexter: A boosting-based system for text categorization. *Machine learning*, 39(2-3):135–168.
- Gaurav Singh, James Thomas, Iain James Marshall, John Shawe-Taylor, and Byron C. Wallace. 2018. Structured multi-label biomedical text tagging via attentive neural tree decoding. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*, pages 2837–2842.
- Nitish Srivastava, Geoffrey E. Hinton, Alex Krizhevsky, Ilya Sutskever, and Ruslan Salakhutdinov. 2014. Dropout: a simple way to prevent neural networks from overfitting. *Journal of Machine Learning Research*, 15(1):1929–1958.
- Richard S Sutton, David A McAllester, Satinder P Singh, and Yishay Mansour. 1999. Policy gradient methods for reinforcement learning with function approximation. In *Advances in Neural Information Processing Systems 12, [NIPS Conference]*, pages 1057–1063.
- Grigorios Tsoumakas and Ioannis Katakis. 2006. Multi-label classification: An overview. *International Journal of Data Warehousing and Mining*, 3(3).
- Grigorios Tsoumakas, Ioannis Katakis, and Ioannis Vlahavas. 2011. Random k-labelsets for multilabel classification. *IEEE Transactions on Knowledge and Data Engineering*, 23(7):1079–1089.
- Oriol Vinyals, Samy Bengio, and Manjunath Kudlur. 2016. Order matters: Sequence to sequence for sets. In *4th International Conference on Learning Representations, Conference Track Proceedings*.
- Pengcheng Yang, Xu Sun, Wei Li, Shuming Ma, Wei Wu, and Houfeng Wang. 2018. SGM: Sequence generation model for multi-label classification. In *Proceedings of the 27th International Conference on Computational Linguistics*, pages 3915–3926.
- Min-Ling Zhang and Zhi-Hua Zhou. 2006. Multilabel neural networks with applications to functional genomics and text categorization. *IEEE Transactions on Knowledge and Data Engineering*, 18(10):1338–1351.
- Min-Ling Zhang and Zhi-Hua Zhou. 2007. ML-KNN: A lazy learning approach to multi-label learning. *Pattern recognition*, 40(7):2038–2048.
- Min-Ling Zhang and Zhi-Hua Zhou. 2014. A review on multi-label learning algorithms. *IEEE transactions on knowledge and data engineering*, 26(8):1819–1837.
- Guangxiang Zhao, Jingjing Xu, Qi Zeng, and Xuancheng Ren. 2018. Review-driven multi-label music style classification by exploiting style correlations. *arXiv preprint arXiv:1808.07604*.