Improved Sequence Generation Model (SGM) for Multi-label Classification with Label Correlation

Junha Choi
Simon Fraser University
junhac@sfu.ca

Zhiheng Zhang
Simon Fraser University
zhihengz@sfu.ca

Weiling Zheng
Simon Fraser University
weilingz@sfu.ca

Sicong Liu
Simon Fraser University
sla255@sfu.ca

Hao Zheng
Simon Fraser University
hza89@sfu.ca

Abstract

Multi-label classication (MLC) is an important yet challenging problem in natural language processing that expects the learning algorithm to take the hidden correlation of the labels into account. In this work, we propose a importance-based sorting method and a learn-able transformation matrix module to extend Sequence Generation Model(SGM) [1]. Experimental results demonstrate that both importance based sorting method and transformation matrix module outperform SGM when a dataset is large enough.

1 Introduction

Sequence-to-sequence (Seq2Seq) model has achieved great success in machine translation in recent studies [3, 4, 5]. SGM proposed a sequence generation model to solve MLC problem with the correlations between labels accounted extending Seq2Seq model [1]. However, solving MLC with sequence generation model faces the problem of ordering labels in the sequence. SGM sorted the target labels of each sample according to the frequency of labels in training set.

In this work, inspired by the fact that some MLC datasets such as US-WIDE [6], have ground truth label hierarchy, we propose sorting labels according to the importance of labels to extend SGM. The proposed sorting method gives higher weight to label when its training sample has fewer target labels. For example, it gives weight 1 to label $People$ in sequence $< People >$ and gives $1/3$ to all three labels in label sequence $< People, Military, Police >$. This approach might be better at capturing labels hierarchy information. Furthermore, inspired by Rethinking net [2], we introduced a transformation matrix module to extend SGM, which memorizes correlation coefficients between each pair of labels.

The whole paper is organized as follow. We describe the overview of SGM and our methods in Section 2. In Section 3, we display the experiments and make analysis and conclusion of this paper.

2 Proposed Method

We introduce our proposed method in detail in this section. First we give an overview of our method in Section 2.1. Then, we introduce the details of the proposed sorting method in Section 2.2. Finally, in Section 2.3, we present our weight matrix module.

2.1 Overview

Given the label space with L labels $\mathcal{L} = l_1, l_2, ..., l_L$, a text sequence x containing m words, the task is to assign a subset y containing n labels in the label space L to x. From the perspective of sequence generation model, the MLC task can be modeled as finding an optimal label sequence y that maximizes the conditional probability $p(y|x)$, which is calculated as follows:

$$p(y|x) = \prod_{i=1}^{n} p(y_i|y_1, y_2, ..., y_{i-1}, x) \tag{1}$$

An overview of our proposed model is shown in Figure 1. First, the target labels of each sample are sorted based on the frequency of the labels in the training set. High-frequency labels are placed in the front. In addition, BOS (Beginning Of Sentence) and EOS (End Of Sentence) symbols are added to the head and tail of the label sequence to indicate the first and the last labels, respectively.

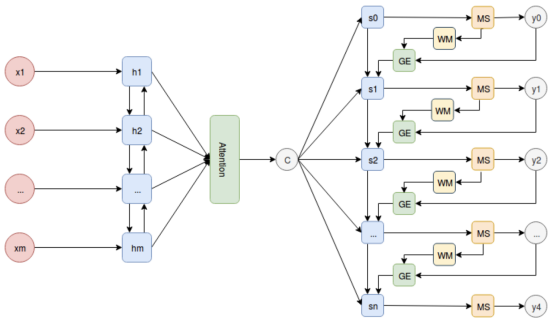

Figure 1: The overview of our proposed model with weight matrix. MS denotes the masked softmax layer. GE denotes the global embedding. WM denotes weight matrix

Bi-directional Encoder: Let x_i be the dense embedding vector of word i. SGM uses a bidirectional LSTM to read the text sequence x from both directions and compute the hidden states for each word,

$$\overrightarrow{h_i} = \overrightarrow{LSTM}(\overrightarrow{h}_{i-1}, x_i) \tag{2}$$

$$\overleftarrow{h_i} = \overleftarrow{LSTM}(\overleftarrow{h}_{i+1}, x_i) \tag{3}$$

The final hidden representation of the i^{th} word is produced by concatenating the hidden states from both directions, $h_i = [\overrightarrow{h_i}; \overleftarrow{h_i}]$.

Attention The attention mechanism assigns the weight α_{t_i} to the i^{th} word at time step t as follows:

$$e_{ti} = v_a^T tanh(W_a s_t + U_a h_i) \tag{4}$$

$$\alpha_{t_i} = \frac{exp(e_{ti})}{\sum_{j=1}^{m} exp(e_{tj})} \tag{5}$$

where W_a, U_a, v_a are weight parameters and s_t is the current hidden state of the decoder at time step t.

Uni-directional Decoder: The hidden state s_t of the decoder at time step t is computed as follows:

$$s_t = LSTM(s_{t-1}, [g(y_{t-1}; c_{t-1})]) \tag{6}$$

$$\overleftarrow{h_i} = \overleftarrow{LSTM}(\overleftarrow{h}_{i+1}, x_i) \tag{7}$$

The final hidden representation of the i^{th} word is produced by concatenating the hidden states from both directions, $h_i = [\overrightarrow{h_i}; \overleftarrow{h_i}]$.

2.2 Importance Based Label Sorting

Solving MLC with seq2seq model requires to sort labels first to generate label sequence. Although SGM's experiment shows that sorting label based on frequency in training set is effective, we argue that it is not the best solution. Inspired by the fact that some MLC datasets like US-WIDE [6] have the ground truth label hierarchy, which is shown in Figure 2, we can improve decoder by adapting a **top-down** predicting approach. Intuitively, if a sample S has short true labels, eg. $< People >$, these labels are very likely be at top level and therefore have high weights. On the contrary, if a sample S has long true labels, eg. $< People, Military, Police >$, these labels are less likely be at top level and therefore have lower weights. We propose importance-based label sorting to give high weights to labels at top level in label hierarchy.

Let S denotes all training samples and L denotes label length of the sample, we define importance as follow:

$$I_l = \sum_{l \subset x_t, x \in S} \frac{1}{L_{x_t}} \tag{8}$$

For a training sample x, we gives one over label length weight to its label. For a certain label l, the importance I_l is the sum of all its weight through the training set.

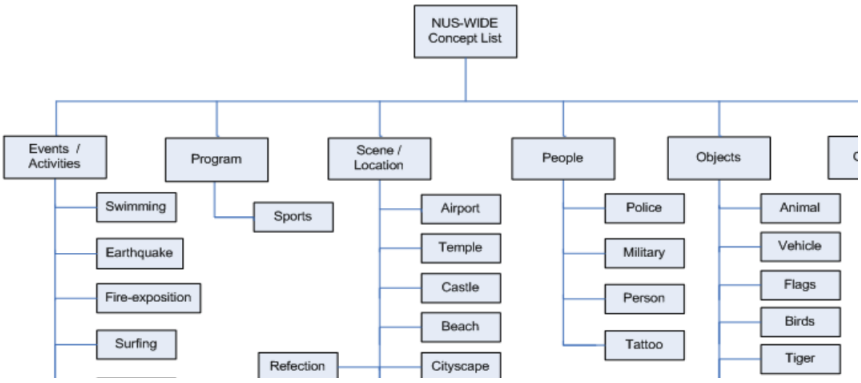

Figure 2: Screenshot of US-WIDE[6]'s label hierarchy

2.3 Transformation Matrix Module

In the sequence generation model mentioned above, the LSTM structure in decoder uses the global embedding, where the embedding vector $g(\mathbf{y}_{t-1})$ at time step t is able to represent the overall information at time step $(t-1)$.

$$g(\mathbf{y}_{t-1}) = (1 - \mathbf{H}) \odot \mathbf{e} + \mathbf{H} \odot \bar{\mathrm{e}} \tag{9}$$

where $\bar{\mathrm{e}}$ is the weighted average embedding at time t which is defined as follows:

$$\bar{\mathrm{e}} = \sum_{i=1}^{L} y_{t-1}^{i} e_i \tag{10}$$

And H is the transform gate controlling the proportion of the weighted average embedding.

RethinkingNet [2] introduces a weight matrix module to memorize label coefficient. Before the output of softmax layer going into global embedding, we add the weight matrix module which has a weight matrix W whose element W_{ij} represents the correlation between the i^{th} label and the j^{th} label. With weight matrix module, $\bar{e}$ is now the output of weight matrix module. With Weight Matrix module, $\bar{\mathrm{e}}$ can be expressed as follows:

$$\bar{\mathrm{e}} = \sum_{i=1}^{L} (W y_{t-1}^{i}) e_i \tag{11}$$

3 Experiments

In this section, we evaluate our proposed methods on two datasets. The main purpose is to evaluate our extension on SGM and compare it with other baselines. For a fair comparison, we try to mimic SGM's experimental setup.

3.1 Datasets

Reuters Corpus Volume I (RCV1-V2): This dataset is provided by [3]. **The large size** dataset consists of over 800,000 manually categorized newswire stories made available by Reuters Ltd for research purposes. Multiple topics can be assigned to each newswire story and there are **103 topics(labels)** in total.

Arxiv Academic Paper Dataset (AAPD): This dataset is provided by [1]. It collects the subjects of 55,840 papers in the computer science field. An academic paper may have multiple subjects and there are **54 subjects(labels)** in total.

Dataset	Total Samples	Label Sets	Words/Sample	Labels/Sample
RCV1-V2	804,414	103	123.94	3.24
AAPD	55840	54	163.42	2.41

Table 1: Dataset Comparison

3.2 Evaluation Metrics

We adopt hamming loss and micro-F1 score as our main evaluation metrics.

Hamming-loss (HL) evaluates the fraction of misclassified instance-label pairs, where a relevant label is missed or an irrelevant is predicted.

Micro-F1 (F1) can be interpreted as a weighted average of the precision and recall. It is calculated globally by counting the total true positives, false negatives, and false positives.

3.3 Results

3.3.1 Importance Based Label Sorting

SGM		
Sorting	HL(-)	F1(+)
w/o	0.0084	0.858
frequency	0.0081	0.869
importance	**0.0069(↓ 14.81%)**	**0.887(↑ 2.07%)**
SGM+GE		
Sorting	HL(-)	F1(+)
w/o	0.0083	0.859
frequency	0.0075	0.878
importance	**0.0070(↓ 6.67%)**	**0.8865(↑ 0.96%)**

Table 2: Ablation study for sorting on RCV1-V2 test set.

Table 2 and table 3 show the comparison between importance based label sorting and frequency based label sorting. GE denotes global embedding. The symbol "+" indicates that the higher the value is, the better the model performs. The symbol "-" is the opposite.

The experimental results of importance based label sorting and the baseline on dataset RCV1-V2 are shown in Table 2. The results demonstrate that our proposed method has better performance in both SGM and SGM with global embedding. Our proposed model without global embedding achieves a reduction of 14.81% hamming-loss and an improvement of 2.07% micro-F_1 score compare to the

SGM		
Sorting	HL(-)	F1(+)
w/o	N/A	N/A
frequency	0.0251	0.699
importance	**0.02521(↑ 0.44%)**	**0.7030(↑ 0.57%)**
SGM+GE		
Sorting	HL(-)	F1(+)
w/o	N/A	N/A
frequency	0.02521	0.7030
importance	**0.02538(↑ 0.67%)**	**0.7044(↑ 0.20%)**

Table 3: Ablation study for sorting on AAPD test set.

baseline which uses the order of frequency. Besides, with global embedding, our method achieves a reduction of 6.67% hamming-loss and an improvement of 0.96% micro-F_1 score.

In addition, our model with importance based label sorting is significantly improved without using global embedding compared to the baseline.

Table 3 presents the results of the proposed methods and the baseline on the AAPD test set. Compared to the experimental results on the RCV1-V2 test set, our proposed method has a slight reduction of 0.44% and 0.67% hamming score in SGM with and without global embedding respectively. However, for micro-F_1 score, results show that our model has an improvement of 0.57% and 0.20% on the test set with and without global embedding.

3.3.2 Transformation Matrix Module

We compare our proposed transformation matrix module with the following baselines: Binary Relevance(BR) [7], Classifier Chains (CC) [8], Label Powerset (LP) [9], CNN [10] and CNN-RNN [11]. To analyze the effect of weight matrix module only, we use frequency based label sorting.

Models	HL(-)	F1(+)
BR	0.0086	0.858
CC	0.0087	0.857
LP	0.0087	0.858
CNN	0.0089	0.855
CNN-RNN	0.0085	0.856
SGM	0.0081	0.869
+GE	0.0075	0.878
+WM	**0.0069(↓ 8.00%)**	**0.886(↑ 0.91%)**

Table 4: Performance on the RCV1-V2 test set.

Models	HL(-)	F1(+)
BR	0.0316	0.646
CC	0.0306	0.654
LP	0.0312	0.634
CNN	0.0256	0.664
CNN-RNN	0.0278	0.664
SGM	0.0251	0.699
+GE	0.0245	0.710
+WM	**0.0252(↑ 2.00%)**	**0.698(↓ 1.60%)**

Table 5: Performance on the AAPD test set.

Table 4 and table 5 compares the proposed method with all baselines on two datasets. GE denotes the global embedding. WM denotes the weight matrix. The symbol "+" indicates that the higher the value is, the better the model performs. The symbol "-" is the opposite.

Table 4 presents the experimental results of our proposed method and the baselines on RCV1-V2 dataset. Results show that our proposed method gives the best performance. Using weight matrix with SGM achieves a reduction of 8% hamming-loss and an improvement of 0.91% micro-F_1 score compare to the second best method which is SGM with GE. Besides, our methods outperform other traditional deep-learning models by a large margin.

However, results in table 5 show weight matrix does not work well for the smaller dataset AAPD, and our proposed method did not improve the SGM method. Using weight matrix with SGM, hamming-loss increased by 2% and micro-F_1 score decreased by 1.6%.

3.4 Analysis and Conclusion

3.4.1 Impact of importance sorting

Compared with original SGM, proposed importance-based sorting outperforms frequency-based sorting in the large dataset (RCV1-V2), while performs similarly in the small dataset (AAPD). The reason might be that RCV1-V2 not only has more training samples, its label set is almost twice as large as AAPD. It seems like importance-based sorting has a higher potential than frequency-based sorting, but it requires a large label space.

3.4.2 Exploration of weight matrix

As shown in table 4, weight matrix can improve performance of SGM on the large dataset (RCV1-V2) with memorizing useful information in it. On the smaller dataset (AAPD), model overfits during experiment. The experiment indicates that weight matrix module increases the complexity of the model.

4 Contributions

Junha Choi (junhac@sfu.ca): Propose importance-based sorting and Two models to incorporate weight matrix into SGM (with and without Global Embedding). Implement the proposed ways to extend SGM and test them. Write poster and revise the report.

Zhiheng Zhang (zhihengz@sfu.ca): Propose topic and choose two main reference papers, explore embedding and cost sensitive loss function implementation, write poster and report.

Weiling Zheng (weilingz@sfu.ca): Initialize topic of MLC, explore various word embeddings, write poster and report.

Sicong Liu (sla255@sfu.ca): Explore various MLC solutions and SGM. Write poster and report.

Hao Zheng(hza89@sfu.ca): Explore SGM and Music auto-tagger using keras. Write poster and report.

5 References

[1] P. Yang, X. Sun, W. Li, S. Ma, W. Wu, and H. Wang, SGM: Sequence Generation Model for Multi-label Classification, arXiv:1806.04822 [cs], Jun. 2018.

[2] Y.-Y. Yang, Y.-A. Lin, H.-M. Chu, and H.-T. Lin, Deep Learning with a Rethinking Structure for Multi-label Classification, arXiv:1802.01697 [cs, stat], Feb. 2018.

[3] T. Luong, H. Pham, and C. D. Manning, Effective Approaches to Attention-based Neural Machine Translation, in Proceedings of the 2015 Conference on Empirical Methods in Natural Language Processing, Lisbon, Portugal, 2015, pp. 14121421.

[4] X. Sun, B. Wei, X. Ren, and S. Ma, Label Embedding Network: Learning Label Representation for Soft Training of Deep Networks, arXiv:1710.10393 [cs], Oct. 2017.

[5] D. Bahdanau, K. Cho, and Y. Bengio, Neural Machine Translation by Jointly Learning to Align and Translate, arXiv:1409.0473 [cs, stat], Sep. 2014.

[6] T.-S. Chua, J. Tang, R. Hong, H. Li, Z. Luo, and Y. Zheng, NUS-WIDE: a real-world web image database from National University of Singapore, in Proceeding of the ACM International Conference on Image and Video Retrieval - CIVR 09, Santorini, Fira, Greece, 2009, p. 1.

[7] Matthew R. Boutell, Jiebo Luo, Xipeng Shen, and Christopher M. Brown. 2004. Learning multi-label scene classification. Pattern Recognition, 37(9):17571771.

[8] Jesse Read, Bernhard Pfahringer, Geoff Holmes, and Eibe Frank. 2011. Classifier chains for multi-label classification. Machine learning, 85(3):333.

[9] Alexander M. Rush, Sumit Chopra, and Jason Weston. 2015. A neural attention model for abstractive sentence summarization. In Proceedings of the 2015 Conference on Empirical Methods in Natural Language Processing, EMNLP 2015, Lisbon, Portugal, September 17-21, 2015, pages 379389.

[10] Yoon Kim. 2014. Convolutional neural networks for sentence classification. In Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing, EMNLP 2014, October 25-29, 2014, Doha, Qatar, A meeting of SIGDAT, a Special Interest Group of the ACL, pages 17461751.

[11] 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 Inter- national Joint Conference on Neural Networks, IJCNN 2017, Anchorage, AK, USA, May 14-19, 2017, pages 23772383.