BdLAN:BERTdoc Label Attention Networks for Multi-label text classification

Anonymous ACL submission

Abstract

Multi-label text classification (MLTC) brings us new challenge in Natural Language Processing (NLP) which aims at assigning multiple labels for a given document. Many real-world tasks can be view as MLTC, such as tag recommendation, information retrieval, etc. However, several flinty problems are placed in the presence of researchers about how to establish connections between labels or distinguish similar sub-labels, which haven't been solved thoroughly by current endeavor. Therefore, we proposed a novel framework named BdLAN, BERTdoc Label Attention Networks in this paper, consist of the BERTdoc layer, the label embeddings layer, the doc encoder layer, the doc-label attention layer and the prediction layer. We apply a powerful technique BERT to pretrain documents to capture their deep semantic features and encode them via Bi-LSTM to obtain a two-directional contextual representation of uniform length. Then we create label embeddings and feed them together with encoded-pretrained-documents to doc-label attention mechanism to obtain interactive information between documents and their corresponding labels, finally using MLP to make prediction.We carry out experiments on three real-world datasets, the empirical results indicating our proposed model outperforms all state-of-the-art MLTC benchmarks. Moreover, we conduct a case study, visualizing real application of our BdLAN model vividly.

1 Introduction

Text classification is a basic data mining task in Natural Language Processing (NLP), including multi-class text classification and multi-label text classification. Multi-class classification only assigns one label to a given document with over two labels in the whole documents, while multi-label text classification divides a document into different topics at the same time. For example, the sentence "Young boys are playing football" can be categorized as topic "Youth" and "Sports", while a news report such as "The cultural industry will become the pillar industry of the national economy in 2020" belong to either "Economy" or "Culture" as well as the movie "Twilight City" is classified as a romance movie and a fantastic magic movie.

MLTC aims at exploring multiple best-matched document-label pairs according to a specific document and its several corresponding labels, which has many practical scenarios, such as tag recommendation (I. et al., 2008), information retrieval (S. and Y., 2010), etc. For example, it always appears on the homepage of news websites, social platforms such as Weibo and Twitter, introductions and reviews of books or movies, and online shopping malls such as Taobao and Jingdong. It principally devotes itself to reducing hunting zone progressively, facilitating humans to select their required information precisely and improving the quality of automatic recommendations in the background, so as to provide a fast retrieval for users to efficiently search for target information with filtering out redundant and irrelevant counterparts.

However, tremendous difficulties impede our progress to solve the MLTC task accurately. Several tough problems of MLTC are summarized as follows. Firstly, the number of labels of a given text is uncertain with some samples may have only one label but others may belong to dozens or even hundreds of topics. Secondly, there is a mutual dependence between labels so that a big difficulty hinders researchers about how to solve the dependency problem between labels. Thirdly, some low-level labels are difficult to distinguish, such as "news" and "broadcast", "economics" and "finance". Meanwhile, some documents may be very long, including complex semantic hierarchical information hidden in the redundant content. In addition, most documents belong to a few tags while a large number of "tail tags" contain only a few documents.

2 Related work

Problem transformation methods convert the MLTC task into multiple single-label text classification tasks, such as BR (Boutell et al., 2004) ignoring label dependencies and building a separate classifier for each label, LP (I. et al., 2008) creating a binary classifier for each label combination, and CC (Read et al., 2011) converting the MLTC task into a binary classification problem chain.

Algorithm adaptive methods aim at modifying specific algorithms to solve MLTC, including local methods and global methods. Local methods such as ML-DT (A. and R., 2001) which constructs a decision tree, Rank-SVM (A. and J., 2002) which uses SVM similar to a learning system, ML-KNN (Zhang and Zhou, 2007) which applies the k-nearest neighbor algorithm and the maximum posterior probability to determine the label set of each sample, CBM (Li et al., 2016) which simplifies the task by transforming it into multiple binary problems. Global methods such as Clus-HMC (Vens et al., 2008) that uses a single decision tree to process the entire hierarchical category structure, HMC-LMLP (Cerri et al., 2016) that trains a set of neural networks with each neural network predicting a given level of categories, CML (Ghamrawi and McCallum, 2005) which aims at joint learning algorithm (Li et al., 2015) . However, the above-mentioned work mainly focuses on the local or global structure to capture low-order label correlation, ignoring the hierarchical dependencies between different levels of labels, facing thorn difficulties when computing higher-order label correlation.

Neural networks have made significant improvement in MLTC recently. For example, BPMLL (Zhang and Zhou, 2006) applies a fully connected network and pairwise ranking loss to perform classification. Nam et al. (J. et al., 2014) further replaced pairwise ranking loss with a cross-entropy loss function. Kurata, Xiang and Zhou (Kurata et al., 2016) proposed an initialization method, using neurons to model label correlation. Chen et al. (Chen et al., 2017) proposed a joint approach combined with CNN and RNN to capture local and global semantic information. Bahdanau et al. (D. et al., 2017) proposed a method to train a neural network to generate sequences using the actor-critic method. Besides, SGM (Yang et al., 2018) and MDC (Lin et al., 2018) also apply LSTM-based Seq2Seq structure which one applies global embedding to propose a novel decoder, and the other create information-enhanced representations with additional semantic units based on mixed attention mechanism.

3 Proposed Method

3.1 Task description

The MLTC task in this research can be summarized as a tuple set $S = \{(d_i, l_i)\}_{i=1}^{N}$ with d_i and l_i represents the i-th document denoted as $D = \{d_i | d_i = \{d^1, d^2, \cdots, d^n\}$ and its corresponding label sets denoted as $L = \{l_i | l_i = \{l^1, l^2, \cdots, l^m\}$. N, n and m are the total number of documents, the length of the i-th document and the number of labels of the i-th document, respectively. Our proposed BdLAN model aims at assigning all suitable labels to its corresponding documents based on the conditional probability $Pr(l_i|d_i)$ to solve the MLTC task.

3.2 Overview of proposed model

Our proposed BdLAN model consists of five layers, i.e, the BERTdoc layer, the label embeddings layer, the doc encoder layer, the doc-label attention layer and the prediction layer shown in Figure 1. The BERTdoc layer refers to pre-train documents via BERT to extract their semantic features while the label embeddings layer means map each label to a high-dimensional space with GloVe (Pennington et al., 2014). The doc encoder layer denotes encoding each pre-trained word in documents via Bi-LSTM to obtain text representation forward and backward of uniform length. The doc-label attention layer means an interactive strategy capturing mutual features of encoded pre-trained document representation and label embeddings, which then feed into prediction layer (MLP) to complete final multi-label classification. The overall proposed model is trained end-to-end.

3.3 BERTdoc layer

In this layer, we use base-BERT with 12 transformer blocks, 768 dimension of hidden state, 12 head per layer of multi-head attention and 110M parameters to pre-train documents to capture their deep information. We preprocess documents as BERT input representation, which are the sum of the token embeddings aiming at different words, the segmentation embeddings distinguishing each sentence in a paragraph and the position embeddings outputing position of words, then pass them

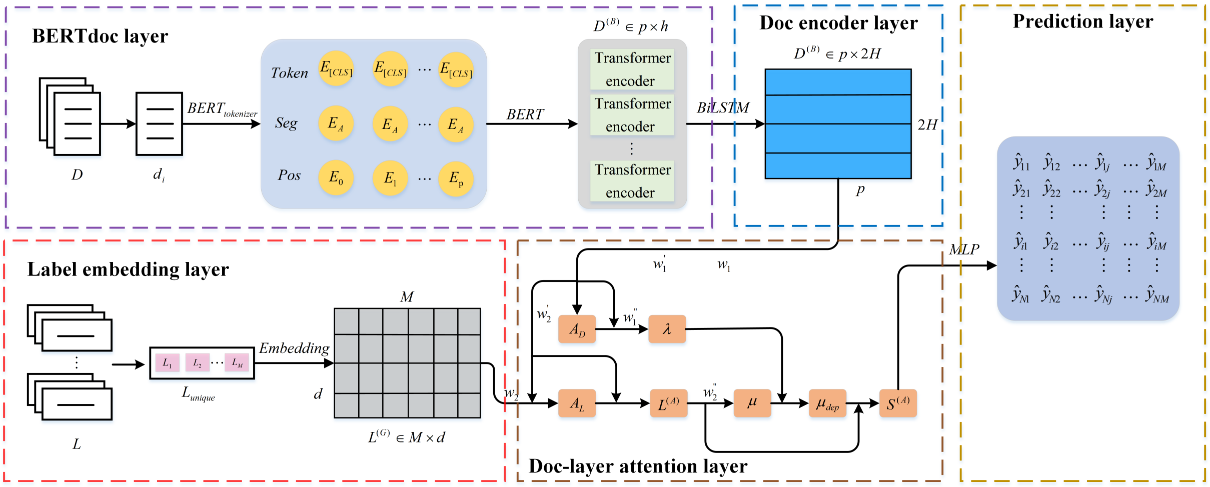


Figure 1: Architecture of our proposed BdLAN model

to transformers mechanism in BERT. Each embedding of BERT input representation is differentiated via slicing which then fed into BERT model to output pre-trained contextual representation of documents. The process of document pre-training can be elaborated as follows:

$$D^{(B)} = BERT(token, seg, pos) \tag{1}$$

3.4 Doc encoder layer

To obtain forward and backward contextual representation of given documents, we adopt bidirectional LSTM (Bi-LSTM) to encode pre-trained documents as $2H$-dimensional vectors. Through the encoder layer, we also unify the length of documents to get encoded pre-trained representation $D^{(B)} \in R^{p \times 2H}$ where p means maximum length of each input document of BERT. The hidden state $h_t \in R^H$ is randomized. The specific equations are shown as follows:

$$\overrightarrow{D_t^{(B)}} = LSTM(\overrightarrow{D_{t-1}^{(B)}}, h_t) \tag{2}$$

$$\overleftarrow{D_t^{(B)}} = LSTM(\overleftarrow{D_{t-1}^{(B)}}, h_t) \tag{3}$$

$$D_t^{(B)} = [\overrightarrow{D_t^{(B)}}; \overleftarrow{D_t^{(B)}}] \tag{4}$$

$$D^{(B)} = \{D_t^{(B)}\}_{t=1}^{T} \tag{5}$$

3.5 Label embeddings layer

For the reason that each label contains latent semantic information besides documents, we convert labels $L = \{l_i | l_i = \{l^1, l^2, \cdots, l^m\}$ to embedding vectors $L^{(G)} \in R^{M \times d}$ via GloVe (Pennington et al., 2014) with M representing the total number of labels , fully establishing contextual relationship among labels.

$$L^{(G)} = Embedding_{label}(L) \in R^{M \times d} \tag{6}$$

3.6 Doc-label attention layer

In the MLTC task, a single document belongs to several labels and vice versa, so it's intuitive and vital to capture interactive features between documents and their corresponding labels. Suppose two sentences "The clock in the high church tower struck and the sound made him remember his parents' early love for him." and "Dad was always there to play the mandolin for his family, sacrificing his time and efforts to see that his family had enough in their life.", the former can be summarized as "clock striking" and "family affection" while the latter belongs to "instrument playing" and "family affection", both belonging to two labels and "family affection" being able to be distributed to the above-mentioned two sentences. Therefore, we adopt a doc-label attention mechanism to fuse information between documents and labels. The details can be described as follows:

Firstly, we apply self-attention mechanism on documents to obtain an independent weight vector λ which implies contribution of documents in doc-label pairs:

$$A_D = softmax(W_1' tanh(W_1 {D^{(B)}}^T)) \tag{7}$$

$$\lambda = \sigma((A_D D^{(B)}) W_1'') \tag{8}$$

Then we apply doc-label attention mechanism to get attention label representation $L^{(A)}$ and its independent weight vector μ:

$$A_L = (W_2 L^{(G)})(W_2^{'} D^{(B)^T}) \tag{9}$$

$$L^{(A)} = A_L D^{(B)} \tag{10}$$

$$\mu = \sigma(L^{(A)} W_2^{''}) \tag{11}$$

The final doc-label representation $S^{(A)}$ is calculated by multiplying dependent label weight vector μ_{dep} via normalization:

$$\mu_{dep} = \frac{\mu}{\mu + \lambda} \tag{12}$$

$$S^{(A)} = \mu_{dep} L^{(A)} \tag{13}$$

Here, W_1, $W_1^{'}$, $W_1^{''}$, W_2, $W_2^{'}$, $W_2^{''}$ are trainable parameters. σ is sigmoid activation function (the same below).

3.7 Prediction layer

Finally, a MLP classifier in the prediction layer is used for the final doc-label representation $S^{(A)}$ to make multi-label text classification:

$$\hat{y} = \sigma(W_p^{'} tanh(W_p S^{(A)})) \tag{14}$$

where W_p, $W_p^{'}$ are trainable parameters.

We adopt cross-entropy loss as the loss function in our work which has been proved suitable for the MLTC task [30]:

$$\begin{aligned} \underset{\Theta}{min} \sum_{i=1}^{N} \sum_{j=1}^{M} y^{(ij)} \log(\sigma(\hat{y}^{(ij)})) \\ +(1 - y^{(ij)}) \log(1 - \sigma(\hat{y}^{(ij)})) \end{aligned} \tag{15}$$

where $y^{(ij)} \in \{0, 1\}$ denotes the j-th ground truth label of the i-th document while $\hat{y}^{(ij)} \in [0, 1]$ indicates the predicted probability of the above-mentioned doc-label pairs.

4 Experiments setup

4.1 Datasets

In this research, we utilize three multi-label text datasets with the detailed statistics shown in Table 1. Specifically, W, N_{train}, N_{test} and M denote the number of total words, training documents, test documents and total unique labels, respectively.

RCV1-V2 (Lewis et al., 2004) contains 804,414 newswire stories, including 643,531 training documents and 160,883 test ones. Each story belongs to several topics with the total number of labels 103.

Dataset	W	N_{train}	N_{test}	M
RCV1-V2	47,236	23,149	781,265	103
AAPD	69,399	54,840	1,000	54
Reuters-21578	18,637	8,630	2,158	90

Table 1: Statistics of three datasets

Reuters-21578 is a collection of 10,788 documents and 90 labels from Reuters News Wire in 1987 with 8,630 for training and 2,158 for testing.

AAPD (Yang et al., 2018) is a combination of 55,840 abstracts and their corresponding topics in the field of computer science from Arxiv in 2018, which consists of 54,840 abstracts as training data and 1,000 ones as test data.

4.2 Baseline

We compare our proposed model with the following nine benchmarks:

BR (Boutell et al., 2004) establishes multiple binary classifiers for each label, ignoring dependency between labels.

CC (Read et al., 2011) converts the MLTC task into a chain of binary classification problems with consideration of high-order label correlation.

LP (Tsoumakas and Katakis, 2006) creats a multi-class classifier for all unique label combinations.

CNN (Kim, 2014) adopts multiple convolution kernals to extract contextual information with activation function to ouput probability distribution.

CNN-RNN (Chen et al., 2017) utilizes a combination of CNN and RNN to capture global and local semantic features as well as label correlation.

S2S (Sutskever et al., 2014) is the pure sequence-to-sequence model which can be used on the MLTC task.

S2S+Attn (Bahdanau et al., 2015) adds attention mechanism on the basis of RNN-oriented Seq2Seq model.

SGM (Yang et al., 2018) is a label sequence generation model with attention mechanism to solve the MLTC task based on LSTM-oriented Seq2Seq model.

MDC (Lin et al., 2018) uses hybrid attention based on LSTM-oriented Seq2Seq model to capture information-enhanced features.

4.3 Evaluation metrics

Inspired by the previous work (Zhang and Zhou, 2007; Chen et al., 2017), we evaluate our proposed model and other nine benchmarks with Hamming Loss, micro-Precision, micro-Recall and micro-F1.

Hamming loss (HL) calculates the percentage of mislabeled documents whose predicted labels are not adequate or irrelevant.

micro-Precision (mP) interprets global precision with True Positives and False Positives of the i-th given label, i.e., FP_i and TP_i.

micro-Recall (mR) describes global recall with True Positives and False Negatives of the i-th given label, i.e., FP_i and FN_i.

micro-F1 (mF1) weights the global precision and recall of the total categories which can be represented as follows:

4.4 Hyper parameters and training

We carry out our experiments on NVIDIA TESLA V100 GPU with Pytorch. In the BERTdoc layer and the label embeddings layer, we set the maximum length of each document as 500 in the pre-training process with BERT and adjusted the embedding size of labels as 300. As for the doc encoder layer, the dimension of hidden state in BiLSTM is set to 300. When it comes to training process, we use Adam optimizer with β_1 = 0.9 and β_2 = 0.999. The batch size is adjusted to 128 and the learning rate is initialized to 0.0001. We evaluate model performance on test sets after 200 epochs with early stopping when the validation loss stops decreasing by 10 epochs.

5 Experimental results

5.1 Model comparison

We compare our proposed HdLAN model with other nine benchmarks on three datasets evaluated with HL, mP, mR, $mF1$ shown in Table 2, Table 3 and Table 4. Due to unreported results of some models on Reuters-21578, we only use three benchmarks for comparison, i.e., BR, CC, CNN-RNN, whose results are complete in the above-mentioned four evaluation metrics. Moreover, $(+)$ in Table 2, Table 3 and Table 4 means the higher the value is, the better performance of the model, such as mP, mR and $mF1$ while $(-)$ indicates the opposite, such as HL.

The nine benchmarks can be divided into three categories referred to as machine learning methods (i.e., BR, CC, LP), conventional deep learning models (i.e., CNN, CNN-RNN) and Seq2Seq-based approaches (i.e., S2S, S2S+Attn, SGM, MDC). For the reason that a good portion of model results of are missing on Reuters-21578, we prominently make analysis on RV1-V2 and AAPD. As shown in Table 2 and Table 3, we can see that generally conventional deep learning methods outperform machine learning models, which strongly demonstrates conventional deep learning model are superior in extracting deep semantic information than feature-engineering-driven traditional machine learning methods dependent on burdensome handcrafts. Surprisingly, CNN performs best on the above-mentioned two datasets with mP possibly due to the function of convolution kernels which exactly manage to capture accurate features but needing validation on more datasets.

A milestone of the MLTC task is sequence-to-sequence (Seq2Seq) model, followed by a bundle of Seq2Seq-based models like S2S+Attn, SGM, MDC, etc. The average results of Seq2Seq-based models show an advantage over that of conventional deep learning models, undoubtedly indicating that Seq2Seq-based models are capable of exploring latent label orders with global embedding which beat conventional deep learning solutions overwhelmingly. Akin to the comparison between conventional deep learning models and machine learning methods, conventional deep learning models perform just plain better than Seq2Seq-based models with mp on these two datasets, which needs more corpora for interpretation. Moreover, MDC is the state-out-of-art solution which applies attention mechanism based on Seq2Seq model suitable for creating information-enhanced contextual representations.

Most importantly, the experiment results show that our model HdLAN has the best performance on all three datasets, outperforming the current state-of-the-art model MDC on RV1-V2 and AAPD, meanwhile defeating all reported models on Reuters-21578 ($P < 0.05$ on student t-test for all above comparisons, the same below), which can be attributed to the pre-training process on documents with BERT, encoding of documents, label embeddings and doc-label attention mechanism.

5.2 Ablation study

To analyze the contributions of each component of our proposed model, we carry out ablation study of

Datasets	**RCV1-V2**			
Metrics	**HL(-)**	**mP(+)**	**mR(+)**	**mF1(+)**
BR	0.0086	0.904	0.816	0.858
CC	0.0087	0.887	0.828	0.857
LP	0.0087	0.896	0.824	0.858
CNN	0.0089	**0.922**	0.798	0.855
CNN-RNN	0.0085	0.889	0.825	0.856
S2S	0.0082	0.883	0.849	0.866
S2S+Attn	0.0081	0.889	0.848	0.868
SGM	0.0075	0.897	0.860	0.878
MDC	**0.0072**	0.891	**0.873**	**0.882**
HdLAN	**0.0068**	**0.925**	**0.894**	**0.909**

Table 2: Comparisons of ten models on RCV1-V2

Datasets	**AAPD**			
Metrics	**HL(-)**	**mP(+)**	**mR(+)**	**mF1(+)**
BR	0.0316	0.664	0.648	0.646
CC	0.0306	0.657	0.651	0.654
LP	0.0323	0.662	0.608	0.634
CNN	0.0256	**0.849**	0.545	0.664
CNN-RNN	0.0280	0.718	0.618	0.664
S2S	0.0255	0.743	0.646	0.691
S2S+Attn	0.0261	0.720	0.639	0.677
SGM	0.0245	0.748	0.675	0.710
MDC	**0.0240**	0.752	**0.681**	**0.715**
HdLAN	**0.0236**	**0.822**	**0.674**	**0.741**

Table 3: Comparisons of ten models on AAPD

Datasets	**Reuters-21578**			
Metrics	**HL(-)**	**mP(+)**	**mR(+)**	**mF1(+)**
BR	0.0032	**0.940**	0.823	0.878
CC	**0.0031**	0.937	**0.828**	**0.879**
CNN-RNN	0.0038	0.902	0.813	0.855
HdLAN	**0.0025**	**0.974**	**0.877**	**0.923**

Table 4: Comparisons of four models on Reuters-21578

five derived models which remove or change any layer on RV1-V2 shown in Table 5. Because of similar tendency on the other two datasets, we only take results on RV1-V2 as an example.

Specifically, w/o BERTdoc and BERTdoc to EMBdoc represents derived models without pre-training on documents with BERT and applying traditional GloVe technique to establish document embeddings instead of BERT, respectively, both affecting the performance compared with proposed HdLAN model by a wide margin, which indicate the powerful capabilities of BERT in capturing deep semantic information. With multiple embeddings such as token embeddings, segment embeddings and position embeddings as well as transformers containing multi-head attention, the pre-training model BERT manages to extract global semantic information of documents undoubtedly. When we remove the doc-label attention layer away from the final model named w/o Doc-label attention, the results also decrease, demonstrating its function of extracting interactive features between documents and their corresponding labels via establishing contextual connection of the two parts, which also clarifies attention mechanism is able to model long sequences, fully finding semantic interaction of document-label pairs at any distance. W/o Label embeddings means feeding only encoded pre-trained documents to the doc-label attention layer without label embeddings, which also has a negative effect on model performance, because label embeddings take all unique labels into consideration, establishing relationship among labels which aims at exploring latent combinations of labels corresponding to given documents. For the derived model without BiLSTM encoder for documents named w/o Doc encoder, we can see that the performance has also a large distance with the proposed HdLAN model, possibly because the doc encoder layer further takes the contextual information of documents into consideration, enhancing the global semantic interaction.

Above all, each component of the proposed model BdLAN has indispensable abilities separately and the organic combination of these layers jointly make tremendous contributions to its state-of-the-art performance.

5.3 Parameters sensitivity

To increase the robustness of our proposed BdLAN model, we carry out a series of experiments to an-

Datasets	**RCV1-V2**			
Metrics	**HL(-)**	**mP(+)**	**mR(+)**	**mF1(+)**
w/o BERTdoc	0.0083	0.8662	0.8247	0.8856
w/o Doc encoder	0.0091	0.8547	0.8365	0.8455
w/o Label embeddings	0.0088	0.8763	0.8572	0.8666
w/o Doc-label attention	0.0086	0.8985	0.8163	0.8554
BERTdoc to EMBdoc	0.0076	0.9175	0.8456	0.8801
HdLAN (ours)	**0.0068**	**0.9248**	**0.8942**	**0.9092**

Table 5: Ablation study of five derived models on RCV1-V2

alyze the impact of the length of input documents in the BERTdoc layer and the dimension of hidden state in the doc encoder layer of our proposed BdLAN model on the RV1-V2 dataset with results shown in Figure 2. Due to the similar trend of parameters on three above-mentioned datasets, we just take one as an example.

Figure 2 shows the turning points of dimension of hidden state (Figure 2(a)) is 300 both on the training set and test set, the larger of the dimension in the doc encoder layer when less than 300, the better performance of the proposed model achieving. When it comes to the effect of document length on the proposed model (Figure 2(b)), there are two peaks at 250 and 500, respectively, indicating that BERT manages to capture more significant information when it learns from more input documents within acceptable limits.

5.4 Case study

Next, we make a case study to further interpret how to classify multi-label documents with our proposed model. Take a certain document from AAPD dataset labeled $cs.sy$ and $math.oc$ as an example with detailed content shown in Figure 3.

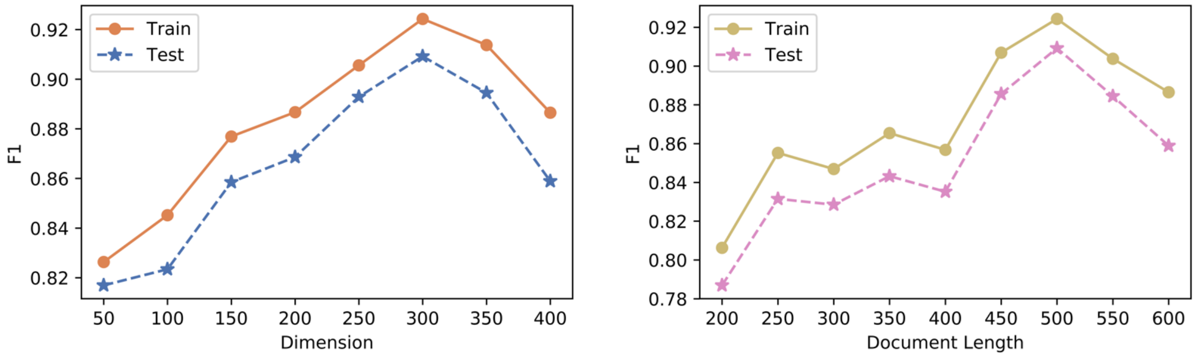
Above all, we aim to explore different contributions of each word to the whole document displayed in color according to its belonging labels $cs.sy$ and $math.oc$, respectively. For the first label $cs.sy$, it's not difficult to find that words such as $systems$, $engineers$ and their variants covered with deep red facilitate the proposed model to predict the correct category while words like $operation$, $dynamical$ and their different forms with less deep red also motivate multi-label text classification, catering for human perception. High contribution words to the second label $math.oc$ such as $dynamical$, $convex$ as well as less high correlation words like $correct$, $engineers$ are also conducive for predicting the target label from human perspective.

Next, we reveal different probabilities of all unique labels calculated by our proposed BdLAN through a heatmap shown in Figure 4 with the probabilities of correct labels $cs.sy$ and $math.oc$ obtaining 0.85 and 0.88 which substantially exceed other labels averaged by 0.2 to 0.7. Furthermore, some less related labels prefixed by cs and $math$ have a probabilities between 0.4 and 0.7 while other almost irrelevant labels such as $physics.soc-ph$, $q-bio.nc$ only occupy 0.2 to 0.3.

From this concrete example, it's intuitionistic for researchers of Natural Language Processing to clarify the mechanism within our propose BdLAN model on how to classify multi-label documents into multiple categories.

6 Conclusion

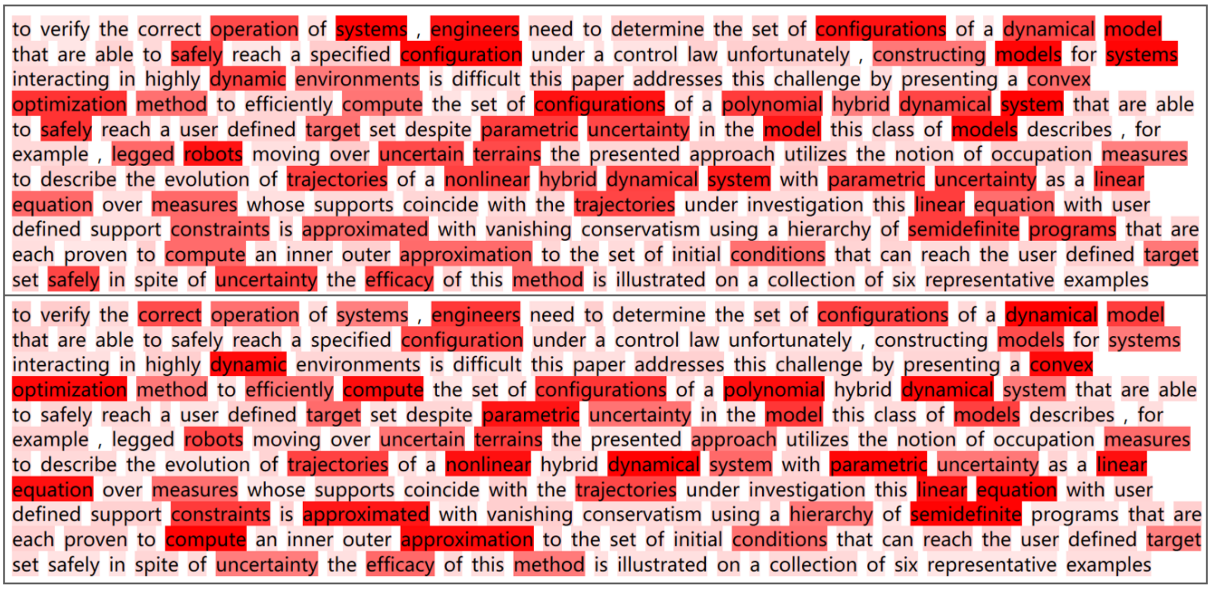
MLTC is a great challenge in text classification. To automate the multi-label text classification progress, we propose a novel solution named BdLAN, BERTdoc Label Attention Networks. We use BERT to pre-train documents and adopt BiLSTM to explore the contextual information of documents forward and backward. Next, we ultilize GloVe to construct label embeddings, applying doc-label attention mechanism to obtain interactive information between documents and labels, followed by a MLP classifier to make final prediction. We carry out experiments on three datasets with four common evaluation metrics, the results indicating our proposed model outperforms all state-of-the-art MLTC models with a case study further visualizing its applications. In the future, we will generalize our model with more datasets to increase its robustness and enlarge its applications in more scenarios.



(a)

(b)

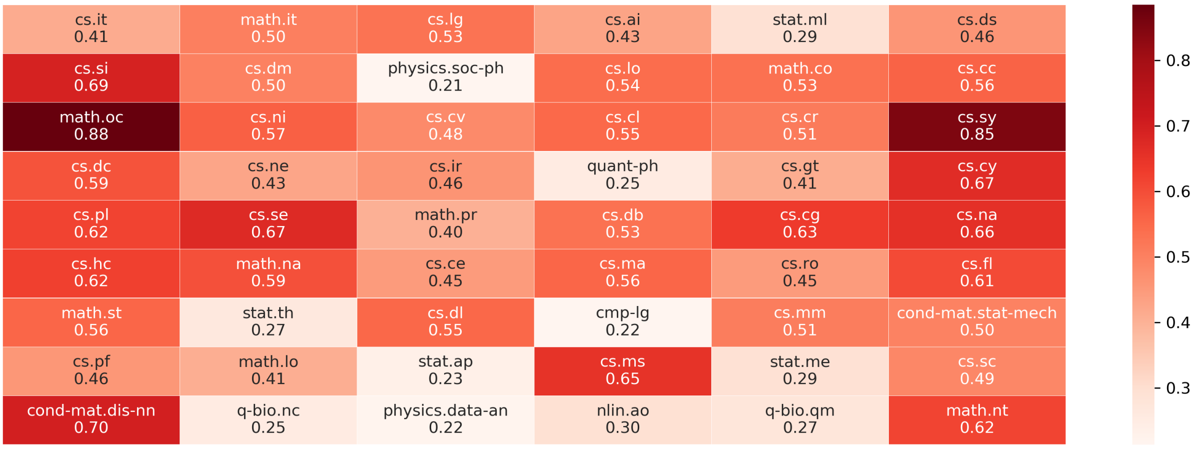
Figure 2: Influence of dimension of hidden state and document length on the RV1-V2 dataset



to verify the correct operation of systems , engineers need to determine the set of configurations of a dynamical model that are able to safely reach a specified configuration under a control law unfortunately , constructing models for systems interacting in highly dynamic environments is difficult this paper addresses this challenge by presenting a convex optimization method to efficiently compute the set of configurations of a polynomial hybrid dynamical system that are able to safely reach a user defined target set despite parametric uncertainty in the model this class of models describes , for example , legged robots moving over uncertain terrains the presented approach utilizes the notion of occupation measures to describe the evolution of trajectories of a nonlinear hybrid dynamical system with parametric uncertainty as a linear equation over measures whose supports coincide with the trajectories under investigation this linear equation with user defined support constraints is approximated with vanishing conservatism using a hierarchy of semidefinite programs that are each proven to compute an inner outer approximation to the set of initial conditions that can reach the user defined target set safely in spite of uncertainty the efficacy of this method is illustrated on a collection of six representative examples

to verify the correct operation of systems , engineers need to determine the set of configurations of a dynamical model that are able to safely reach a specified configuration under a control law unfortunately , constructing models for systems interacting in highly dynamic environments is difficult this paper addresses this challenge by presenting a convex optimization method to efficiently compute the set of configurations of a polynomial hybrid dynamical system that are able to safely reach a user defined target set despite parametric uncertainty in the model this class of models describes , for example , legged robots moving over uncertain terrains the presented approach utilizes the notion of occupation measures to describe the evolution of trajectories of a nonlinear hybrid dynamical system with parametric uncertainty as a linear equation over measures whose supports coincide with the trajectories under investigation this linear equation with user defined support constraints is approximated with vanishing conservatism using a hierarchy of semidefinite programs that are each proven to compute an inner outer approximation to the set of initial conditions that can reach the user defined target set safely in spite of uncertainty the efficacy of this method is illustrated on a collection of six representative examples

Figure 3: Visual analysis of our proposed model on a MLTC task with label $cs.sy$ (above) and $math.oc$ (below)



cs.it 0.41	math.it 0.50	cs.lg 0.53	cs.ai 0.43	stat.ml 0.29	cs.ds 0.46
cs.si 0.69	cs.dm 0.50	physics.soc-ph 0.21	cs.lo 0.54	math.co 0.53	cs.cc 0.56
math.oc 0.88	cs.ni 0.57	cs.cv 0.48	cs.cl 0.55	cs.cr 0.51	cs.sy 0.85
cs.dc 0.59	cs.ne 0.43	cs.ir 0.46	quant-ph 0.25	cs.gt 0.41	cs.cy 0.67
cs.pl 0.62	cs.se 0.67	math.pr 0.40	cs.db 0.53	cs.cg 0.63	cs.na 0.66
cs.hc 0.62	math.na 0.59	cs.ce 0.45	cs.ma 0.56	cs.ro 0.45	cs.fl 0.61
math.st 0.56	stat.th 0.27	cs.dl 0.55	cmp-lg 0.22	cs.mm 0.51	cond-mat.stat-mech 0.50
cs.pf 0.46	math.lo 0.41	stat.ap 0.23	cs.ms 0.65	stat.me 0.29	cs.sc 0.49
cond-mat.dis-nn 0.70	q-bio.nc 0.25	physics.data-an 0.22	nlin.ao 0.30	q-bio.qm 0.27	math.nt 0.62

Figure 4: Weights of all labels of the given document

References

Clare A. and King R. 2001. Knowledge discovery in multi-label phenotype data. In *European Conference on Principles of Data Mining and Knowledge Discovery*, page 42–53. Springer.

Elisseeff A. and Weston J. 2002. A kernel method for multi-labelled classification. In *Advances in neural information processing systems*, page 681–687.

D. Bahdanau, K. Cho, and Y. Bengio. 2015. Neural machine translation by jointly learning to align and translate. In *Proc. Int. Conf. Learn. Represent.*

M. R. Boutell, J. Luo, X. Shen, and C. M Brown. 2004. Learning multi-label scene classification.

R. Cerri, R. C. Barros, A. C. de Carvalho, and Y. Jin. 2016. Reduction strategies for hierarchical multi-label classification in protein function prediction.

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 *International Joint Conference on Neural Networks*, page 2377–2383, Anchorage, AK, USA. IJCNN.

Bahdanau D., Brakel P., Xu K., Goyal A., Lowe R., Pineau J., Courville A., and Bengio Y. 2017. An actorcritic algorithm for sequence prediction. In *ICLR*.

Nadia Ghamrawi and Andrew McCallum. 2005. Collective multi-label classification. In *Proceedings of the 14th ACM international conference on Information and knowledge management*, pages 195–200. ACM.

Katakis I., Tsoumakas G., and Vlahavas I. 2008. Multilabel text classification for automated tag suggestion. In *Proceedings of the ECML/PKDD*.

Nam J., Kim J., Gurevych I., and Furnkranz J. 2014. Large-scale multi-label text classification — revisiting neural networks. In *Proceedings of the 2014th European Conference on Machine Learning and Knowledge Discovery in Databases*, Volume Part II, page 437–452.

Yoon Kim. 2014. Convolutional neural networks for sentence classification. In *Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing*. EMNLP.

Gakuto Kurata, Bing Xiang, and Bowen Zhou. 2016. Improved neural network-based multilabel classification with better initialization leveraging label cooccurrence. In *Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, page 521–526, San Diego California, USA. NAACL HLT.

David D Lewis, Yiming Yang, Tony G Rose, and Fan Li. 2004. Rcv1: A new benchmark collection for text categorization research.

C. Li, B. Wang, V. Pavlu, and J. A. Aslam. 2016. Conditional bernoulli mixturesfor multi-label classification. In *Proc. Int. Conf. Mach. Learn.*, page 2482–2491.

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*, page 835–839.

J. Lin, X. Su, P. Yang, S. Ma, and Q. Su. 2018. Semantic-unit-based dilated convolution for multilabel text classification. In *Proc. Empirical Methods Natural Lang. Process.*, page 4554–4564. IJCNN.

Jeffrey Pennington, Richard Socher, and Christopher Manning. 2014. Glove: Global vectors for word representation. In *Proceedings of the 2014 conference on empirical methods in natural language processing (EMNLP)*, page 1532–1543.

Jesse Read, Bernhard Pfahringer, Geoff Holmes, and Eibe Frank. 2011. Classifier chains for multi-label classification.

Gopal S. and Yang Y. 2010. Multilabel classification with meta-level features. In *Proceedings of the 33rdinternational ACM SIGIR conference on Research and development in information retrieval*, page 315–322. ACM.

I. Sutskever, O. Vinyals, and Q. V. Le. 2014. Sequence to sequence learning with neural networks. In *Proc. Neural Inf. Process. Syst.*, page 3104–3112.

G. Tsoumakas and I Katakis. 2006. Multi-label classification: An overview.

C. Vens, J. Struyf, L. Schietgat, S. Džeroski, and H. Blockeel. 2008. Decision trees for hierarchical multi-label classification.

P. Yang, X. Su, W. Li, S. Ma, W. Wu, and H. Wang. 2018. Sgm: Sequence generation model for multilabel classification. In *Proc. Int. Conf. Comput.Linguistics*, page 3915–3926.

Min-Ling Zhang and Zhi-Hua Zhou. 2006. Multilabel neural networks with applications to functional genomics and text categorization.

Min-Ling Zhang and Zhi-Hua Zhou. 2007. Ml-knn: A lazy learning approach to multilabel learning.