# Multi-Aspect co-Attentional Collaborative Filtering for Extreme Multi-label Text Classification

Anonymous ACL submission

## Abstract

This work proposes a general and effective architecture for the extreme multi-label text classification (XMTC), and reformate the learning task to an interaction function between document and label. Recently, there are many studies trying to enhance text representation or reduce the number of labels to optimize the problem of lack of information in a text or the sparsity of the possibility vector. In the field of recommendation, a similar problem is already defined and studied for a quite long time. It is worthy to learn methods from recommendation to XMTC for finding matching relations in large size of dataset accurately. With co-attention mechanism and neural collaborative filtering, we not only learn informative label representation enhanced by document-specific label group vector and label-specific text feature vector but also build an effective interaction function to get matching score. After extensive comparison experiments with various models, results demonstrate the architecture we proposed outperforms most of the methods and achieves significant improvement on basic document encoders.

## 1 Introduction

Text classification is one of the fundamental tasks in natural language processing (NLP). There is a wide range of application scenarios such as sentiment analysis, news filtering, web page tagging, and so on. Normally researchers can extract features of text by a convolutional neural network (CNN), recurrent neural network (RNN). Since 2018, a large number of pre-trained models such as ELMO, BERT have shown an outstanding performance in several tasks in NLP. Recently, with the growth of data scale, multi-label text classification (MLTC) has attracted more attention, since automatically labeling multiple labels of documents can effectively reduce labor costs. To distinguish from multi-class classification, MLTC specifically refers to classification tasks where the text has multiple labels, rather than choosing one from multiple possible candidate labels. For MLTC, in most cases, we convert this task into several binary classification problems on each label.

However, there is still no good solution for extreme multi-label text classification (XMTC) which is described as text with its most relevant multiple labels from an extremely large-scale label set (You et al., 2018). Different from regular MLTC tasks, because of the large space of possible labels in XMTC, expanding the dimension of the output vector will result in unnecessary computational costs in time and space.

Existing studies for XMTC mainly focus on learning enhanced documents (Liu et al., 2017) and modeling label dependency (Zhang et al., 2018) to optimize this problem. Although we can utilize various models to explore information from the content of documents or label correlations, existing works still focus on mining obtaining more information to optimize a multi-label cross-entropy after a fully connected (FC) layer (Xun et al., 2020). It is straightforward and easy to understand, but this kind of method has some backward. Firstly, the correlations among labels are not reflected from it, while the relationship between labels in a big-size database is informative. Secondly, also because of the number of labels, it is hard to precisely map the feature of text to several positions within a large solution space.

To resolve these problems, inspired by some solutions in the recommender system, which is always trying to retrieve information and establish mapping relations between different types of entities in large size database, we consider reformatting the objective of XMTC to learn an interaction function between document and label by a general learning architecture: Multi-Aspect co-Attentional Collaborative Filtering Plus (MAACF+). we model features of text and label with co-attention mechanism and learn an interaction function by Neural Collaborative Filtering (NCF) (He et al., 2017) . Plus means an alternative document encoder, any kind of text encoder can be ensembled into MAACF+. In our architecture, an informative feature of text can be modeled by an advanced document encoder and considering information in the label group. Meanwhile, to enhance the representation of the label, a high co-occurrence label group is introduced and fuse with the attention mechanism. Finally, we exploit the NCF component to simulate the interaction function learning. We evaluate our architecture on three public real-world XTMC datasets, and the results illustrate its effectiveness and verify the significance of each component.

We conclude the contributions in this paper as follows:

- We propose a novel architecture MAACF+ to learn an interaction function to model the relations between labels and documents from a large space of labels. It utilizes multiple co-attention mechanisms to extract information from the label statistical nearest group and enhance text representation.
- MAACF+ is a general and independent architecture that can be integrated with any document encoder without changing other parts of the model.
- Extensive experiments and visualization on one Multi-lingual benchmark dataset and two English datasets illustrate the effectiveness of MAACF+ based on three popular text encoders in XMTC task: XMLCNN, BiLSTM, BERT. Compared with other state-of-the-art models, MAACF+BERT can outperform them in most of indicators. Besides, results also confirm the success and necessity of introducing the co-occurrence label group.

## 2 Model

The architecture of our proposed model is presented in Figure 1. The MAACF+ is composed of three main components: 1) label-group information extractor 2) multiple document encoders 3) neural collaborative filter. To be more specific, the label-group information extractor aims to explore statistical features and supplement information from the label's neighbor group. The setting of multi-encoders is expected to capture multi-modal distribution in label groups, which is similar to the previous exploration in social recommendation (Wang et al., 2021).

### 2.1 Problem Formulation

For input set $\mathcal{I} = (X, Z, G)^{|N|}$, $X$ consists of $N$ document $\boldsymbol{x_i}$ and $Z$ is labels $\boldsymbol{x_i}$ interacted with which has $\boldsymbol{z_i} \in \{0,1\}^{|\mathrm{C}|}$, where C is the total number of labels. Differ to other works in multi-label text classification, we define an additional matrix $\mathcal{G}^{|C|\times|\mathrm{K}|}$, where $K$ indicates the number of statistical nearest neighbors of each label in $Z$. Each document $\boldsymbol{x_i}$ contains L words, and we wish to learn a mapping function between document and most relevant labels.

### 2.2 Architecture of MAACF+

Figure 1 illustrates the overall architecture of our proposed method. For the input, it consists of three components: document $\boldsymbol{x_i}$, label $\boldsymbol{z_i}$ and label group $\boldsymbol{g_i}$. Given $\boldsymbol{x_i}$ with $L$ words, we firstly utilize $n$ document encoders to get $\mathrm{n}$ sequence vector $\boldsymbol{s_{in}} \in \mathcal{R}^{\mathrm{L}\times\mathrm{D}}$, where D is the dimension of sequence vector. Note that, for different basic document encoders, different word embedding approaches are utilized. In our experiments, after preprocessing in text, XMLCNN and BiLSTM use the Continuous Bag-of-Words Model (CBOW) which belongs to word2vec methods to get the embedding vector of each word. And BERT uses its tokenizer to get the hidden vector of each token. Each document encoder gets input in form of vectorized text, then output feature information as a vector $\boldsymbol{s_{in}}$.

Modeling on labels is going in parallel. we input target label $\boldsymbol{z_i}$ and its label group $\boldsymbol{g_i}$ into the label encoder. Firstly, for each label, we get its hidden feature vector by embedding layer. Then for the label group, the same number of attention layers are employed to generate comprehensive but exclusive representation $\boldsymbol{h_{in}}$ with dimension D of label group from different perspectives by querying with each $\boldsymbol{s_{in}}$. The formula is as follows:

$$\alpha_{ink} = \frac{\exp(\boldsymbol{h_{ink}}\boldsymbol{s_{in}^{T}})}{\sum_k \exp(\boldsymbol{h_{ink}}\boldsymbol{s_{in}^{T}})} \tag{1}$$

$$\boldsymbol{h_{in}} = \sum_k \alpha_{ink}\boldsymbol{h_{ink}} \tag{2}$$

where $\alpha_{ink}$ indicates how informative the k-th label friend is for the whole label group $\boldsymbol{g_i}$ in the n-th aspect.

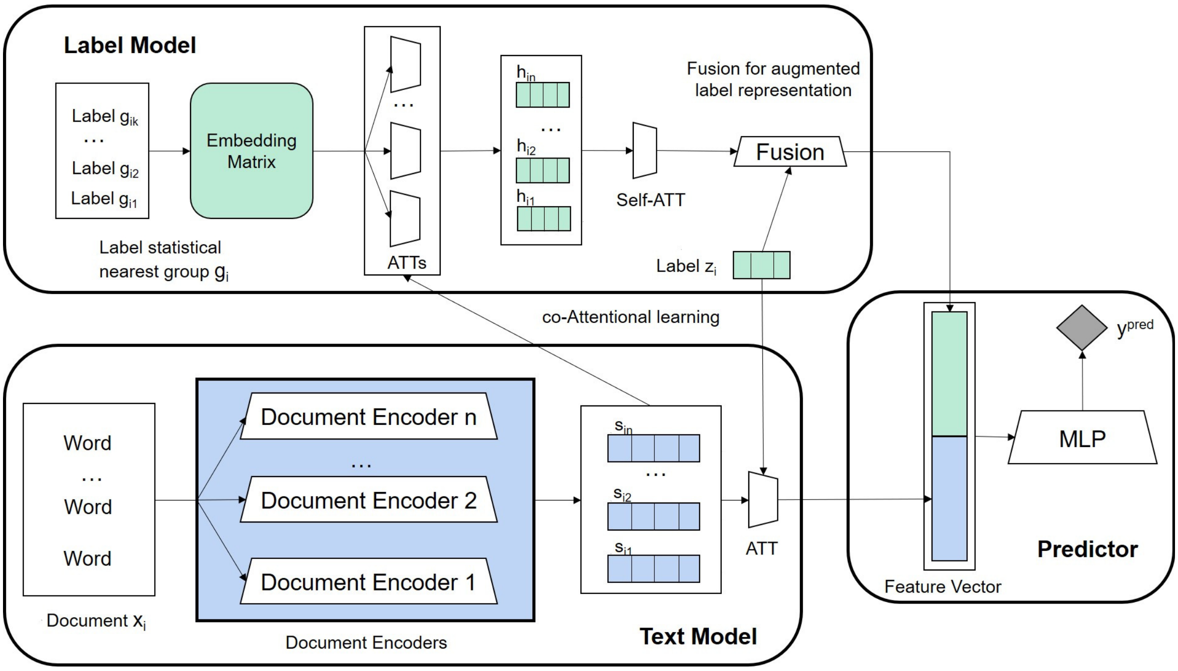


Figure 1: The architecture of proposed MAACF+. Embedding Matrix denotes the matrix consists of representations of the target label's statistical label group. ATT means attention mechanism. In predictor, we utilize a MLP (Multi-Layer Perceptron) to do prediction.

After that, a self-attention layer is used to combine all aspects of information from the label group as $\boldsymbol{H}_{gi}$ in (3)(4). The final representation of label $\boldsymbol{H}_i$ is learned through a linear transformation on a vector of label group and target label. The formula of this linear fusion function is defined as (5).

$$\alpha_{in} = \text{Softmax}(\mathcal{W}^T \boldsymbol{h_i} + b) \tag{3}$$

$$\boldsymbol{H}_{gi} = \sum_n \alpha_{in} \boldsymbol{h_i} \tag{4}$$

$$\boldsymbol{H}_i = \mathcal{W}_f^T [\boldsymbol{h_i}, \boldsymbol{H}_{gi}] + b_f \tag{5}$$

In the document encoder side, the target label's hidden vector $\boldsymbol{h_i}$ is used to compute the label-aware attention values in all of the aspect-level document representations with a similar attention mechanism.

When both the feature of label and document are extracted, I input them into a multi-layer perceptron to get the predicted score $y^{pred}$. The proposed architecture's goal of optimization is minimizing its loss for each document on its most relevant labels and maximizing the loss of negative samples which are randomly selected. It can be formulated as:

$$\min_\theta L_\theta(x, z, g) \tag{6}$$

$$L_\theta = \sum_p^{Pos} y^p \log(\hat{y}^p) + (1 - y^p)\log(1 - \hat{y}^p) + \\ \sum_n^{Neg} y^n \log(\hat{y}^n) + (1 - y^n)\log(1 - \hat{y}^n) \tag{7}$$

## 3 Experiment

### 3.1 Experiment Setting

#### 3.1.1 Datasets

We evaluate the proposed architecture on one multi-lingual benchmark dataset EUR-Lex (Mencia and Fürnkranz, 2008a) and two English benchmark datasets: AAPD (Yang et al., 2018) and AmazonCat-13K (McAuley and Leskovec, 2013). The detailed description of the dataset is shown in Table 1.

#### 3.1.2 Evaluation Metrics

We reformate the matching problem to the top-k rank problem, so we utilize precision at topK (P@K) and Normalized Discounted Cumulated Gains (NDCG) at topK (N@K) for evaluation. Specifically, P@K measures the precision of predicted matching relations between label and text within K highest possible candidates. And NDCG@K indicates the result of the order of

| Dataset | $N_{\text{train}}$ | $N_{\text{test}}$ | $D$ | $L$ | $\bar{L}$ | $\bar{L}$ | $\bar{W}_{\text{train}}$ | $\bar{W}_{\text{test}}$ |
|---|---|---|---|---|---|---|---|---|
| EUR-Lex | 15,449 | 3,865 | 186,104 | 3,956 | 5.30 | 20.79 | 1248.58 | 1230.40 |
| AAPD | $54,840$ | 1,000 | $69,399$ | 54 | 2.41 | 2444.04 | 163.42 | 171.65 |
| AmazonCat-13K | $1,186,239$ | 306,782 | 203,882 | 13,330 | 5.04 | 448.57 | 246.61 | 245.98 |

Table 1: Dataset statistics, $N_{train}$ and $N_{test}$ denote the number of documents in train and test sets respectively. $D$ is the vocabulary size of the input text. $L$ is the number of labels, $\bar{L}$ is the average number of labels for each document, $\widetilde{L}$ is the the average number of instances for each label. $\bar{W}_{\text{train}}$ and $\bar{W}_{\text{test}}$ are the number of words in each train and test document.

rank(Yu et al., 2018) .

$$Precision@K = \frac{\sum_{i=1}^{N} Hit_i@K}{\sum_{i=1}^{N} NumInTest} \quad (8)$$

$$DCG@K = \sum_{j=1}^{K} \frac{2^{rel_j} - 1}{\log(j+1)} \quad (9)$$

$$NDCG@K = \frac{DCG@K}{IDCG@K} \quad (10)$$

#### 3.1.3 Implementation Details

Based on the design of fair comparison, we split the dataset in the same way as its publisher. Considering the implementation of different document encoders, we use two methods to preprocess documents. For XMLCNN and BiLSTM, word2vec models (Mikolov et al., 2013) are used to initiate 256-dimensional word vectors. For BERT we use its tokenizer. To satisfy the input requirement of the attention mechanism, we align the dimension of representation for both label and document to 256. we select the Adam optimizer method (Kingma and Ba, 2014) to minimize the binary cross-entropy loss, and a negative sample strategy is deployed for training. In case of over-fitting, an early-stop strategy and partial freeze on the document encoder are used as well.

### 3.2 Experiments Contents

#### 3.2.1 Baselines

To evaluate the performance of our proposed architecture, we compared it with several classical or state-of-the-art models on XMTC. Note that, to show the outstanding performance of our model, except for some models that are not open source or no testing on datasets we used, we directly cited the results from their papers. In addition to that, we did a wide range of ablation experiments to validate the significance of each component of our architecture.

Basic document encoders:

- XMLCNN (Liu et al., 2017): a CNN-based model with dynamic pooling to capture high-level features of document for XMTC.
- BiLSTM(Cornegruta et al., 2016): A basic BiLSTM model with a self-attention layer to get a representation of text. It is a widely used sequence encoder.
- BERT (Kenton and Toutanova, 2019): One layer bi-direction Encoder Representation from Transformers. We use different pre-trained weights to refer to the language of the dataset.

Advanced comprehensive models:

- DXML (Zhang et al., 2018): It uses deep metric learning to learn the embedding of text and uses the graph representation learning method to learn the embedding of the label.
- AttentionXML (You et al., 2018): A label tree-based model with multi-label attention to exploring most informative words in the text.
- LSAN (Xiao et al., 2019): A label-specific attention network to build multiple text representations and adaptive fusion them using a self-attention mechanism.
- CorNetAttentionXML (Xun et al., 2020): An architecture with AttentionXML as text encoder that able to exploit the correlation information among different labels.
- LDGN (Ma et al., 2021): A model using graph network to extract label-specific components from text and internal interaction among these components.

For each model or variant, to get their best performance, we choose the scores from their best parameters after lots of experiments.

|  | EUR-Lex |  |  |  |  | AAPD |  |  |  |  |
|---|---|---|---|---|---|---|---|---|---|---|
| Model | P@1 | P@3 | P@5 | N@3 | N@5 | P@1 | P@3 | P@5 | N@3 | N@5 |
| XMLCNN | 70.40 | 54.98 | 44.86 | 58.62 | 53.10 | 74.38 | 53.84 | 37.79 | 71.12 | 75.93 |
| MAACF+XMLCNN | 72.92 | 60.67 | 49.71 | 64.93 | 59.33 | 76.27 | 55.23 | 38.95 | 72.51 | 78.25 |
| Improvement | 3.55% | 10.2% | 10.7% | 10.2% | 11.7% | 2.54% | 2.58% | 3.07% | 1.95% | 3.06% |
| BiLSTM | 53.51 | 35.97 | 29.83 | 42.71 | 35.72 | 50.12 | 41.74 | 30.83 | 49.17 | 52.35 |
| MAACF+BiLSTM | 70.72 | 53.46 | 43.85 | 63.81 | 52.32 | 73.36 | 56.25 | 40.62 | 71.27 | 75.71 |
| Improvement | 32.2% | 48.8% | 47.0% | 49.4% | 46.5% | 26.2% | 34.8% | 36.2% | 45.0% | 44.7% |
| BERT | 63.51 | 48.37 | 40.24 | 54.81 | 48.92 | 66.82 | 50.61 | 34.40 | 62.17 | 69.25 |
| MAACF+BERT | 81.16 | 65.32 | 54.46 | 73.32 | 68.13 | 87.10 | 62.32 | 42.53 | 84.47 | 86.32 |
| Improvement | 26.9% | 37.3% | 35.8% | 30.1% | 37.6% | 24.8% | 23.1% | 23.6% | 35.9% | 24.7% |

Table 2: Augment test result on EUR-Lex and AAPD

| Model | P@1 | P@3 | P@5 | N@3 | N@5 |
|---|---|---|---|---|---|
| XMLCNN | 92.07 | 75.29 | 60.53 | 87.34 | 84.29 |
| MAACF+XMLCNN | 93.62 | 78.2 | 63.06 | 88.04 | 85.9 |
| Improvement | 1.68% | 3.87% | 4.18% | 0.80% | 1.91% |
| BiLSTM | 68.75 | 52.1 | 42.53 | 57.66 | 55.91 |
| MAACF+BiLSTM | 93.47 | 79.62 | 62.51 | 86.31 | 85.35 |
| Improvement | 35.96% | 52.82% | 46.98% | 49.69% | 52.66% |
| BERT | 74.78 | 65.78 | 57.51 | 73.57 | 68.58 |
| MAACF+BERT | 94.82 | 79.92 | 66.40 | 91.82 | 89.73 |
| Improvement | 26.80% | 21.50% | 15.46% | 24.81% | 30.84% |

Table 3: Augment test result on AmazonCat-13K

### 3.2.2 Augment Test

The results of the three datasets are presented in Tables 2, 3. We calculate each indicator with $k = 1, 3, 5$. As we can observe from Tables 2, 3, MAACF++ can consistently improve the performance of all popular basic document encoders in XMTC in all metrics.

Among all of the basic encoders, XMLCNN has the slightest improvement, and we think it is because of the dynamic pooling mechanism. A complex feature extractor enables the encoder to exploit sufficient and informative features from documents. But with a larger size of label space, MAACF will bring a more significant improvement because there exist more correlations in the label side.

In addition, MAACF+ architecture shows a more significant improvement on metrics with $k = 3$ or 5. We speculate that it is due to the introduction of the label group. Different from $k = 1$, the nearest label group works more often as an anchor to augment the prediction of other possible candidates. Besides, on datasets with the higher average number of labels for each document, the results of $k = 3$ show more promising on metrics than $k = 5$, because the limitation of the real true number of labels of each text is normally less than $k$.

### 3.2.3 Performance Comparison

Tables 4, 5 demonstrate the performance of all state-of-the-art methods on three datasets. To ensure a fair comparison, we cited results from their source paper directly if it is available. Otherwise, for those experiments that have not been done before, we performed them based on their open-source codes if applicable, or versions implemented on our own.

By observing results in Table 3 and 4, we can find that methods that do not utilize label correlations to enhance the learning process of text representation have worse performance. Specifically, on the AAPD dataset, AttentionXML promotes the P@1 of the DXML from 80.54% to 83.02%, the increase is nearly 3.08%. For example, although DXML tries to model information in label space by deep embedding methods, AttentionXML can pay attention to more semantic relevant parts in the document for each label.

But compared with other previous methods exploiting label correlations, LSAN owns a better performance. We think that is because of its mul-

|  | EUR-Lex |  |  |  |  | AAPD |  |  |  |  |
|---|---|---|---|---|---|---|---|---|---|---|
| Model | P@1 | P@3 | P@5 | N@3 | N@5 | P@1 | P@3 | P@5 | N@3 | N@5 |
| DXML | 75.63 | 60.13 | 48.65 | 63.96 | 53.60 | 80.54 | 56.30 | 39.16 | 77.23 | 80.99 |
| AttentionXML | 67.34 | 52.52 | 47.72 | 56.21 | 50.78 | 83.02 | 58.72 | 40.56 | 78.01 | 82.31 |
| LSAN | 79.17 | 64.99 | 53.67 | 68.32 | 62.47 | 85.28 | 61.12 | 41.84 | 80.84 | 84.78 |
| CorNetAttentionXML | 79.02 | 65.49 | 53.94 | 68.92 | 62.97 | 85.71 | 61.55 | 42.50 | 80.31 | 85.73 |
| LDGN | 81.03 | 64.99 | 56.36 | 71.81 | 66.09 | 86.24 | 61.95 | 42.29 | 83.32 | 86.85 |
| MAACF+BERT | **81.16** | **65.32** | **54.46** | **73.32** | **68.13** | **87.10** | **61.32** | **42.53** | **84.47** | **86.32** |

Table 4: Comparison results on EUR-Lex and AAPD.

| Model | P@1 | P@3 | P@5 | N@3 | N@5 |
|---|---|---|---|---|---|
| DXML | 91.05 | 71.86 | 61.32 | 88.29 | 81.12 |
| AttentionXML | 92.12 | 72.15 | 62.71 | 88.56 | 82.37 |
| LSAN | 92.34 | 74.81 | 63.38 | 89.08 | 81.13 |
| CorNetAttentionXML | 92.17 | 74.36 | 63.83 | 89.11 | 84.54 |
| MAACF+BERT | **94.82** | **79.92** | **66.40** | **91.82** | **89.73** |

Table 5: Comparison results on AmazonCat-13K. Restricted by computing resource, experiments on LDGN does not present.

tiple learning space mechanism and considering semantic correlations between text and label simultaneously. Multiple learning space mechanism is helpful to stabilize the adaptive fusion by attention mechanism, and adaptive fusion learns label-specific text representation.

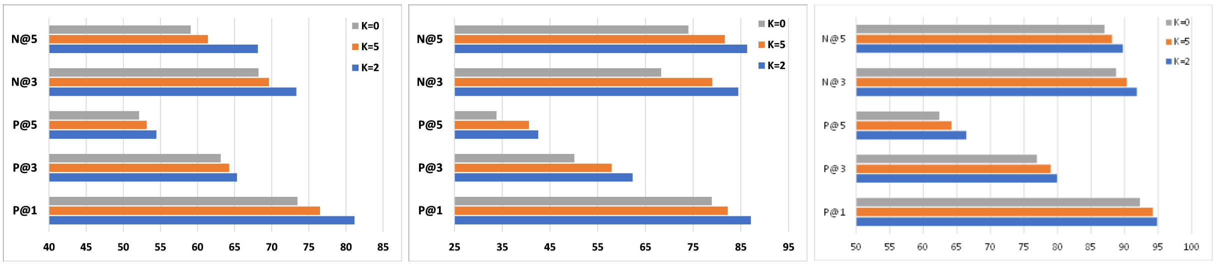
Both CorNetAttentionXML and LDGN are recent research works in XMTC. While CorNetAttentionXML implemented a general architecture to output augmented label prediction, it did not further explore the possible solution of exploiting abundant correlations in the large label space but augmented final representation for prediction. And LDGN utilizes a graph neural network to model correlations in a label. But this work has the following obvious shortcomings: 1) Due to the huge computational expense of extra graph neural networks and introducing label information into the text learning process, it is hard to deploy on large dataset such as AmazonCat-13K; 2) It does not open its source codes, thus we cannot measure its reliability.

The architecture MAACF+ we proposed outperforms previous works on three datasets. To be specific, compared with those works that share source codes, the BERT augmented by MAACF achieves a better performance on all of the indicators. On AmazonCat-13k, MAACF+BERT boosts P@3 and N@5 from 92.17% to 94.82%, 84.54% to 89.73% respectively. Compared with LDGN, on EUR-Lex and AAPD, the model we proposed still outperforms it on most indicators. In conclusion, by introducing a co-attention mechanism and multiple aspect learning, we augment the basic text encoder with the most semantic relevant label group information and this architecture can effectively optimize computational expense compared with LDGN.

### 3.2.4 Ablation Test

We perform a series of ablation experiments to validate the effects and significance of introducing a statistical nearest label group on both three datasets. We set the test value of the population of label group $k$ as [0, 2, 5]. When $k = 0$, it means modeling without label group information. We just present the visualization of the results of MAACF+BERT.

From the results in Figure 2, we can see that model with a positive number of k normally outperforms in most datasets than a model without label group information, which indicates the model with co-attention label correlations extracting mechanism achieves more accurate prediction and effectively information capture. It proves that the significance of introducing label group modeling as well. Meanwhile, the nearest label group that has two labels is constantly the best choice, due to the model with $k = 2$ having better performance than $k = 5$. Thus, it is necessary to explore the effect of introducing label nearest group information extractor.



(a)EUR-Lex (b)AAPD (c)AmazonCat-13K

Figure 2: The visualization of ablation test results on three datasets.

# 4 Related Work

## 4.1 Text-Specific Methods in XMTC

Deep learning advanced in many fields in recent years, deep-based XMTC models also attracted many researchers to work on it. Compared with traditional methods (Liu et al., 2017), deep models take a sequence as input rather than bag-of-words which seems to contain less semantics. XMLCNN utilizes a dynamic pooling mechanism to capture complex level feature of the text, each filter represent one semantic pattern.

Recently, some studies (You et al., 2018)(Xiao et al., 2019)have used attention mechanisms to explore interactions between labels and words. To be more specific, they try to enhance the representation of the document with a label for classification. Some works attempt to utilize multiple fundamental text encoders to extract features from input sequence (Liu et al., 2017). Since 2018, pretrained models showed outstanding performance on most NLP tasks. X-BERT (Chang et al., 2020) utilized BERT as a text encoder with a multi-head self-attention mechanism for XMTC.

## 4.2 Lable-Specific Methods in XMTC

Unlike traditional text classification tasks, XMTC has a bigger number of labels. Training for each label or mapping the feature vector of text to a higher dimensional vector to output the possibility of each label is a high computational expense. Previously there were some researchers want to solve this task by building a tree structure to minimize the number of label candidates and reduce computational cost (Jain et al., 2016)(Jasinska et al., 2016)(Khandagale et al., 2020). But if there is no hierarchy structure within the label group in some datasets, tree-based XMTC models will not be able to undertake this task.

To build a relation graph in the label set, there are also some other models (Bhatia et al., 2015)(Tagami, 2017) that focus on using embedding of labels to search the similarity within their feature space. For example, AnneXML treated this problem as a weak-supervised task and employed KNN on a label to get less available label candidates. But these methods are not able to perform well in datasets that have no hierarchy relation between labels.

Recently, because of the increasing popularity of graph neural networks (GNN), some studies utilized GNN to explore interactions within labels or extract label-specific information from a document by treating relations between label and text as multi-relation graph (Ma et al., 2021).

However, existing studies were still classifying one feature vector. There are no approaches updated methods in learning interactions between text and label. Thus, our goal is to propose a novel classification method to explore label-specific components of text and information in label groups more accurately.

# 5 Conclusion

In this work, we propose a novel architecture named MAACF++ to promote the performance of basic document encoders in XMTC. It is an independent and general architecture and can be integrated with any deep encoders. Extensive experiments on three real-world benchmark datasets have demonstrated the effectiveness of it and it can achieve state-of-the-art performance. In the future, we will do more research on improving computing efficiency. And besides, we will pay attention to extracting as much as possible semantic information from the content of the label for XMTC.

# References

Kush Bhatia, Himanshu Jain, Purushottam Kar, Manik Varma, and Prateek Jain. 2015. Sparse local embeddings for extreme multi-label classification. In *NIPS*, volume 29, pages 730–738.

Wei-Cheng Chang, Hsiang-Fu Yu, Kai Zhong, Yiming Yang, and Inderjit S Dhillon. 2020. Taming pretrained transformers for extreme multi-label text classification. In *Proceedings of the 26th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining*, pages 3163–3171.

Savelie Cornegruta, Robert Bakewell, Samuel Withey, and Giovanni Montana. 2016. Modelling radiological language with bidirectional long short-term memory networks. *arXiv preprint arXiv:1609.08409*.

George Eason, Benjamin Noble, and Ian Naismith Sneddon. 1955. On certain integrals of lipschitz-hankel type involving products of bessel functions. *Philosophical Transactions of the Royal Society of London. Series A, Mathematical and Physical Sciences*, 247(935):529–551.

Xiangnan He, Lizi Liao, Hanwang Zhang, Liqiang Nie, Xia Hu, and Tat-Seng Chua. 2017. Neural collaborative filtering. In *Proceedings of the 26th international conference on world wide web*, pages 173–182.

Himanshu Jain, Yashoteja Prabhu, and Manik Varma. 2016. Extreme multi-label loss functions for recommendation, tagging, ranking & other missing label applications. In *Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, pages 935–944.

Kalina Jasinska, Krzysztof Dembczynski, Róbert Busa-Fekete, Karlson Pfannschmidt, Timo Klerx, and Eyke Hullermeier. 2016. Extreme f-measure maximization using sparse probability estimates. In *International conference on machine learning*, pages 1435–1444. PMLR.

Jacob Devlin Ming-Wei Chang Kenton and Lee Kristina Toutanova. 2019. Bert: Pre-training of deep bidirectional transformers for language understanding. In *Proceedings of NAACL-HLT*, pages 4171–4186.

Sujay Khandagale, Han Xiao, and Rohit Babbar. 2020. Bonsai: diverse and shallow trees for extreme multi-label classification. *Machine Learning*, 109(11):2099–2119.

Diederik P Kingma and Jimmy Ba. 2014. Adam: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980*.

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.

Qianwen Ma, Chunyuan Yuan, Wei Zhou, and Songlin Hu. 2021. Label-specific dual graph neural network for multi-label text classification. In *Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing (Volume 1: Long Papers)*, pages 3855–3864.

Julian McAuley and Jure Leskovec. 2013. Hidden factors and hidden topics: understanding rating dimensions with review text. In *Proceedings of the 7th ACM conference on Recommender systems*, pages 165–172.

Eneldo Loza Mencia and Johannes Fürnkranz. 2008a. Efficient pairwise multilabel classification for large-scale problems in the legal domain. In *Joint European Conference on Machine Learning and Knowledge Discovery in Databases*, pages 50–65. Springer.

Eneldo Loza Mencia and Johannes Fürnkranz. 2008b. Efficient pairwise multilabel classification for large-scale problems in the legal domain. In *Joint European Conference on Machine Learning and Knowledge Discovery in Databases*, pages 50–65. Springer.

Tomas Mikolov, Ilya Sutskever, Kai Chen, Greg S Corrado, and Jeff Dean. 2013. Distributed representations of words and phrases and their compositionality. In *Advances in neural information processing systems*, pages 3111–3119.

Yashoteja Prabhu, Anil Kag, Shrutendra Harsola, Rahul Agrawal, and Manik Varma. 2018. Parabel: Partitioned label trees for extreme classification with application to dynamic search advertising. In *Proceedings of the 2018 World Wide Web Conference*, pages 993–1002.

Yukihiro Tagami. 2017. Annexml: Approximate nearest neighbor search for extreme multi-label classification. In *Proceedings of the 23rd ACM SIGKDD international conference on knowledge discovery and data mining*, pages 455–464.

Jiyao Wang, Zhengnan Dong, and Liyang Chen. 2021. An multi-aspect attentional model to capture multistratal influence in social group. In *2021 International Conference on Electronic Information Engineering and Computer Science (EIECS)*, pages 315–320. IEEE.

Lin Xiao, Xin Huang, Boli Chen, and Liping Jing. 2019. Label-specific document representation for multi-label text classification. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*, pages 466–475.

Guangxu Xun, Kishlay Jha, Jianhui Sun, and Aidong Zhang. 2020. Correlation networks for extreme multi-label text classification. In *Proceedings of* *the 26th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining*, pages 1074–1082.

Pengcheng Yang, Xu Sun, Wei Li, Shuming Ma, Wei Wu, and Houfeng Wang. 2018. Sgm: sequence generation model for multi-label classification. *arXiv preprint arXiv:1806.04822*.

Ronghui You, Zihan Zhang, Ziye Wang, Suyang Dai, Hiroshi Mamitsuka, and Shanfeng Zhu. 2018. Attentionxml: Label tree-based attention-aware deep model for high-performance extreme multi-label text classification. *arXiv preprint arXiv:1811.01727*.

Wenhui Yu, Huidi Zhang, Xiangnan He, Xu Chen, Li Xiong, and Zheng Qin. 2018. Aesthetic-based clothing recommendation. In *Proceedings of the 2018 world wide web conference*, pages 649–658.

Wenjie Zhang, Junchi Yan, Xiangfeng Wang, and Hongyuan Zha. 2018. Deep extreme multi-label learning. In *Proceedings of the 2018 ACM on International Conference on Multimedia Retrieval*, pages 100–107.