多标签长尾数据分类平衡

摘要

多标签文本分类一直是一个挑战，尤其是对于 *长尾* 型数据，这就变得更加具有挑战性。重采样与重加权是解决这类问题的常用方法，但是对于类不均衡的标签依赖下，重采样和重加权会导致标签的过采样，于是本文介绍了平衡损失函数在多标签文本分类中的应用。

数据集：90个label的通用域 *Reuters-21758数据集/18211个label的特定域PubMed数据集*

本文发现了一个分布平衡的损失函数，内在的解决了类不平衡和标签连接的问题，优于常用损失函数。该分布平衡方法已成功应用于图像识别领域，此次将其运用在自然语言处理中，展示其有效性。

背景

多标签文本分类作为自然语言处理的核心问题之一，一直在搜索和产品分类等应用了得到广泛应用(Prabhuet al.,2018)(Agrawal et al.,2013)。旨在将具有多个标签的文本数据中从一组固定的标签找到相关的标签。
当标签存在长尾分布（长尾效应，英文名称Long Tail Effect。“头”（head）和“尾”（tail）是两个统计学名词。正态曲线中间的突起部分叫“头”；两边相对平缓的部分叫“尾”。从人们需求的角度来看，大多数的需求会集中在头部，而这部分我们可以称之为流行，而分布在尾部的需求是个性化的，零散的小量的需求。而这部分差异化的、少量的需求会在需求曲线上面形成一条长长的“尾巴”，而所谓长尾效应就在于它的数量上，将所有非流行的市场累加起来就会形成一个比流行市场还大的市场。）（类别不平衡）和标签连接的情况时，多标签分类变得复杂。
当标签的一小部分含有许多实例，而大多数标签每个标签仅有少部分实例，但是这些大多数标签的少部分实例在总实例中占有可观比例，就会发生类间不平衡。例如，Reuters数据集中，有一半的标签为 *铜*、 *战略金属* 和 *镍* ，但是他们只出现在少于5%的数据中。当一些头标签与罕见标签和尾标签出现共现时，标签贡献和标签连接就是一个挑战，导致了对于头标签的分类偏向。例如，镍的标签出现的频率较小，但是联合标签 *镍/铜*、*镍/战略金属* 的信息对于精确建模很重要。为此提出的解决方案有对较少出现频率的标签进行重新采样，在建模时考虑到共现信息，或者使用多任务架构提供头尾类别的混合解决方案，但是这些方案在nlp领域中不适合不平衡的数据集或者依赖于模型体系结构。
多标签数据在CV领域中的目标识别利用损失函数进行代价敏感学习，语义分割、医学成像中， *平衡损失函数[Focal Loss/Class-balanced loss/distribution-balanced loss]* 对CV中的多标签分类有了提升。

由此CV领域，迁移到NLP领域中的损失函数也进行了探索，Li Dice-based loss function 从医学图像分割任务，标准交叉熵损失函数在NLP任务中得到显著改进。
本文主要任务是将平衡损失函数引入到NLP领域中的多标签文本分类任务，并认为平衡损失函数对于多标签文本分类任务提供了一个健壮的解决方案。

损失函数

NLP领域中的 *BCE损失函数(Binary Cross Entropy loss)* 得到广泛应用。给定一个数据集 $\left(x^1, y^1\right), ..., \left(x^n, y^n\right)$ 其中含有N个训练实例，其中每个实例含有k个标签，$y^k = [y_1^2, ..., y_C^k] \in 0, 1^C$,$C$是类别种类。分类输出为$z^k = [z_1^k, ..., z_C^k] \in R$。BCE被定义为：

$$L_{BCE} = \begin{cases} -log(p_i^k), & if\ y_i^k = 1 \\ -log(1 - p_i^k), & otherwise. \end{cases} \tag{1}$$

其中， *sigmod function* 用来计算 p_i^k,$p_i^k = \sigma(z_i^k)$ 。普通的 BCE 容易收到标签不平衡的影响，于是本文介绍了三种用来解决长尾文本数据分类的方法，主要思想是对 BCE进行重加权，使稀有标签得到合理的“关注”

Focal loss (FL)

通过增加一个调制因子 γ $(\gamma\ \geq 0)$ *Focal loss* 增加里那些 *“hard-to-classify”* 的实例的权重。

Focal Loss 被定义为：

$$L_{FL} = \begin{cases} -(1 - p_i^k)^\gamma log(p_i^k), & if\ y_i^k = 1 \\ -(p_i^k)^\gamma log(1 - p_i^k), & otherwise. \end{cases} \tag{2}$$

Class-balanced focal loss(CB)

通过估计有效样本数， *Class-balanced focal loss* 进一步重加权 *FL*，来获得数据的边际效益递减，从而减少头部类的冗余信息。对于多标签任务，每个标签其出现的频率 n_i 都有其平衡项:

$$r_{CB} = \frac{1 + \beta}{1 + \beta^{n_i}} \tag{3}$$

$\beta \in [0, 1)$

此时损失函数为：

$$L_{CB} = \begin{cases} -r_{CB}(1-p_i^k)^{\gamma} log(p_i^k), & if\ y_i^k = 1 \\ -r_{CB}(p_i^k)^{\gamma} log(1-p_i^k), & otherwise. \end{cases} \tag{4}$$

Distribution-balanced loss(DB)

通过整合重加权和负容忍正则化 *Negative tolerant regularization(NTR)* , *DB loss* 首先减少标签共现的冗余信息，在 *easy to classify* 的负实例上分配更低的权重。

首先，为了平衡权重，在单标签场景中，实例通过重采样来加权，$P_i^C = \frac{1}{C}\frac{1}{n_i}$ ；对于多标签场景，采取相同的策略进行重加权时， $P^I = \frac{1}{C}\sum_{y_i^k=1}\frac{1}{n_i}$,此时可能过采样，为避免过采样，重平衡权值设置为 $r_{DB} = \frac{P_i^C}{P^I}$，平滑处理后为：$\hat{r}_{DB} = \alpha + \sigma(\beta + (r_{DB} - \mu))$，$r_{DB} \in [\alpha, \alpha+1]$，重平衡 $Focal\ loss$被定义为：

$$L_{R-FL} = \begin{cases} -\hat{r}_{DB}(1-p_i^k)^{\gamma} log(p_i^k), & if\ y_i^k = 1 \\ -\hat{r}_{DB}(p_i^k)^{\gamma} log(1-p_i^k), & oterwise. \end{cases} \tag{5}$$

然后，NTR对积极标签和消极标签进行处理，引入标度因子λ和类差异参数v_i来降低尾类与之，避免过度抑制。

$$L_{NTR-FL} = \begin{cases} -(1-q_i^k)^{\gamma} log(q_i^k), & if\ y_i^k = 1 \\ -\frac{1}{\lambda}(q_i^k)^{\gamma} log(1-q_i^k), & otherwise. \end{cases} \tag{6}$$

其中：$q_i^k = \sigma(z_i^k - v_i)$是积极实例，$q_i^k = \sigma(\lambda(z_i^k - v_i))$是消极样例。在训练开始时，通过尺度因子$k$和先前分类$p_i = \frac{n_i}{N}$，最小化损失函数得到$v_i$，于是，

$$\hat{b}_i = -log(\frac{1}{p_i} - 1), v_i = -k \times \hat{b}_i \tag{7}$$

最终结果，$DB\ Loss$就是：

$$L_{DB} = \begin{cases} -\hat{r}_{DB}(1-q_i^k)^{\gamma} log(q_i^k), & if\ y_i^k = 1 \\ -\hat{r}_{DB}\frac{1}{\lambda}(q_i^k)^{\gamma} log(1-q_i^k), & otherwise. \end{cases} \tag{8}$$

实验

数据集

Table 1: Dataset Statistics

Dataset	Statistic
Reuters-21578	
Number of documents	10788
Number of labels	90
Average number of labels per instance	1.24
Average number of instances per label	148.11
PubMed	
Number of documents	224897
Number of labels	18211
Average number of labels per instance	12.30
Average number of instances per label	151.88

#pic_center

实验设置

classical multilabel classification baseline：*SVM one-vs-rest*

两个数据集，不同的 *loss function* 对整个标签集(*头部/中部/尾部*)计算出最好的 *micro-F1* 和 *macro-F1 scores* 。

损失函数参数，分类模型及实现细节于附录A。

实验结果

Table 2: Micro and macro F1 scores for multi-label classification of Reuters-21578 (left) and PubMed (right) using the SVM model or different loss functions. The F1 scores are reported for the total set of labels as well as for the head, medium and tail label sets, with the number of instances given in parenthesis. The experiments are performed with the SVM one-vs-rest model (SVM), the binary cross entropy (BCE), focal loss (FL), class balanced focal loss (CB), rebalanced focal loss (R-FL), negative-tolerant regularization FL (NTR-FL), distribution balance with no FL (DB-0FL), class balanced FL with negative regularization (CB-NTR) and distribution balanced loss (DB).

Model/ Loss Function	**Reuters Total miF/maF**	**Reuters Head($\geq$35) miF/maF**	**Reuters Med(8-35) miF/maF**	**Reuters Tail($\leq$8) miF/maF**	**PubMed Total miF/maF**	**PubMed Head($\geq$50) miF/maF**	**PubMed Med(15-50) miF/maF**	**PubMed Tail($\leq$15) miF/maF**
SVM	87.60/51.63	89.87/78.47	66.92/61.00	22.54/13.83	58.54/13.31	60.77/34.33	19.78/5.62	6.94/0.67
BCE	89.14/47.32	91.75/82.81	66.28/57.26	0.00/0.00	26.17/0.02	27.61/0.06	0.00/0.00	0.00/0.00
FL	89.97/56.83	91.83/82.64	76.16/70.63	27.40/15.37	58.30/13.94	60.43/33.69	26.39/8.15	8.58/0.86
CB	89.23/52.96	91.56/80.44	71.64/66.61	23.08/9.93	58.57/13.67	60.75/33.40	24.50/7.39	9.92/1.01
R-FL	89.47/54.35	91.59/80.39	72.86/66.69	25.00/14.22	57.90/14.66	59.85/34.09	30.32/9.70	11.45/1.15
NTR-FL	90.70/60.70	92.37/82.65	79.35/75.34	39.51/22.33	60.92/16.99	**63.15**/38.85	33.14/11.39	15.86/1.82
DB-0FL	89.45/57.98	91.21/82.05	77.33/71.11	31.17/19.05	58.95/15.15	60.99/34.92	31.06/10.02	14.23/1.49
CB-NTR	**90.74**/63.31	**92.46**/83.28	78.42/72.98	46.32/**32.31**	**61.07**/18.40	63.02/39.95	37.18/13.43	24.15/2.97
DB	90.62/**64.47**	92.14/**83.48**	**80.25/77.01**	**48.89**/31.39	60.63/**19.19**	62.39/**40.48**	**41.14/15.33**	**24.19/3.08**

Reuter Dataset 1000 documents，90 labels，150 instances/each label

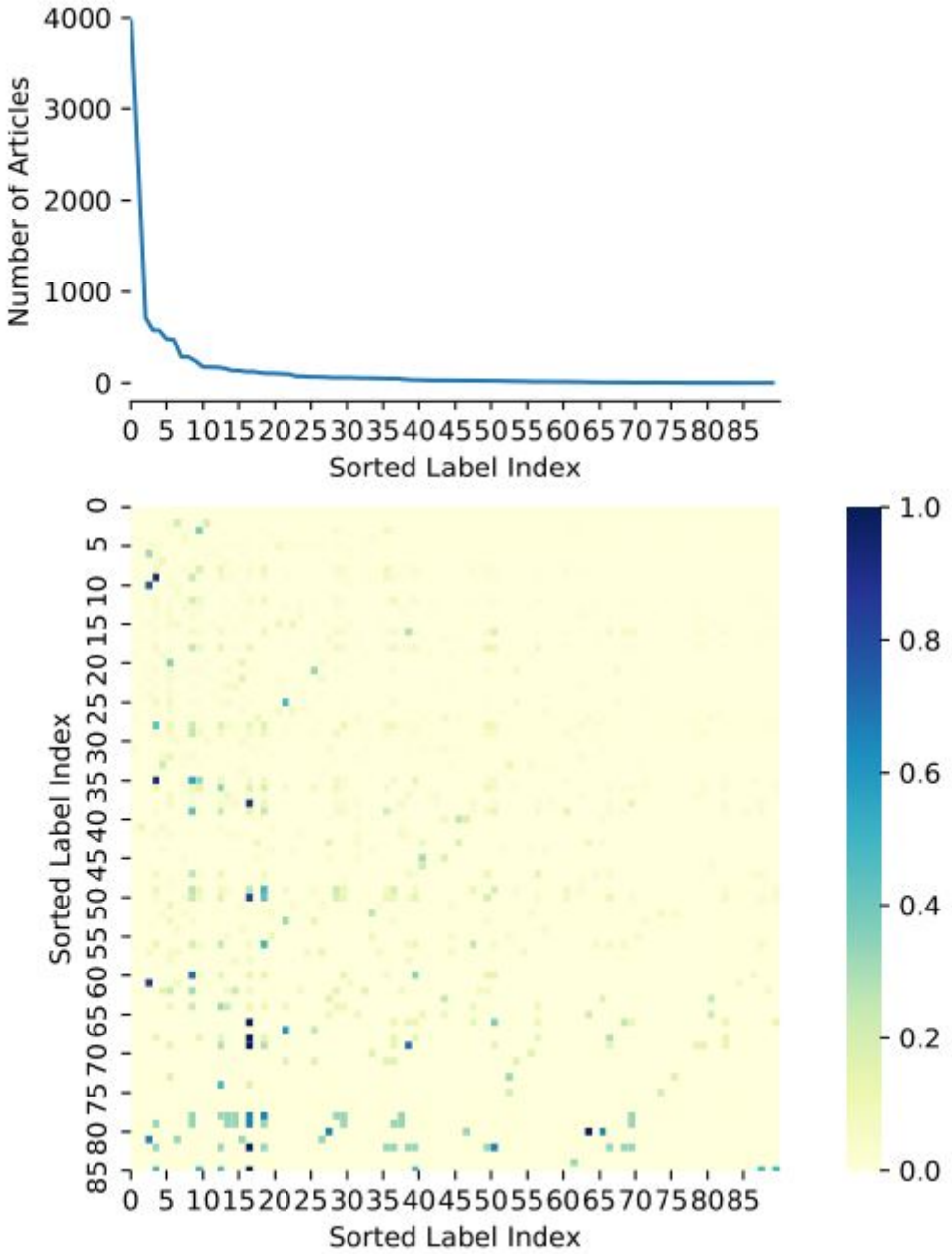

Figure 2: The long-tailed distribution and label co-occurrence for the Reuters-21578 dataset. The co-occurence matrix is color coded based on the conditional probability $p(i|j)$ of class in the i^{th} column on class in the j^{th} row.

Fig.2 展示了长尾分布，其中少量标签含有大量文章，头部标签和其他标签有很高共现性。偏态分布的影响也可以从micro-F1和Macro-F1分数的比较中看出来。而且，在损失函数中，BCE在Reuter dataset

表现最差，macro-F1：47/tail F1：0。

PubMed dataset 22500 documents, 18000 labels

长尾分布不平衡在大数据集中更加明显，total micro-F1 score(60)/total macro-F1 score(15)。
总体上，SVM在数据集的表现都不如提出的分布均衡损失函数。

Retuers-21578 Experience

损失函数 $FL, CB, R-FL, NTR-FL$ 在头部类的表现与 BCE 相似，但是在中/尾部表现优于 BCE，展现了其处理分类不平衡的优势。 DB 在尾部分类提供了最大改进，尾部 $micro-F1$ 比 FL 增加了21.49，比 CB 增加了25.81。使用损失函数优于之前的常用操作，比如，
$Binary\ Relevance, EncDec, CNN, CNN-RNN, Optimal\ Completion\ Distillation, attetion-based\ GNN$，使得 *micro-F1<89.9*。

PubMed dataset Experience

PubMed数据集基于生物医学领域，数据量更大，类别更不平衡。这个数据集，BCE表现不好，于是本文使用FL作为 *strong baseline*。FL在中/尾测试micro-F1 score为26/9。所有其他损失函数，在中/尾类都优于FL，表现了平衡标签分布的优势。DB在所有类别上都性能最好，在中部和尾部级别上的改进最明显(medium：micro-F1:41/tail：micro-F1:24)。

Ablation Study

通过对比R-FL、NTR-FL和DB-0FL的结果，我们进一步研究了三层DB layer的贡献。
如Table2所示，去除NTR layer(R-FL)或者是Focal layer(DB-0FL)会降低所有子集的模型性能。去除重新加权layer(NTR-FL),total micro-layer(Retuers:90/PubMed:60)与之前相似，但是中/尾部macro-F1 score比DB高，展现了增加重平衡权重的价值。本文还通过将NTR与CB积分来测试NTR的贡献，得到了新的损失函数CB-NTR，CB-NTR在所有集合上表现都比CB要好。CB-NTR和DB之间唯一的区别就是CB权重 r_{CB} 的使用，而不是重平衡权重 $\hat{r}_{DB}$ 。DB在中/尾部的性能非常接近甚至超过CB-NTR，表明了权重 $\hat{r}_{DB}$ 对于解决共现现象是有用的。

Error Analysis

本文进行错误分析，观察的最常见的错误是由于标签的相似带来的不正确分类。在Retuers数据集中，最容易混淆的类别为 *platinum/gold,yen/money-fx,platinum/copper* 。在PubMed数据集中，最常见错误为： *Pandemics/Betacoronavirus,Pandemics/SARS-CoV-2/Pneumonia,Viral/Betacoronavirus* ，并且所有损失函数中，BCE有明显更多的错误。

结论

本文提出并比较了一系列平衡损失函数在多标签文本分类中解决类别不平衡问题的应用。首先将DB引入到NLP领域，然后设计了一个新的损失函数CB-NTR。实验表明，考虑到长尾分布和标签共现的DB损失函数优于其他函数，并且在Retuers数据集和PubMed数据集中有较好鲁棒性。本研究表明，通过损失函数解决类不平衡和标签共现问题是解决多标签文本分类的有效办法。可应用于所有类型的基于神经网络的模型，其也有可能是其他NLP任务中的强大策略，例如 *part-of-speech tagging/named entity recognition/machine reading comprehension/paraphrase identification/coreference resolution* 这些任务通常会收到长尾分布的影响。

致谢

Appendix

Experience Setting

评价指标(Evaluation metrics)

对于每个数据集和方法，本文选择验证集micro-F1最好的阈值作为最终模型，使用marco-F1和micro-F1 score来评估测试集上的性能。

损失函数参数(Loss function parameters)

本文比较了DB在不同损失函数BCE及其修改的表现。操作包括：

- BCE所有实例和标签权重相同
- FL，$\gamma = 2$
- CB，$\beta = 0.9$
- R-FL，$\alpha = 0.1\ \beta = 10\ \mu = 0.9(Retures - 21578)/0.05(PubMed)$
- NTR-FL，$k = 0.05$，λ
- DB，same parameters with R-FL/NTR-FL

实现细节(Implementation Details)

Retuers-21578 dataset，使用了_BertgForSequenceClassifition_作为_transformers_主干，_bert-base-cased_预训练模型，PubMed dataset 使用了_biobert-base-cased-v1.1预训练模型。
bert-base-cased和biobert-base-cased-v1.1是基于BERT模型，具有1.1亿参数，训练数据最大长度512，batch_size为32。
采用全职衰减为0.01的AdamW作为优化器，通过超参数确定学习率。使用PyTotch框架，Retuers-21578数据集，使用one-GPU(V100)，每个epoch花费5分钟；PubMed数据集，使用obe-GPU(A100)，

每个epoch花费1小时。对于SVM的one-vs-rest模型，本文使用具有TF-IDF特征的_scikit-learn_库。通过超参数搜索，在每个验证集上应用线性内核和超平面移位优化。

额外有效性检查(Additional Effectiveness Check)

我们进一步研究损失函数对于每个实例标签数量的有效性(Table3)。
Retuers dataset：将测试实例分为2组，其中2583个实例只有1个标签，436个实例具有多个标签。单标签实例中，从BCE道DB具有相似性能，但是在多标签实例中，BCE性能下降幅度大于DB。DB的micro-F1在多标签组中优于其他函数，macro-F1优于所有函数。
PubMed dataset：数据集仅有少于0.1% 的实例为单标签，于是我们根据标签数量分为3组。在每组中，NTR-FL，CB-NTR，DB在所有指标上都优于其他函数。

Table 3: Micro and macro F1 scores for multi-label classification of Reuters-21578 (left) and PubMed (right) using different loss functions. The F1 scores are reported for the total set of labels as well as for groups split by the number of labels per instance. The experiments are performed with the binary cross entropy (BCE), focal loss (FL), class balanced focal loss (CB), rebalanced focal loss (R-FL), negative-tolerant regularization FL (NTR-FL), distribution balance with no FL (DB-0FL), class balanced FL with negative regularization (CB-NTR) and distribution balanced loss (DB).

Loss Function	**Reuters Total miF/maF**	**Reuters Single-label miF/maF**	**Reuters Multi-label miF/maF**	**PubMed Total miF/maF**	**PubMed $\leq$ 9 labels miF/maF**	**PubMed 10-14 labels miF/maF**	**PubMed $\geq$ 15 labels miF/maF**
BCE	89.14/47.32	94.11/41.44	76.26/33.11	26.17/0.02	16.48/0.01	27.36/0.02	30.36/0.03
FL	89.97/56.83	94.81/50.33	77.54/40.07	58.30/13.94	53.72/7.44	59.02/10.27	59.72/8.63
CB	89.23/52.96	94.10/44.72	77.27/38.80	58.57/13.67	54.41/7.40	59.21/10.11	59.82/8.51
R-FL	89.47/54.35	95.21/47.45	74.29/38.79	57.90/14.66	53.08/7.67	58.60/10.50	59.45/8.81
NTR-FL	90.70/60.70	**95.42**/51.33	78.85/44.37	60.92/16.99	**58.51**/9.07	**61.86**/12.31	61.12/10.20
DB-0FL	89.45/57.98	94.48/51.80	76.63/42.26	58.95/15.15	55.14/8.11	59.84/10.90	59.85/8.94
CB-NTR	**90.74**/63.31	95.17/51.08	79.56/49.94	**61.07**/18.40	58.29/9.67	61.72/12.97	**61.72**/10.77
DB	90.62/**64.47**	94.49/**54.31**	**81.17/50.12**	60.63/**19.19**	57.81/**9.76**	61.53/**13.49**	61.08/**11.23**

Appendix

混淆矩阵：

混淆矩阵		预测	
		真（正）	假（负）
实际	真（正）	TP	FN
	假（负）	FP	TN

TP:正预测为正
FN:正预测为负
FP:负预测为正
TN:负预测为负

microF1

$$microPrecision = \frac{\overline{TP}}{\overline{TP}+\overline{FP}}$$

$$microRecall = \frac{\overline{TP}}{\overline{TP}+\overline{FN}}$$

$$microF1 = \frac{2 \times microP \times microR}{microP + microR}$$

macroF1

$$macroF1 = \frac{micro-F1-score_1 + micro-F1-score_2 + micro-F1-score_3 + ... + micro-F1-score_N}{N}$$