分层文本分类

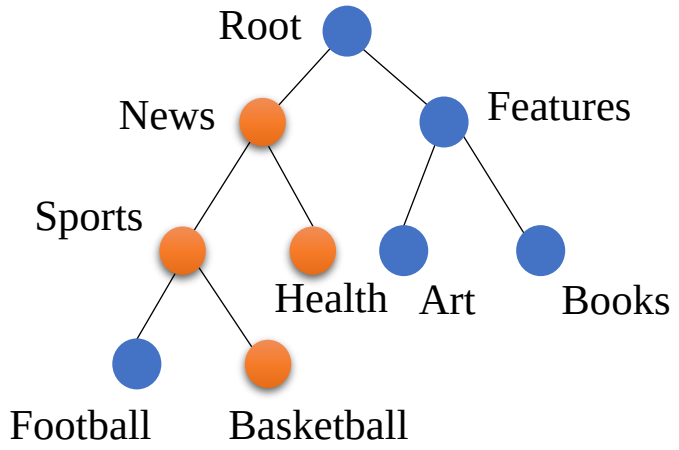
Hierarchical Text Classification

刘家伟

2020.12.03

Introduction

Hierarchical text classification (HTC) aims to categorize a textual description within a set of labels that are organized in a structured class hierarchy



e.g. Sentence: Durant couldn‘t play the basketball game because of an Achilles injury.

Labels: News、Sports、Health、Basketball

Hierarchy-Aware Global Model for Hierarchical Text Classification

Jie Zhou[1,2*], **Chunping Ma**[2], **Dingkun Long**[2], **Guangwei Xu**[2],
Ning Ding[3], **Haoyu Zhang**[4], **Pengjun Xie**[2], **Gongshen Liu**[1†]
[1]School of Electronic Information and Electrical Engineering, Shanghai Jiao Tong University
[2]Alibaba Group, [3]Tsinghua University, [4]National University of Defense Technology
{sanny02,lgshen}@sjtu.edu.cn,
{kunka.xgw,chengchen.xpj}@taobao.com
{chunping.mcp,dingkun.ldk}@alibaba-inc.com
{dingn18}@mails.tsinghua.edu.cn,{zhanghaoyu10}@nudt.edu.cn

Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics, pages 1106–1117
July 5 - 10, 2020.

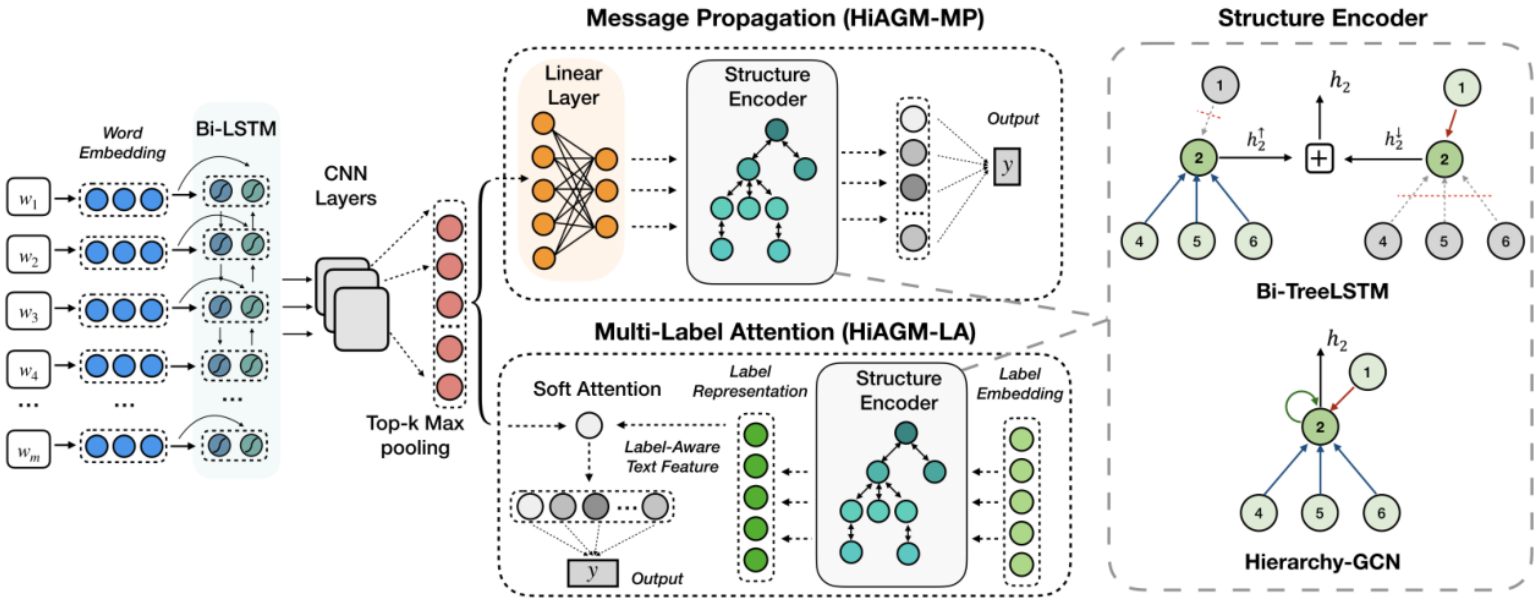
motivation

- 目前还没有一种全局方法对整体标签结构进行编码。
- 当前的方法仍然以一种浅层的方式利用了层次结构，而忽略了更为有效的细粒度标签相关信息。

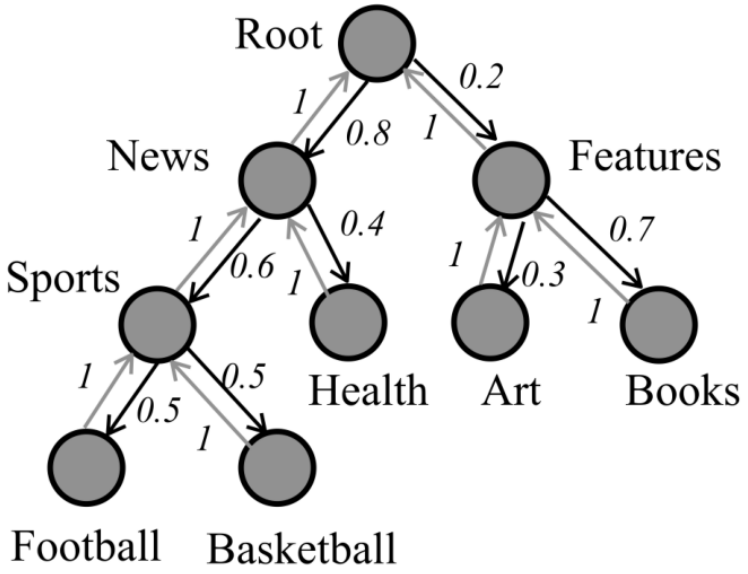
contributions

- 基于先验层级信息，采用结构编码器对标签依赖关系进行自顶向下和自底向上的建模。
- 提出了一种端到端的层级感知全局模型（HiAGM）。在此基础上进一步提出了两个变体--层级感知的多标签注意力模型（HiAGM-LA）和层级感知的文本特征传播模型（HiAGM-TP）。

Hierarchy-Aware Global Model (HIAGM)



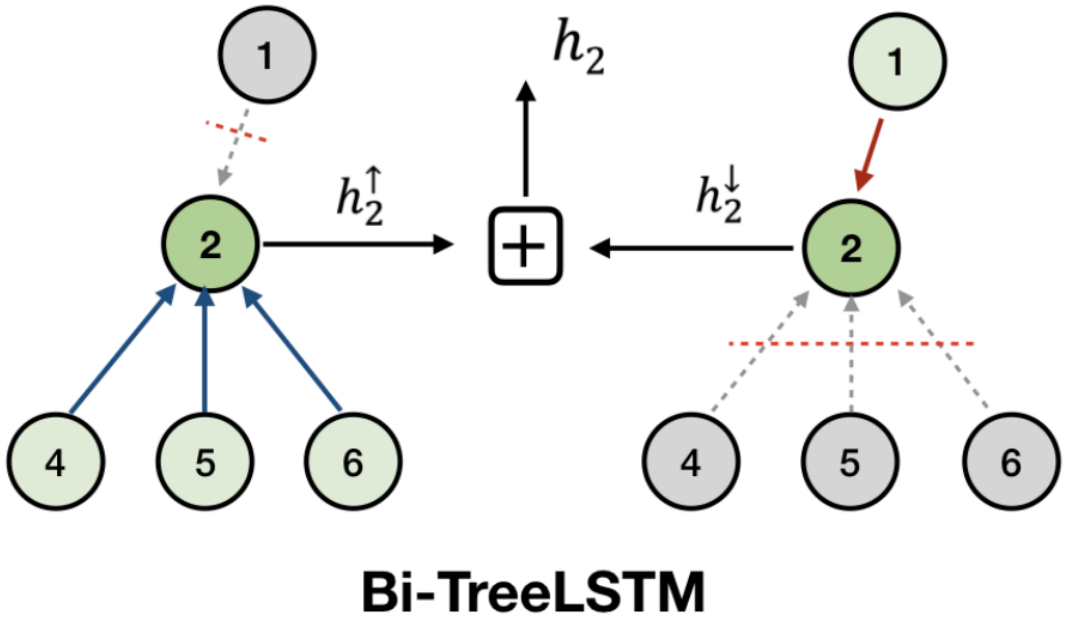
Prior Hierarchy Information



$$P(U_j|U_i) = \frac{P(U_j \cap U_i)}{P(U_i)} = \frac{P(U_j)}{P(U_i)} = \frac{N_j}{N_i},$$

$$P(U_i|U_j) = \frac{P(U_i \cap U_j)}{P(U_j)} = \frac{P(U_j)}{P(U_j)} = 1.0,$$

Structure Encoder

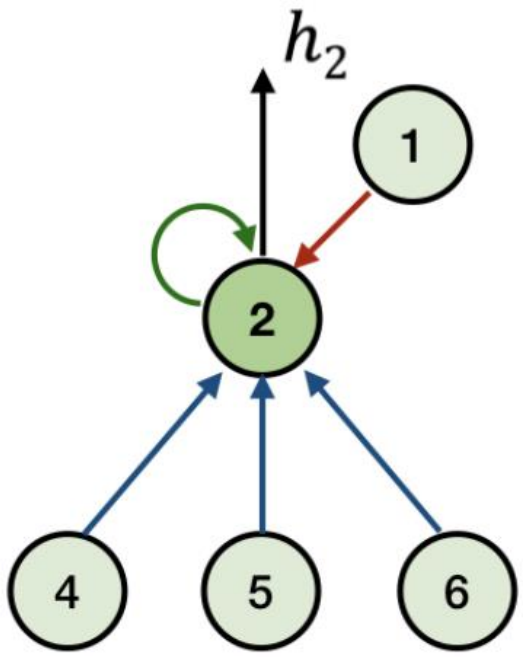


Bi-TreeLSTM

$$
\begin{aligned}
\boldsymbol{i}_k &= \sigma(\boldsymbol{W}_{(i)}\,\boldsymbol{v}_k + \boldsymbol{U}_{(i)}\,\widetilde{\boldsymbol{h}}_k + \boldsymbol{b}_{(i)}),\\
\boldsymbol{f}_{k,j} &= \sigma(\boldsymbol{W}_{(f)}\,\boldsymbol{v}_k + \boldsymbol{U}_{(f)}\,\boldsymbol{h}_j + \boldsymbol{b}_{(f)}),\\
\boldsymbol{o}_k &= \sigma(\boldsymbol{W}_{(o)}\,\boldsymbol{v}_k + \boldsymbol{U}_{(o)}\,\widetilde{\boldsymbol{h}}_k + \boldsymbol{b}_{(o)}),\\
\boldsymbol{u}_k &= tanh(\boldsymbol{W}^{(u)}\,\boldsymbol{v}_k + \boldsymbol{U}^{(u)}\,\widetilde{\boldsymbol{h}}_k + \boldsymbol{b}^{(u)}),\\
\boldsymbol{c}_k &= \boldsymbol{i}_k \odot \boldsymbol{u}_k + \sum\nolimits_j \boldsymbol{f}_{k,j} \odot \boldsymbol{c}_j,\\
\boldsymbol{h}_k &= \boldsymbol{o}_k \odot tanh(\boldsymbol{c}_k),
\end{aligned}
$$

$$
\begin{aligned}
\widetilde{\boldsymbol{h}}_k^{\uparrow} &= \sum\nolimits_{j \in child(k)} f_p(e_{k,j})\,\boldsymbol{h}_j^{\uparrow},\\
\widetilde{\boldsymbol{h}}_k^{\downarrow} &= f_c(e_{k,p})\,\boldsymbol{h}_p^{\downarrow},\\
\boldsymbol{h}_k^{bi} &= \boldsymbol{h}_k^{\uparrow} \oplus \boldsymbol{h}_k^{\downarrow},
\end{aligned}
$$

Structure Encoder



Hierarchy-GCN

$$\boldsymbol{u}_{k,j} = a_{k,j}\boldsymbol{v}_j + \boldsymbol{b}_l^k,$$
$$\boldsymbol{g}_{k,j} = \sigma(\boldsymbol{W}_g^{d(j,k)}\boldsymbol{v}_k + \boldsymbol{b}_g^k),$$
$$\boldsymbol{h}_k = \mathrm{ReLU}(\sum\nolimits_{j \in N(k)} \boldsymbol{g}_{k,j} \odot \boldsymbol{u}_{k,j}),$$

$$N(k) = \{n_k, child(k), parent(k)\}$$

Datasets

Dataset	$\|L\|$	Depth	Avg($\|L_i\|$)	Train	Val	Test
RCV1	103	4	3.24	20,833	2,316	781,265
WOS	141	2	2.0	30,070	7,518	9,397
NYT	166	8	7.6	23,345	5,834	7,292

Data Statistics: |L| is the number of classes. Avg(|L i |) is the average number of classes per sample.Depth indicates the maximum level of hierarchy

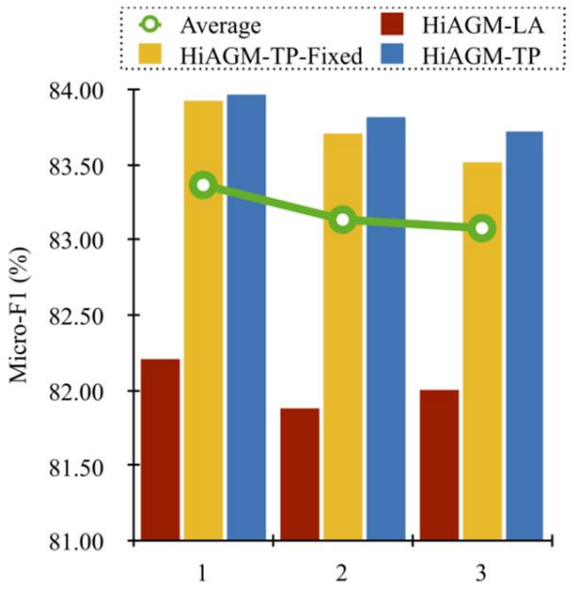
Experiments

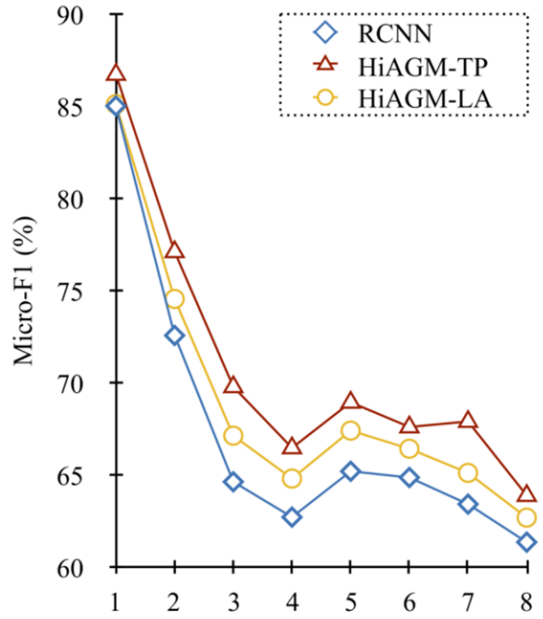
Model	RCV1-V2		RCV1-V2-R		WOS		NYT	
	Micro-F1	Macro-F1	Micro-F1	Macro-F1	Micro-F1	Macro-F1	Micro-F1	Macro-F1
Global Text Classification Baseline								
TextRNN	81.10	51.09	87.78	70.42	77.94	69.65	70.29	53.06
TextCNN	79.37	55.45	84.97	68.06	82.00	76.18	70.11	56.84
TextRCNN	81.57	59.25	88.32	72.23	83.55	76.99	70.83	56.18
HiAGM-LA								
GCN	82.21	61.65	88.49	73.14	84.61	79.37	72.35	58.67
TreeLSTM	82.54	61.90	88.47	72.81	84.82	79.51	72.50	58.86
HiAGM-TP								
GCN	**83.96**	**63.35**	88.64	74.00	**85.82**	**80.28**	**74.97**	**60.83**
TreeLSTM	83.20	62.32	**88.86**	**74.16**	85.18	79.95	74.43	60.76

Model	HiAGM-LA			HiAGM-TP		
	Micro	Macro	Time	Micro	Macro	Time
TreeLSTM	82.54	61.90	1.0 ×	83.24	62.60	3.2×
GCN	82.21	61.65	**0.9**×	**83.92**	**63.01**	1.1×

Experiments

Top-Down	Bottom-Up	Fixed		Trainable	
		Micro	Macro	Micro	Macro
Edge-Wise Matrix		-	-	82.75	60.81
Randomly Initialized		-	-	83.86	62.12
Randomly Initialized*		-	-	82.80	62.51
1	1	83.77	62.31	83.86	62.96
P	P	83.61	**63.65**	83.83	63.14
1	P	83.65	62.46	83.95	63.23
P	1	**83.92**	63.01	**83.96**	**63.35**
P*	1*	-	-	83.33	62.86





Efficient strategies for hierarchical text classification: External knowledge and auxiliary tasks

Kervy Rivas Rojas, Gina Bustamante, Arturo Oncevay[‡], Marco A. Sobrevilla Cabezudo[†]
Research Group on Artificial Intelligence, Pontificia Universidad Católica del Perú, Peru
[‡]School of Informatics, University of Edinburgh, Scotland
[†]Instituto de Ciências Matemáticas e de Computação, Universidade de São Paulo, Brazil
k.rivas@pucp.pe, gina.bustamante@pucp.edu.pe,
a.oncevay@ed.ac.uk, msobrevillac@usp.br

Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics, pages 2252–2257
July 5 - 10, 2020.

motivation

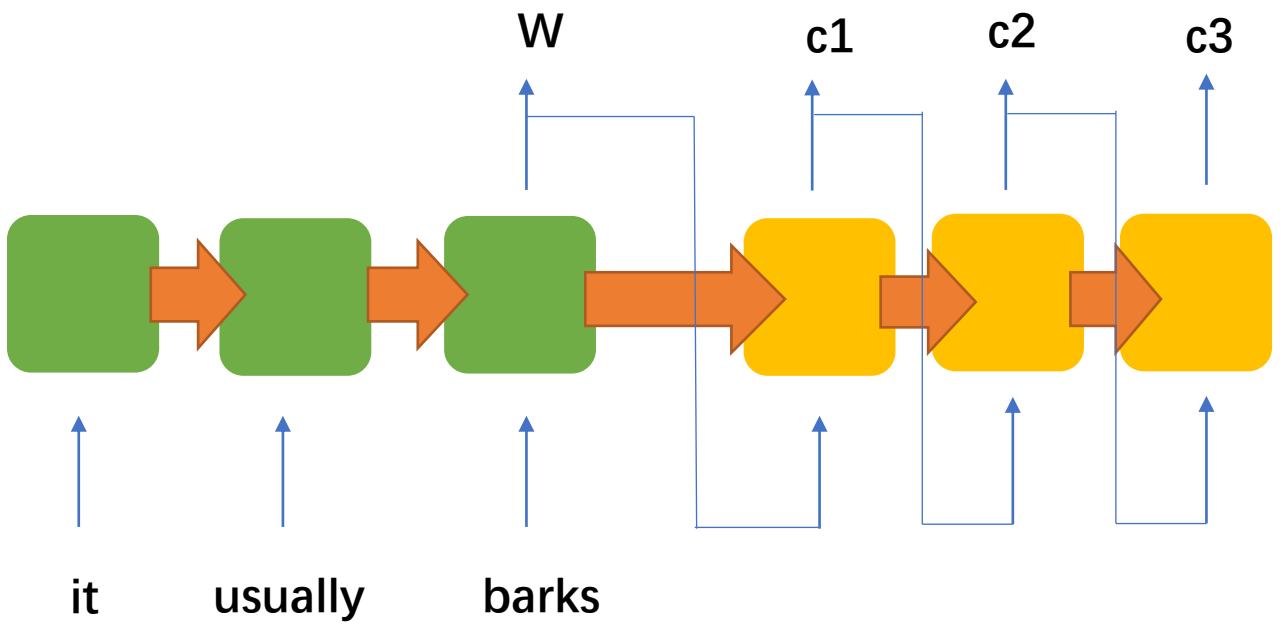
- 大多数的研究都集中在开发新的神经网络结构来处理层次结构，而没有重视采用合理的措施来提高已有模型的性能。
- 以往的HTC模型常采用自上而下的的预测顺序，从而导致比较严重的误差传播。

contributions

- 将分层文本分类看成一个sequence-to-sequence 问题。
- 为了减少误差传播，提出了一种自底向上分类的辅助任务。
- 引入类别的定义知识，作为下一层类别判断的条件。

1. Model Architecture:

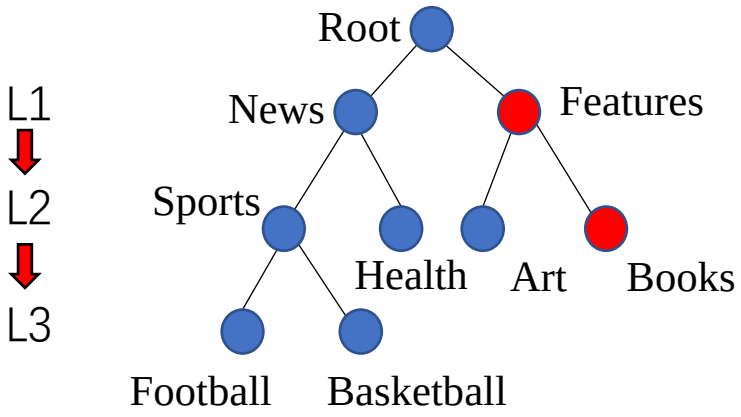
HTC as a sequence-to-sequence problem using a bidirectional GRU unit.

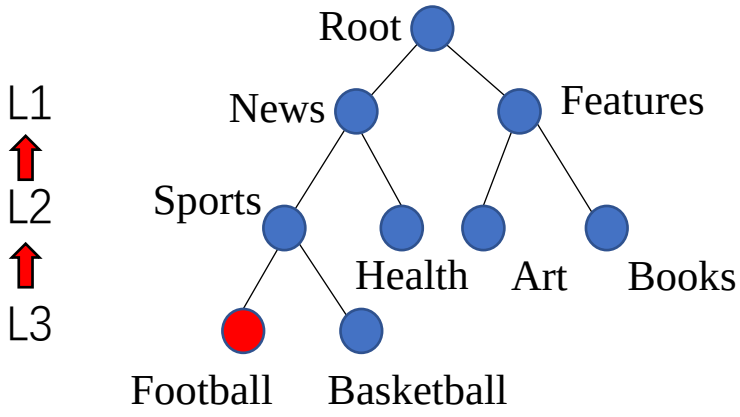


2. Auxiliary task:

The usual HTC predicts in this order: L1→L2→L3

We also predict inverse order of the class hierarchy:L3→L2→L1

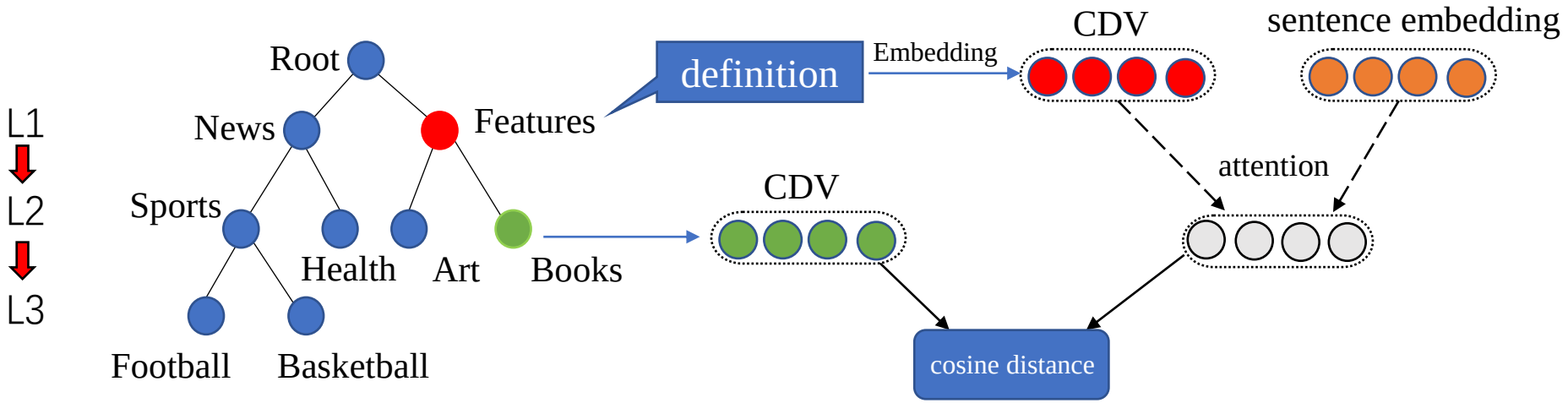




3. Class-definition embeddings for external knowledge integration

For each class c^{ij} in any level l_j of the hierarchy, we could obtain a raw text definition and compute a vector representation(CDV)

- Parent node conditioning (PNC):
- Adapted beam search



$$\sum_{i=0}^{T} logP(y^i|x, y^1, ..., y^{t-1}) + CD(z, y^i)$$

Datasets

	WOS	DBpedia
Number of documents	46,985	342,782
Classes in level 1	7	9
Classes in level 2	143	70
Classes in level 3	NA	219

Experiments

		WOS	DBpedia
Individual strategies	seq2seq baseline	78.84 ± 0.17	95.12 ± 0.01
	Auxiliary task	*78.93 ± 0.52	*95.21 ± 0.16
	Parent node conditioning (PNC)	*79.01 ± 0.18	*95.26 ± 0.09
	Beam search (original)	*78.90 ± 0.25	*95.25 ± 0.01
	Beam search (modified)	*78.90 ± 0.28	*95.26 ± 0.01
Combined strategies	Auxiliary task + PNC [7M params.]	*79.79 ± 0.45	*95.23 ± 0.13
	Beam search (original) + PNC	*79.18 ± 0.19	***95.30** ± 0.10
	Beam search (modified) + PNC	*79.18 ± 0.23	*95.30 ± 0.11
	Auxiliary task + PNC + Beam search (orig.)	***79.92** ± 0.51	*95.26 ± 0.12
	Auxiliary task + PNC + Beam search (mod.)	*79.87 ± 0.49	*95.26 ± 0.12
Previous work	HDLTex (Kowsari et al., 2017) [5B params.]	76.58	92.10
	Sinha et al. (2018) [34M params.]	77.46	93.72

MixText: Linguistically-Informed Interpolation of Hidden Space for Semi-Supervised Text Classification

Jiaao Chen
Georgia Tech
jchen896@gatech.edu

Zichao Yang
CMU
zichaoy@cs.cmu.edu

Diyi Yang
Georgia Tech
dyang888@gatech.edu

Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics, pages 2147–2157

motivation

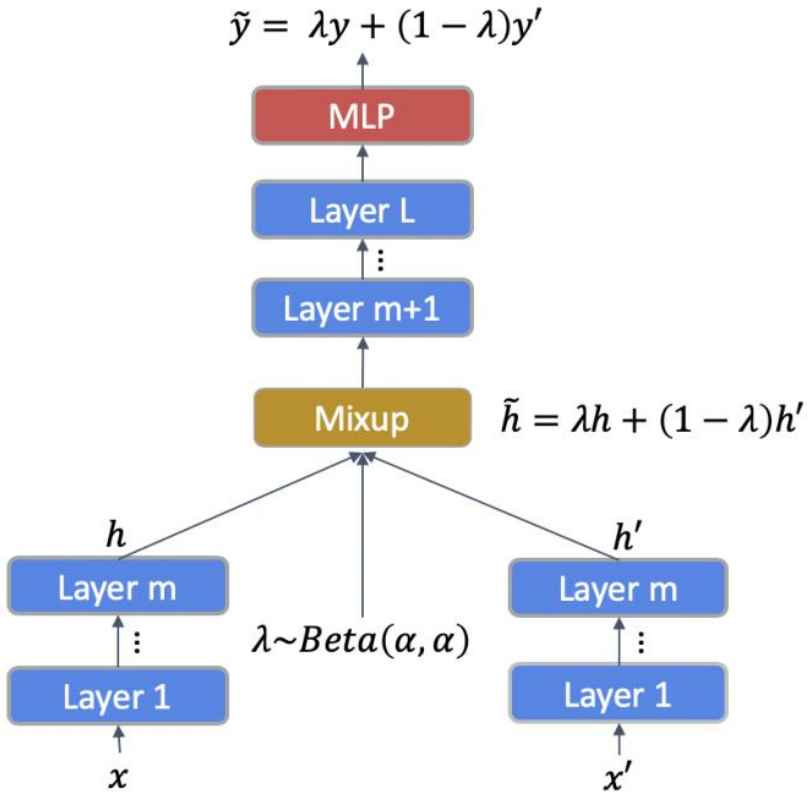
现有的半监督文本分类模型中，有标签数据和无标签数据是分开训练的，在训练中往往会出现有标签数据已经过多轮迭代、而无标签数据还处于欠拟合状态的局面。因此，大多数半监督模型仍然很容易对极为有限的标记数据过度拟合。

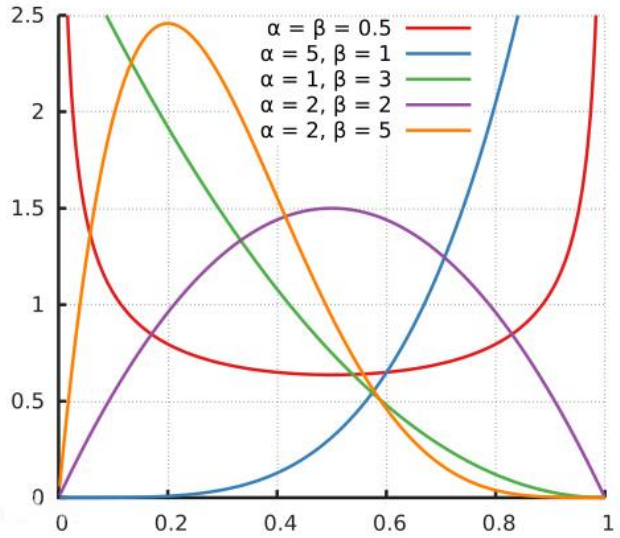
contributions

- 受到图像分类算法Mixup的启发，作者提出了一种新的数据增强算法Tmix ——在文本隐藏空间中进行插值（interpolation），从而生成大量新的训练数据，极大地避免了过拟合的产生。

- 作者重点研究Tmix在文本分类上的应用，进一步提出了基于Tmix与consistency training 的半监督文本分类模型MixText。该模型在标记数据有限时的优质性能，能够显著提升分类准确率、缓解过拟合问题。

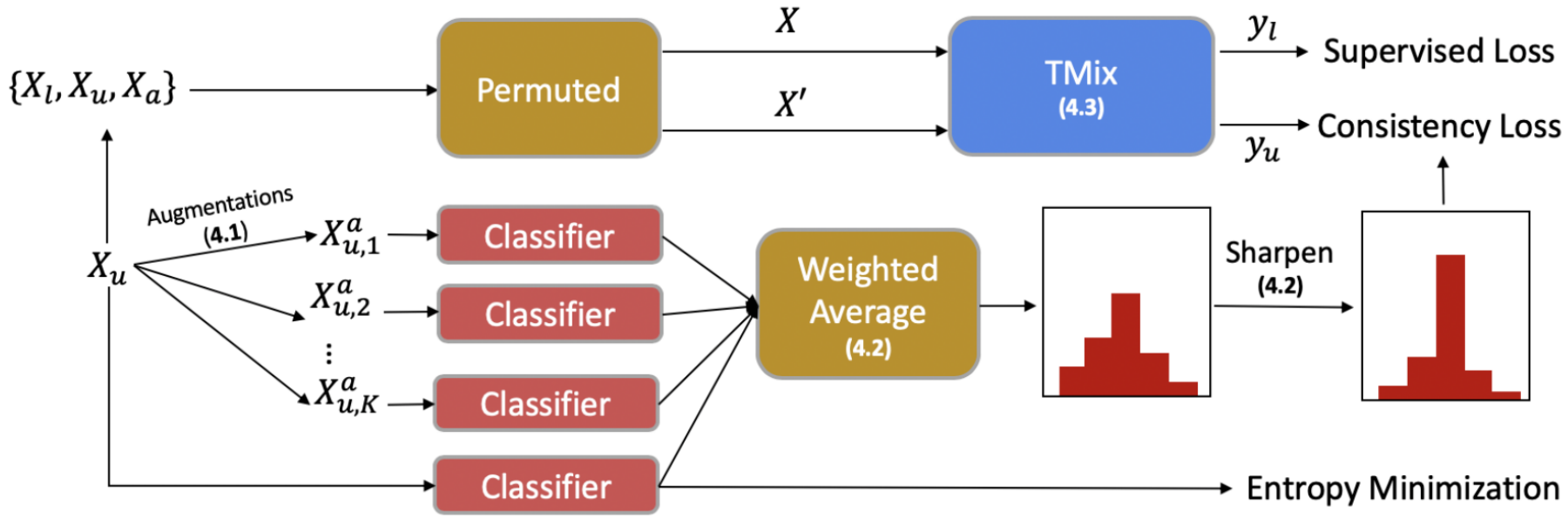
TMix



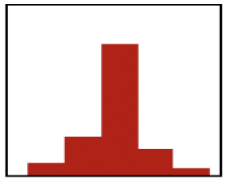


$$\lambda \sim \text{Beta}(\alpha, \alpha),$$
$$\lambda = \max(\lambda, 1-\lambda)$$

MixText



Weighted Average



$$\mathbf{y}_i^u = \frac{1}{w_{ori} + \sum_k w_k}(w_{ori} p(\mathbf{x}_i^u) + \sum_{k=1}^{K} w_k p(\mathbf{x}_{i,k}^a)))$$

$$\text{Sharpen}(\mathbf{y}_i^u, T) = \frac{(\mathbf{y}_i^u)^{\frac{1}{T}}}{||(\mathbf{y}_i^u)^{\frac{1}{T}}||_1}$$

TMix loss $L_{\text{TMix}} = \mathbb{E}_{\mathbf{x},\mathbf{x}' \in \mathbf{X}} \text{KL}(\text{mix}(\mathbf{y}, \mathbf{y}') || p(\text{TMix}(\mathbf{x}, \mathbf{x}'))$

Entropy Minimization $L_{\text{margin}} = \mathbb{E}_{\mathbf{x} \in \mathbf{X}_u} \max(0, \gamma - ||\mathbf{y}^u||_2^2)$

MixText loss $L_{\text{MixText}} = L_{\text{TMix}} + \gamma_m L_{\text{margin}}$

Datasets

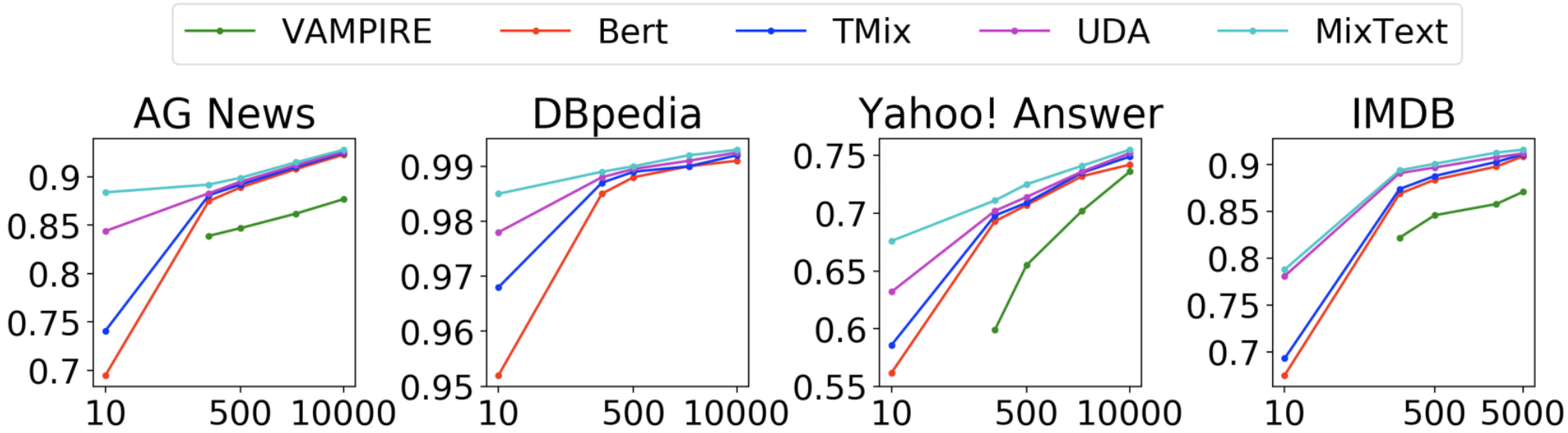
Dataset	Label Type	Classes	Unlabeled	Dev	Test
AG News	News Topic	4	5000	2000	1900
DBpedia	Wikipeida Topic	14	5000	2000	5000
Yahoo! Answer	QA Topic	10	5000	5000	6000
IMDB	Review Sentiment	2	5000	2000	12500

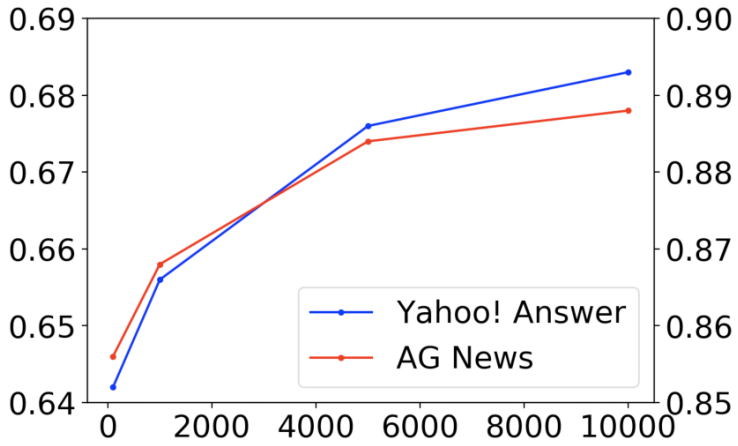
Dataset statistics and dataset split. The number of unlabeled data, dev data and test data in the table means the number of data per class.

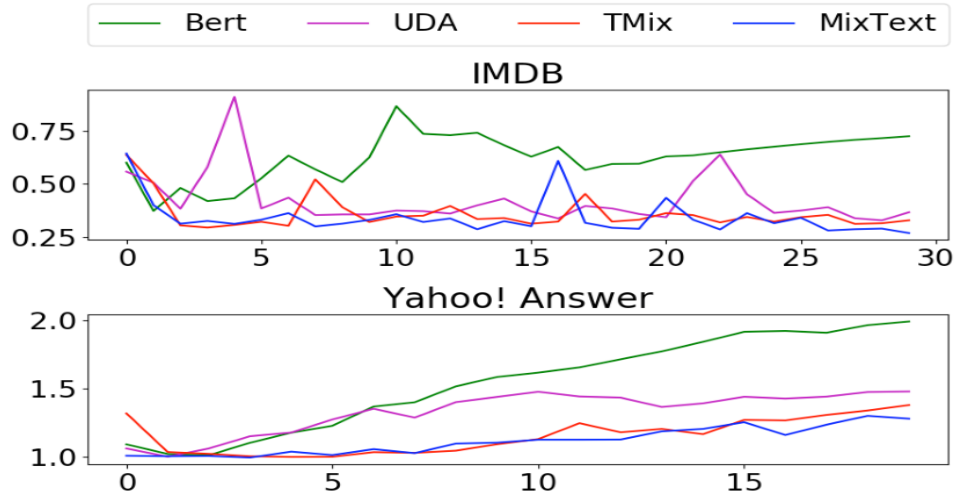
Experiments

Datset	Model	10	200	2500	Dataset	Model	10	200	2500
AG News	VAMPIRE	-	83.9	86.2	DBpedia	VAMPIRE	-	-	-
	BERT	69.5	87.5	90.8		BERT	95.2	98.5	99.0
	TMix*	74.1	88.1	91.0		TMix*	96.8	98.7	99.0
	UDA	84.4	88.3	91.2		UDA	97.8	98.8	99.1
	MixText*	**88.4**	**89.2**	**91.5**		MixText*	**98.5**	**98.9**	**99.2**
Yahoo!	VAMPIRE	-	59.9	70.2	IMDB	VAMPIRE	-	82.2	85.8
	BERT	56.2	69.3	73.2		BERT	67.5	86.9	89.8
	TMix*	58.6	69.8	73.5		TMix*	69.3	87.4	90.3
	UDA	63.2	70.2	73.6		UDA	78.2	89.1	90.8
	MixText*	**67.6**	**71.3**	**74.1**		MixText*	**78.7**	**89.4**	**91.3**

Experiments







Experiments

Mixup Layers Set	Accuracy(%)
∅	69.5
{0,1,2}	69.3
{3,4}	70.4
{6,7,9}	71.9
{7,9,12}	**74.1**
{6,7,9,12}	72.2
{3,4,6,7,9,12}	71.6

Table 3: Performance (test accuracy (%)) on AG News with 10 labeled data per class with different mixup layers set for TMix. ∅ means no mixup.

Model	Accuracy(%)
MixText	**67.6**
- weighted average	67.1
- TMix	63.5
- unlabeled data	58.6
- all	56.2

Table 4: Performance (test accuracy (%)) on Yahoo! Answer with 10 labeled data and 5000 unlabeled data per class after removing different parts of MixText.

研究表明，多层编码器（例如BERT）能够捕获不同类型和层次的信息，例如3、4层为表面信息，6、7层为句法，7、9、12层为语义。

Challenges

- 数据集中各个类别的数量相差很大，缺少解决数据不平衡的有效方法
- 在专业性较强的数据集中分类效果有待提高
- 过于复杂的模型不能满足任务的时效性

- THE END