短文本分类
(Short Text Classification)

郭晨亮

Short Text Classification

- 一组预定义的类别标签，对一些短的文本进行分类
- 应用：Web搜索、广告匹配、情感分析、对话系统
- 方法分类：

1.显示特征表示（依靠人设计）：传统的NLP步骤建模、分块、标记、语法解析

2.隐式特征表示（使用机器学习、深度神经网络）

（1）BOW、n-gram、embedding（word2vec、fasttext、glove等）

（2）主题模型的方法

（3）朴素贝叶斯、SVM、决策树，最大熵（ME），K最近邻（KNN）

（4）CNN、RNN、LSTM、注意力机制

（5）融合外部知识 the Angles won the World Series. (Angles: baseball team name)

- 问题：

(1) 短文本并不总是遵循自然语言的语法

(2) 短文本缺乏上下文

(3) 短文本通常比较模棱两可，因为它们包含多义和错别字

Radical-Aware Attention-based Four Granularity model (RAFG)

- 中文与英文差异巨大，中文是一种从象形文字衍生的语言，部首也是语义的重要载体。

(1)如果几个字共享一个共同的部首，那么该部首通常是它们之间的核心语义关联。

- (2)如果几个词共享一个共同的字，那么这个字通常是它们之间的核心语义关联。
- 中文没有空格作为分隔符，单词粒度定义不清楚，需要分词。

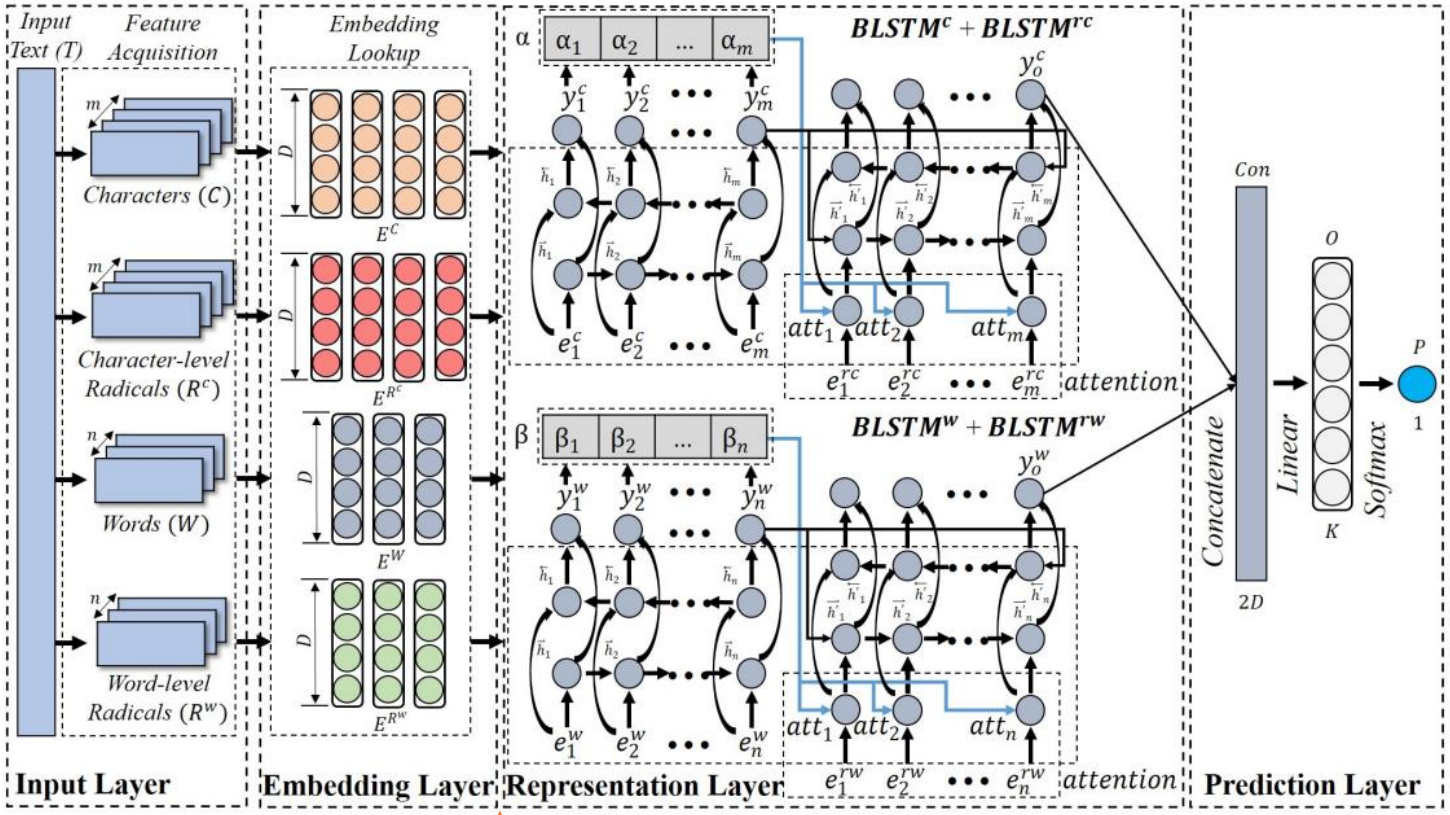
Chinese Characters	Radical	English
蝇	虫	fly
蚊	虫	mosquito
蜂	虫	bee
虱	虫	louse
蚁	虫	ant

Chinese Words	Chinese Characters	English
公牛	公 (male) + 牛 (cattle)	bull
母牛	母 (female) + 牛 (cattle)	cow
牛奶	牛 (cattle) + 奶 (milk)	milk
牛肉	牛 (cattle) + 肉 (meat)	beef
牛角	牛 (cattle) + 角 (horn)	horn

- 从字、词、字的部首、词的部首四个粒度对中文文本分类。（数字字母映射到’–’）

[1]A Radical-Aware Attention-Based Model for Chinese Text Classification, AAAI, 2019.

Radical-Aware Attention-based Four Granularity model (RAFG)



$$y_\epsilon^c \in Y^c = \{y_1^c, y_2^c, ..., y_m^c\}$$

$$y_\theta^w \in Y^w = \{y_1^w, y_2^w, ..., y_n^w\}$$

$$\alpha' = [\alpha_1', ..., \alpha_\epsilon', ..., \alpha_m'],\ \alpha_\epsilon' = f(y_\epsilon^c, e_i^{rc}), 1 \leqslant \epsilon \leqslant m,$$

$$\beta' = [\beta_1', ..., \beta_\theta', ..., \beta_n'],\ \beta_\theta' = f(y_\theta^w, e_j^{rw}), 1 \leqslant \theta \leqslant n,$$

$$\alpha_i = \frac{exp\,(\alpha_\epsilon')}{\sum_{\epsilon=1}^{m} exp\,(\alpha_\epsilon')},\ where \sum_{i=1}^{m} \alpha_i = 1,$$

$$\beta_j = \frac{exp\,(\beta_\theta')}{\sum_{\theta=1}^{n} exp\,(\beta_\theta')},\ where \sum_{j=1}^{n} \beta_j = 1,$$

$$\widetilde{e}_i^{rc} = \sum_{\epsilon=1}^{m} \alpha_\epsilon y_\epsilon^c,\ \widetilde{e}_j^{rw} = \sum_{\theta=1}^{n} \beta_\theta y_\theta^w,$$

$$\overrightarrow{h_t} = LSTM(\overrightarrow{h}_{t-1}, x_t),$$

$$\overleftarrow{h_t} = LSTM(\overleftarrow{h}_{t+1}, x_t),$$

$$y_t = [\overrightarrow{h_t}, \overleftarrow{h_t}],$$

$$O = sigmoid\,(Con \times W),$$

$$P = argmax\,(softmax\,(O)).$$

$$Loss = -\sum_{T \in Corpus} \sum_{i=1}^{K} p_i(T) \log p_i(T),$$

[1]A Radical-Aware Attention-Based Model for Chinese Text Classification, AAAI, 2019.

Radical-Aware Attention-based Four Granularity model (RAFG)

- 使用新华字典，共20849个字和270种部首
- 新闻标题分类
- Dataset#2是删除非中文比例大于20%的数据

Dataset		Count	Len (Avg. / Max)	Class
Dataset#1	Train	47,952	17.8 / 56	32
	Test	15,986	17.7 / 56	
Dataset#2	Train	36,431	16.7 / 46	32
	Test	12,267	16.7 / 43	

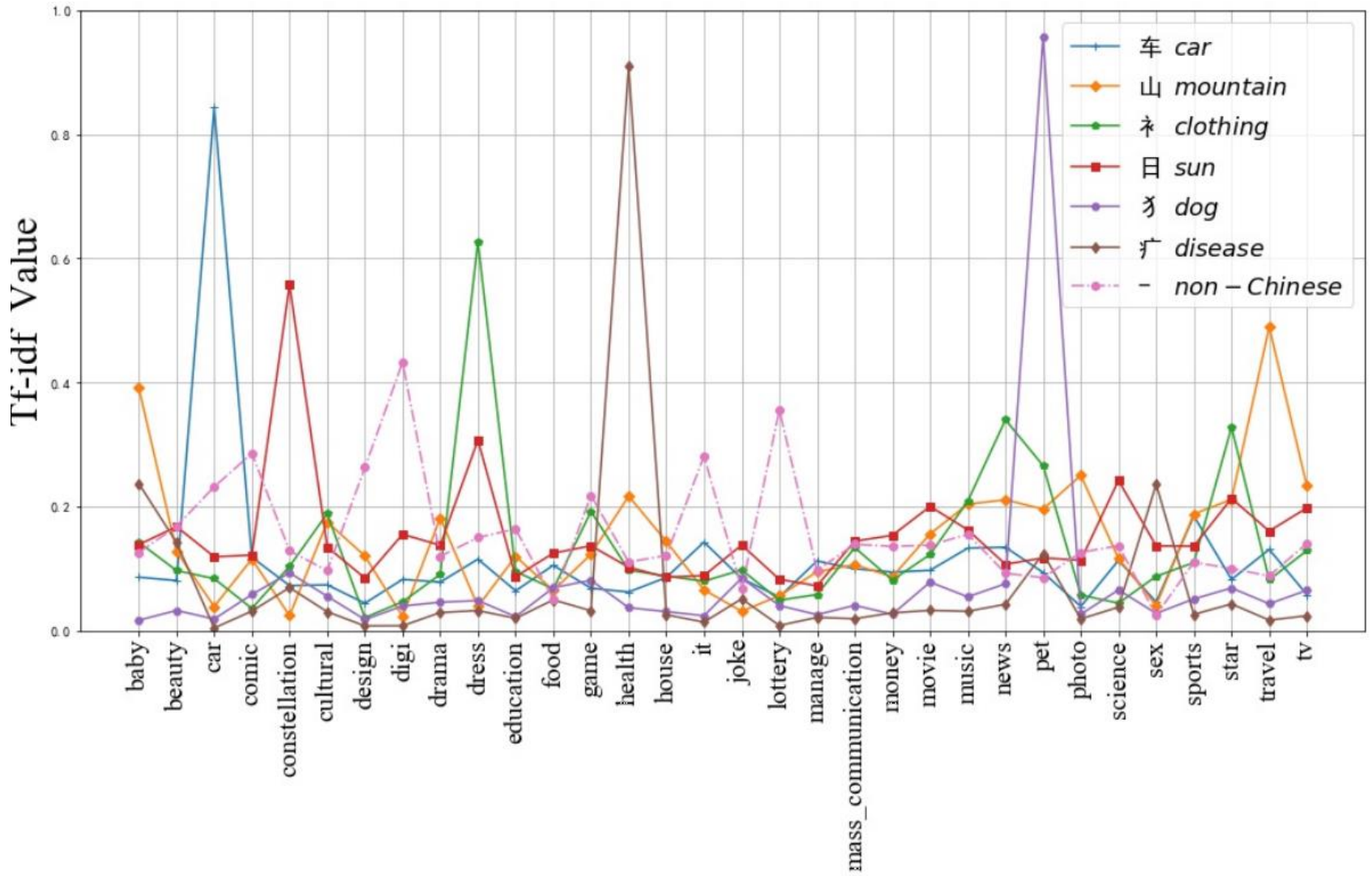
health	夏末秋初红眼病高发 白领群体患病居高不下
joke	损人真是件爆笑又过瘾滴事
digi	北京公交WiFi网络实测" 免费午餐不好拿"
joke	这是何必呢
movie	丸山隆平确定出演《圆桌》 扮演芦田爱菜班主
star	调查：朱莉为防癌切除乳腺你如何看待?
science	大蒜，你为什么那么受欢迎?
photo	只有美国人不爱无反相机?
photo	300幅摄影作品诠释人文澄迈(图)
pet	的哥养蝈蝈陪驾来解闷
photo	瑞典摄影师获第56届“荷赛”年度奖
music	《星搭档》蔡妍阿兰淘汰 李泉彩排情绪低

Methods	*Dataset#1*	*Dataset#2*
	F_1 (P, R)	F_1 (P, R)
SVM + BOW (W)	0.7552 (0.7639, 0.7514)	0.7341 (0.7459, 0.7303)
SVM + BOW (C)	0.7421 (0.7440, 0.7420)	0.7252 (0.7268, 0.7255)
SVM + BOW (R^w)	0.6834 (0.6913, 0.6800)	0.6762 (0.6858, 0.6729)
SVM + BOW (R^c)	0.4697 (0.4652, 0.4809)	0.4691 (0.4636, 0.4813)
LSTM (E^C)	0.7077 (0.7108, 0.7077)	0.6871 (0.6926, 0.6887)
LSTM (E^W)	0.8029 (0.8034, 0.8031)	0.7875 (0.7893, 0.7885)
Four LSTMs ($E^W + E^C + E^{R^w} + E^{R^c}$)	0.8072 (0.8078, 0.8074)	0.7904 (0.7912, 0.7910)
Four BLSTMs ($E^W + E^C + E^{R^w} + E^{R^c}$)	0.8098 (0.8103, 0.8103)	0.7915 (0.7925, 0.7921)
C-LSTMs ($E^W + E^C$)	0.8112 (0.8118, 0.8115)	0.7931 (0.7944, 0.7929)
C-BLSTMs ($E^W + E^C$)	0.8128 (0.8135, 0.8131)	0.7956 (0.7951, 0.7972)
Ours (RAFG)	**0.8181** (0.8181, 0.8187)	**0.7999** (0.7993, 0.8010)

- 部首无法反映足够的语义，在词中组合时才可以。
- BLSTM比单向LSTM效果好。

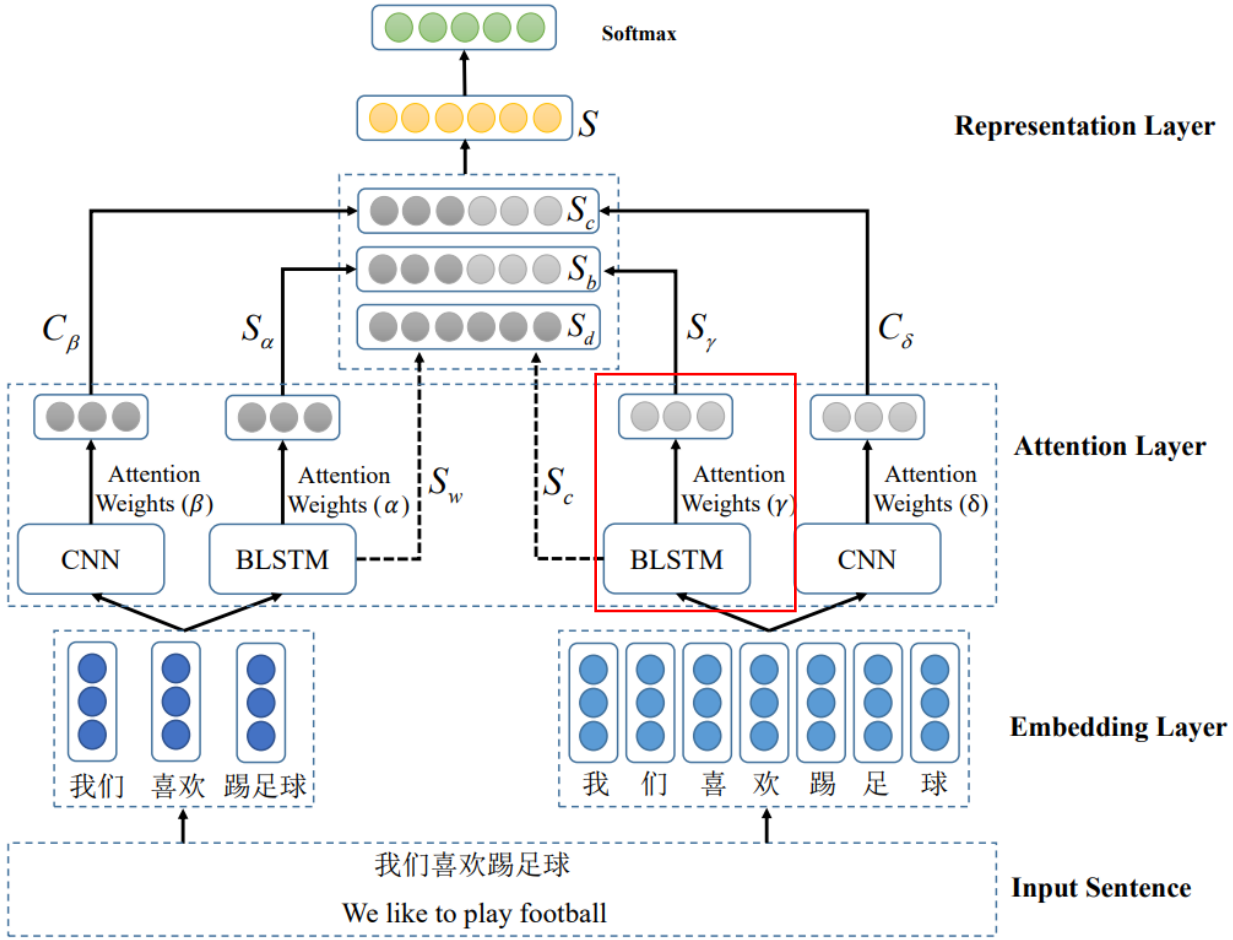
[1]A Radical-Aware Attention-Based Model for Chinese Text Classification, AAAI, 2019.

Radical-Aware Attention-based Four Granularity model (RAFG)



[1]A Radical-Aware Attention-Based Model for Chinese Text Classification, AAAI, 2019.

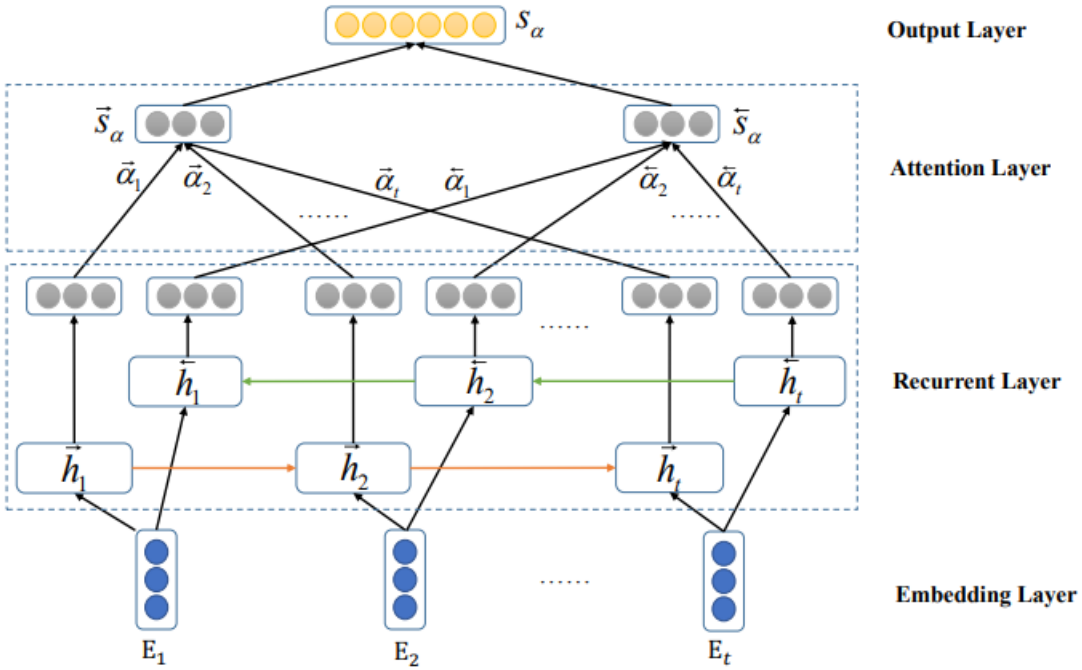
Hybrid Attention Networks(HANs)



$$u_i = \tanh(W_h h_i + b_h), \qquad v_i = \tanh(W_c C_i + b_c),$$

$$\alpha_i = softmax(W_\alpha u_i). \qquad \beta_i = softmax(W_\beta v_i),$$

$$S_\alpha = \sum_{i=1}^{l} \alpha_i h_i, \qquad C_\beta = \sum_{i=1}^{l} \beta_i C_i.$$



[2] Hybrid attention networks for Chinese short text classification. Computación y Sistemas, 2017.

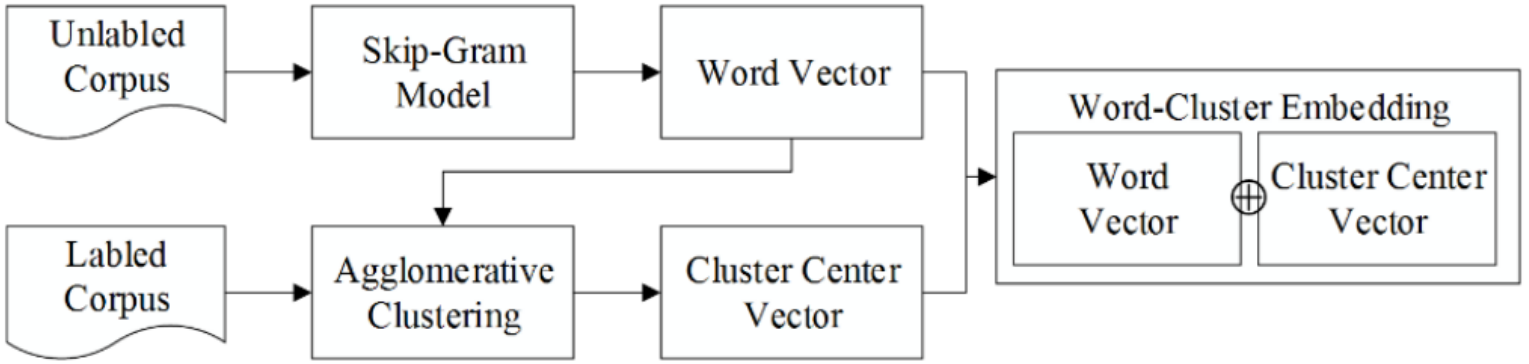
Hybrid Attention Networks(HANs)

Data Set	Class	$\|W\|$	$\|C\|$	Training	Test	Avg. #L	Max #L
Chinese news titles	32	62,167	4,721	47,850	15,950	18	64
Douban movie reviews	5	85,770	6,027	56,945	27,689	89	100
Weibo rumors	5	19,324	3,805	6,031	2,159	128	201

Methods	F_1 (Precision, Recall)		
	News Title	Movie Review	Weibo Rumor
CNN-char [21]	74.0 (74.2, 74.2)	9.2 (5.7, 23.9)	15.0 (28.6, 23.0)
CNN-word [21]	76.8 (77.8, 76.8)	35.4 (43.2, 35.5)	77.8 (79.2, 78.1)
LSTM-char [21]	77.3 (77.8, 77.4)	11.7 (26.9, 24.5)	17.6 (14.4, 26.4)
LSTM-word [21]	80.0 (80.2, 80.1)	37.3 (44.6, 35.8)	78.3 (78.9, 78.4)
BLSTM-char [21]	77.4 (77.7, 77.4)	35.6 (39.5, 36.5)	78.6 (79.7, 78.6)
BLSTM-word [21]	80.0 (80.4, 79.9)	39.5 (42.6, 38.6)	82.8 (85.4, 83.0)
RCNN-char [9]	76.7 (76.9, 76.9)	36.4 (38.0, 37.1)	80.4 (81.3, 80.5)
RCNN-word [9]	78.8 (79.5, 78.8)	40.1 (44.3, 39.5)	82.5 (84.3, 82.3)
FastText-char [6]	77.3 (77.4, 77.4)	34.0 (40.1, 33.1)	79.3 (80.0, 79.3)
FastText word [6]	78.5 (78.8, 78.5)	38.3 (42.6, 37.3)	80.2 (80.8, 80.1)
C-LSTMs [21]	82.1 (82.2, 82.1)	38.4 (45.1, 37.3)	81.0 (82.5, 81.3)
C-BLSTMs [21]	81.9 (82.1, 82.1)	39.5 (44.6, 38.2)	86.0 (86.8, 86.0)
HANs-CNN	81.9 (82.2, 81.8)	37.8 (44.5, 37.2)	81.4 (81.7, 81.5)
HANs-LSTM	82.3 (82.9, 82.4)	41.3 (46.3, 42.0)	86.4 (87.3, 86.3)
HANs-LSTM+CNN	82.5 (82.8, 82.5)	40.8 (45.3, 39.8)	83.8 (84.0, 83.8)
HANs BLSTM	82.7 (83.0, 82.6)	**41.6** (44.0, 42.4)	**87.9** (88.2, 87.9)
HANs-BLSTM+CNN	**83.1** (83.2, 83.2)	41.0 (45.7, 40.0)	84.1 (85.1, 84.1)

[2] Hybrid attention networks for Chinese short text classification. Computación y Sistemas, 2017.

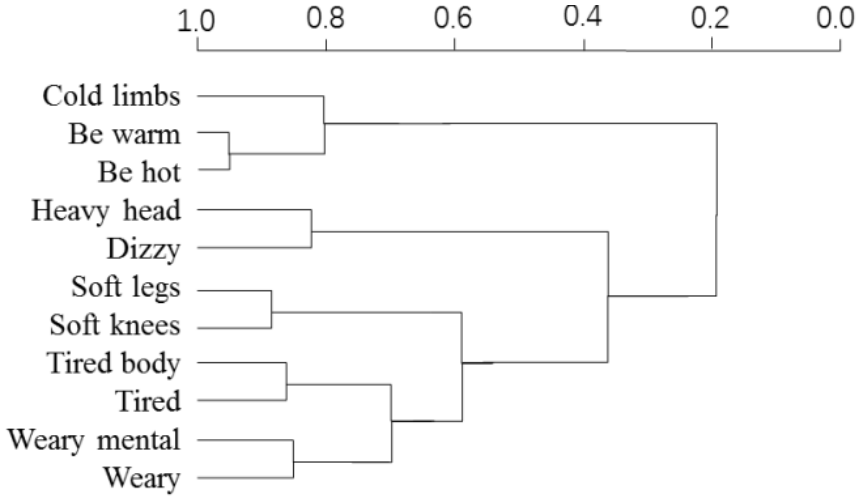
HAC-LSTM

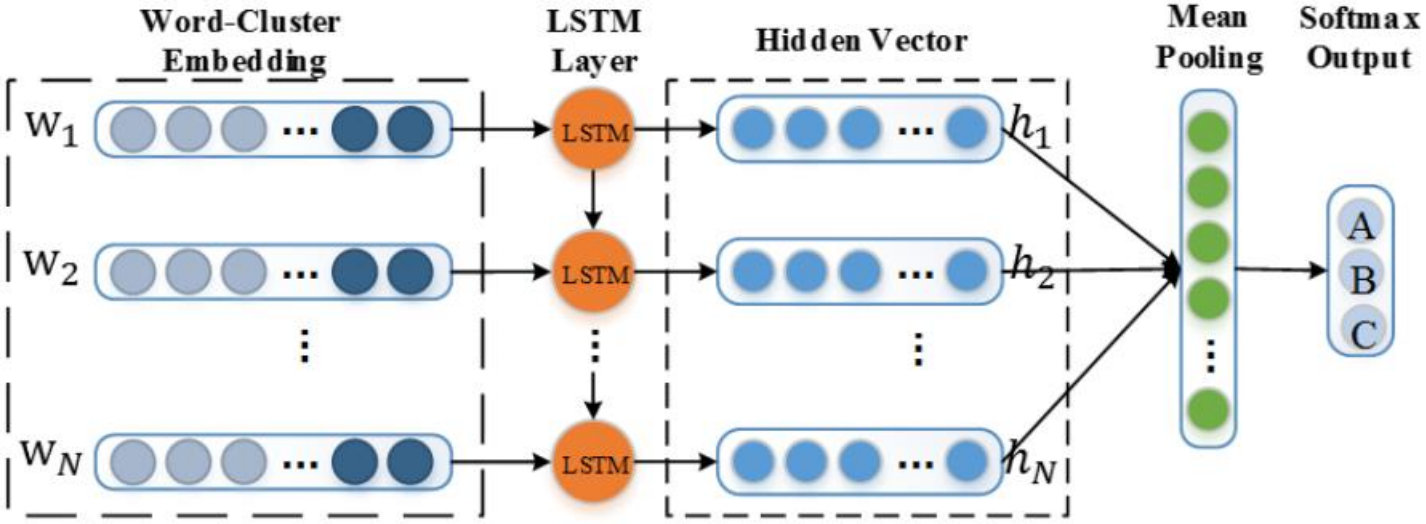


$$sim(u,v) = \frac{1}{1+\sqrt{\sum_{j=1}^{n}(u_j - v_j)^2}}$$

$$sim(A,B) = \frac{\sum_{u\in A, v\in B} sim(u,v)}{size(A)*size(B)}$$

$$C = \frac{1}{m}\sum_{i=1}^{m} V_i$$





[3] Improving medical short text classification with semantic expansion using word-cluster embedding. International Conference on Information Science and Applications. Springer, Singapore, 2018.

HAC-LSTM

Corpus	Words(million)
Baidu Encyclopedia (medical)[1]	37.8
Hudong Encyclopedia (medical)[2]	39.2
Chinese Wikipe[3]	44
General EMR	312.7
Stroke EMR	1.2
7th edition of Internal Medicine	0.4

Model	Accuracy	Reported in
LibLinear	0.908	Fan et al. 2008
C-LSTM	0.929	Zhou et al.,2015
RCNN	0.933	Lai et al.,2015
HAC-CNN	0.928	Our method
HAC-LSTM	**0.947**	**Our method**

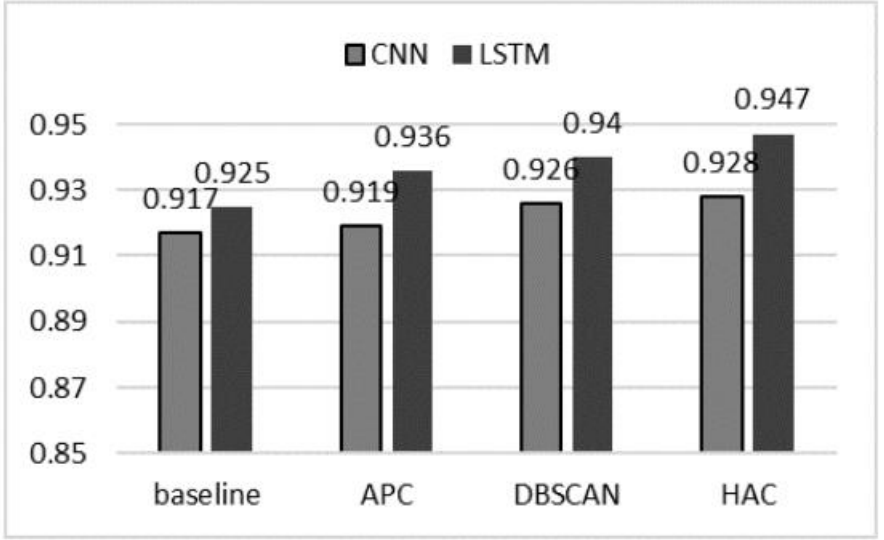
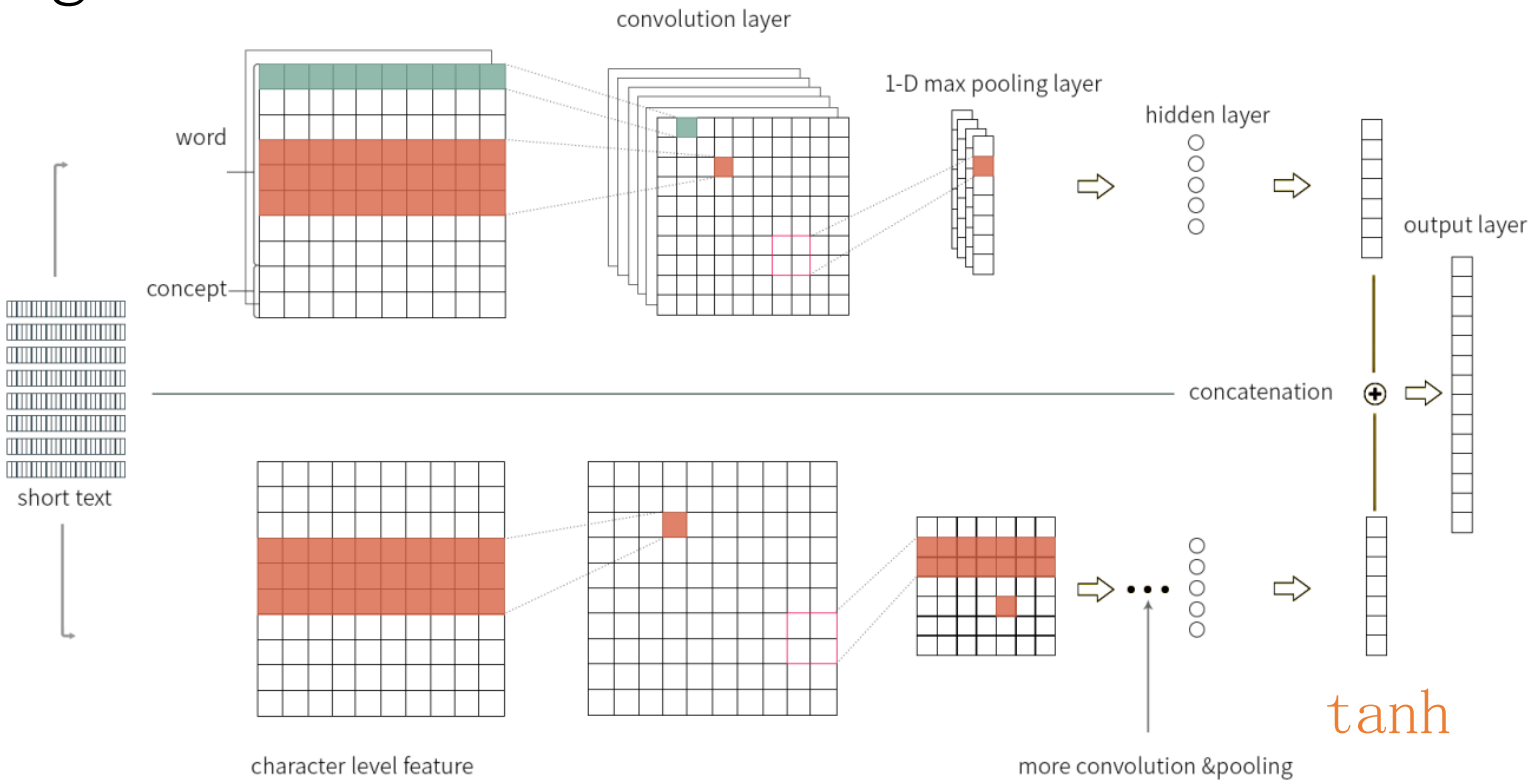


Table 3. The results of short text classification on TREC data sets

Model	Accuracy	Reported in
SVMs	0.95	Silva et al.,2011
TFIDF+SVMs	0.943	Wang et al.,2015
CNNs-non-static	0.936	Kim,2014
CNNs-multichannel	0.922	Kim,2014
DCNN	0.93	Kalchbrenner et al.,2014
DCNNs	0.956	Ma et al.,2015
Tree CNN	0.96	Komninos et al.,2016
Semantic-CNN	0.956	Wang et al.,2015
C-LSTM	0.946	Zhou et al.,2015
RCNN	0.96	Lai et al.,2015
HAC-LSTM	**0.977**	**Our method**

[3] Improving medical short text classification with semantic expansion using word-cluster embedding. International Conference on Information Science and Applications. Springer, Singapore, 2018.

Knowledge Powered Convolutional Neural Network (KPCNN)



$$\Theta_t = \Theta_{t-1} - \frac{\alpha}{\sqrt{\sum_{i=1}^{t} g_i}} g_t$$

- 将显性和隐性表示相结合，限制输入序列长度<256
- Probase的concept涵盖范围更广，可获得concept与短文本的相关性。
- “CNOOC Signed a PSC with ROC”

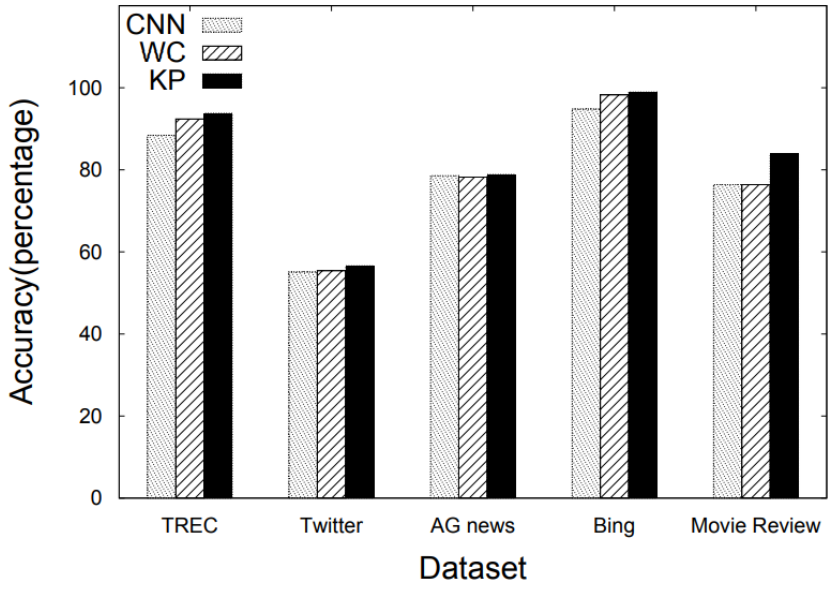
<client,0.9>, <channel,0.6>, <mythological creature,0.6>, <international well-known enterprise,0.2>, <chinese oil major,0.2>

[4] Combining Knowledge with Deep Convolutional Neural Networks for Short Text Classification, IJCAI, 2017.

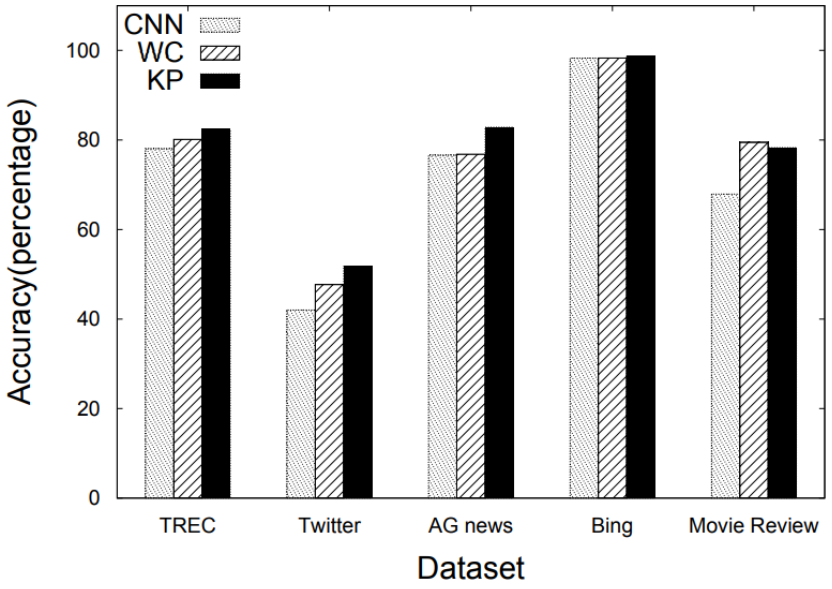
Knowledge Powered Convolutional Neural Network (KPCNN)

Datasets	#class	Training/Test set	Avg. Len
TREC	6	5952/500	10
Twitter	3	8,204/3,005	19
AG News	4	120,000/7,600	7
Bing	4	31,383/3,488	8
Movie Review	2	8,530/2,132	20

Model	TREC	Twitter	AG news	Bing	Movie Review
WC + LR	52.8	57.57	61.56	74.48	60.44
BoW + SVM	85.66	56.23	72.7	80.33	77.52
CNN	89.33	57.24	86.11	96.2	81.52
CharCNN	76	44.96	78.27	85.64	77.01
WCCNN	91.21	57.87	85.57	96.22	**83.77**
KPCNN	**93.46**	**59.84**	**88.36**	**99.17**	83.25



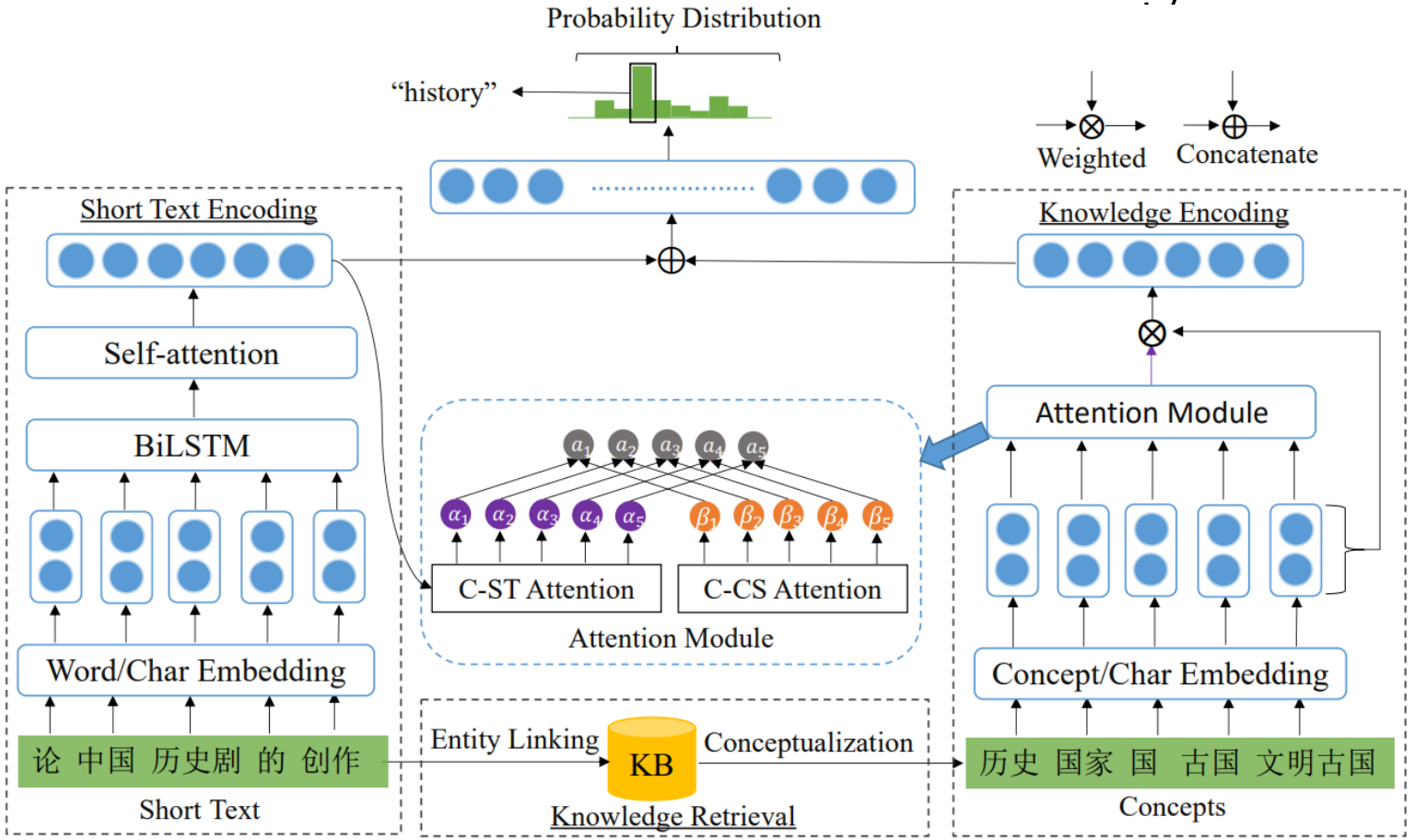
(a) Glove embedding



(b) Word-Concept embedding

[4] Combining Knowledge with Deep Convolutional Neural Networks for Short Text Classification, IJCAI, 2017.

Short Text Classification with Knowledge Powered Attention(STCKA)



$$A = Attention(Q, K, V) = softmax(\frac{QK^T}{\sqrt{2u}})V$$

Concept towards Short Text (C-ST)

$$\alpha_i = softmax(w_1^T f(W_1[c_i; q] + b_1))$$

Concept towards Concept Set (C-CS)

$$\beta_i = softmax(w_2^T f(W_2 c_i) + b_2)$$

$$a_i = softmax(\gamma\alpha_i + (1-\gamma)\beta_i)$$

$$= \frac{exp(\gamma\alpha_i + (1-\gamma)\beta_i)}{\sum_{k\in[1,m]} exp(\gamma\alpha_k + (1-\gamma)\beta_k)}$$

$$\gamma = \sigma(w^T[\alpha; \beta] + b)$$

$$p = \sum_{i=1}^{m} a_i c_i$$

- 将知识库中的先验知识结合起来丰富语义信息的注意力模型
- 考虑知识库中概念的粒度和相对重要性，加入注意力机制
- 衡量短文本及其相应概念之间的语义相似性，衡量每个概念相对于整个概念集的重要性

[5] Deep Short Text Classification with Knowledge Powered Attention, AAAI, 2019.

Short Text Classification with Knowledge Powered Attention(STCKA)

Datasets	# Class	Training/Validation/Test set	Avg. Chars	Avg. Words	Avg. Ent	Avg. Con
Weibo	7	3771/665/500	26.51	17.23	1.35	3.01
Product Review	2	7648/1350/1000	64.71	40.31	1.82	4.87
News Title	18	154999/27300/10000	20.63	12.02	1.35	2.72
Topic	20	6170/1090/700	15.64	7.99	1.77	4.50

Model	Weibo	Topic	Product Review	News Title
CNN	0.3900	0.8243	0.7290	0.7706
RCNN	0.4040	0.8257	0.7280	0.7853
CharCNN	0.4100	0.8500	0.7010	0.7493
BiLSTM-MP	0.4160	0.8186	0.7290	0.7719
BiLSTM-SA	0.4120	0.8200	0.7310	0.7802
KPCNN	0.4240	0.8643	0.7340	0.7878
STCKA	**0.4320**	**0.8814**	**0.7430**	**0.8011**

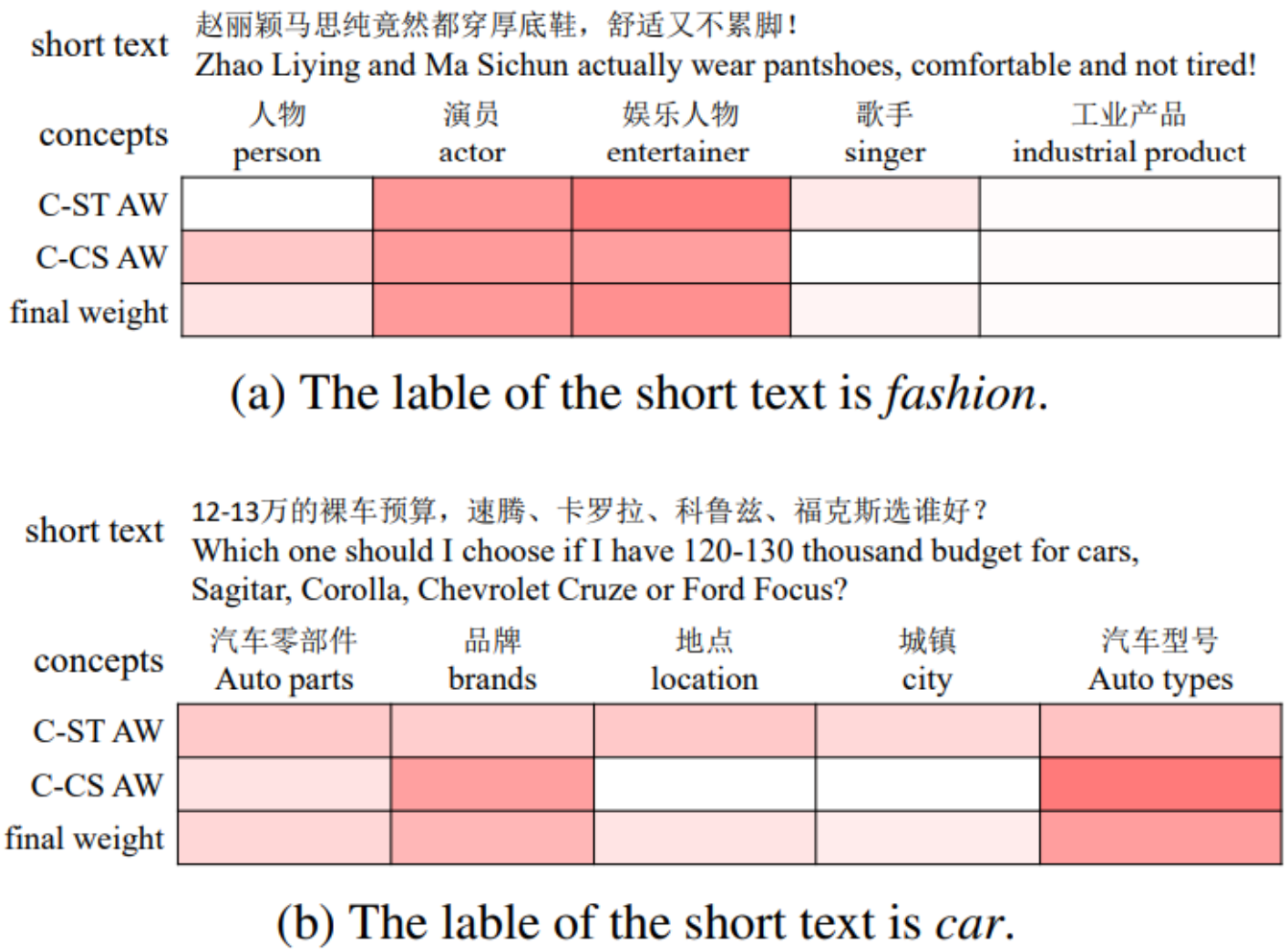
Model	Weibo	Topic	Product Review	News Title
STCKA($\lambda = 0.00$)	0.4280	0.8600	0.7390	0.7972
STCKA($\lambda = 0.25$)	**0.4320**	0.8700	**0.7430**	**0.8007**
STCKA($\lambda = 0.50$)	0.4260	**0.8786**	0.7380	0.8002
STCKA($\lambda = 0.75$)	0.4220	0.8643	0.7380	0.7959
STCKA($\lambda = 1.00$)	0.4160	0.8557	0.7360	0.7965

Model	Weibo	Topic	Product Review	News Title
STCKA-rand	0.3780	0.8414	0.7290	0.7930
STCKA-static	0.4240	0.8600	0.7350	0.7889
STCKA-non-static	**0.4320**	**0.8814**	**0.7430**	**0.8011**

- λ=0.25时通常最好，但和数据集相关
- STACK-non-static最好
- 小数据上static好，大量数据上rand好

[5] Deep Short Text Classification with Knowledge Powered Attention, AAAI, 2019.

Short Text Classification with Knowledge Powered Attention(STCKA)



Short text: 辛亥革命是犯了“激进主义”错误吗？
Did the Revolution of 1911 make a mistake of “radicalism”?

Concepts: 历史 history; 历史事件 historical event; 历史书籍 history book

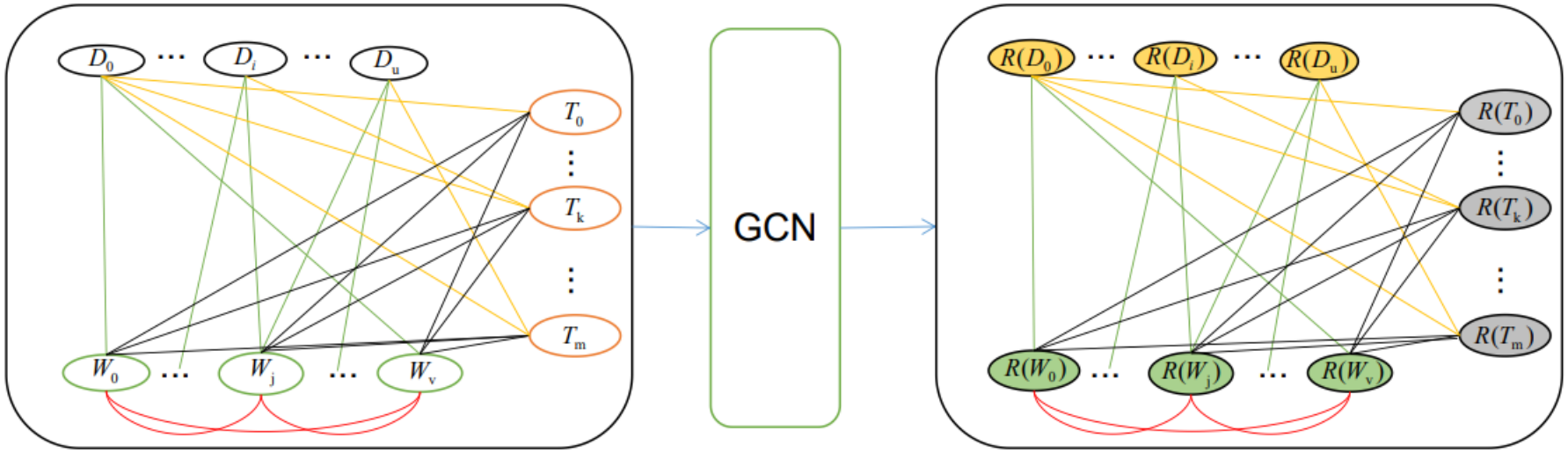
Short text: 南宁 铁路分局创安全新纪录
Nanning Railway Branch creates a new safety record.

Concepts: 交通术语 transportation term; 组织结构 organization; 城市 city

Figure 3: Two examples for power of knowledge. Underlined phrases are the entities, and the class labels of these two short texts are *history* and *transport* respectively.

[5] Deep Short Text Classification with Knowledge Powered Attention, AAAI, 2019.

short text graph convolutional network(STGCN)



biterm topic model (BTM)

$$b = (w_i, w_j) \qquad P(b) = \sum_t P(t)P(w_i|t)P(w_j|t) = \sum_t \theta_t \phi_{i|t} \phi_{j|t} \qquad P(B) = \prod_{i,j} \sum_t \theta_t \phi_{i|t} \phi_{j|t}$$

$$A_{ij} = \begin{cases} PMI(i,j) & i, j \text{ are words} \\ TF\text{-}IDF_{ij} & i \text{ is document, } j \text{ is word} \\ word\text{-}topic_{ij} & i \text{ is word, } j \text{ is topic} \\ doc\text{-}topic_{ij} & i \text{ is document, } j \text{ is topic} \\ 1 & i = j \\ 0 & otherwise \end{cases}$$

$$L^{(1)} = \rho(\tilde{A}XW_0) \quad X = I$$

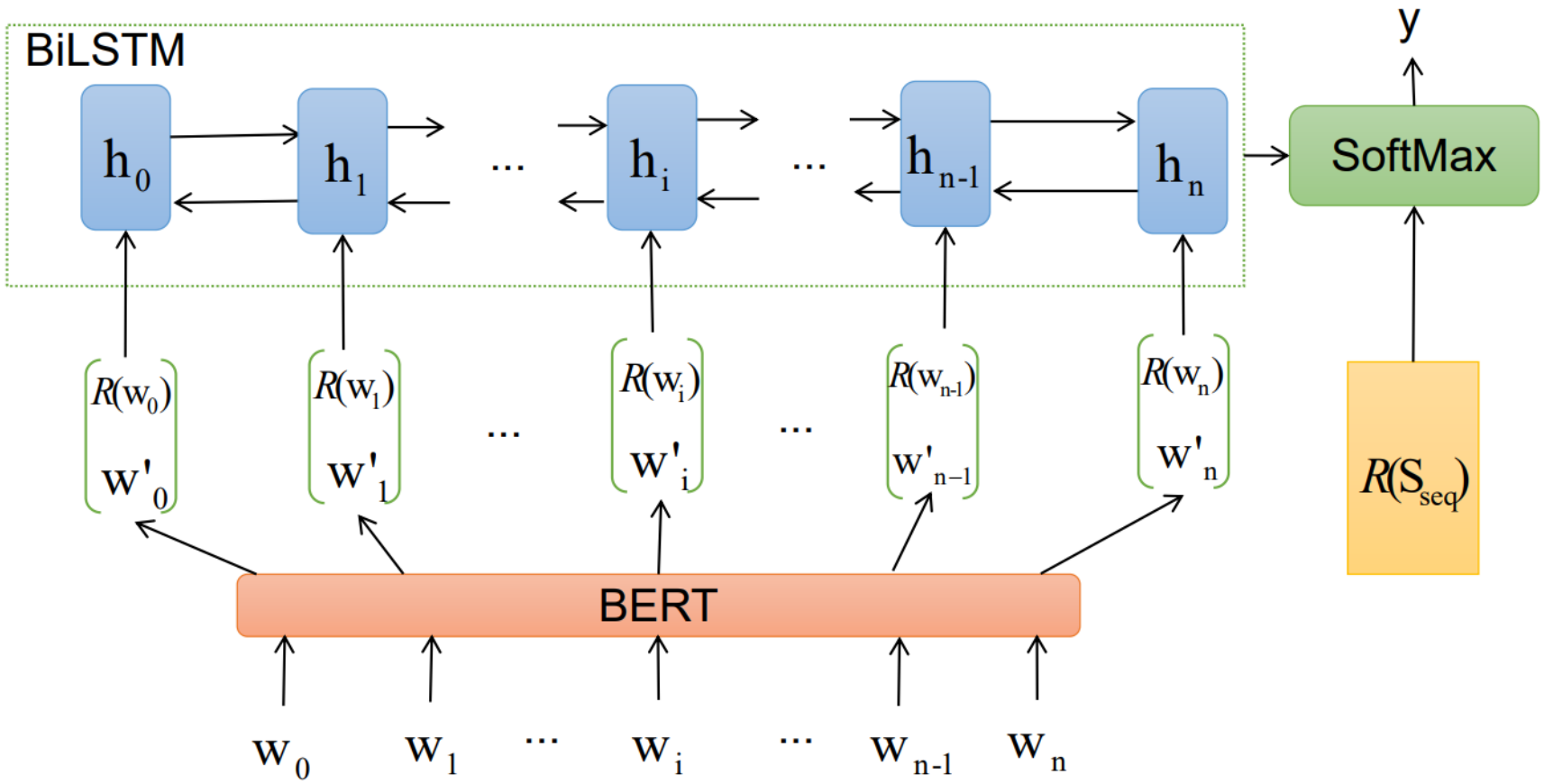
$$\tilde{A} = D^{-\frac{1}{2}} A D^{-\frac{1}{2}}$$

$$L^{(l+1)} = \rho(\tilde{A}L^{(l)}W_l)$$

$$\mathcal{L} = -\sum_{d \in y_D} \sum_{f=1}^{F} Y_{df} ln Z_{df}$$

[6] Document and Word Representations Generated by Graph Convolutional Network and BERT for Short Text Classification, ECAI, 2020.

short text graph convolutional network(STGCN)



[6] Document and Word Representations Generated by Graph Convolutional Network and BERT for Short Text Classification, ECAI, 2020.

short text graph convolutional network(STGCN)

Dataset	#Docs	#Classes	#Avg len	Vocab size
MR	10,662	2	20	18,764
Weibo	35,000	50	7	10,220
StackOverflow	20,000	20	8	6,762
Biomedical	20,000	20	18	6,004
R8	7674	8	66	7,688

Model	MR	Weibo	StackOverflow
Text GCN	0.767	0.534	0.814
LDA	0.772	0.530	0.815
NTM	0.776	0.540	0.825
NVDM	0.773	0.541	0.822
BTM	0.782	0.542	0.835

Models	MR	Weibo	StackOverflow	Biomedical	R8
TF-IDF+LR	0.746	0.501	—	—	0.937
CNN	0.777	0.524	0.823	0.701	0.957
LSTM	0.751	0.514	0.821	0.702	0.937
Bi-LSTM	0.777	0.522	0.821	0.705	0.963
fastText	0.751	0.524	—	—	0.961
Text GCN	0.767	0.534	0.814	0.680	0.970
Fine-tuning BERT	0.857	0.522	0.908	0.726	0.982
STGCN	0.782	0.542	0.835	0.690	0.972
STGCN+BiLSTM	0.785	0.546	0.857	0.728	—
STGCN+BERT+BiLSTM	**0.864**	**0.550**	**0.914**	**0.740**	**0.985**

- 主题信息有帮助
- 应该充分利用文档、短文本的特征
- Bert对效果有帮助

[6] Document and Word Representations Generated by Graph Convolutional Network and BERT for Short Text Classification, ECAI, 2020.