文本分类

袁浩达

目录

目录

问题定义

- 文本分类是在预定义的分类体系下根据文本的特征，将给定的文本与一个或多个类别相关联的过程。
- $\Phi: D \times C \rightarrow \{T, F\}$
  - $D = \{d_1, d_2, \ldots, d_{|D|}\}$表示要分类的文档
  - $C = \{c_1, c_2, \ldots, c_{|C|}\}$表示预定义得到分类体系下的类别集合
  - T表示对于$< d_i, c_j >$来说，文档d_i属于类c_j，F表示对于$< d_i, c_j >$来说，文档d_i不属于类c_j
- 文本分类中有两个关键问题，一个是文本的表示，另一个是分类器设计

宗成庆. 统计自然语言处理[M]. 清华大学出版社, 2013:416-417.

目录

评价指标

- 精确率（accuracy）
  - 分类器正确分类的样本数与总样本数之比
- 正确率、召回率、F值
  - 正确率 $P = \frac{TP}{TP+FP}$
  - 召回率 R $= \frac{TP}{TP+FN}$
  - F值 $F_\beta = \frac{(\beta^2+1)\times P\times R}{\beta^2\times P+R}$，$F_1 = \frac{P\times R}{P+R}$
- 多分类的评价指标
  - 宏平均：计算每个类别的的P、R、F值，再取平均
  - 微平均：按照公式直接计算P、R、F值

目录

传统方法

- 传统的文本表示方法
  - 基于one-hot表示使用tf-idf权重的词袋模型
    - One-hot表示：每个词使用向量中的一个维度来表示，假设所有的单词都是无关的
      - “狗”：[0, 0, ……, 0, 1, 0, 0, 0, ……, 0, 0]
      - “猫”：[0, 0, ……, 1, 0, 0, 0, 0, ……, 0, 0]
      - “汽车”：[0, 0, ……, 0, 0, 0, 0, 1, ……, 0, 0]
    - 词袋模型：构造词表，通过文本为词表中的词赋值，不考虑词之间的语义关系
      - “我爱吃西瓜”：{我，爱，吃，西瓜}
      - “我不爱吃西瓜”：{我，不，爱，吃，西瓜}
    - 评价
      - 数据稀疏：要采取复杂的平滑策略
      - 但是这种表示仍然有很强的判别能力，是一个很强的baseline

传统方法

- 传统的文本表示方法
  - 字符级的N-gram的表示（word hashing)
    - “good” -> ”#good#” -> ”#go”,”goo”,”ood”,”od#”
    - 可以对词表进行压缩
    - 可以处理未登录词（之前要用平滑）
    - 仍然没有语义信息
  - 还有其他特征和表示
    - LDA
    - BM25
    - ……

传统方法

- 分类器
  - 朴素贝叶斯
  - 支持向量机（tfidf+ SVM经常被用作baseline）
  - K-近邻
  - 决策树
  - ……

宗成庆. 统计自然语言处理[M]. 清华大学出版社, 2013:424-428.

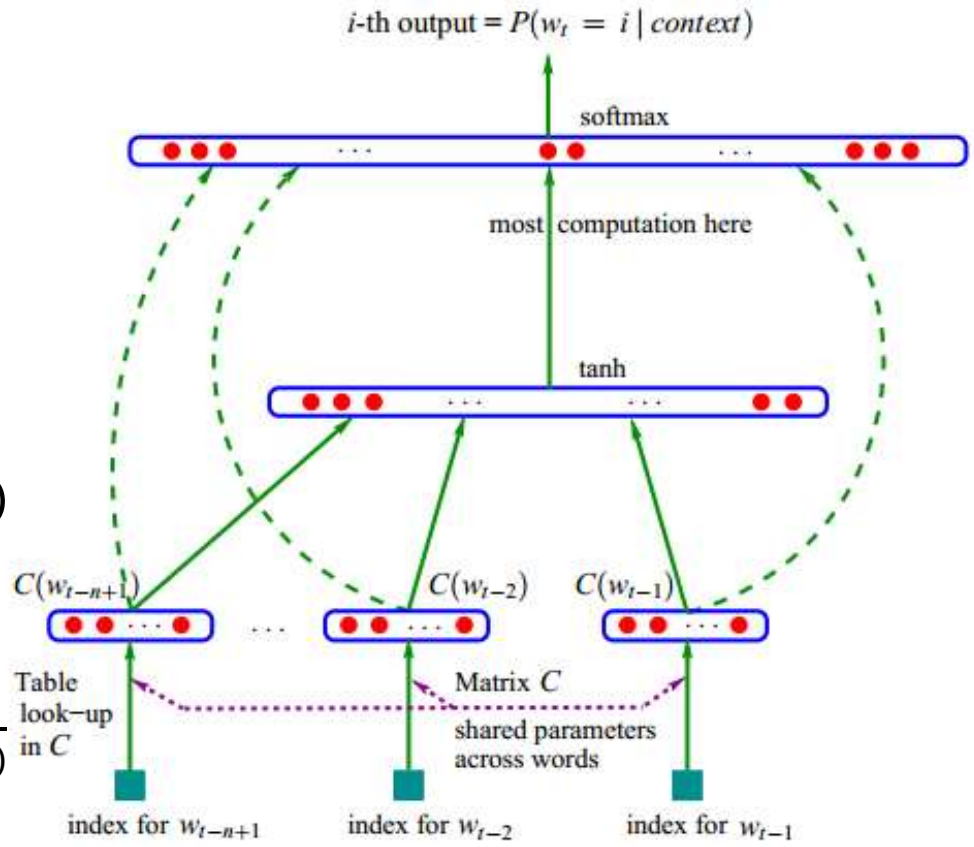
目录

深度学习的方法

- 分布式表示
  - 传统的文本表示的不足之处
    - 维度灾难，神经网络不擅长处理这类的数据
    - 不能表示词与词之间的语义上的联系
  - 分布式表示
    - 与传统one-hot表示中每个词占用向量空间中一个维度不同的是，分布式表示中每个词占用向量空间中的全部维度。

深度学习的方法

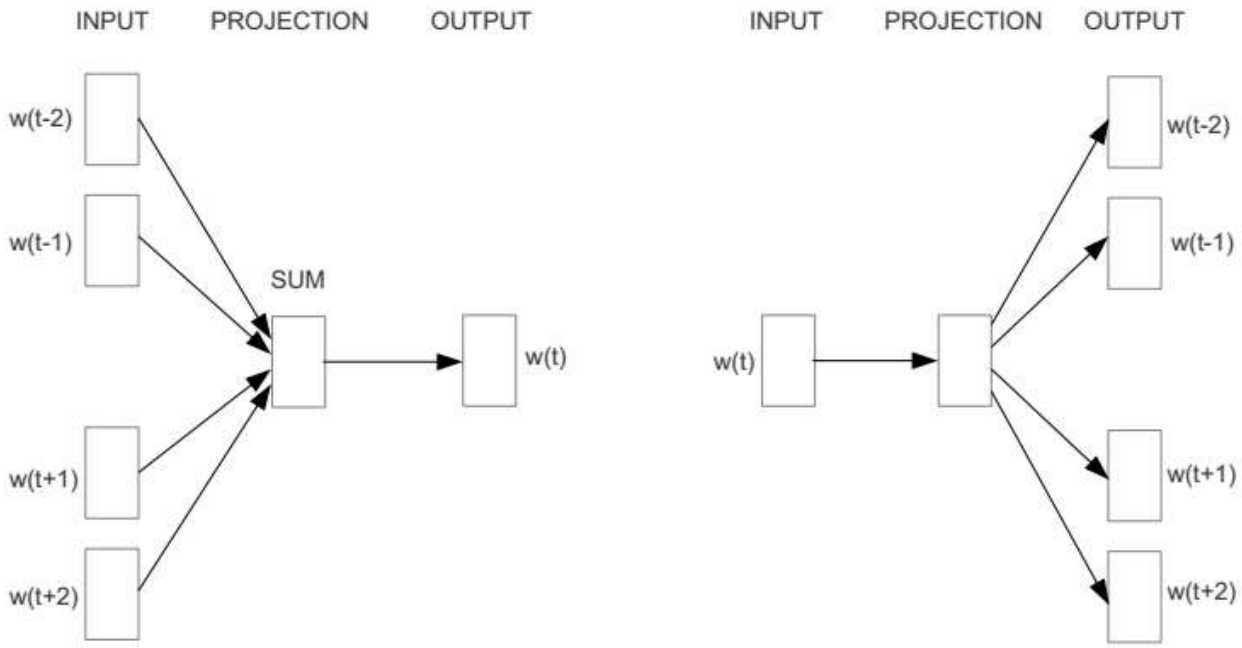
- 分布式表示
  - Neural Probabilistic Language Model(NPLM)
    - Embedding层+一个隐层的神经网络
    - Embedding层
      - Matrix C $w_i \longrightarrow C(w_i)$
    - History: $x = (C(w_{t-1}), C(w_{t-2}), \dots, C(w_{t-n+1}))$
    - 非线性隐层
      - $y = b + Wx + Utanh(d + Hx)$
    - Sotfmax: $P(w_t|w_{t-1}w_{t-2} \dots w_{t-n+1}) = \frac{\exp(y_{w_i})}{\sum_i \exp(y_i)}$
    - 复杂度过高



Bengio Y, Ducharme R, Vincent P, et al. A neural probabilistic language model[J]. Journal of machine learning research, 2003, 3(Feb): 1137-1155.

深度学习的方法

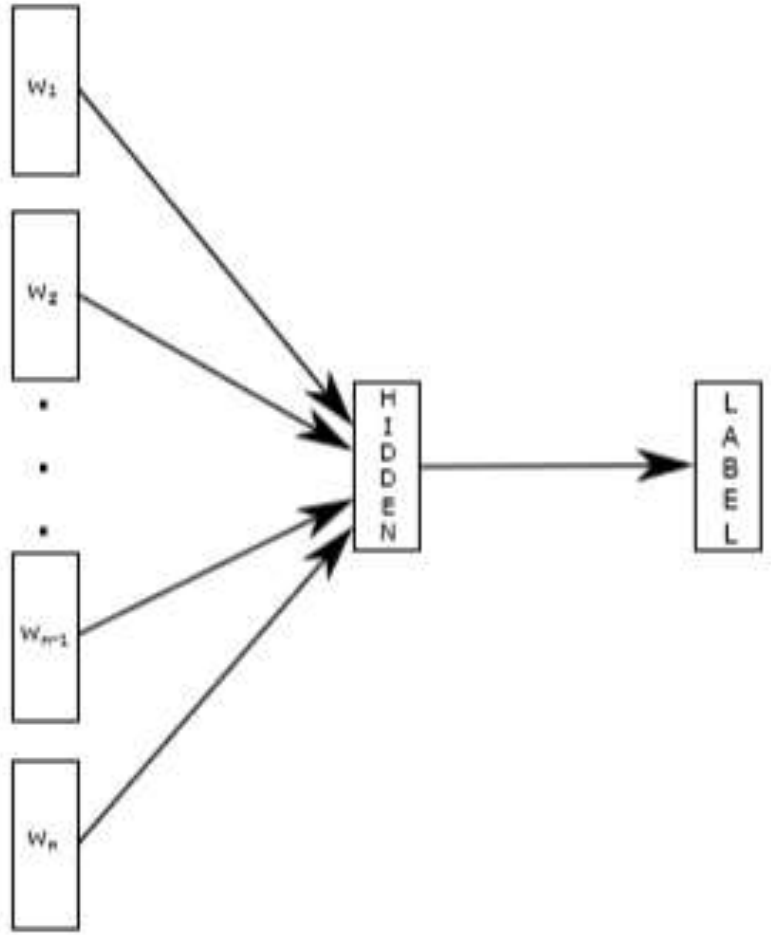
- 分布式表示
  - Word2vec
    - 更简单的网络模型
    - Continuous Bag-of-Words(CBOW)
      - 用上下文预测中间的词
      - $\boldsymbol{h} = \sum_{t=-c}^{c} \boldsymbol{w}(t-c)$
      - $p(w_i|\boldsymbol{c}) = \frac{\exp(\boldsymbol{w}_i \cdot \boldsymbol{h}_i)}{\sum_{\boldsymbol{w}'} \exp(\boldsymbol{w}' \cdot \boldsymbol{h}_i)}$
    - Skip-gram
      - 用一个词去预测其他的词



Mikolov T, Chen K, Corrado G, et al. Efficient estimation of word representations in vector space[J]. arXiv preprint arXiv:1301.3781, 2013.

深度学习的方法

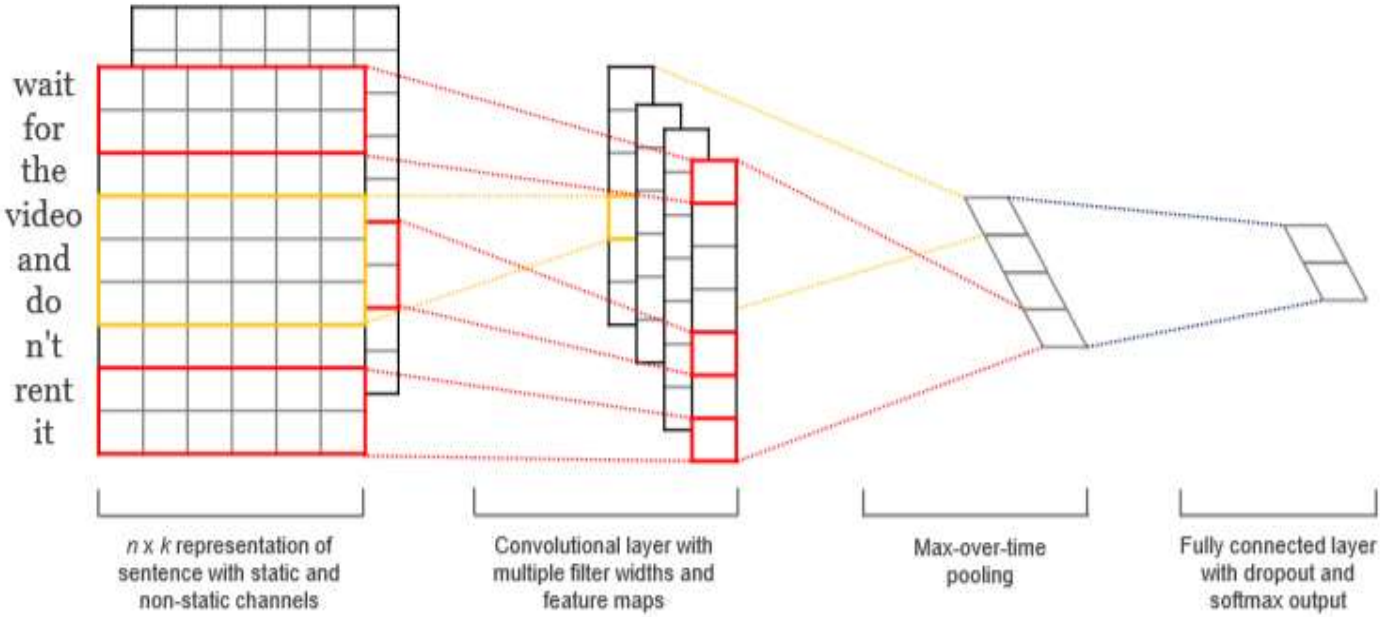
- 文本分类模型- Fast Text
  - 模型结构
    - 只有一个隐层的神经网络
    - 分层softmax
    - N-gram 特征
  - 效果
    - 用了一个简单得模型和一些trick在特定数据集上取得了不错的效果
    - 训练的速度很快
    - 基于词袋假设只能在对词序不敏感的数据集上才会表现得好



Joulin A, Grave E, Bojanowski P, et al. Bag of Tricks for Efficient Text Classification[J]. 2016:427-431.

深度学习的方法

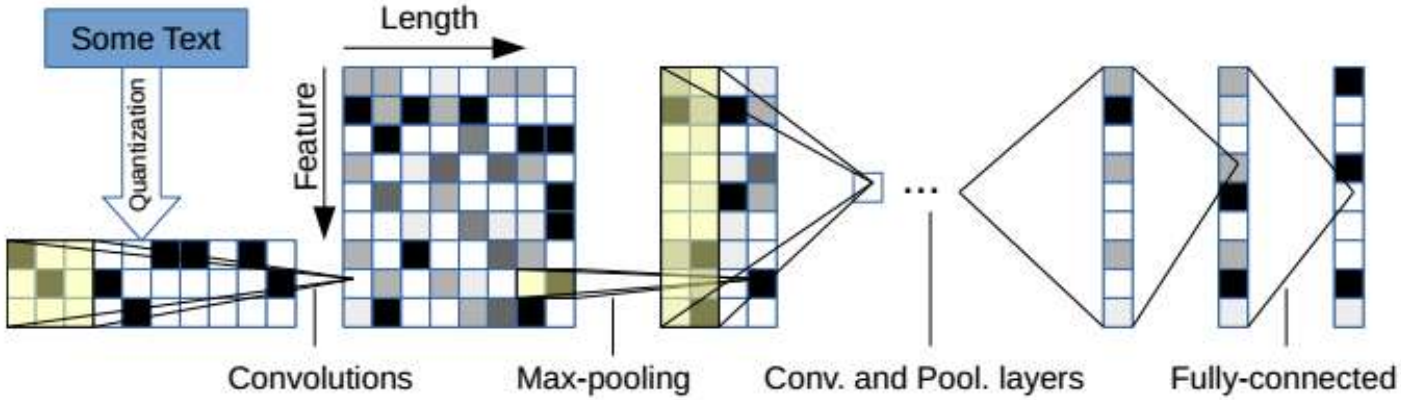
- 文本分类模型- CNN Text
  - 动机
    - 线性模型不能利用上下文信息
  - 模型结构
    - 特征：预训练的word2vec
    - 通道：静态词向量、fine-tunning
    - 卷积：不同大小的filters提取出不同的特征
    - 池化：最大值池化
    - 全连接和softmax



Kim Y. Convolutional Neural Networks for Sentence Classification[J]. Eprint Arxiv, 2014.

深度学习的方法

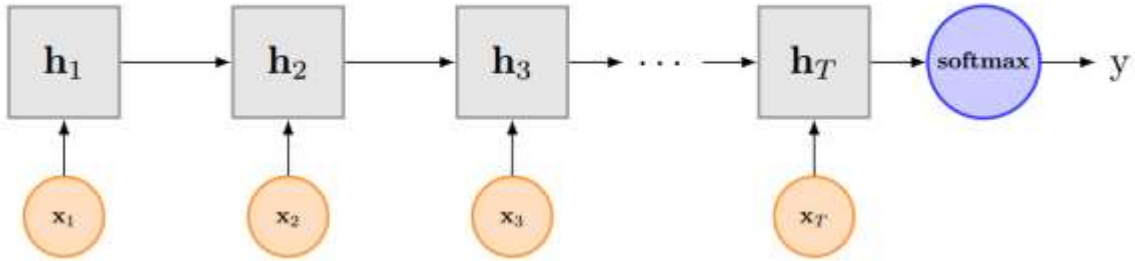
- 文本分类模型- Char-CNN
  - 动机
    - 字符级的模型不需要预训练词向量，也不需要句法结构
    - 可以推广到所有的语言
  - 模型结构
    - 字符编码
    - 同义词替换
    - 卷积-池化



Zhang X, Zhao J, LeCun Y. Character-level convolutional networks for text classification[C]//Advances in neural information processing systems. 2015: 649-657..

深度学习的方法

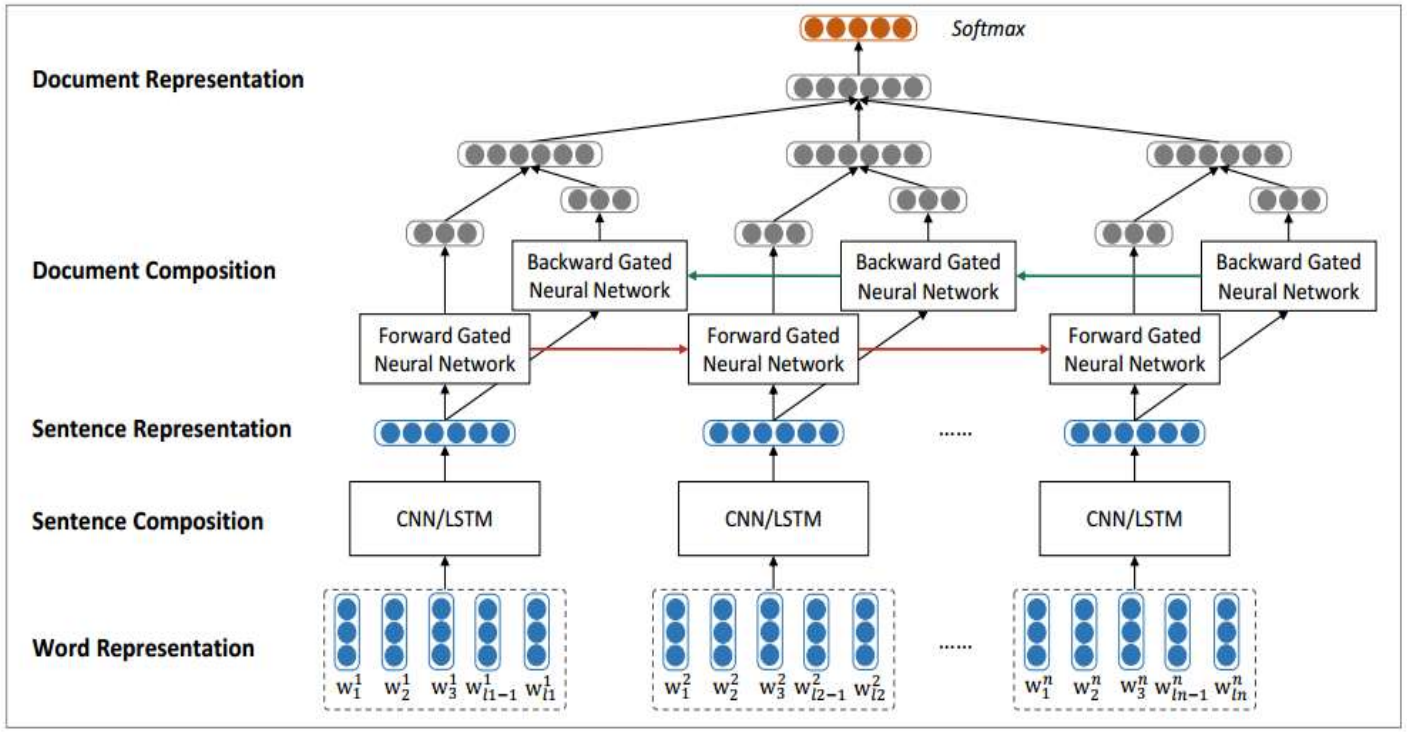
- 文本分类模型- RNN Text
  - 动机
    - CNN受卷积核的限制
    - RNN处理不定长度的序列数据
  - 模型结构
    - LSTM
    - 输入门 $\boldsymbol{i}_t = \sigma(\boldsymbol{W}_i\boldsymbol{x}_t + \boldsymbol{U}_i\boldsymbol{h}_{t-1} + \boldsymbol{V}_i\boldsymbol{c}_{t-1})$　　σ是sigmod函数
    - 遗忘门 $\boldsymbol{f}_t = \sigma(\boldsymbol{W}_f\boldsymbol{x}_t + \boldsymbol{U}_f\boldsymbol{h}_{t-1} + \boldsymbol{V}_f\boldsymbol{c}_{t-1})$
    - 输出门 $\boldsymbol{o}_t = \sigma(\boldsymbol{W}_o\boldsymbol{x}_t + \boldsymbol{U}_o\boldsymbol{h}_{t-1} + \boldsymbol{V}_o\boldsymbol{c}_t)$
    - $\tilde{\boldsymbol{c}}_t = \tanh(\boldsymbol{W}_c\boldsymbol{x}_t + \boldsymbol{U}_c\boldsymbol{h}_{t-1})$
    - 记忆单元 $\boldsymbol{c}_t = \boldsymbol{f}_t \odot \boldsymbol{c}_{t-1} + \boldsymbol{i}_t \odot \tilde{\boldsymbol{c}}_t$　　⊙表示对应元素分别相乘
    - 隐状态 $\boldsymbol{h}_t = \boldsymbol{o}_t \odot \tanh(\boldsymbol{c}_t)$

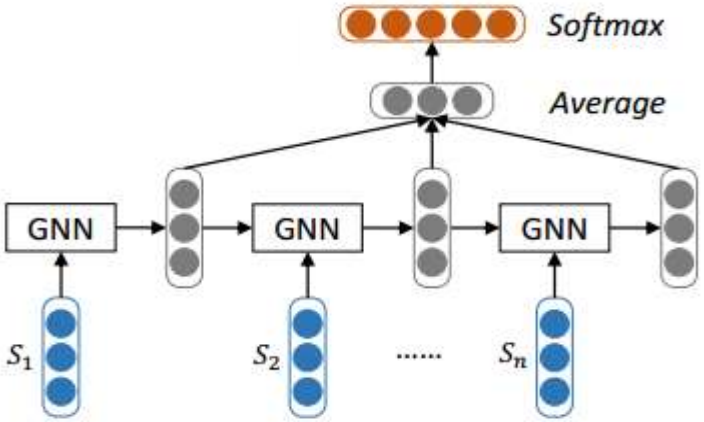


Liu P, Qiu X, Huang X. Recurrent Neural Network for Text Classification with Multi-Task Learning[J]. 2016:2873-2879.

深度学习的方法

- 文本分类模型- Gate RNN
  - 动机
    - 捕捉文章内句子间的语义关系
    - RNN建模长序列时梯度消失或爆炸
  - 模型结构
    - Step1：用CNN/LSTM由word vector得到sentence vector
    - Step2：GatedRNN自适应地编码句子的语义和句子之间的关系

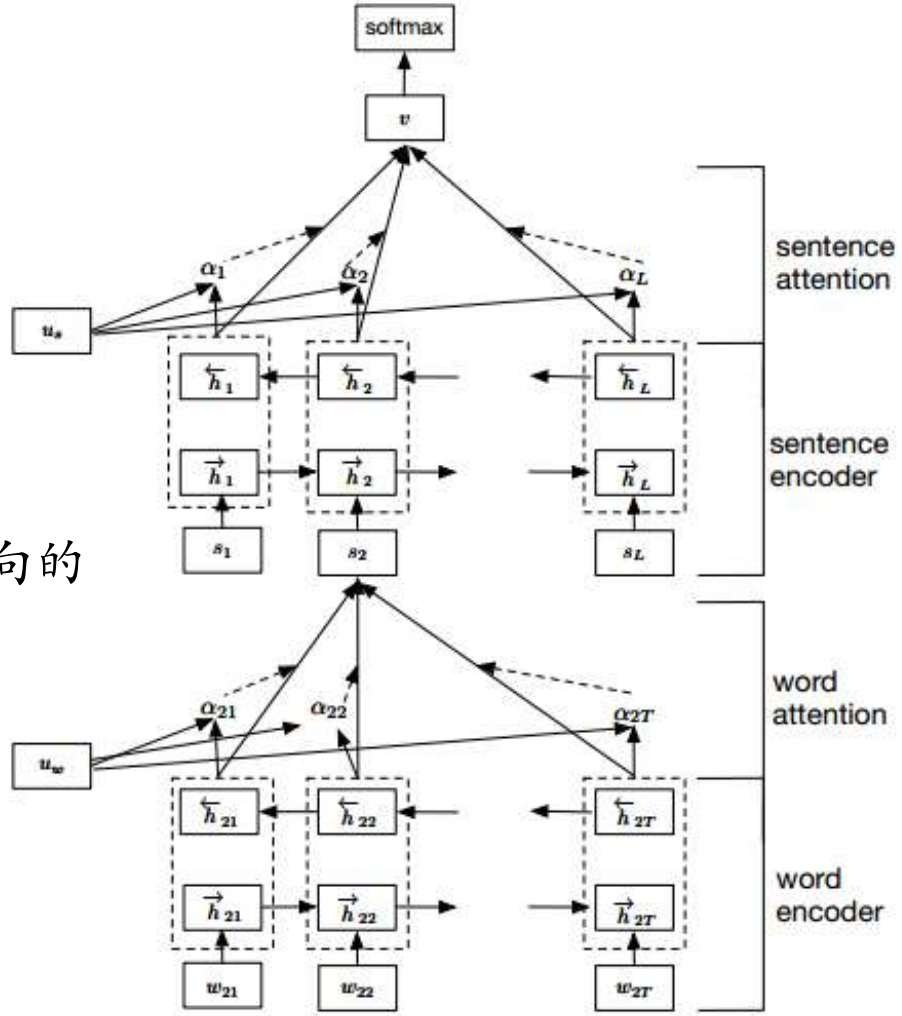




Tang D, Qin B, Liu T. Document modeling with gated recurrent neural network for sentiment classification[C]//Proceedings of the 2015 conference on empirical methods in natural language processing. 2015: 1422-1432.

深度学习的方法

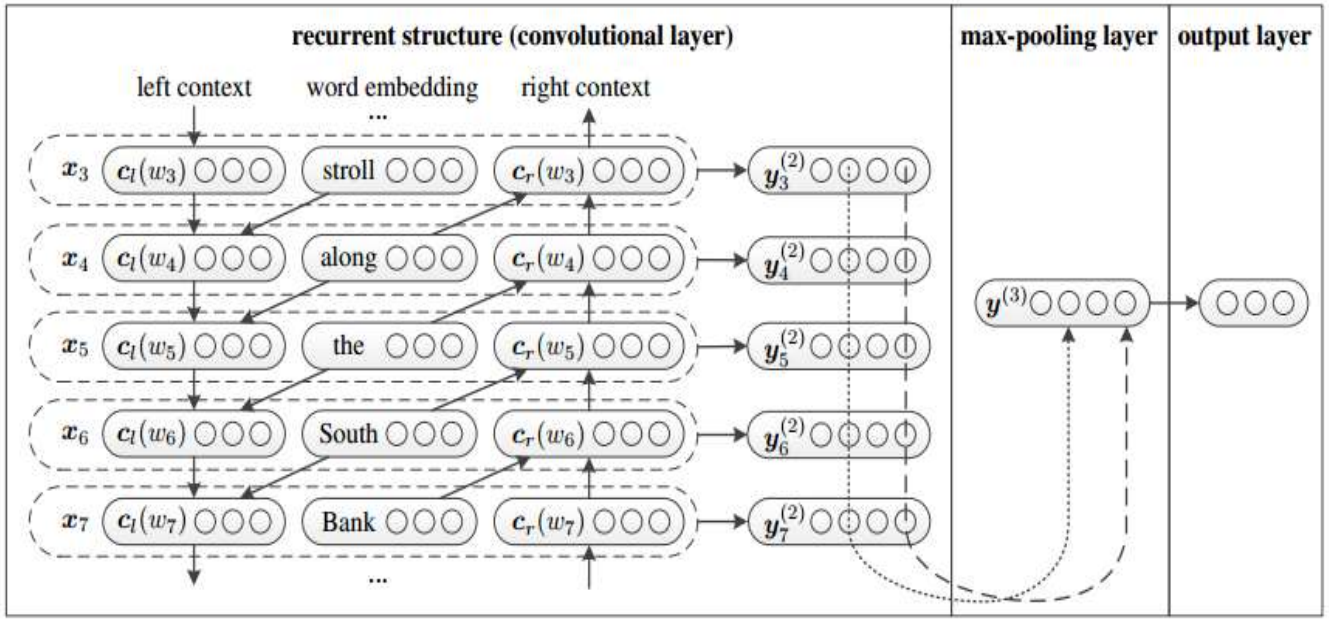
- 文本分类模型- RNN with Attention
  - 动机
    - 为了提高模型的可解释性
  - 模型结构
    - 双向GRU作为Encoder，最终的隐状态由两个方向的隐状态拼接起来
    - 词粒度的attention和句子粒度的attention
    - 得到文章向量表示v，接softmax得到分类结果
    - 注意力机制最大的好处是能够直观的解释各个句子和词对分类类别的重要程度



Yang Z, Yang D, Dyer C, et al. Hierarchical Attention Networks for Document Classification[C]// Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies. 2017:1480-1489.

深度学习的方法

- 文本分类模型- RCNN
  - 动机
    - 为了能够更加精确地捕捉到关键信息
  - 模型结构
    - 用一种循环结构作为卷积层
    - 词表示：词向量和上下文向量



上文向量 $c_l(w_i) = f(\boldsymbol{W}^{(l)}\boldsymbol{c}_l(w_{i-1}) + \boldsymbol{W}^{(sl)}\boldsymbol{e}(w_{i-1}))$

下文向量 $c_r(w_i) = f(\boldsymbol{W}^{(r)}\boldsymbol{c}_r(w_{i+1}) + \boldsymbol{W}^{(sr)}\boldsymbol{e}(w_{i+1}))$

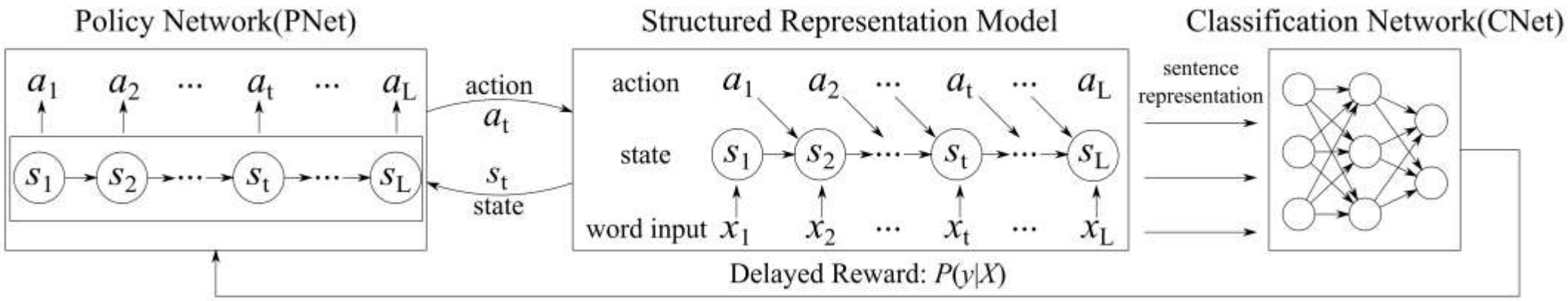
词w_i的向量表示 $\boldsymbol{x}_i = [c_l(w_i); \boldsymbol{e}(w_i); c_r(w_i)]$

- 句子表示：最大值池化

Lai S, Xu L, Liu K, et al. Recurrent Convolutional Neural Networks for Text Classification[C]//AAAI. 2015, 333: 2267-2273.

深度学习的方法

- 文本分类模型- LSTMs
  - 动机
    - 用RL的方法捕捉结构结构信息和语义信息，得到文本表示
    - 用ID-LSTM捕捉语义上的信息，HS-LSTM捕捉结构信息



Zhang T, Huang M, Zhao L. Learning Structured Representation for Text Classification via Reinforcement Learning[C]. AAAI, 2018.

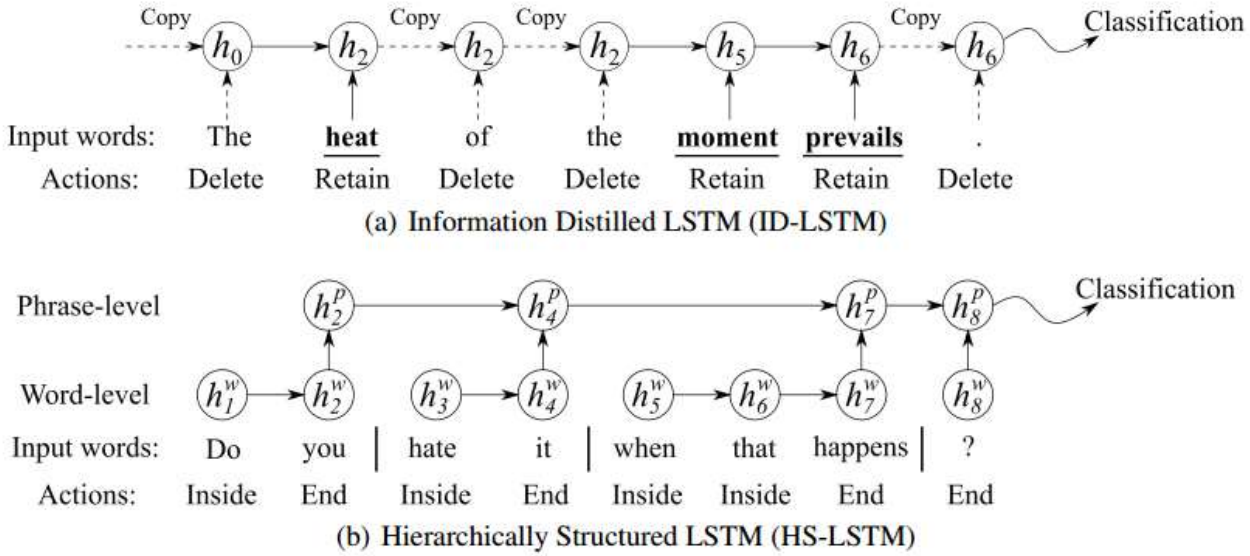
深度学习的方法

- 文本分类模型- LSTMs
  - 强化学习框架：PNet
    - 策略 $\pi(a_t|\boldsymbol{s}_t;\Theta)$选择action a_t的概率
    - State
      - 不同的LSTM不同
    - Action & Policy
      - ID-LSTM: $\{Retain, Remove\}$
      - HS-LSTM: $\{Inside,\ End\}$
      - $a_t^* = \underset{a}{\operatorname{argmax}}\,\pi(a|\boldsymbol{s}_t;\Theta) = \underset{a}{\operatorname{argmax}}\,\sigma(\boldsymbol{W} * \boldsymbol{s}_t + \boldsymbol{b})$
    - Reward
      - 不同的LSTM的reward计算不同
    - 目标函数
      - 最大化excepted reward

Zhang T, Huang M, Zhao L. Learning Structured Representation for Text Classification via Reinforcement Learning[C]. AAAI, 2018.

深度学习的方法

- 文本分类模型- LSTMs
  - 强化学习框架：SRM
    - ID-LSTM
      - $X = x_1 x_2 \ldots x_L$
      - $A = a_1 a_2 \ldots a_L$
      - Copy机制
      - State $s_t = c_{t-1} \oplus h_{t-1} \oplus x_t$
      - Reward $R_L = logP(c_g|X) + \gamma \frac{L'}{L}$
    - HS-LSTM
      - 分层的LSTM
      - State $s_t = c_{t-1}^P \oplus h_{t-1}^P \oplus c_t^w \oplus h_t^w$
      - Reward $R_L = logP(c_g|X) + \gamma(\frac{L'}{L} + 0.1\frac{L}{L'})$



Zhang T, Huang M, Zhao L. Learning Structured Representation for Text Classification via Reinforcement Learning[C]. AAAI, 2018.

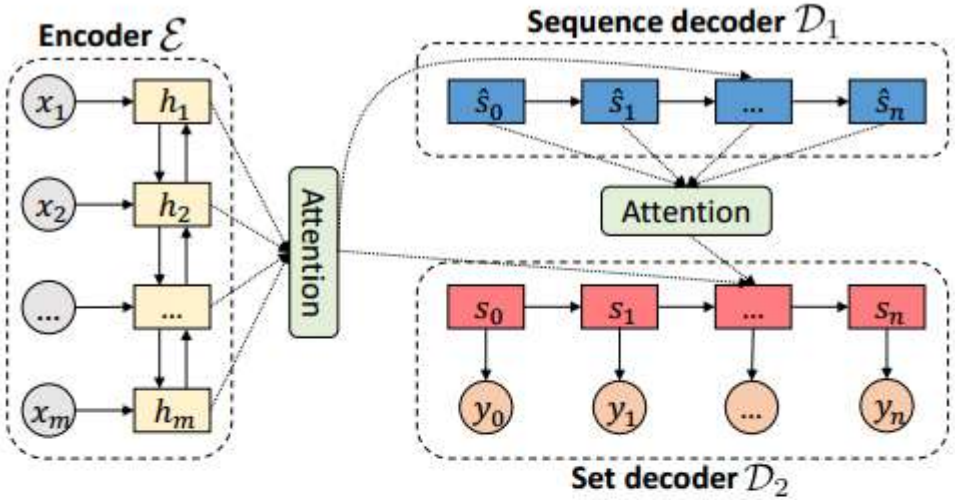
深度学习的方法

- 文本分类模型- LSTMs
  - 强化学习框架：CNet
    - $P(y|X) = softmax(\boldsymbol{W}_s h + \boldsymbol{b}_s)$

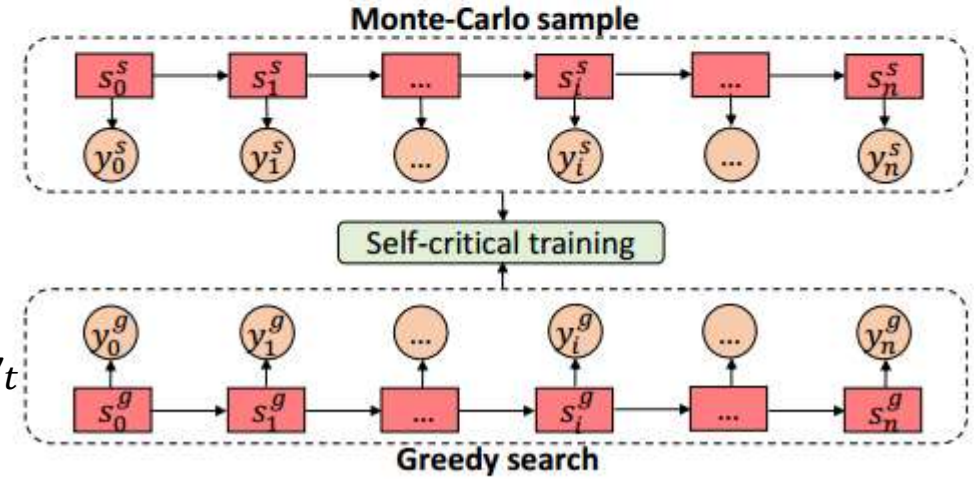
Zhang T, Huang M, Zhao L. Learning Structured Representation for Text Classification via Reinforcement Learning[C]. AAAI, 2018.

深度学习的方法

- 文本分类模型- SequenceToSet
  - 动机
    - 多标签分类
  - 模型结构
    - 一个Encoder($\mathcal{E}$)，两个Decoder($\mathcal{D}_1$、$\mathcal{D}_2$)
    - $\mathcal{E}$与$\mathcal{D}_1$组成一个注意力机制的seq2seq模型
    - $\mathcal{D}_2$读$\mathcal{E}$与$\mathcal{D}_1$的隐状态，加入注意力机制
    - $\mathcal{D}_2$建模为Reinforce Learning问题
      - $\mathcal{D}_2$是一个agent，在时刻t的state是生成的标签y_t
      - θ是Decoder采取的policy
      - 当标签全都被生成时，得到reward r
      - 目标：最小化$L(\theta) = -\mathbb{E}_{\boldsymbol{y}\sim\boldsymbol{p_\theta}}[r(\boldsymbol{y})]$



(a) Neural sequence-to-set model



(b) Self-critical training method

Yang P, Ma S, Zhang Y, et al. A Deep Reinforced Sequence-to-Set Model for Multi-Label Text Classification[J]. arXiv preprint arXiv:1809.03118, 2018.

目录

- 问题定义
- 评价指标
- 传统方法
- 深度学习的方法
- Benchmark
- 总结

Benchmark

- Char-CNN和fastText的数据集

Dataset	Classes	Train Samples	Test Samples
AG's News	4	120, 000	7, 600
Sogou News	5	450, 000	60, 000
DBPedia	14	560, 000	70, 000
Yelp Review Polarity	2	560, 000	38, 000
Yelp Review Full	5	650, 000	50, 000
Yahoo! Answers	10	1, 400, 000	60, 000
Amazom Review Full	5	3, 000, 000	650, 000
Amazon Review Polarity	2	3, 600, 000	400, 000

Benchmark

- Char-CNN和fastText的数据集
  - 精确度

Model	AG	Sougo	DBP	Yelp P.	Yelp F.	Yah. A.	Amz. F.	Amz. P.
Bow	88.8	92.9	96.6	92.2	58.0	68.9	54.6	90.4
N-grams	92.0	97.1	98.6	95.6	56.3	68.5	54.3	92.0
N-grams tfidf	92.4	**97.2**	**98.7**	95.4	54.8	68.5	52.4	91.5
Char-CNN	87.2	95.1	98.3	94.7	62.0	71.2	59.5	94.5
VDCNN	91.3	96.8	98.7	**95.7**	**64.7**	**73.4**	**63.0**	**95.7**
fastText	**92.5**	96.8	98.6	**95.7**	63.9	72.3	60.2	94.6

Benchmark

- Char-CNN和fastText的数据集
  - 每个epoch的训练时间

Model	AG	Sougo	DBP	Yelp P.	Yelp F.	Yah. A.	Amz. F.	Amz. P.
Small char-CNN	1h	-	2h	-	-	8h	2d	2d
Big char-CNN	3h	-	5h	-	-	1d	5d	5d
VDCNN(depth=9)	8h	8h30	9h	9h20	9h40	20h	2d7h	2d7h
VDCNN(depth=17)	12h20	13h40	14h50	14h30	15h	1d7h	3d15h	3d16h
VDCNN(depth=29)	17h	18h40	20h	23h	1d	1d17h	5d20h	5d20h
fastText	3s	36s	8s	15s	18s	27s	33s	52s

NVIDA Tesla K40 GPU

Benchmark

- Text CNN的数据集

Dataset	Classes	Average Length	Dataset Size	Vocabulary Size
MR(Movie Review)	2	20	10662	18765
SST-1	5	18	11855	17836
SST-2	2	19	9613	16185
Subj(subjective or objective)	2	23	10000	21323
TREC(Question dataset)	6	10	5952	9592
CR(Customer review, positive or negative)	2	19	3775	5340
MQPA(Opinion polarity detection)	2	3	10606	6246

Benchmark

- Text CNN的数据集
  - 精确度

Model	MR	SST-1	SST-2	Subj	TREC	CR	MPQA
CNN-rand	76.1	45.0	82.7	89.6	91.2	79.8	83.4
CNN-static	81.0	45.5	86.8	93.0	92.8	84.7	**89.6**
CNN-non-static	**81.5**	48.0	87.2	93.4	93.6	84.3	89.5
CNN-multichannel	81.1	47.4	**88.1**	93.2	92.2	**85.0**	89.4
RAE (Socher et al., 2011)	77.7	43.2	82.4	–	–	–	86.4
MV-RNN (Socher et al., 2012)	79.0	44.4	82.9	–	–	–	–
RNTN (Socher et al., 2013)	–	45.7	85.4	–	–	–	–
DCNN (Kalchbrenner et al., 2014)	–	48.5	86.8	–	93.0	–	–
Paragraph-Vec (Le and Mikolov, 2014)	–	**48.7**	87.8	–	–	–	–
CCAE (Hermann and Blunsom, 2013)	77.8	–	–	–	–	–	87.2
Sent-Parser (Dong et al., 2014)	79.5	–	–	–	–	–	86.3
NBSVM (Wang and Manning, 2012)	79.4	–	–	93.2	–	81.8	86.3
MNB (Wang and Manning, 2012)	79.0	–	–	**93.6**	–	80.0	86.3
G-Dropout (Wang and Manning, 2013)	79.0	–	–	93.4	–	82.1	86.1
F-Dropout (Wang and Manning, 2013)	79.1	–	–	**93.6**	–	81.9	86.3
Tree-CRF (Nakagawa et al., 2010)	77.3	–	–	–	–	81.4	86.1
CRF-PR (Yang and Cardie, 2014)	–	–	–	–	–	82.7	–
SVM_S (Silva et al., 2011)	–	–	–	–	**95.0**	–	–

Benchmark

- RNN with Attention(HN)和GateRNN的实验结果
  - 精确度

	HN	Conv-GRNN	LSTM-GRNN	TextCNN	CharCNN	SVM	BoW
Yelp'13	68.2	63.7	65.1	59.7	-	59.8	-
Yelp'14	70.5	65.5	67.1	61.0	-	61.8	-
Yelp'15	71.0	66.0	67.6	61.5	62.0	62.4	59.9
Yahoo Answer	75.8	-	-	-	71.2	-	71.0
Amazon	63.6	-	-	-	59.6	-	55.3

Benchmark

- RCNN的数据集

Dataset	Classes	Dataset Size	Average Length
20News	4	13, 919	429
Fudan(CH)	20	19, 636	298
ACL	5	202, 979	25
SST	5	11, 855	19

Benchmark

- RCNN数据集上的结果
  - 精确度

	RCNN	CNN	ClassifyLDA-EM	Labeled-LDA	CFG	C&J	SVM
20News	96.49	94.79	93.60	-	-	-	92.43
Fudan	95.20	94.04	-	90.80	-	-	93.02
ACL	49.19	47.47	-	-	39.20	49.20	45.24
SST	47.21	46.35	-	-	-	-	40.70

目录

- 问题定义
- 评价指标
- 传统方法
- 深度学习的方法
- Benchmark
- 总结

总结

- 简单的模型加上好的特征在特定领域也能取得不错的效果（tf-idf + svm、fastText）
- CNN可以用来提取类似n-gram的特征，不过卷积核的设计很繁琐，另外字符级的特征好像并不是很work
- RNN对于序列数据建模很友好，但是序列太长的话也会有梯度消失或爆炸的问题
- 注意力机制无论是对段落或者文章的结构进行建模还是捕捉关键特征效果都很好
- 强化学习任重道远

请多指教

袁浩达
2018/9/19