2020-2021学年秋季学期

自然语言处理

Natural Language Processing

授课教师： 胡玥

助　教： 于静

自然语言处理
Natural Language Processing

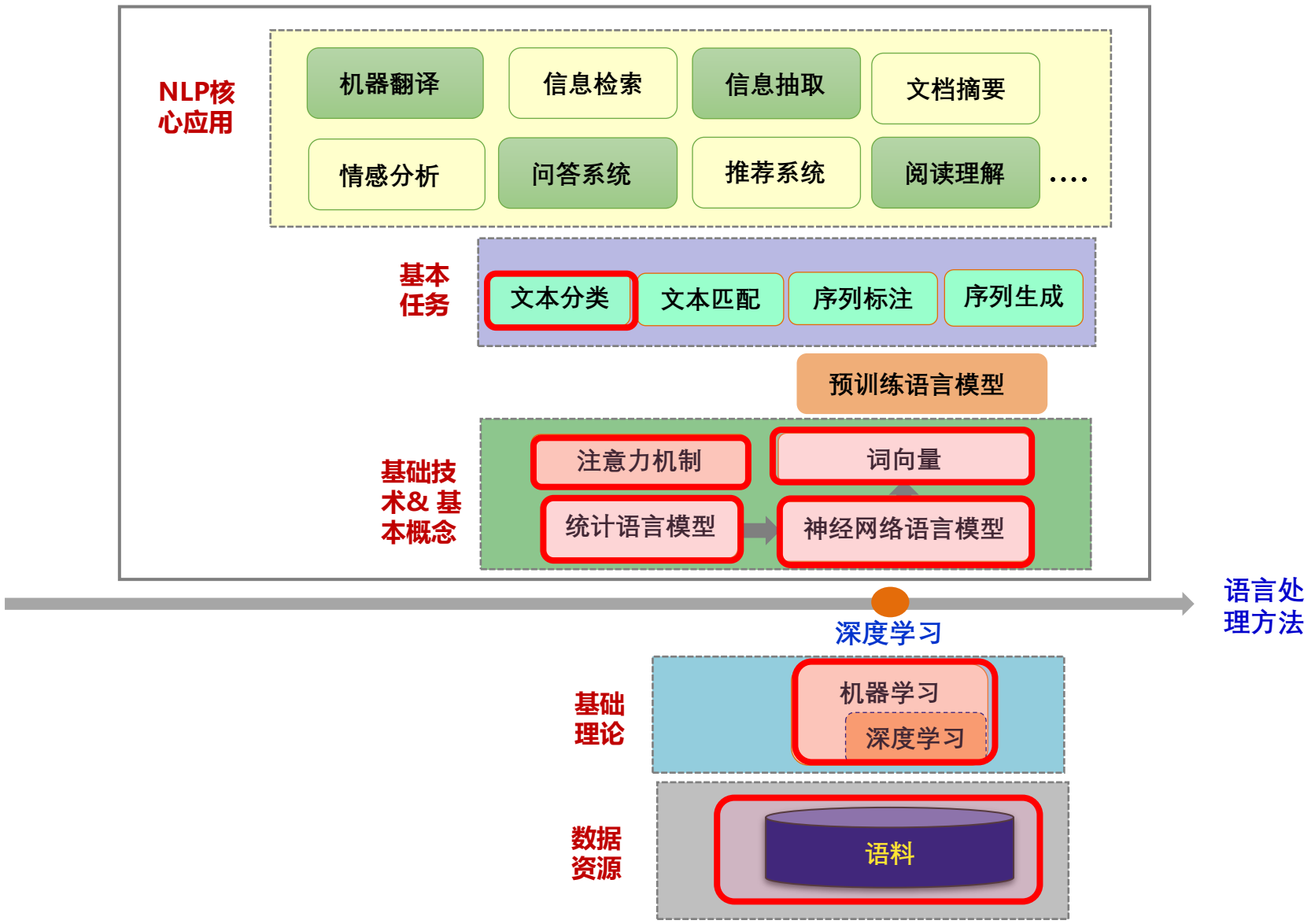
第 11 章　文本分类任务

授课教师：　胡玥

授课时间：2020.9

课程编码 201M4005H　课程名称 自然语言处理　授课团队名单 胡玥 、于静

基于深度学习的自然语言处理课程内容



概 要

本章主要内容：

本章主要围绕句子级/文档级的文本分类任务介绍典型的分类方法及模型包括：词袋模型，各种CNN模型，RNN模型，Attention模型 和图卷积模型等。

本章教学目的：

使学生理解并掌握文本分类的共性问题，并掌握各种文本分类方法

内 容 提 要

11.1 概述

■ 文本分类

文本分类是NLP中的常见的重要任务之一，应用广泛，在很多领域发挥着重要作用，例如垃圾邮件过滤、舆情分析以及新闻分类等。

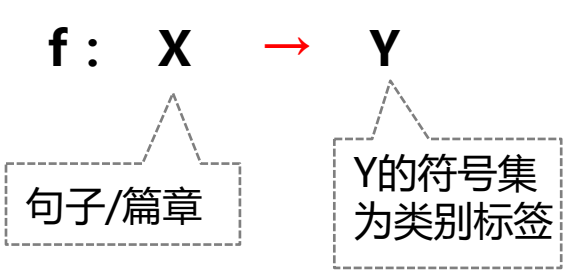
11.1 概述

■ 文本分类问题定义

给定文档 p，将文档分类为n个类别中的一个或多个

处理文档粒度可以是：

- 句子级
- 篇章级



输入：X 句子/篇章

输出：X 所属类别 Y

Y ∈{类别集合}

12.1 概述

■ 分类方法

★ 规则方法：不同任务需要专门构建特征规则

★ 统计方法：特征工程+算法（Naive Bayes/SVM/LR/KNN……）

★ 深度学习方法：

- 基于词袋的文本分类
- 基于卷积神经网络文本分类（TextCNN/DPCNN/Char-CNN/VDCNN）
- 基于循环神经网络文本分类（TextRNN/TextRCNN）
- 基于attention机制文本分类
- 基于图卷积神经网络文本分类

本章讨论：句/文档粒度的基于深度学习的分类方法

内 容 提 要

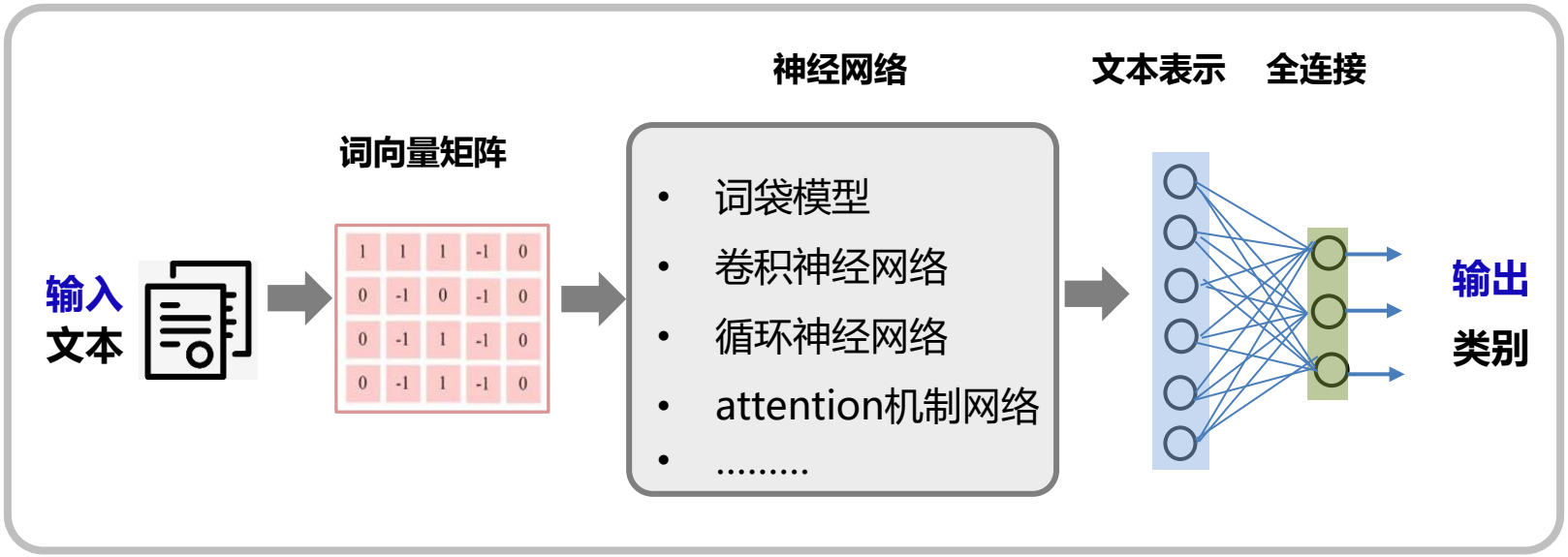
11.1 概述

11.2 传统神经文本分类方法

11.3 图卷积文本分类方法

11.2 传统神经文本分类方法

■ 传统神经文本分类框架



关键问题：如何生成高质量的文本表示

- 句子级文本表示
- 文档级文本表示

句子级文本表示

句子级文本表示

主要研究如何有效地从词嵌入通过不同方式的组合得到句子表示。其中，比较有代表性的方法有四种

1. 神经词袋模型（Bag of words）

简单对文本序列中每个词嵌入进行平均/加总，作为整个序列的表示。这种方法的缺点是丢失了词序信息。对于长文本，神经词袋模型比较有效。但是对于短文本，神经词袋模型很难捕获语义组合信息

2. 卷积神经网络（Convolutional Neural Network）

通过多个卷积层和子采样层，抽取序列的 n-gram特征信息，最终将得到特征信息合并成一个固定长度的向量作为整个序列表示。

句子级文本表示

3. 循环神经网络（Recurrent Neural Network）

将文本序列看作时间序列，不断更新，最后得到整个序列的表示。这种表示中包含的是序列的顺序信息。

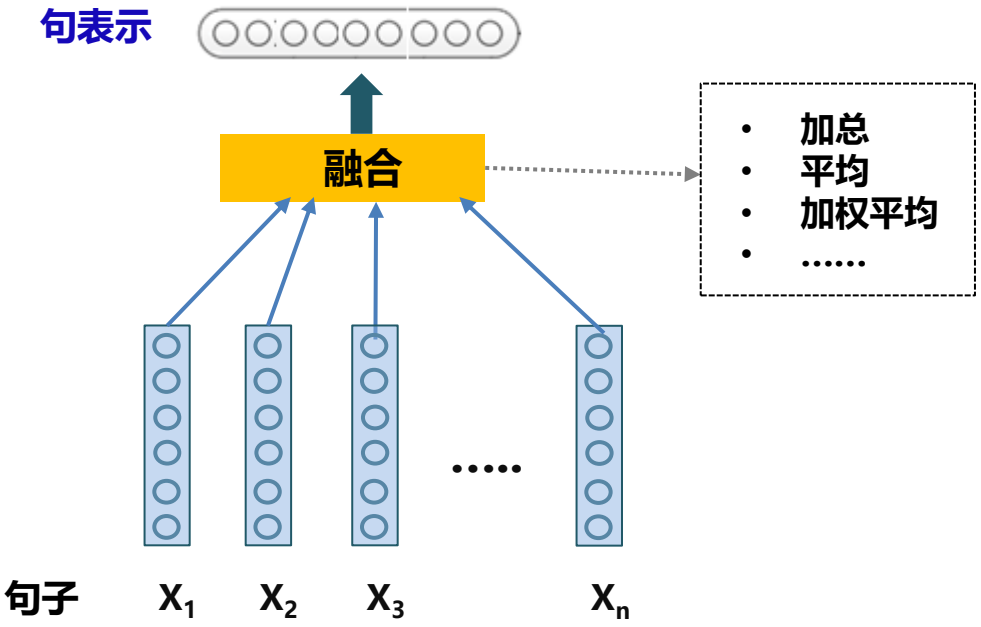
4. 注意力神经网络（Attention Network）

通过注意力机制对序列进行编码，最后得序列的表示，这种表示包含的是词与词之间的关联关系。

在上述基本方法的基础上，很多研究者综合这些方法的优点，结合具体的任务，提出了一些更复杂的组合模型

1. 神经词袋模型（Bag of words）

■ 词袋模型



1. 神经词袋模型（Bag of words）

★ FastText

- **动机：**

良好的文本分类表示是许多应用程序的重要任务，如Web搜索，信息检索，排序和文档分类。 基于神经网络的模型在计算句子表示实践中取得了非常好的表现，但是在训练和测试时间，它们往往相对较慢，限制了其在非常大的数据集上的使用。 Facebook提出了一种简单而有效的文本分类和表示学习方法可以在不到10分钟的时间内使用标准的多核CPU对超过10亿个单词进行快速文本训练，并在不到一分钟的时间内对312K类中的50万个句子进行分类。

Joulin et, al.Bag of Tricks for Efficient Text Classification，2017

1. 神经词袋模型（Bag of words）

■ **模型结构：**

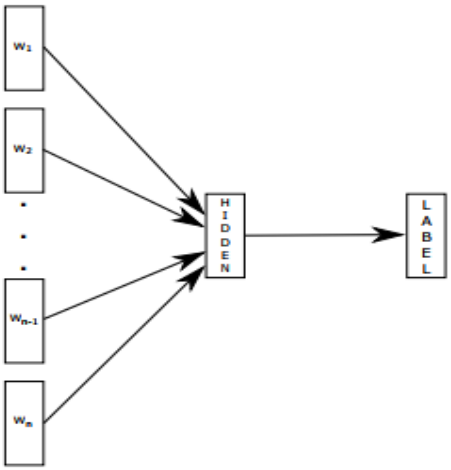


Figure 1: Model architecture for fast sentence classification.

输入：Document中的每个词的词向量

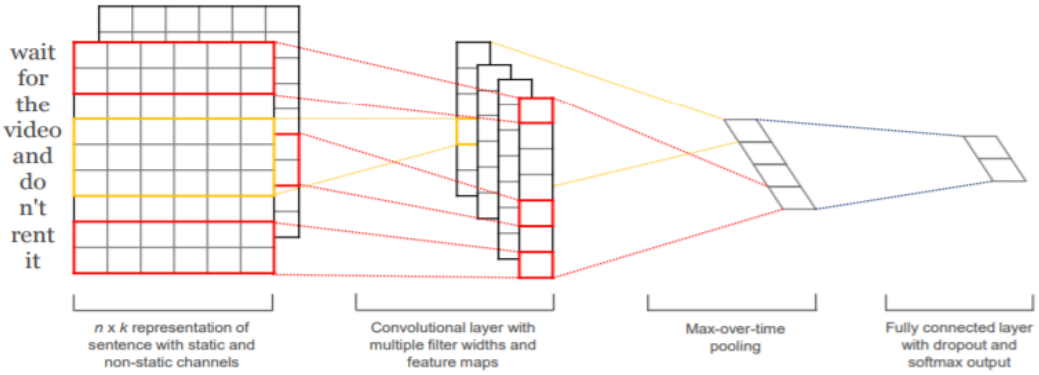
输出：分类标签

- 隐藏层：得到document中每个词的word embedding表示，将向量相加取平均值作为输出层的输入
- 输出层：当类别较少的时候使用softmax，类别较多的时候使用hierarchical softmax

2. 卷积神经网络（Convolutional Neural Network）

★ TextCNN

■ 模型结构



输入： 两套词向量构造出的句子矩阵作为两个通道

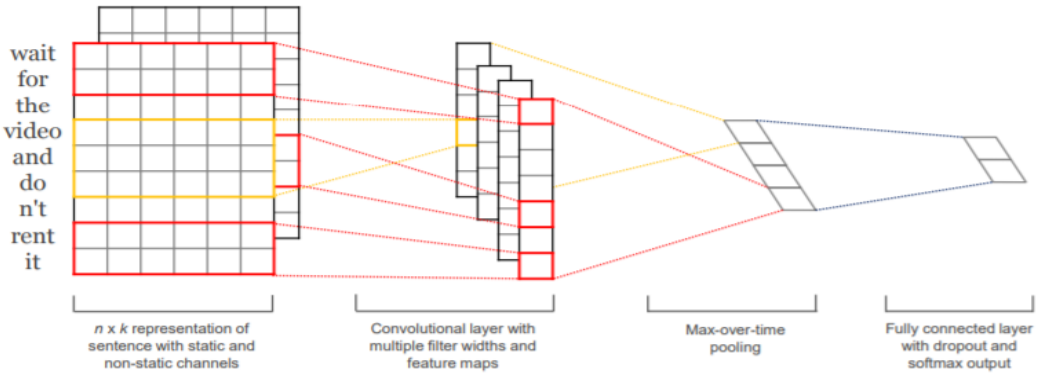
- CNN-rand：所有的词向量都随机初始化（模型训练时做参数进行训练）
- CNN-static：即用word2vec预训练好的向量（模型训练过程中不更新词向量）句中若有单词不在预训练好的词典中，则用随机数来代替。

输出： 句子类别分布

Kim et al. Convolutional Neural Networks for Sentence Classification， 2014

2. 卷积神经网络（Convolutional Neural Network）

■ 模型结构



模型层数：单层CNN

卷积方式：两个通道分别采用不同的卷积核（如2，3，4，5，7）进行卷积构建n-gram特征；然后将Pooling后的特征进行连接作为句表示；将生成的句表示做全连接分类

■ 模型训练

两套词向量构造出的句子矩阵作为两个通道，在误差反向传播时，只更新CNN-rand一组词向量，保持另外一组不变

2. 卷积神经网络（Convolutional Neural Network）

★ DCNN

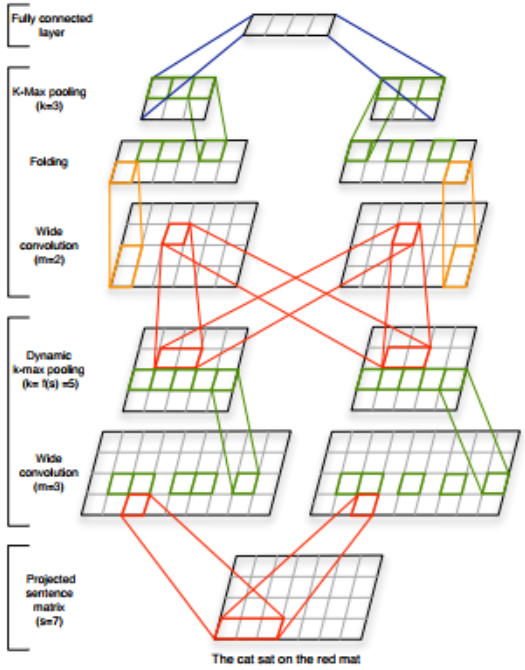


Figure 3: A DCNN for the seven word input sentence. Word embeddings have size $d = 4$. The network has two convolutional layers with two feature maps each. The widths of the filters at the two layers are respectively 3 and 2. The (dynamic) k-max pooling layers have values k of 5 and 3.

■ **模型结构**

输入： 句子的词向量矩阵

输出： 句向量

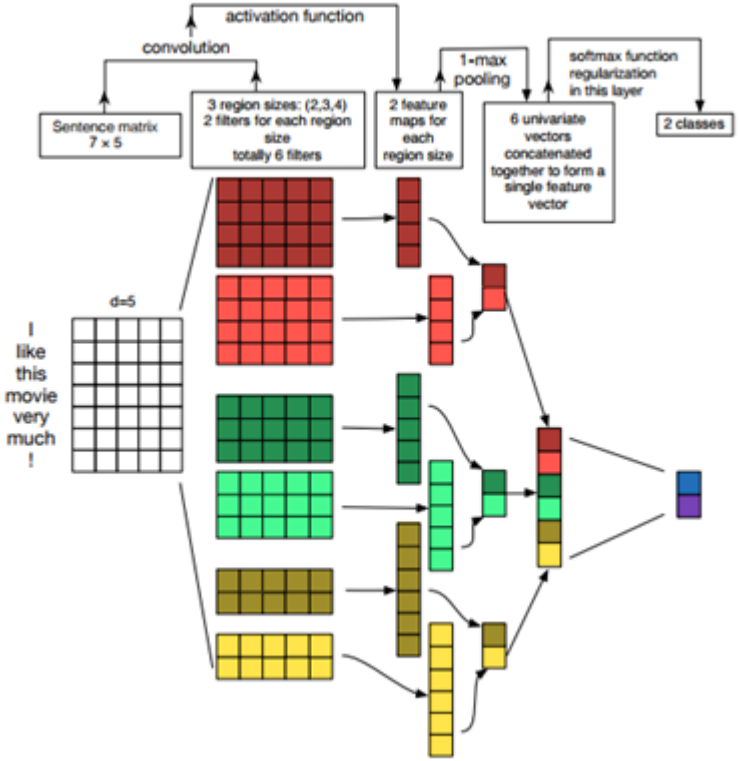
■ **模型特点**

- one-dim宽卷积层
- 动态 K-Max pooling层
- Folding层

Kalchbrenner， A Convolutional Neural Network for Modelling Sentences， 2014

2. 卷积神经网络（Convolutional Neural Network）

★ Sensitivity Analysis of CNN for Sentence Classification



■ 模型结构

输入： 句子的多种词向量矩阵

GloVe、word2vec、one-hot

输出： 句向量

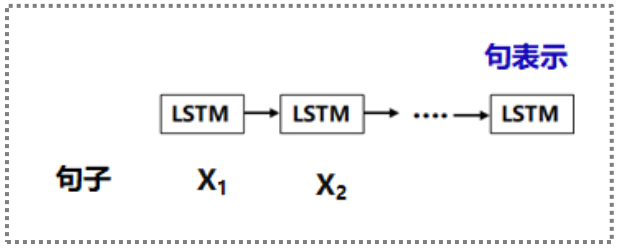
■ 模型特点

- 采用不同尺寸的卷积核
- 不同的池化方法（最大池化，平均池化，K最大池化）
- 采用不同的激活函数（Tanh,ReLU, Sigmoid,Softplus, Iden等7种 ）

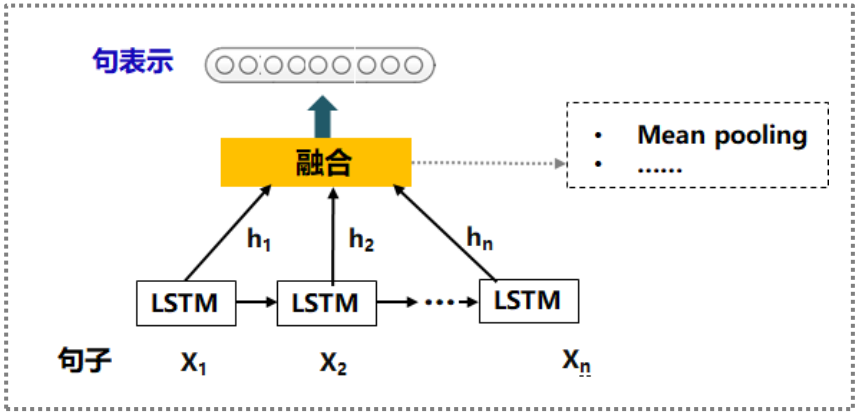
A Sensitivity Analysis of (and Practitioners' Guide to) Convolutional Neural Networks for Sentence Classification, 2015

3. 循环神经网络（Recurrent Neural Network）

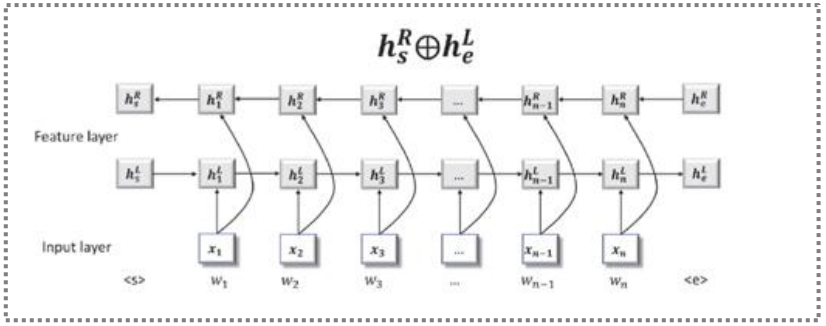
■ 循环神经网络模型



(1)



(2)



(3)

3`. 循环+卷积混合神经网络

■ 循环+卷积神经网络模型

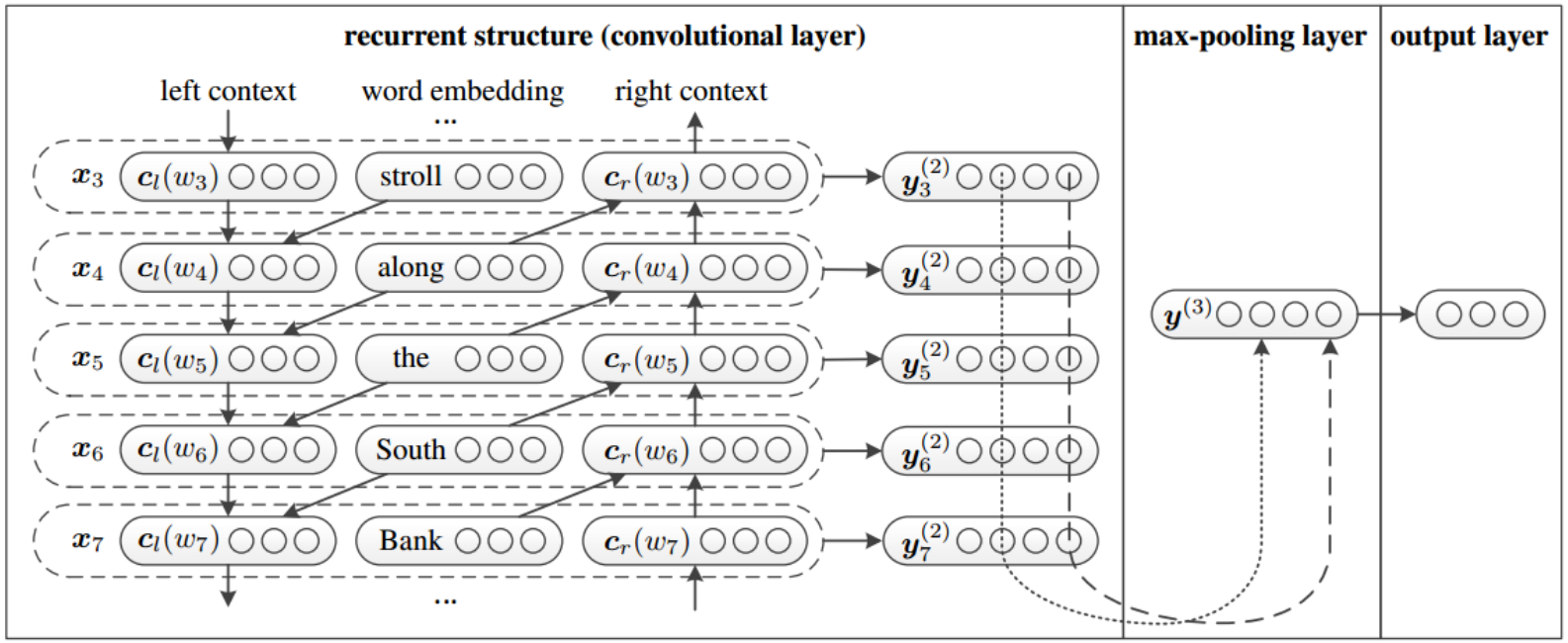
- RNN擅长处理序列结构，能够考虑到句子的上下文信息。
- RNN属于“biased model”，一个句子中越往后的词重要性越高，这有可能影响最后的分类结果，因为对句子分类影响最大的词可能处在句子任何位置。
- CNN属于无偏模型，能够通过最大池化获得最重要的特征。

结合二者的优势生成上下文信息（不仅窗口信息）卷积网络

3`. 循环+卷积混合神经网络

★ Recurrent Convolutional Neural Networks

■ 模型结构



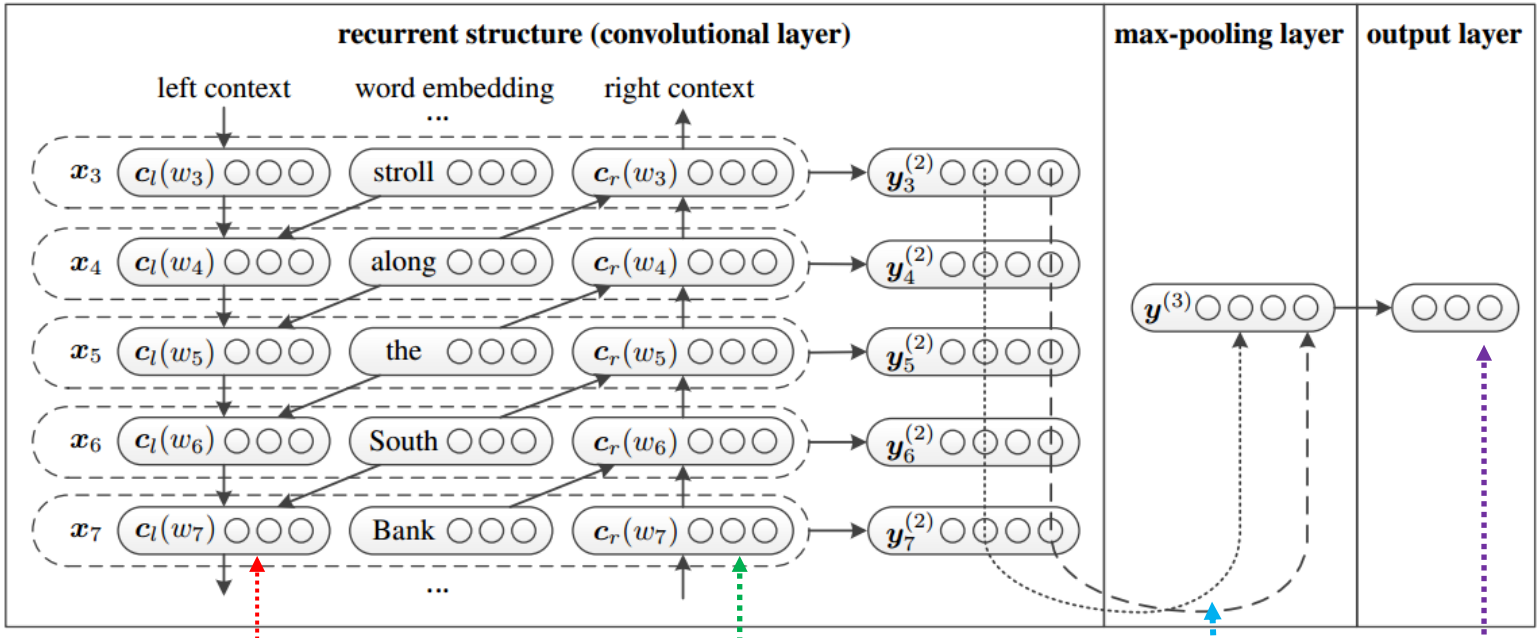
输入：一篇文档D，包含n个词 $\{w_1, w_2, \dots, w_n\}$

输出：文档类别

Lai et al. 2015，Recurrent Convolutional Neural Networks for Text Classification

3`. 循环+卷积混合神经网络

■ 模型结构



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

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

$$\boldsymbol{x}_i = [\boldsymbol{c}_l(w_i); \boldsymbol{e}(w_i); \boldsymbol{c}_r(w_i)]$$

$$\boldsymbol{y}_i^{(2)} = \tanh\left(W^{(2)}\boldsymbol{x}_i + \boldsymbol{b}^{(2)}\right)$$

$$\boldsymbol{y}^{(3)} = \max_{i=1}^{n} \boldsymbol{y}_i^{(2)}$$

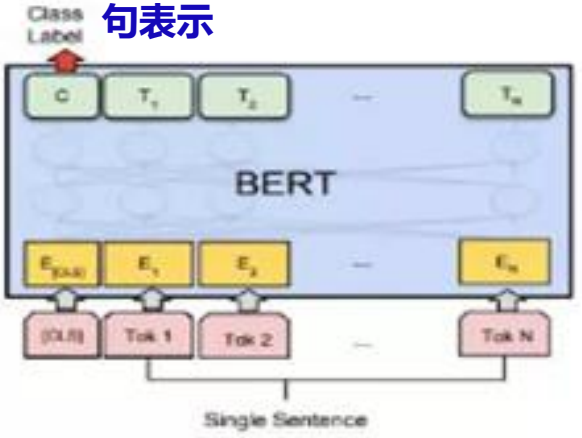
$$\boldsymbol{y}^{(4)} = W^{(4)}\boldsymbol{y}^{(3)} + \boldsymbol{b}^{(4)}$$

$$p_i = \frac{\exp\left(\boldsymbol{y}_i^{(4)}\right)}{\sum_{k=1}^{n}\exp\left(\boldsymbol{y}_k^{(4)}\right)}$$

4. 注意力神经网络（Attention Network）

■ 注意力神经网络

Attention 编码



输入: 句子 $\{w_1, w_2, \ldots, w_n\}$

Pretrained + Fine-tuning

BERT

篇章/文档级文本表示

篇章级文本表示

由于篇章级文本序列更长，为了降低模型复杂度，一般采用层次化的方法，先得到句子编码，然后以句子编码为输入，进一步得到篇章的表示。具体的层次化可以采用以下几种方法：

1. 层次化的卷积神经网络

即用卷积神经网络对每个句子进行建模，然后以句子为单位再进行一次卷积和池化操作，得到篇章表示。

2. 层次化的循环神经网络

即用循环神经网络对每个句子进行建模，然后再用一个循环神经网络建模以句子为单位的序列，得到篇章表示。

篇章级文本表示

3. 混合模型

先用循环神经网络对每个句子进行建模，然后以句子为单位再进行一次卷积和池化操作，得到篇章表示。在上述模型中，循环神经网络因为非常适合处理文本序列，因此被广泛应用在很多自然语言处理任务上。

.

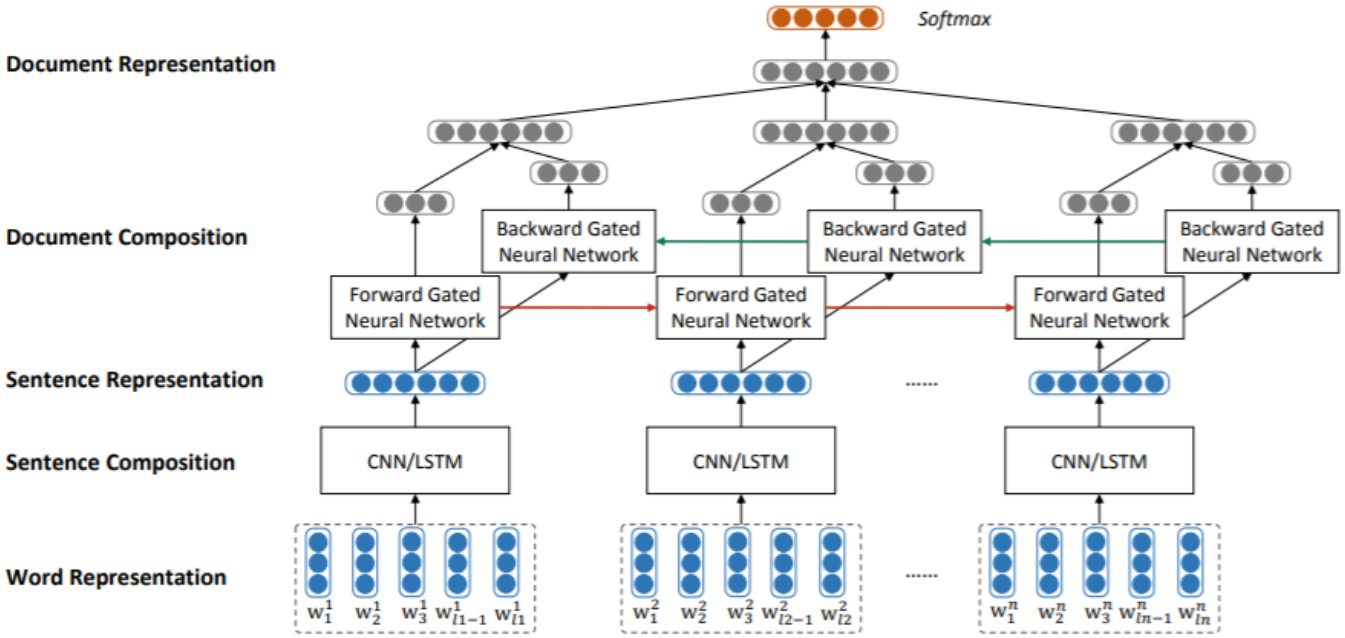
4. 注意力模型

在混合模型中引入注意力机制，该机制能识别不同粒度语言单位中各元素对分类决策的贡献程度，从而能区别对待不同的元素。提高分类性能。注意力机制目前广泛应用在很多自然语言处理任务上。

混合模型

★ LSTM/CNN-GRU

■ 模型结构



输入：篇章中所有句子的词向量矩阵作为输入

所有的词向量都随机初始化（模型训练时做参数进行训练）

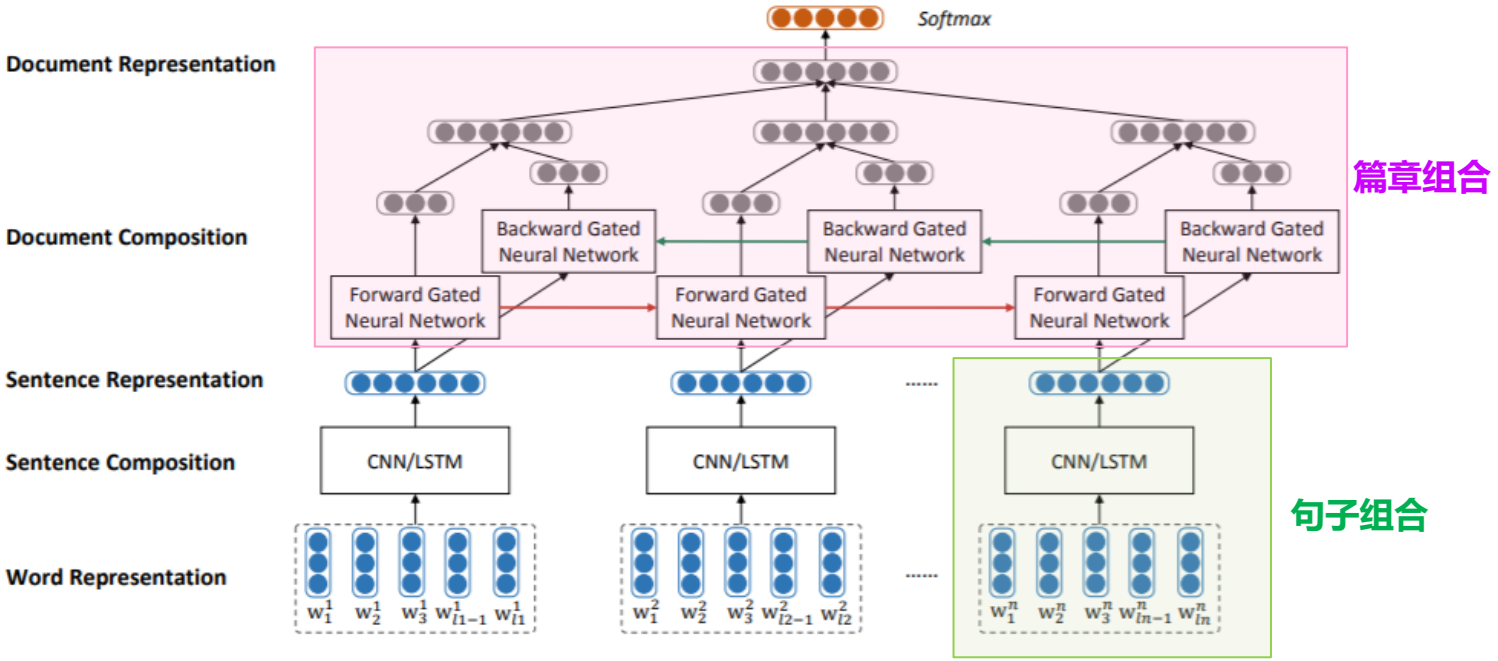
输出：篇章类别分布

Tang et al. Document Modeling with Gated Recurrent Neural Network for Sentiment Classification, EMNLP 2015.

混合模型

★ LSTM/CNN-GRU

■ 模型结构



输入：篇章中所有句子的词向量矩阵作为输入

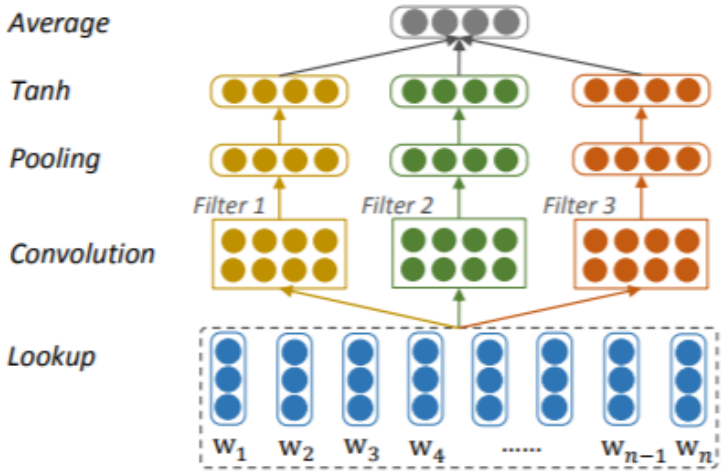
所有的词向量都随机初始化（模型训练时做参数进行训练）

输出：篇章类别分布

Tang et al. Document Modeling with Gated Recurrent Neural Network for Sentiment Classification, EMNLP 2015.

混合模型

■ Sentence Composition（句子组合）



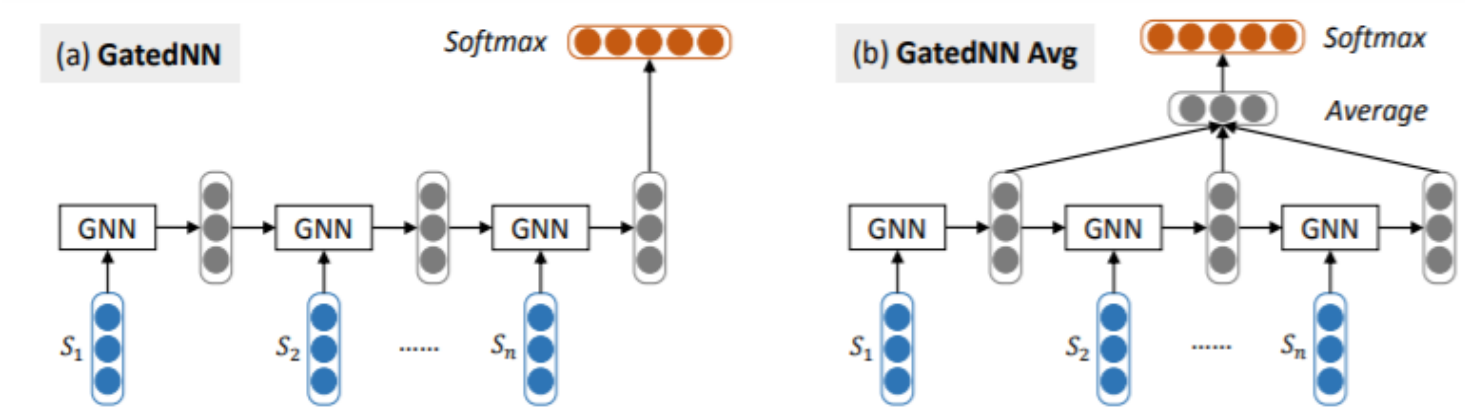
输入：每一个句子的词向量矩阵

- 引入三个卷积核，经过卷积、最大池化、非线性映射和平均池化得到句子级向量表示

输出：句子级向量表示

混合模型

■ Document Composition（篇章组合）



输入： 篇章中所有句子的向量表示

- 引入双向门限RNN，对篇章所有句子进行统一编码

输出： 篇章级向量表示

注意力模型

★ Hierarchical Attention Networks for Document Classification (HAN)

- **动机：**

在文档中，句子是由单词组成，文档是由句子组成，句子对文档的重要性贡献有差异，在句子中单词对句子的重要性贡献也有差异。文章从句子级和文档级两个层次引入了Attention机制来描述这种重要性：在生成句子表示时使用词级Attention机制，在生成文档表示时使用句级Attention机制，该方法能识别出在影响最终分类决策的重要单词和句子，从而提升分类性能。

Yang et al. Hierarchical Attention Networks for Document Classification，2016

注意力模型

■ **模型结构：**

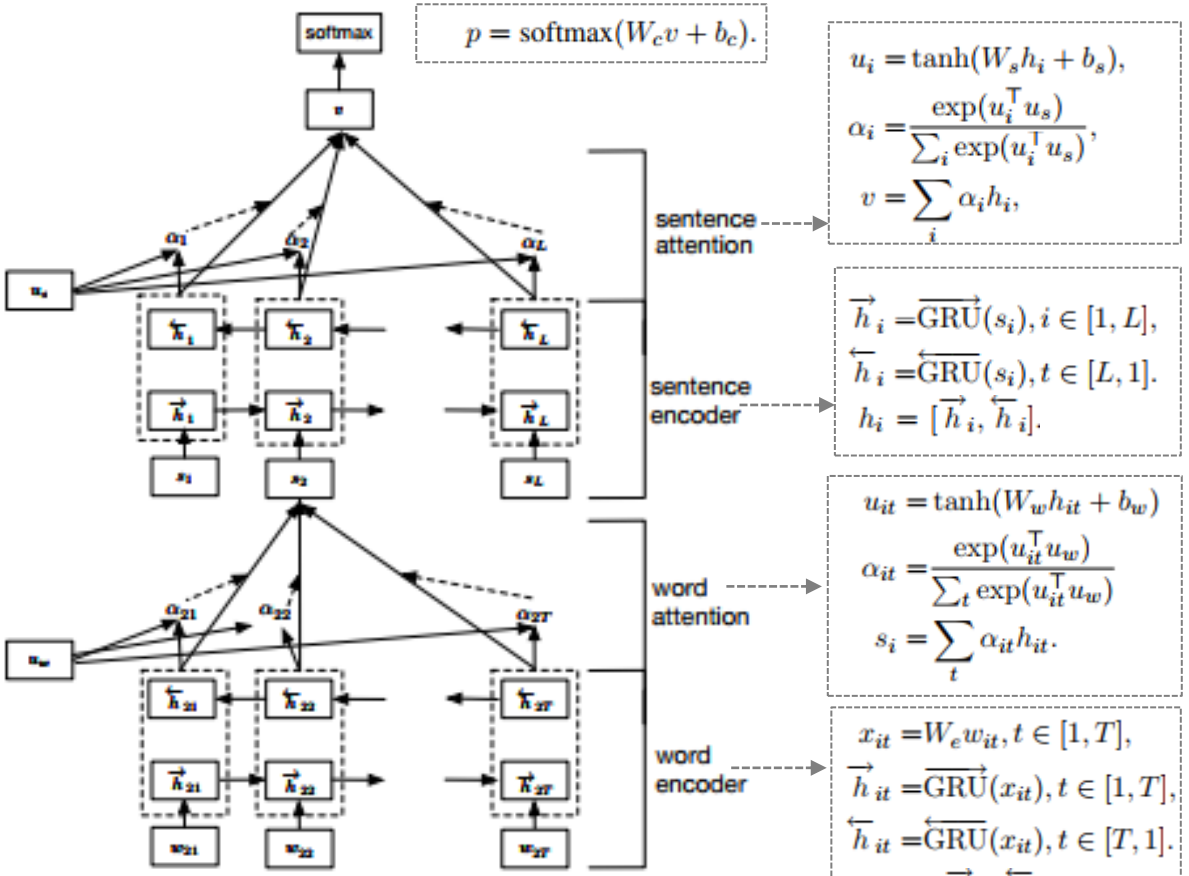


Figure 2: Hierarchical Attention Network.

L 表示篇章中句子数量

T 表示句子中词的数量

输入： 篇章中所有句子的词向量矩阵作为输入（所有的词向量都随机初始化）

输出： 篇章类别分布

内 容 提 要

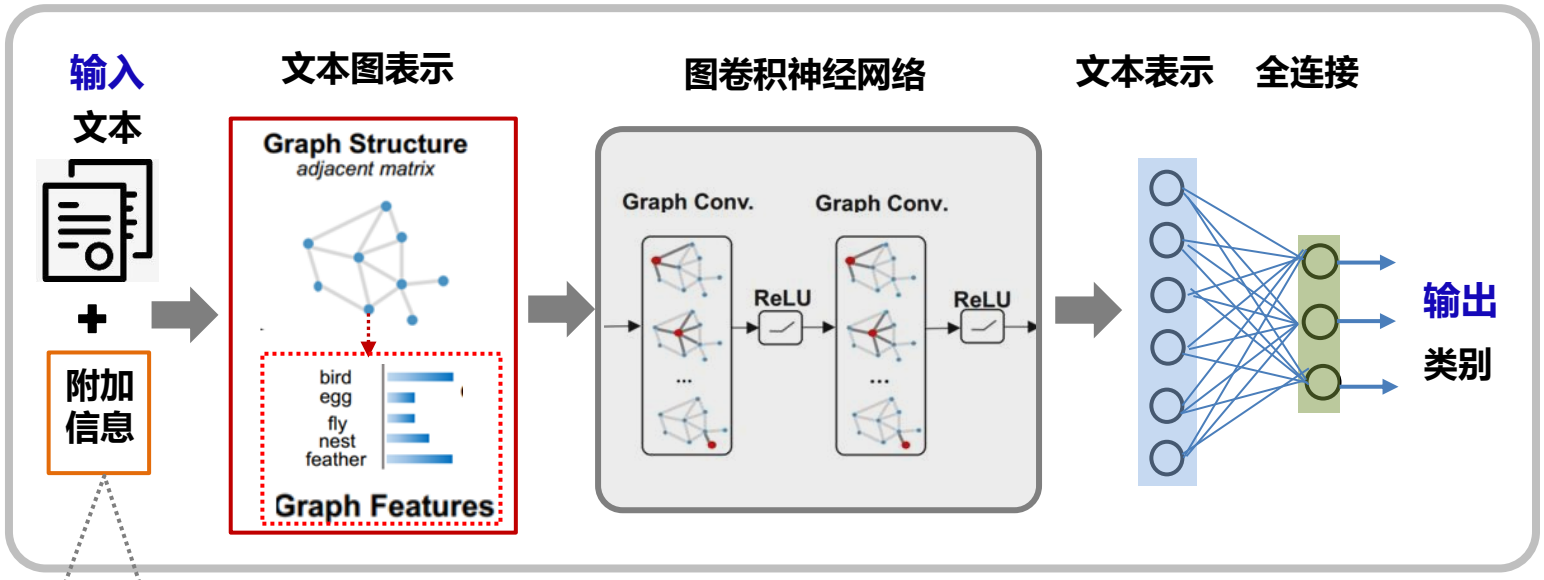
11.1 概述

11.2 传统神经文本分类方法

11.3 图卷积文本分类方法

11.3 图卷积文本分类方法

■ 图卷积文本分类框架



词性POS，句法信息，关联，共现，主题，实体，实体关系，知识图谱.....

★ **图卷积文本分类步骤：**

1. Graph 构建
2. 文本Graph节点特征表示
3. 图卷积算法

11.3 图卷积文本分类方法

1. Graph 构建

在图卷积中可以根据任务对原文本加入附加信息并构建原文本与附加信息的关系图作为图卷积网络的输入

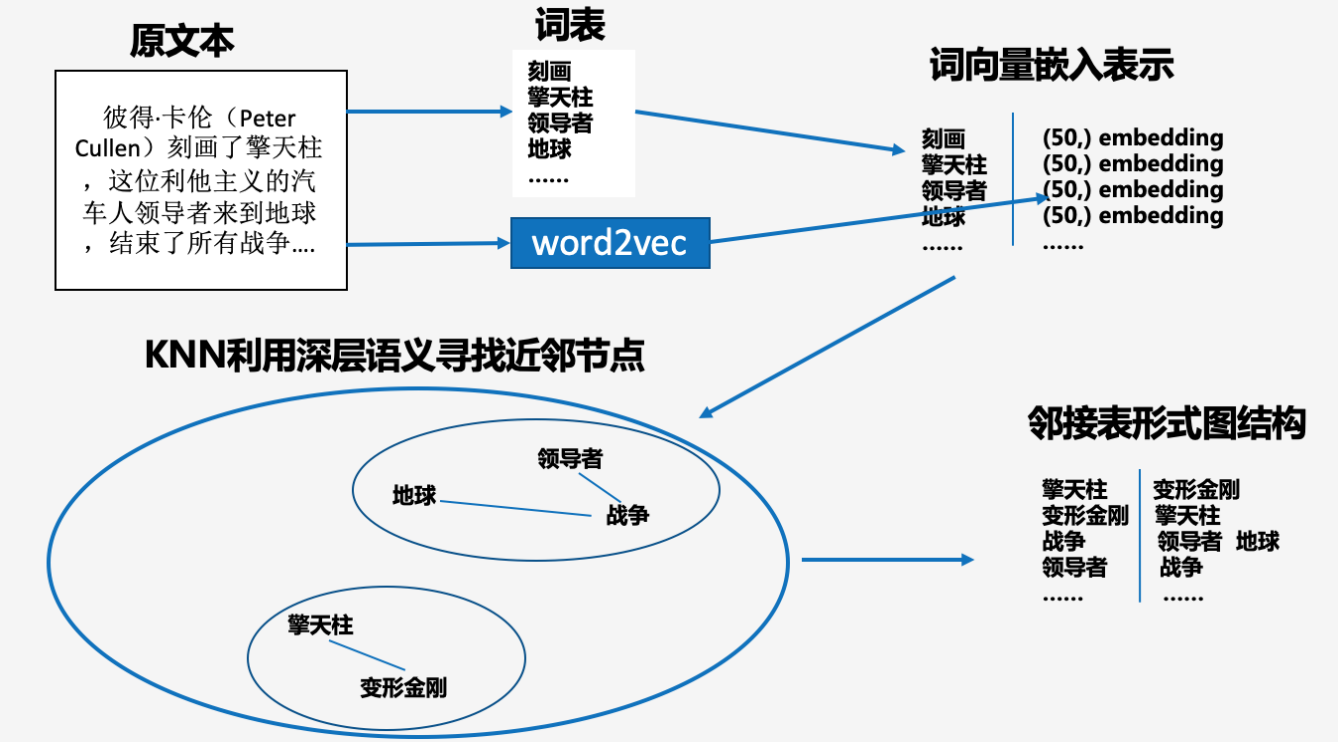
如：对原文本附加词的近义信息，共现信息，先验知识信息等，按照附加信息的不同有不同的建图方法：

- 基于word2vec 词向量的余弦距离建图（KNN）
- 基于词语的共现关系建图（co-occurrence）
- 基于知识图谱的先验知识建图（KG）

……

11.3 图卷积文本分类方法

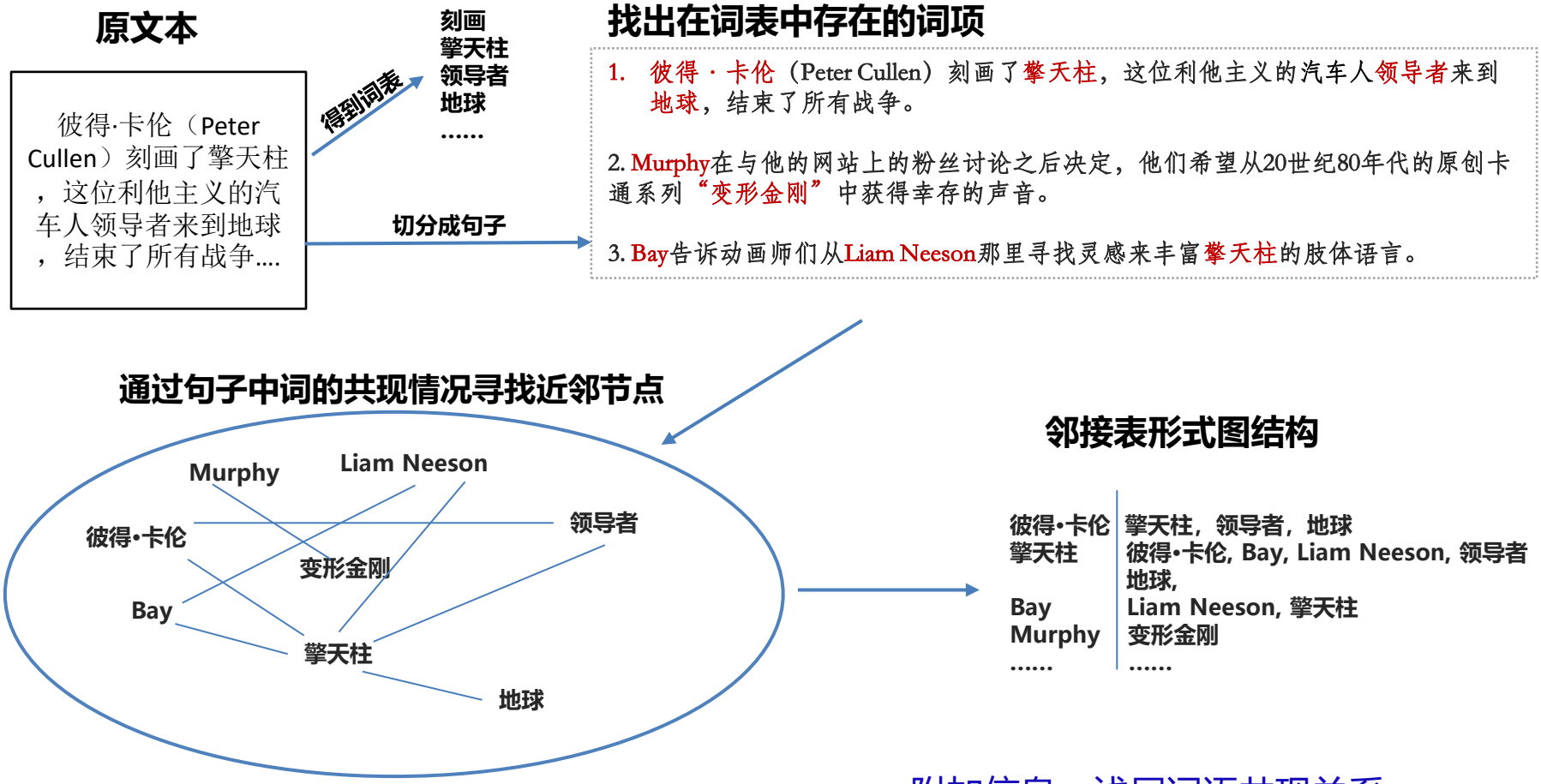
- 基于word2vec 词向量的余弦距离建图（KNN）



附加信息：词义近似关系

11.3 图卷积文本分类方法

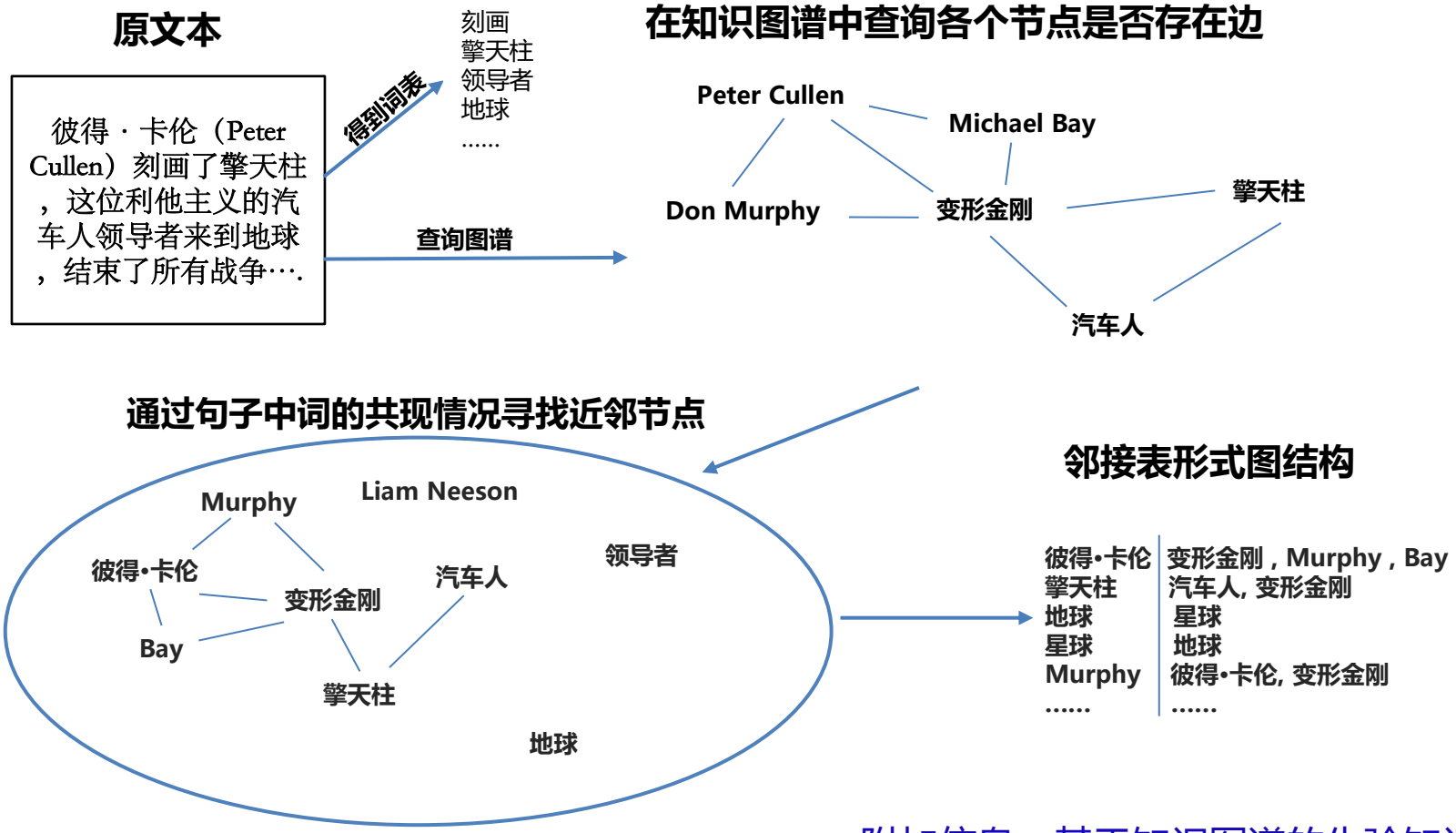
- 基于词语的共现关系建图（co-occurrence）



附加信息：浅层词语共现关系

11.3 图卷积文本分类方法

- 基于知识图谱的先验知识建图（KG）



附加信息：基于知识图谱的先验知识

2. 文本Graph结点特征表示

在图卷积中结点可以根据任务需要采用不同的结点表示方法

如：

- 基于Bag of Words 词语词频表示
- 基于词语所在位置的position编码表示
- 基于Bi-LSTM 的词语隐层向量嵌入表示
- 基于ELMO 的词向量嵌入表示

……

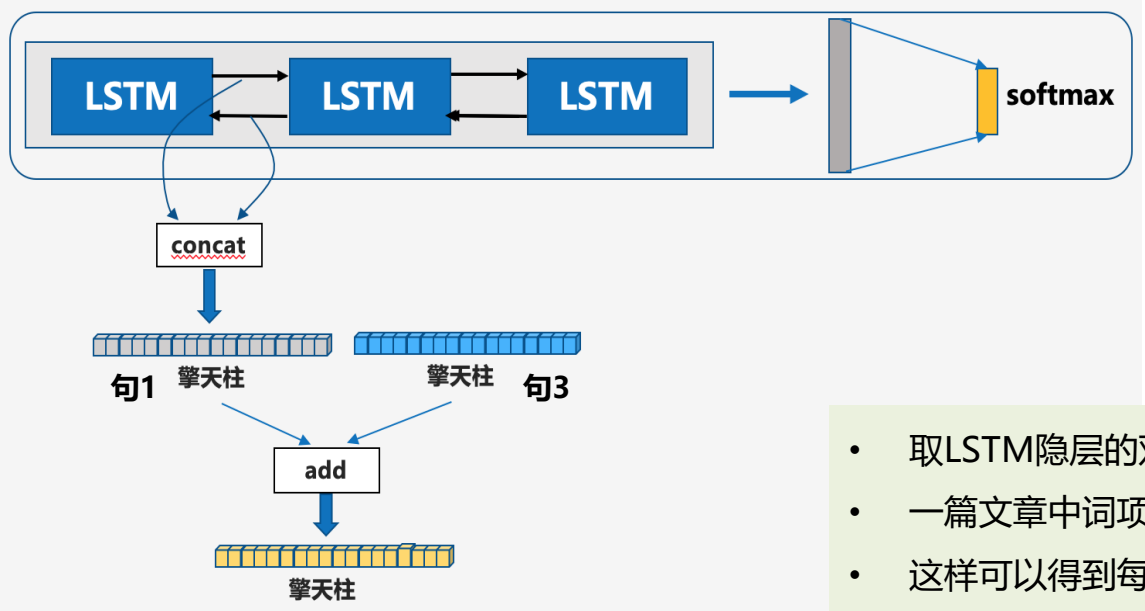
11.3 图卷积文本分类方法

- 基于 Bi-LSTM 的词语隐层向量嵌入表示

文本

1. 彼得·卡伦（Peter Cullen）刻画了擎天柱，这位利他主义的汽车人领导者来到地球，为了结束战争而扑灭了所有火星。
2. Murphy在与他的网站上的粉丝讨论之后决定，他们希望从20世纪80年代的原创卡通系列“变形金刚”中获得幸存的声音
3. Bay告诉动画师们从Liam Neeson那里寻找灵感来丰富擎天柱的肢体语言…

标签 [movie, movie, movie ...] (以句子所属的文章标签作为句子标签)



- 取LSTM隐层的双向输出作为词的嵌入
- 一篇文章中词项多次出现的话将vector对位相加
- 这样可以得到每篇文章单独的词的嵌入表示

11.3 图卷积文本分类方法

LSTM Embedding 优点：

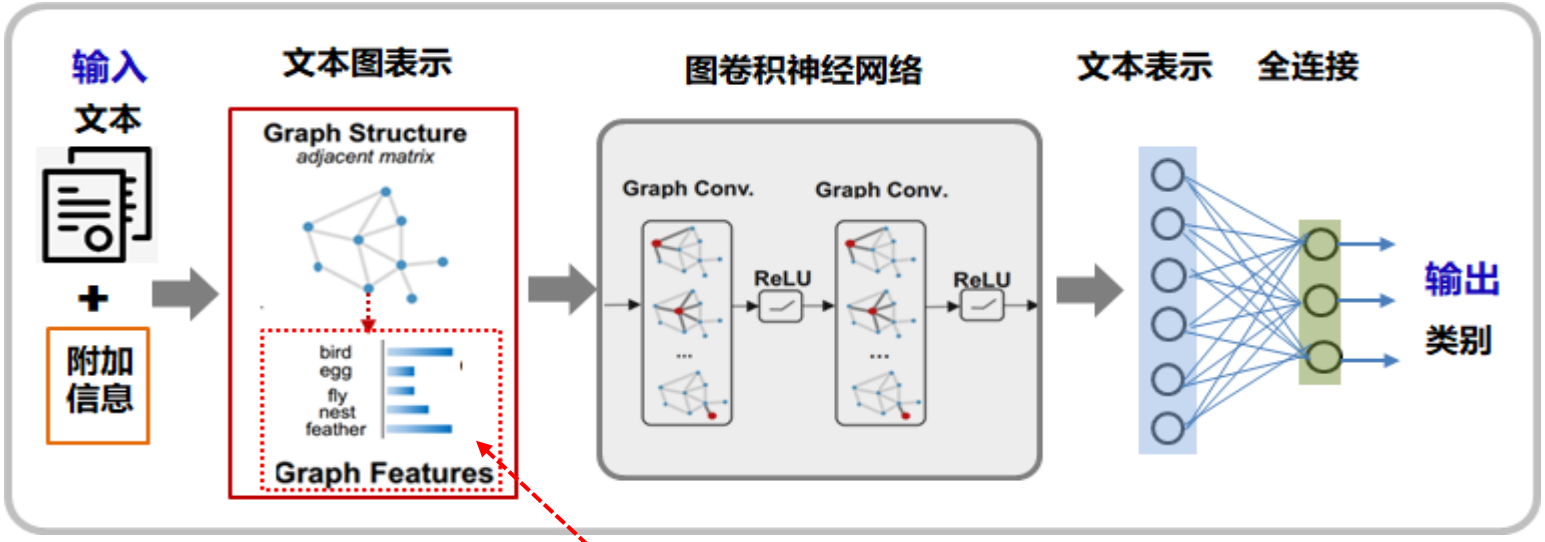
- 可以应用训练样本进行预训练，得到的embedding更贴合数据。
- 可以为每篇文章生成独有的词向量表示，可表述词项在不同语境的不同含义。
- Bi-lstm 的 embedding 结合了上下文信息以及深度语义信息，将词项表示的更加完整。

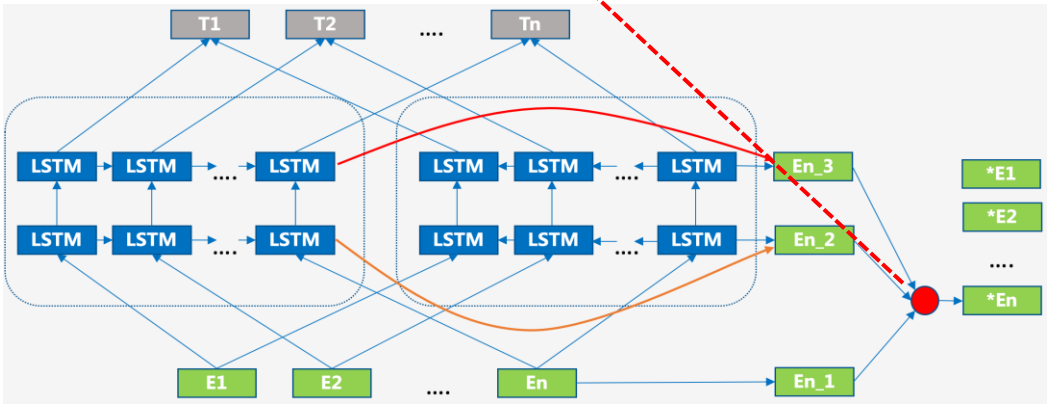
缺点：

- 词向量需要一定维度支持，这在进行图卷积过程中将加大计算量。
- 预训练时的目标函数与最终任务不能完全切合。

11.3 图卷积文本分类方法

- 基于ELMO 预训练的词向量嵌入表示





11.3 图卷积文本分类方法

Elmo Embedding 优点：

- 可以为每篇文章生成独有的词向量表示，可表述词项在不同语境的不同含义，并结合了上下文信息以及深度语义信息。
- ELMo 模型本身经过大量预料预训练，使得词向量的信息非常丰富。
- Language Model 将ELMo接入端到端任务中，fine-tune过程使词向量的训练与最终任务完全切合。

缺点：

- 每个epoch的训练，ELMo 模型都要重新计算词向量，这将加大计算量，使得训练时间大大增加。
- 最终获得的词向量维度较大，使图卷积过程参数增加，训练减慢。

3. 图卷积算法

构建好输入图和图上结点表示后，可以根据不同的任务构建不同的图卷积算法

如：

- 图卷积网络文本分类算法
- 图卷积网络多层加权分类算法

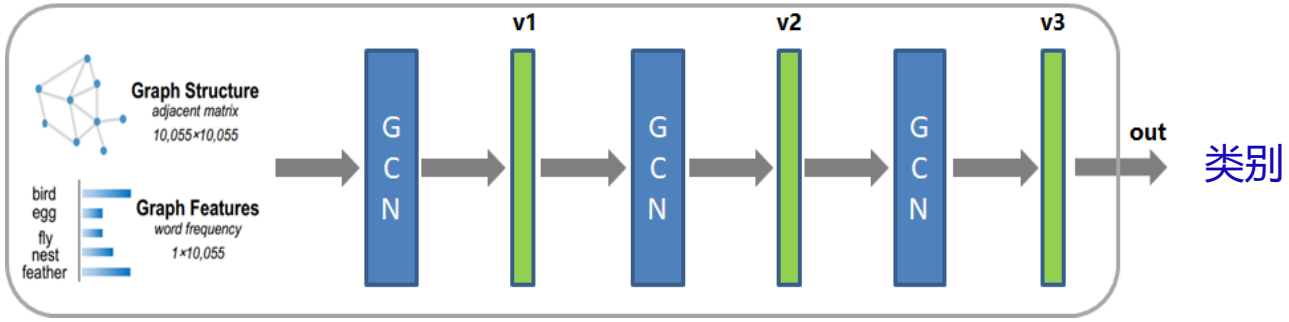
……

11.3 图卷积文本分类方法

- 图卷积网络文本分类算法

原文本

彼得·卡伦（Peter Cullen）刻画了擎天柱，这位利他主义的汽车人领导者来到地球，结束了所有战争….



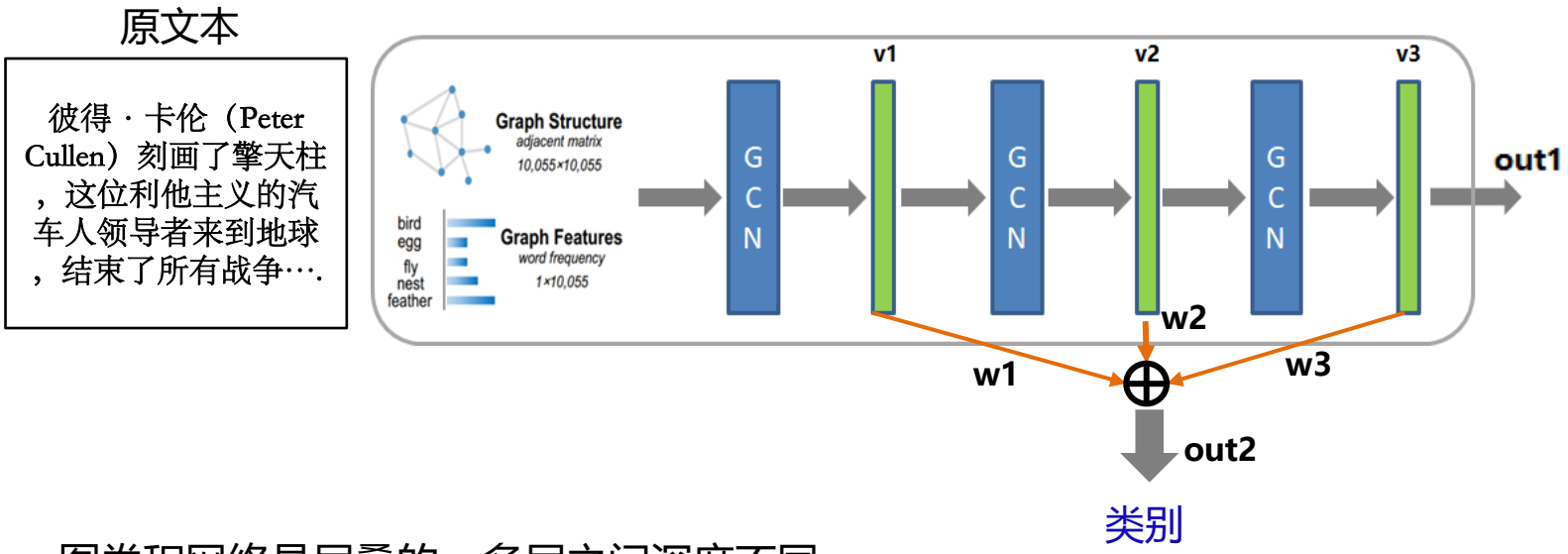
类别

GCN

$$\mathbf{h}_v^k = \sigma\left(\mathbf{W}_k \sum_{u \in N(v) \cup v} \frac{\mathbf{h}_u^{k-1}}{\sqrt{|N(u)||N(v)|}}\right)$$

11.3 图卷积文本分类方法

- 图卷积网络的多层加权



- 图卷积网络是层叠的，多层之间深度不同，表示的语义也不同，可以对多层进行加权，让网络端到端的自适应学习各层权重。

11.3 图卷积文本分类方法

模型参数训练方法：

- 需要先设置以out1为输出，设置w权重均为0，进行预训练
  （直接初始化这些各层的权重，会导致模型无法收敛）
- 预训练完成后，固定网络其他参数，训练各层的wi 参数。

实验结果：

方法	Eng-wiki	Chi-wiki	TVGraz
BOW	85.2	72.4	88.3
Doc2vec	66.5	59.0	89.0
CNN	72.4	58.2	88.1
LSTM	68.1	60.9	85.3
GCN	90.5	86.1	93.0

11.3 图卷积文本分类方法

★ Heterogeneous Graph Attention Networks （HGAT）

■ 动机：

由于短文本标注数据较少，在加上短文本文自身能提供的信息非常有限，一些传统的文本分类模型在短文本分类任务上性能不佳。该文提出一种针对短文本分类的异质图注意力网络 HGAT 来 提升短文本分类任务的性能

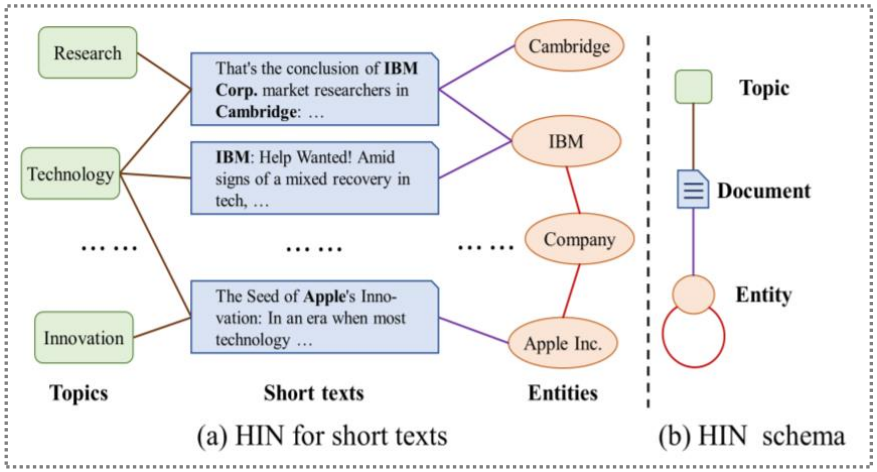
■ 模型结构：

1. 异质图HIN的构建 （图卷积网的输入）
2. 异质图注意力卷积网络 HGAT
   - 异质图卷积网络构建
   - 异质图卷积网络中引入双重注意力机制

11.3 图卷积文本分类方法

1. 异质图HIN的构建

HIN $\mathcal{G} = (\mathcal{V}, \mathcal{E})$； $\mathcal{V}$ 为结点，$\mathcal{E}$为边



- 结点：

  $\mathcal{V} = \boldsymbol{D} \cup \boldsymbol{T} \cup \boldsymbol{E}$ （$\boldsymbol{D}, \boldsymbol{T}, \boldsymbol{E}$ 异质）

  short text $\boldsymbol{D} = \{\mathbf{d_1}, \ldots, \mathbf{d_m}\}$

  topics $\mathbf{T} = \{\mathbf{t_1}, \ldots, \mathbf{t_K}\}$

  entities $\mathbf{E} = \{\mathbf{e_1}, \ldots, \mathbf{e_n}\}$

  附加信息为Topics 和 Entities

  Topics: 使用LDA的方式挖掘短文本的主题分布，然后选择 Top P

  Entities: 在文本中识别实体，然后使用实体链接工具TAGME同时考虑实体之间的关系

- 边：

  如果某篇文档属于主题，则文档与主题词间建立连接

  如果文档包含实体，则文档与实体间建立连接

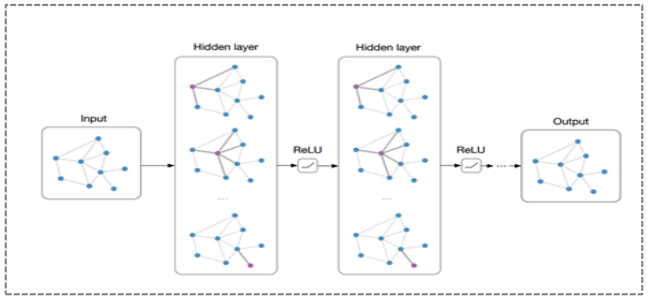
  实体之间如果距离（cosine similarity）在设定阈值内，建立实体间的连接

11.3 图卷积文本分类方法

2. 异质图注意力卷积网络 HGAT

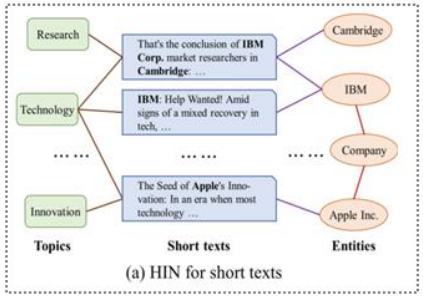
HIN使用图卷积面临的挑战：

GCN



图结点同质

HIN



图结点异质

short text $D = \{d_1, \ldots, d_m\}$

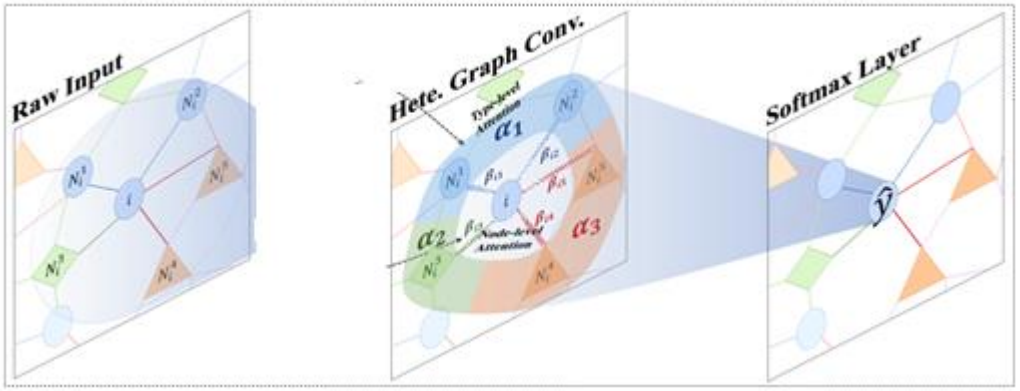
topics $T = \{t_1, \ldots, t_K\}$

entities $E = \{e_1, \ldots, e_n\}$

可以将所有异质结点特征接连，形成同质特征结点，然后用图卷积，但这种方法不同的信息将按同等重要性对待，无法区分不同信息之间的差异。

11.3 图卷积文本分类方法

- 异质图卷积网构建



类型：

$$\mathcal{T} = \{\tau_1, \tau_2, \tau_3\}$$

$$H^{(l+1)} = \sigma(\sum_{\tau \in \mathcal{T}} \tilde{A}_\tau \cdot H_\tau^{(l)} \cdot W_\tau^{(l)})$$

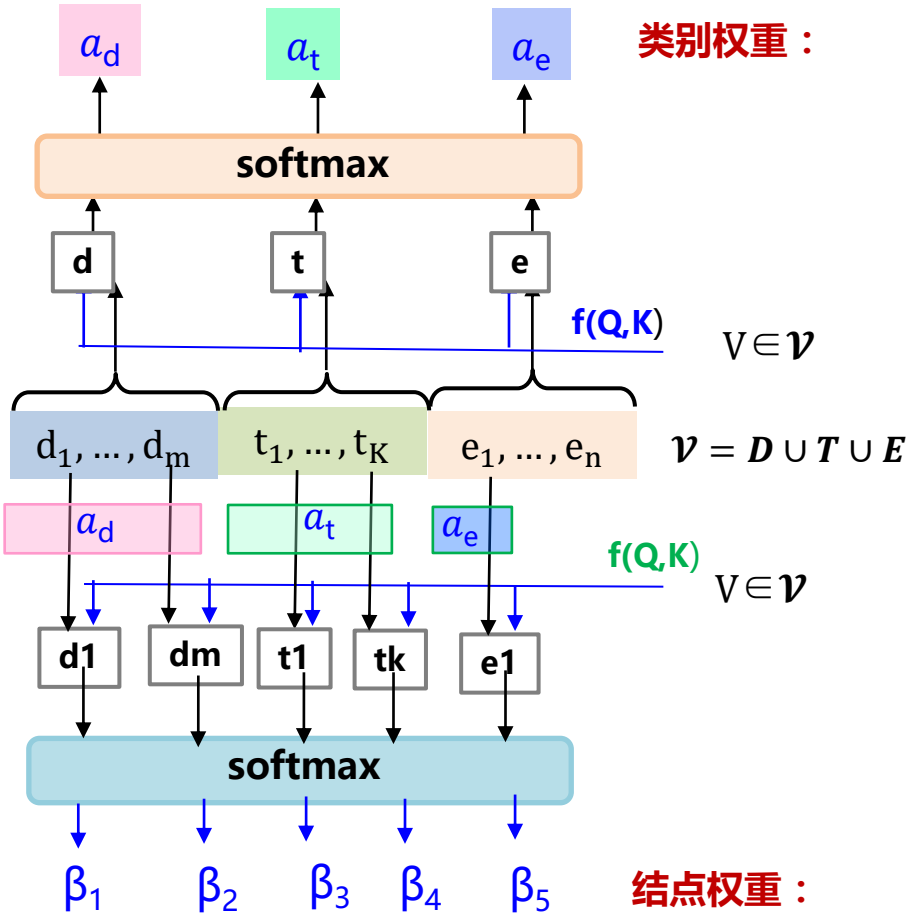
其中：$\tilde{A}_\tau \in \mathbb{R}^{|\mathcal{V}| \times |\mathcal{V}_\tau|}$ 是邻接矩阵 $\tilde{A}$ 的子矩阵，

行：所有结点；列：类型为 τ 的邻接结点

$W_\tau^{(l)} \in \mathbb{R}^{q^{(l)} \times q^{(l+1)}}$ 类型为 τ 的结点参数

11.3 图卷积文本分类方法

- 异质图卷积网络中引入双重注意力机制



Type level attention

$$h_\tau = \sum_{v'} \tilde{A}_{vv'} h_{v'}$$

$$a_\tau = \sigma(\mu_\tau^T \cdot [h_v || h_\tau])$$

$$\alpha_\tau = \frac{\exp(a_\tau)}{\sum_{\tau' \in \mathcal{T}} \exp(a_{\tau'})}$$

μ_τ : attention vector for the type τ

Node level attention

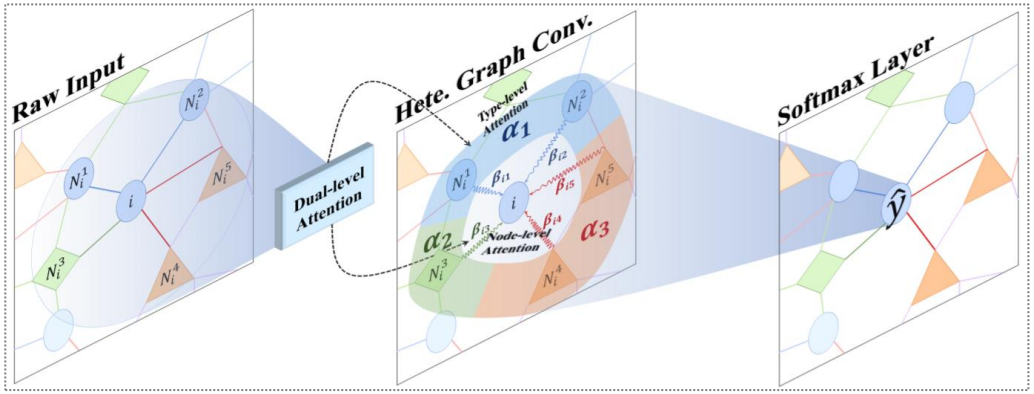
$$b_{vv'} = \sigma(\nu^T \cdot \alpha_{\tau'} [h_v || h_{v'}])$$

$$\beta_{vv'} = \frac{\exp(b_{vv'})}{\sum_{i \in \mathcal{N}_v} \exp(b_{vi})}$$

ν : attention vector for the node-level

11.3 图卷积文本分类方法

异质图注意力卷积网络 HGAT构建



输入： HIN

输出： 短文类别

参数： $W_\tau^{(l)}$ μ_τ ν

异质图卷积网:

$$H^{(l+1)} = \sigma\Big(\sum_{\tau\in\mathcal{T}} \tilde{A}_\tau \cdot H_\tau^{(l)} \cdot W_\tau^{(l)}\Big)$$

HGAT:

$$H^{(l+1)} = \sigma\Big(\sum_{\tau\in\mathcal{T}} \mathcal{B}_\tau \cdot H_\tau^{(l)} \cdot W_\tau^{(l)}\Big)$$

$$Z = \mathrm{softmax}\big(H^{(L)}\big)$$

$\mathcal{B}_\tau$ represents the attention matrix, whose element in the v^{th} row v'^{th} column is $\beta_{vv'}$

Node level attention

$$b_{vv'} = \sigma(\nu^T \cdot \alpha_{\tau'}[h_v||h_{v'}])$$

$$\beta_{vv'} = \frac{\exp(b_{vv'})}{\sum_{i\in\mathcal{N}_v}\exp(b_{vi})}$$

ν : attention vector for the node-level

11.3 图卷积文本分类方法

- **模型学习：**

优化目标： 最小化交叉熵损失函数

$$\mathcal{L} = -\sum_{i \in D_{\text{train}}} \sum_{j=1}^{C} Y_{ij} \cdot \log Z_{ij} + \eta \|\Theta\|_2$$

where C is the number of classes, D_{train} is the set of short text indices for training, Y is the corresponding label indicator matrix, Θ is model parameters, and η is regularization factor. For model optimization, we adopt the gradient descent algorithm.

11.3 图卷积文本分类方法

■ **实验结果：**

Dataset	SVM +TFIDF	SVM +LDA	CNN -rand	CNN -pretrain	LSTM -rand	LSTM -pretrain	PTE	TextGCN	HAN	HGAT
AGNews	57.73	65.16	32.65	67.24	31.24	66.28	36.00	<u>67.61</u>	62.64	**72.10***
Snippets	63.85	63.91	48.34	77.09	26.38	75.89	63.10	<u>77.82</u>	58.38	**82.36***
Ohsumed	41.47	31.26	35.25	32.92	19.87	28.70	36.63	<u>41.56</u>	36.97	**42.68***
TagMyNews	42.90	21.88	28.76	57.12	25.52	<u>57.32</u>	40.32	54.28	42.18	**61.72***
MR	56.67	54.69	54.85	58.32	52.62	<u>60.89</u>	54.74	59.12	57.11	**62.75***
Twitter	54.39	50.42	52.58	56.34	54.80	<u>60.28</u>	54.24	60.15	53.75	**63.21***

Test accuracy (%) of different models on six standard datasets. The second best results are underlined. The note * means our model significantly outperforms the baselines based on t-test ($p < 0.01$).

Dataset	GCN -HIN	HGAT w/o ATT	HGAT -Type	HGAT -Node	HGAT
AGNews	70.87	70.97	71.54	71.76	**72.10***
Snippets	76.69	80.42	81.68	81.93	**82.36***
Ohsumed	40.25	41.31	41.95	42.17	**42.68***
TagMyNews	56.33	59.41	60.78	61.29	**61.72***
MR	60.81	62.13	62.27	62.31	**62.75***
Twitter	61.59	62.35	62.95	62.45	**63.21***

Test accuracy (%) of our variants.

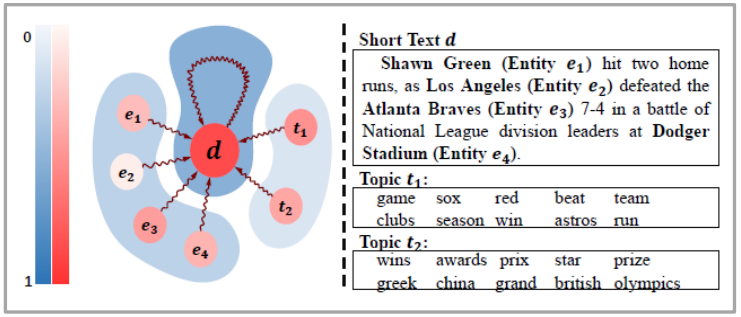


Figure 5: Visualization of the dual-level attention including node-level attention (shown in red) and type-level attention (shown in blue). Each topic t is represented by top 10 words with highest probabilities.

参考文献：

[grave et, al. 2017] Bag of Tricks for Efficient Text Classification

[Kim et, al. 2014] Convolutional Neural Networks for Sentence Classification

[Zhang et al. 2016] Character-level Convolutional Networks for Text Classification

[Liu et al.2016] Recurrent Neural Network for Text Classification with Multi-Task Learning

[Lai et al. 2015] Recurrent Convolutional Neural Networks for Text Classification

[Yang et al. 2016] Hierarchical Attention Networks for Document Classification

[Yu et al.2018] Learning cross-modal correlations by exploring inter-word semantics and stacked co-attention.

[Yu et al.2018] Modeling Text with Graph Convolutional Network for Cross-Modal Information Retrieval.

[Hu et al.2019] Heterogeneous Graph Attention Networks for Semi-superivised Short Text Classification

在 此 表 示 感 谢!

谢谢各位！

Q&A