2021年度暑期强化课程

深度学习与自动摘要

授课人：曹亚男

中国科学院信息工程研究所
INSTITUTE OF INFORMATION ENGINEERING,CAS

9.自动摘要

9.自动摘要

什么是自动摘要?

- Automatic summarization is the process of shortening a text document with software, in order to create a summary with the major points of the original document. Technologies that can make a coherent summary take into account variables such as length, writing style and syntax.

--维基百科

- 所谓自动文摘就是利用计算机自动地从原始文献中提取文摘，文摘是全面准确地反映某一文献中心内容的简单连贯的短文。

--百度百科

为什么做自动摘要?

- WWW的出现创造了大量的数据，产生了信息过载的问题
- 一个简短的摘要，传递文档的核心思想，有助于快速查找相关信息
- 自动摘要还提供了一种将相似文档聚类的方案并提供对应摘要

"Summaries as short as 17% of the full text length speed up decision making twice, with no significant degradation in accuracy."

"Query-focused summaries enable users to find more relevant documents more accurately, with less need to consult the full text of the document."

Mani, Inderjeet, et al. "SUMMAC: a text summarization evaluation." Natural Language Engineering 8.1 (2002): 43-68.

生成式文摘 vs. 抽取式文摘

- **抽取式文摘**：从源文本中选择能够表示摘要的关键词（或关键句），对这些关键词（或关键句）进行重组 ➡ **排序/分类**
- **生成式文摘**：通过对源文本的语义进行理解，生成对应的文摘，文摘中可以出现源文本中未出现的词 ➡ **序列转换**

例 子

2月23日，国家统计局发布2021年1月份70个大中城市商品住宅销售价格变动情况显示，一二三线城市一二手房价涨幅均有所扩大。根据简单算术平均计算，1月，一二三线城市新房价格环比分别上涨0.6%、0.36%、0.18%；二手房价格分别上涨1.33%、0.35%、0.29%。

生成式摘要

一二三线城市1月房价涨幅均扩大

抽取式摘要

一二三线城市一二手房价涨幅均有所扩大

9.自动摘要

传统的方法

- 从源文本中选择能够表示摘要的关键词（或关键句），对这些关键词（或关键句）进行重组
  - 将源文本表示为词频表
  - 根据句子的各个词重要性、句子位置、句子与首句相似度等来选择重要的文本片段
  - 对文本信息进行整合得到抽取的词句，组成摘要

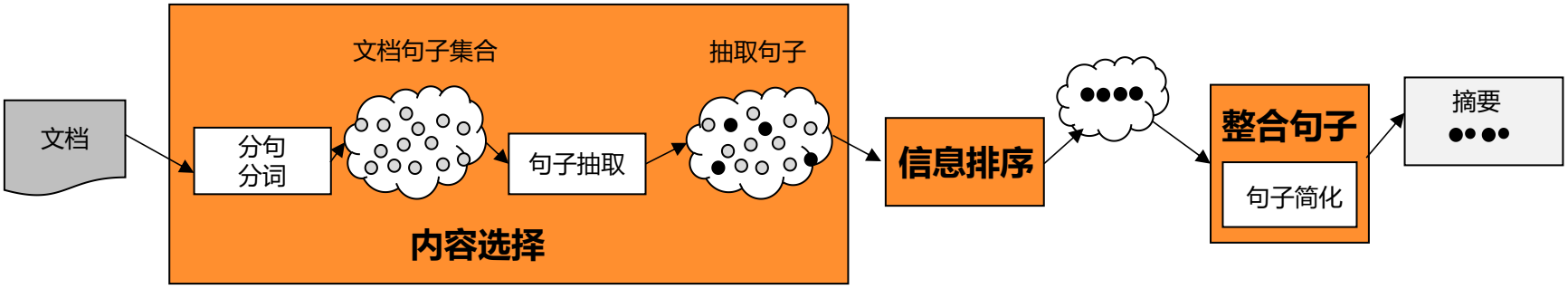
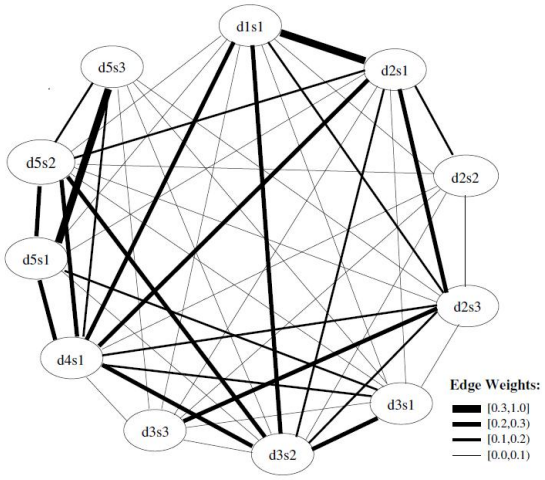


图1 抽取式摘要流程图

Hans Peter Luhn. The automatic creation of literature abstracts. IBM Journal of research and development, 2(2):159–165, 1958.

基于Graph的方案

- 将语料库中的句子构建成图
  - 图上每个节点表示一个句子，每一条边由节点之间的余弦相似性所决定
  - 采用基于图的方法，从这些候选句子中选择关键句



句子s中特定词条的出现次数

$$\text{idf-modified-cosine}(x, y) = \frac{\sum_{w \in x,y} tf_{w,x} tf_{w,y} (idf_w)^2}{\sqrt{\sum_{x_i \in x} (tf_{x_i,x} idf_{x_i})^2} \times \sqrt{\sum_{y_i \in y} (tf_{y_i,y} idf_{y_i})^2}}$$

逆文档频率 $idf_t = \log \frac{N}{df_t}$

G. Erkan and D. R. Radev.2004. LexRank: Graph-based Centrality as Salience in Text Summarization. Journal of Artificial Intelligence Research (JAIR).

基于Bert的抽取式文摘：BertSum

- 输入层
  - 每句话前后插入CLS和SEP符号
  - 用间隔符区分一篇文档中的多个句子
- 摘要层
  - Linear Layer+Sigmoid
  - Inter-sentence Transformer+Sigmoid
  - RNN+Sigmoid

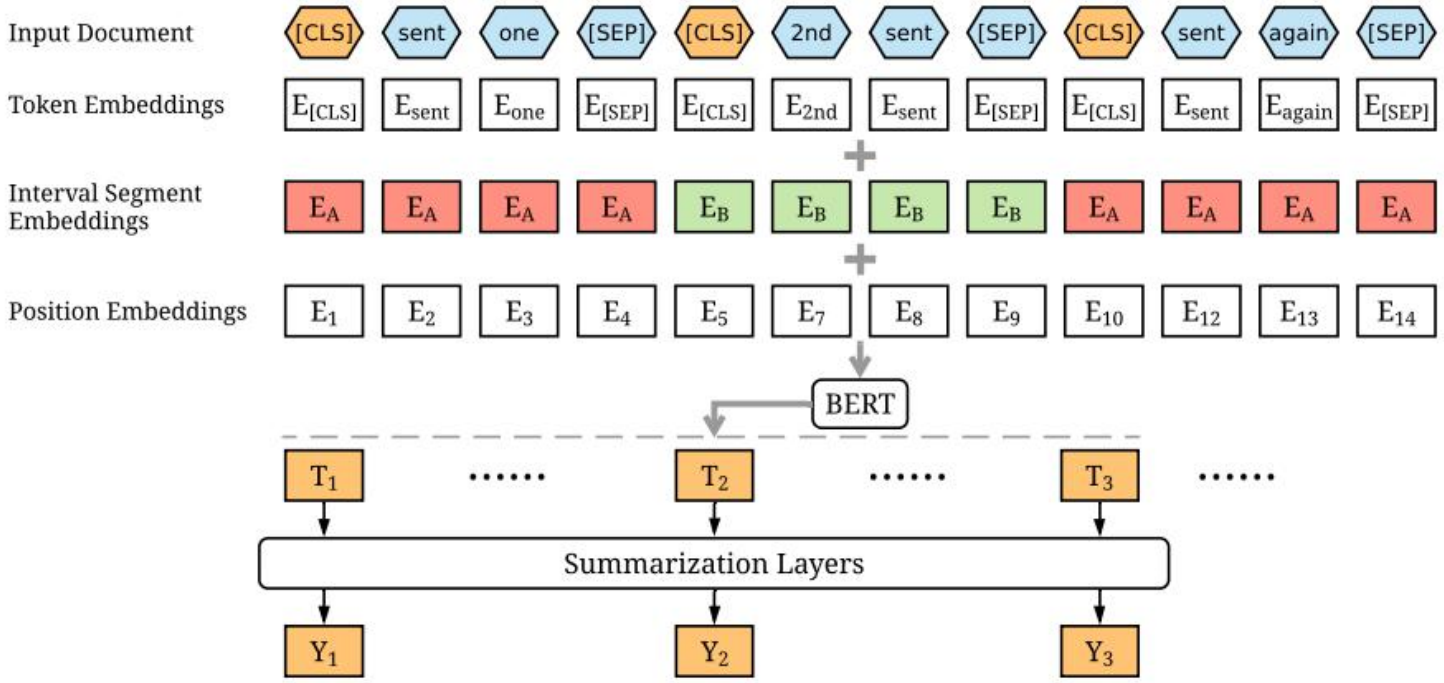


Figure 1: The overview architecture of the BERTSUM model.

Fine-tune BERT for Extractive Summarization, 2019

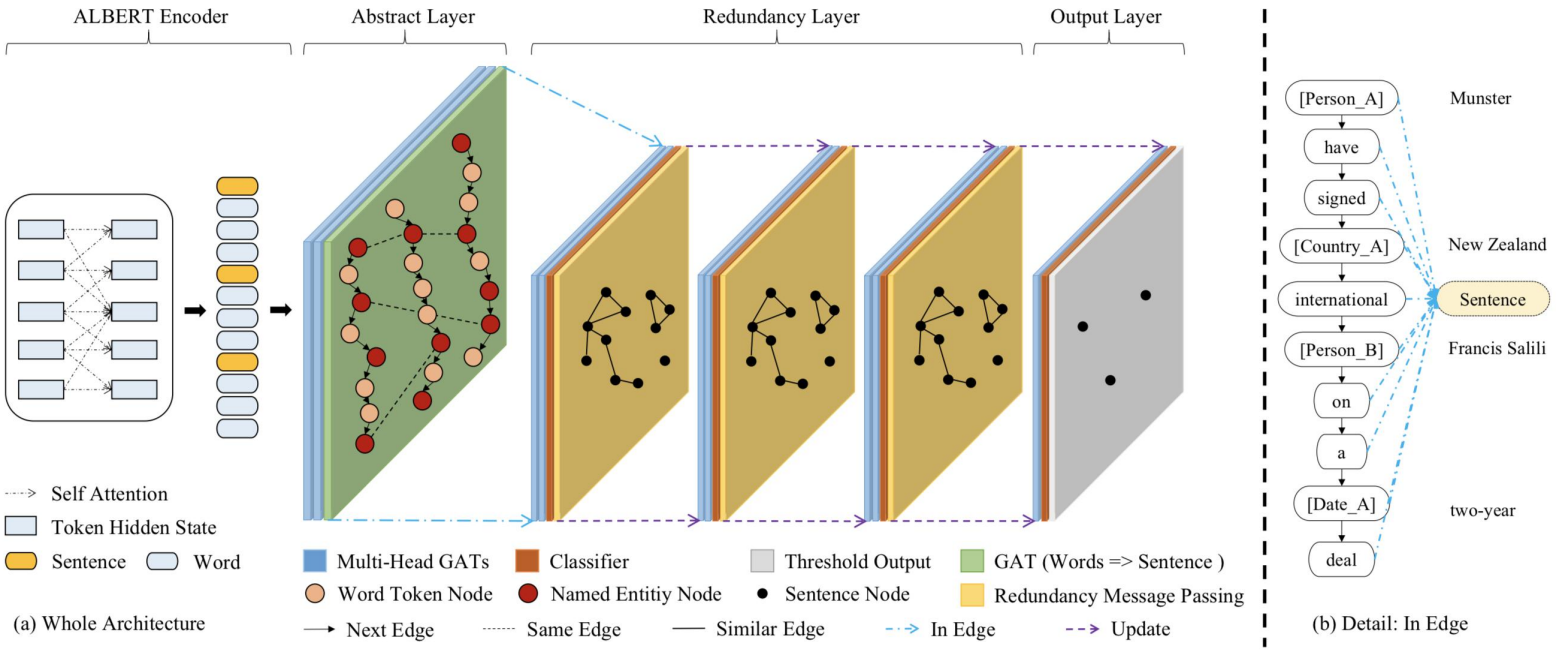
抽取式文摘的主要问题

- 文档中的每句话有一个二进制标签表示本句话是否抽取为摘要，但这些标签并非相互独立，而是全局依赖于其他相似句子的标签
- 相似的句子会有截然相反标签的情况, 增加模型学习难度

	Salience	Label	Sentence
sent1:	0.7	0	Deanna Holleran is charged in murder.
sent2:	0.1	0	Jackson County Prosecutor Jean Peters Baker announced today.
sent3:	0.7	1	Deanna Holleran faces a charge of traffic accident.
sent4:	0.7	1	The fatal traffic accident is a murder.
sent5:	0.2	0	It took the life of Marianna Hernandez near 9th Hardesty.
Summary:			Woman faces a charge of murder for a fatal traffic accident.

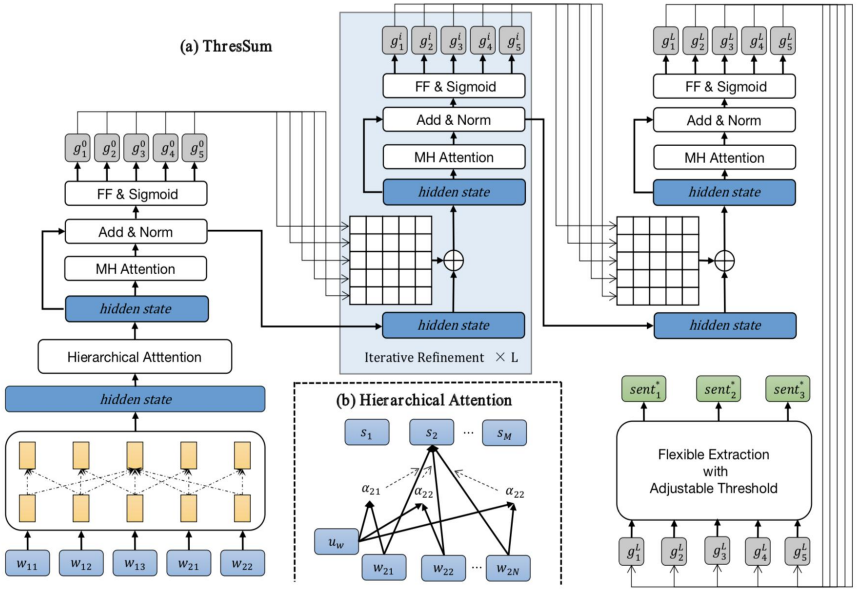
基于层次异构图网络的抽取式摘要 (EMNLP 2020)

- **问题**：抽取式文摘抽到的句子之间存在语义冗余
- **方法**：使用层次级异构图网络对文档进行词级别抽象信息和句子级别冗余信息的建模，能够抽取出更加灵活和准确的摘要。在CNN/DM数据集上进行测试，ROUGE-1达44.68，较传统方法提高13%以上



抽取句子数量灵活的摘要 (AAAI 2021)

- **问题**：传统的抽取式摘要对于不同类型和长度的文档会抽取相同句子数量的摘要
- **方法**：通过对文档句子语义进行预标签和迭代式更新，最终使用一个阈值灵活的控制抽取摘要句子的数量, 大大提高了摘要系统的灵活性



Models	CNN/DM R-1	CNN/DM R-2	CNN/DM R-L	NYT R-1	NYT R-2	NYT R-L
Abstractive						
ABS (2015)	35.46	13.30	32.65	42.78	25.61	35.26
PGC (2017)	39.53	17.28	36.38	43.93	26.85	38.67
TransformerABS (2017)	40.21	17.76	37.09	45.36	27.34	39.53
$T5_{Large}$ (2019)	43.52	21.55	40.69	-	-	-
$BART_{Large}$ (2019b)	44.16	21.28	40.90	48.73	29.25	44.48
$PEGASUS_{Large}$ (2019b)	44.17	21.47	41.11	-	-	-
$ProphetNet_{Large}$ (2020)	44.20	21.17	41.30	-	-	-
Extractive						
Oracle (Sentence)	55.61	32.84	51.88	64.22	44.57	57.27
Lead-3 †	40.42	17.62	36.67	41.80	22.60	35.00
SummaRuNNer † ⋆ (2017)	39.60	16.20	35.30	42.37	23.89	38.74
Exconsumm ‡ ⋆ (2019)	41.7	18.6	37.8	43.18	24.43	38.92
$PNBERT_{Base}$ † ⋆(2019a)	42.69	19.60	38.85	-	-	-
$DiscoBERT_{Base}$ (2020)	43.77	20.85	40.67	-	-	-
$BERTSUMEXT_{Large}$ † ⋆(2019)	43.85	20.34	39.90	48.51	30.27	44.65
$MATCHSUM_{Base}$ ‡ ⋆(2020)	44.41	20.86	40.55	-	-	-
$ThresSum_{Large}$ ‡ •(Ours)	**44.59**	**21.15**	**40.76**	**50.08**	**31.77**	**45.21**

Flexible Non-Autoregressive Extractive Summarization with Threshold: How to Extract a Non-Fixed Number of Summary Sentences

基于深度差分放大器的摘要 (ACL 2021)

- **问题：**摘要数据集中由于抽取出来的句子是少数，因此标记为“1”和标记为“0”的样本存在严重的不均衡问题
- **方法：**采用深度差分放大器框架**增强摘要句子的特征**，使用深度差分放大器模型计算和放大每个句子之间的语义差异，优化目标采用加权交叉熵使得模型更关注句子中的重要信息

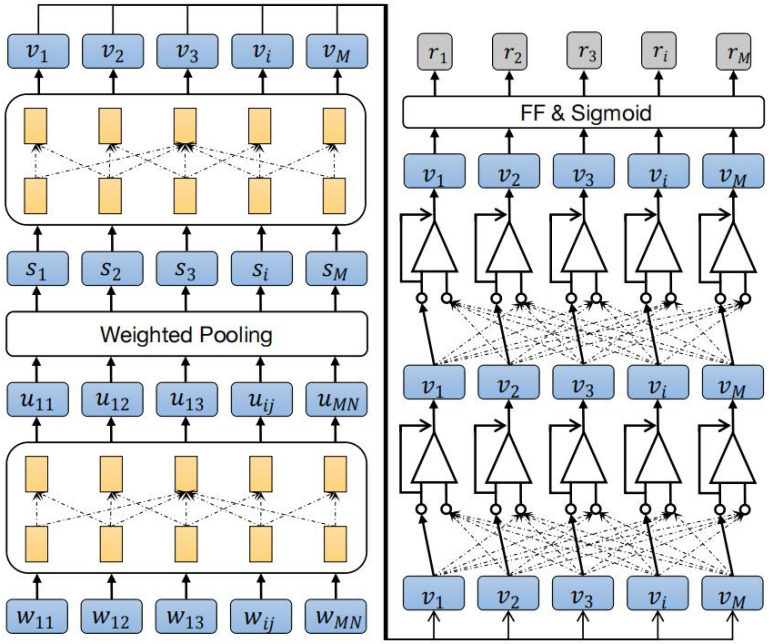


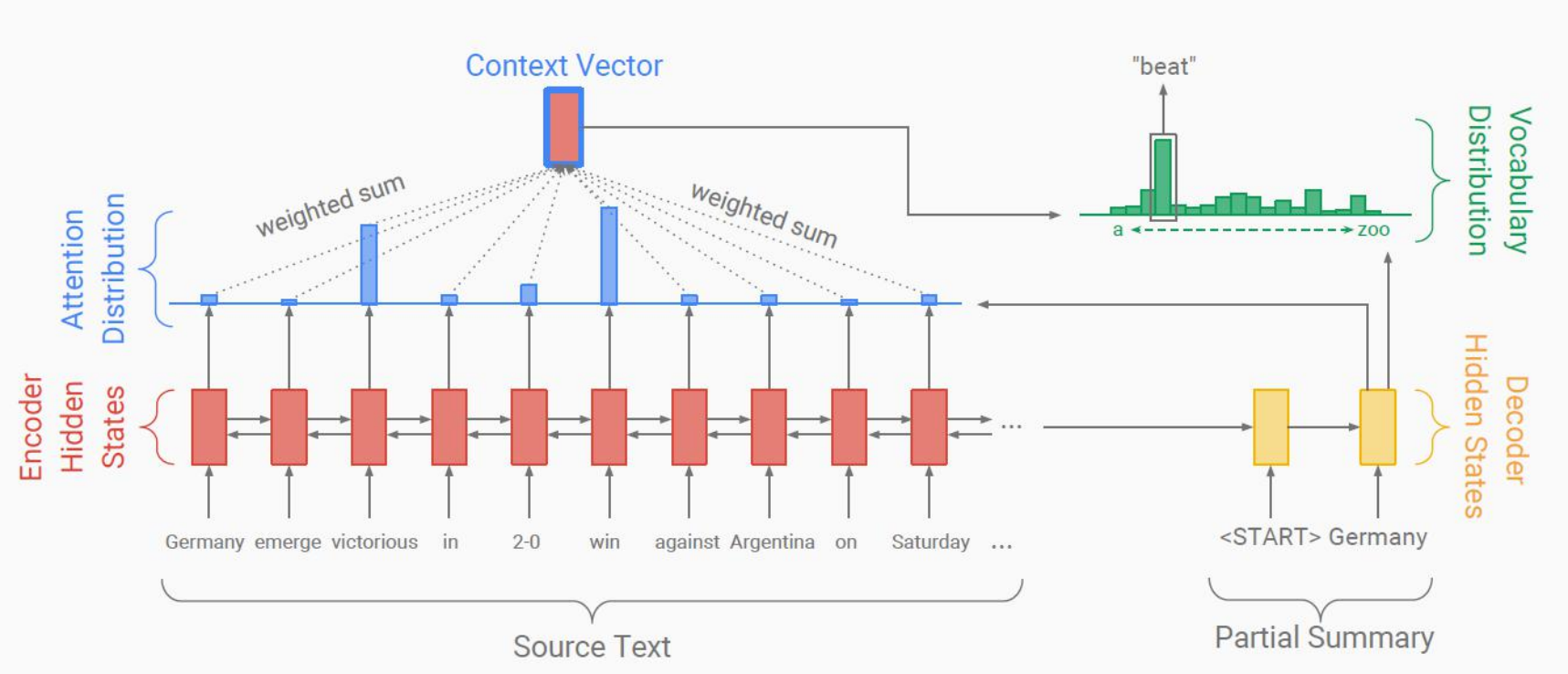
Table 3: ROUGE F1 on NYT.

Models	NYT R-1	NYT R-2	NYT R-L
Abstractive			
ABS (2015)	42.78	25.61	35.26
PGC (2017)	43.93	26.85	38.67
TransformerABS (2017)	45.36	27.34	39.53
BART$_{Large}$ (2019a)	48.73	29.25	44.48
Extractive			
Lead-3	41.80	22.60	35.00
Oracle (Sentence)	64.22	44.57	57.27
SummaRuNNer (2017)	42.37	23.89	38.74
Exconsumm (2019)	43.18	24.43	38.92
JECS (2019)	45.50	25.30	38.20
BERTSUMEXT$_{Base}$ (2019)	46.66	26.35	42.62
HIBERT$_{Large}$ (2019b)	49.47	**30.11**	41.63
DifferSum$_{Large}$	**49.52**	29.78	**43.86**

9.自动摘要

基于深度学习的生成式摘要

- 把摘要生成看作many-to-many的序列转换，与机器翻译的机制类似



Attention Based Seq2seq Model

Alexander M Rush, Sumit Chopra, and Jason Weston. A neural attention model for abstractive sentence summarization. arXiv preprint arXiv:1509.00685, 2015.

基础模型的诸多问题

1. Encoder端仅考虑词向量，输入特征单一
2. 源文中的某些关键词很少出现但很重要，由于模型基于词向量表示，对低频词的处理并不友好
3. 在源文档非常长的数据集中，关键句对于生成摘要也很重要
4. 解码端容易出现相同的单词
5. Decoder词汇表过大而造成softmax层的计算瓶颈
6. 在训练阶段，解码端的前t-1个词是已知的，而测试阶段没有这样的监督，因此导致预测序列的累计错误
7. 摘要本身可以是灵活多样的，可以有不同的句子顺序等，而最大似然估计完全忽略了这个特性

生成式摘要改进方案

模型结构优化

求解目标优化

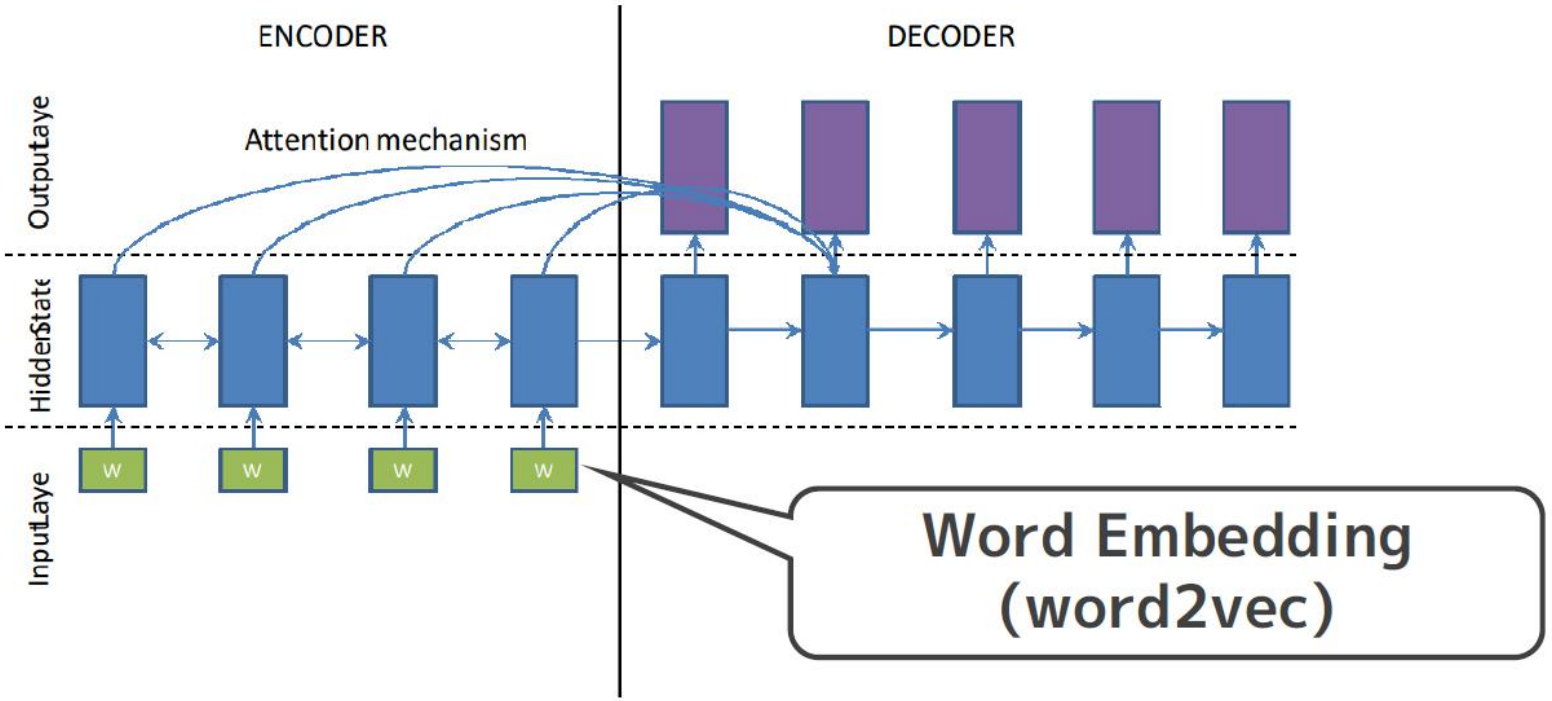
生成式摘要改进方案

模型结构优化

求解目标优化

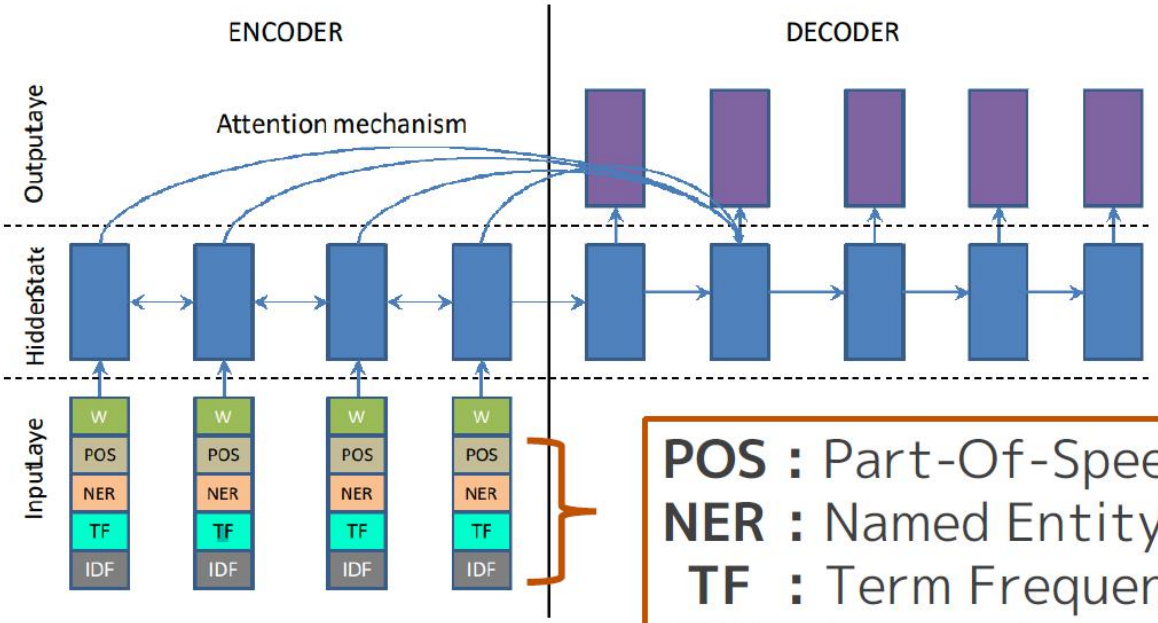
输入层-- Feature-rich Encoder

- 问题：
  - Encoder端只考虑到词向量（Word Embedding），输入特征较少



输入层-- Feature-rich Encoder

- 解决方案：
  - Encoder端的文档表示使用word Embedding+语言特性共同表示



POS：Part-Of-Speech
NER：Named Entity Tag
TF：Term Frequency
IDF：Inverse Document Frequency

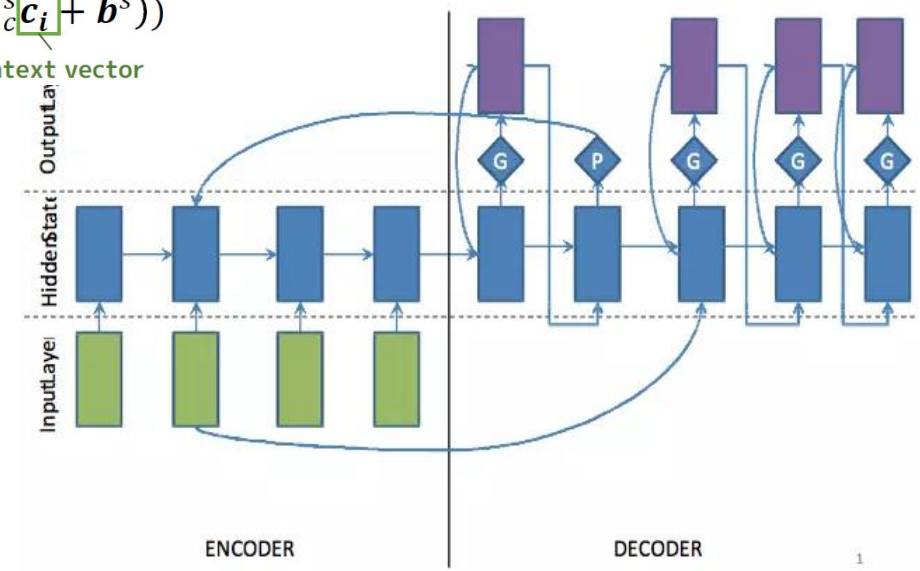
Abstractive Text Summarization using Sequence-to-sequence RNNs and Beyond, 2016

利用关键词-- Switching Generator/Pointer

- 问题：有些关键词频度很低但很重要，由于模型基于word embedding，对低频词的处理并不友好。
- 解决思路：模型中decoder端设置一个开关：
  - Switch -> on（G）：以正常的方式生成一个单词；
  - Switch -> off（P）：decoder生成一个原文单词位置的指针，该指针位置的词嵌入将作为decoder中下一个隐状态的输入

$$P(s_i = 1) = \sigma(\boldsymbol{v}^s \cdot (\boldsymbol{W}_h^s \boldsymbol{h}_i + \boldsymbol{W}_e^s \boldsymbol{E}[o_{i-1}] + \boldsymbol{W}_c^s \boldsymbol{c}_i + \boldsymbol{b}^s))$$

decoder的隐藏层　先前的词向量　Context vector



利用关键句-- Hierarchical Attention

- 动机：
  - 在源文档非常长的数据集中，除了识别文档中的关键词之外，还需要识别关键句
- 解决思想：
  - 在源端使用两个双向RNN来捕获两个级别的特征，分别是单词级和句子级
  - Attention机制同时在单词和句子两个层面上运行

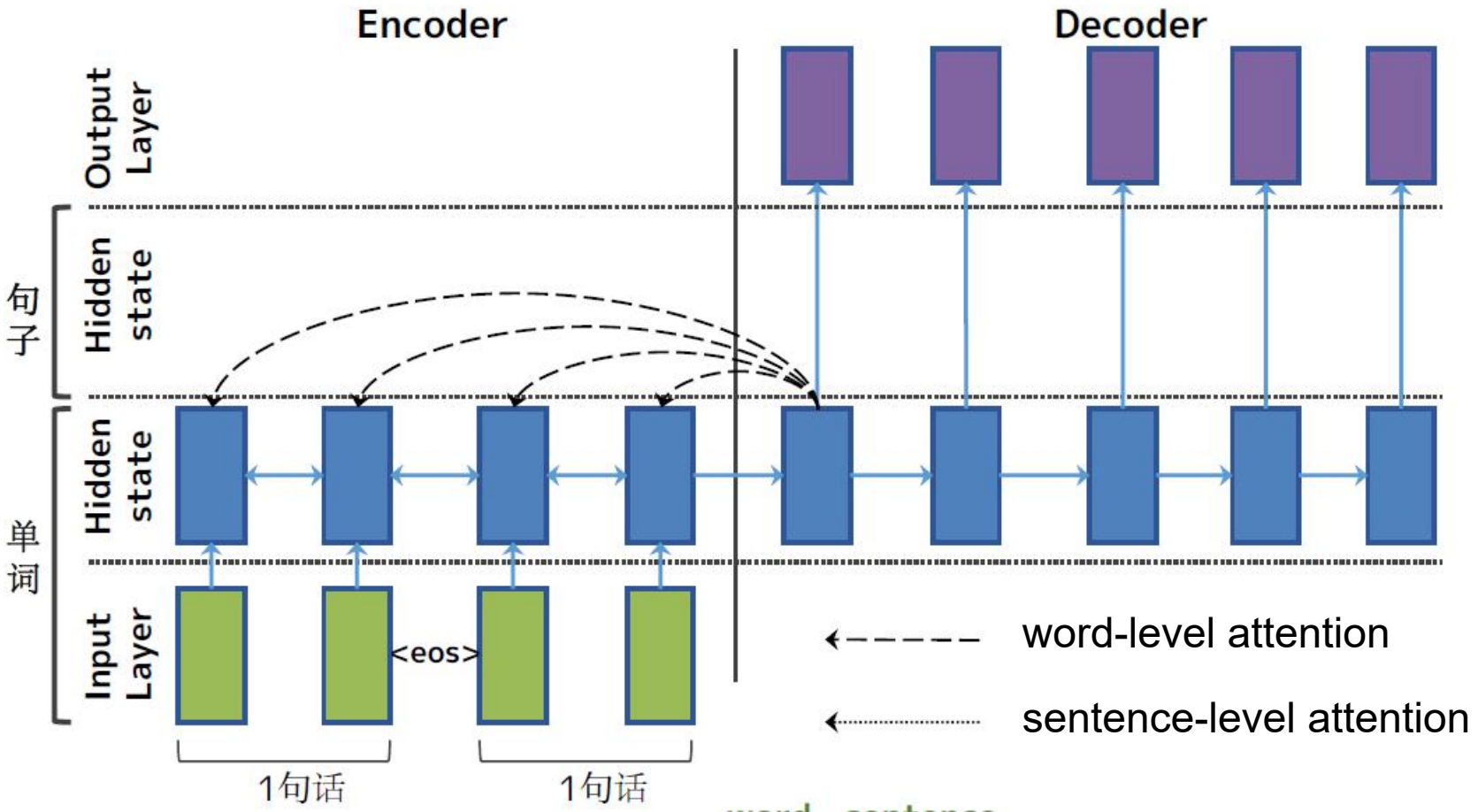
Re-scaled attention

word level sentence level

$$P^a(j) = \frac{P_w^a(j)P_s^a(s(j))}{\sum_{k=1}^{N_d} P_w^a(k)P_s^a(s(k))}$$

Nallapati, Ramesh, et al. "Abstractive Text Summarization using Sequence-to-sequence RNNs and Beyond." CoNLL 2016 (2016): 280.

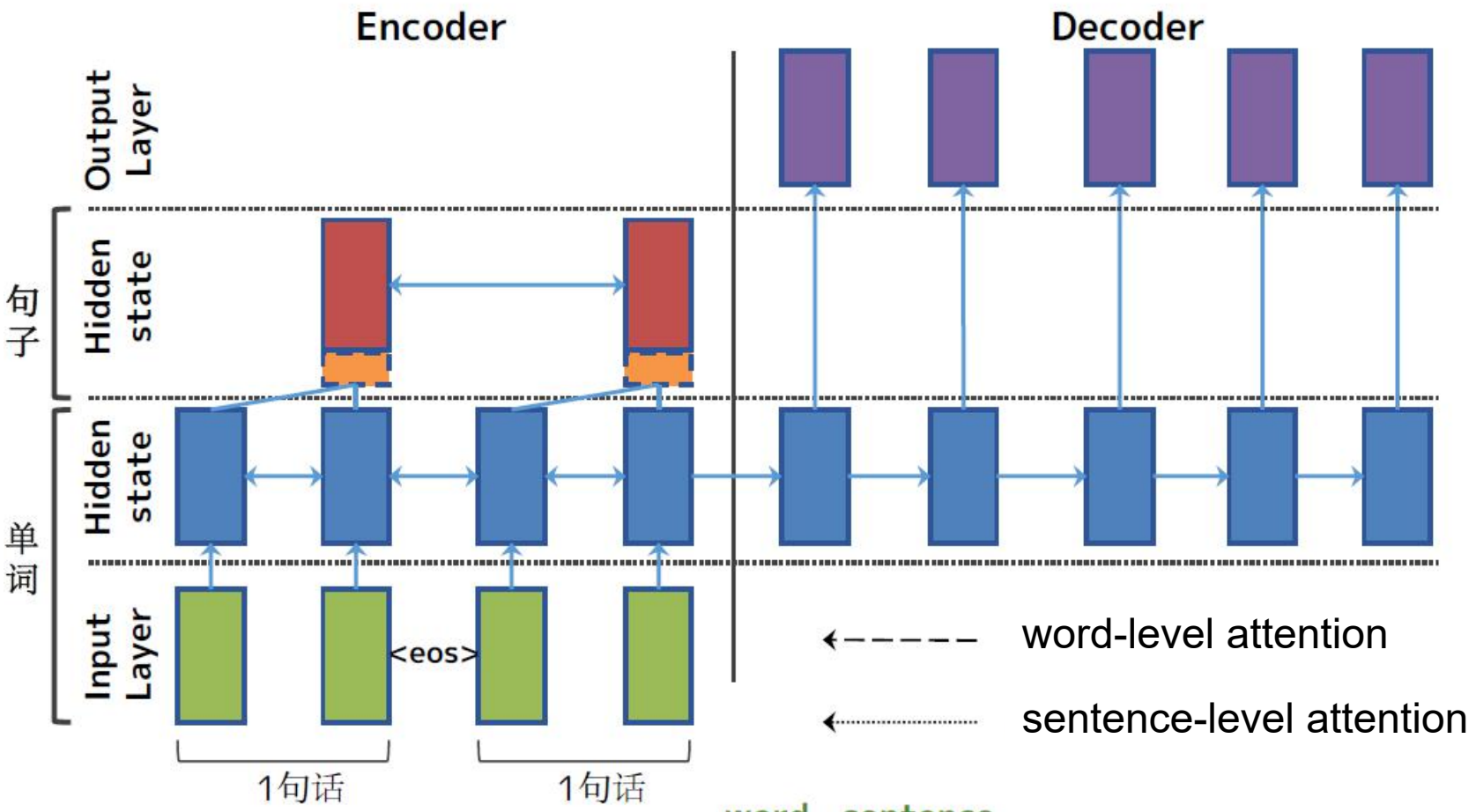
利用关键句-- Hierarchical Attention



$$P^{a}_{new}(j) = \frac{P^{a}_{w}(j)\,P^{a}_{s}(s(j))}{\sum_{k=1}^{N_d} P^{a}_{w}(k)\,P^{a}_{s}(s(k))}$$

word sentence

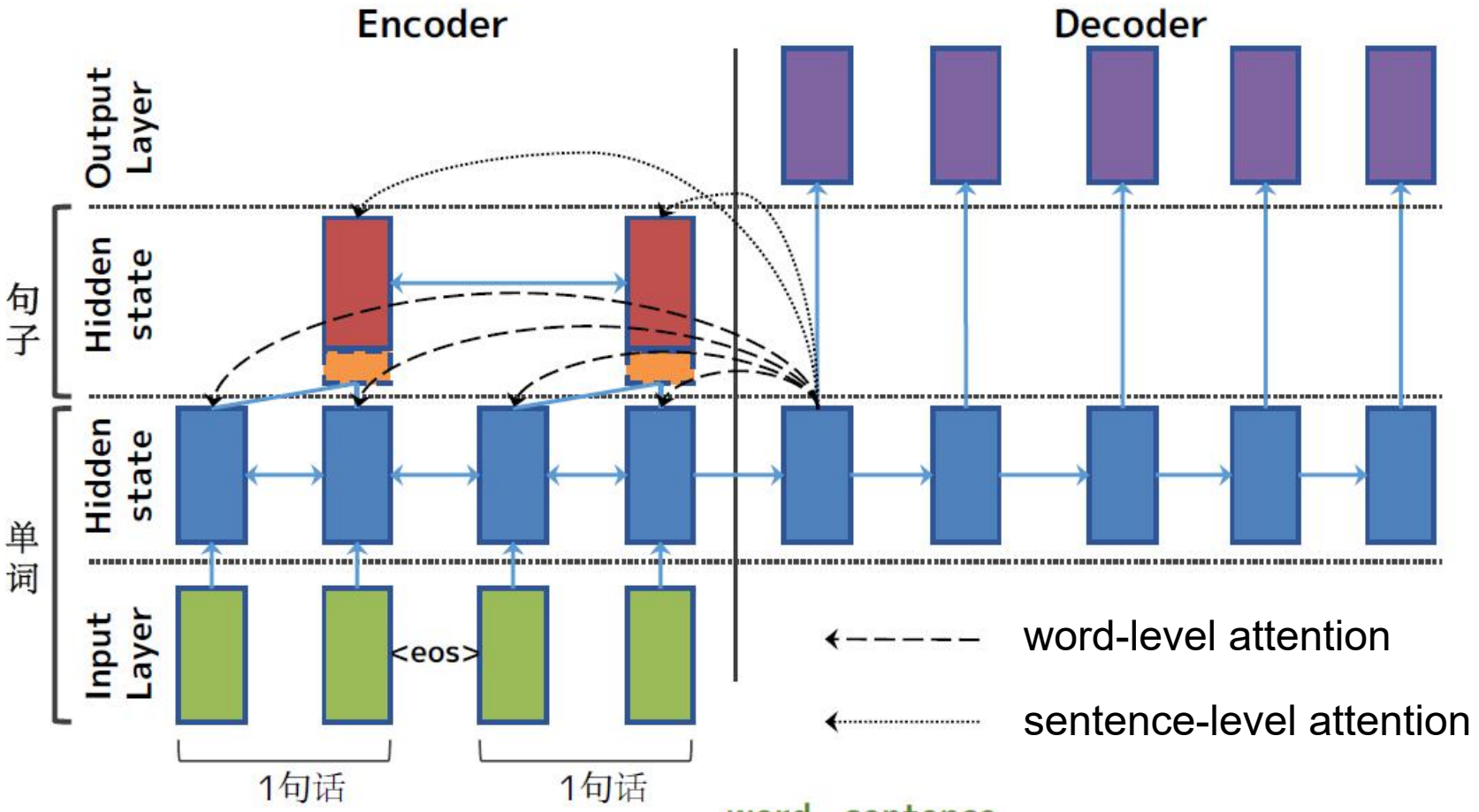
利用关键句-- Hierarchical Attention



$$P^{a}_{new}(j) = \frac{P^{a}_{w}(j) P^{a}_{s}(s(j))}{\sum_{k=1}^{N_d} P^{a}_{w}(k) P^{a}_{s}(s(k))}$$

word sentence

结构优化 -- Hierarchical Attention

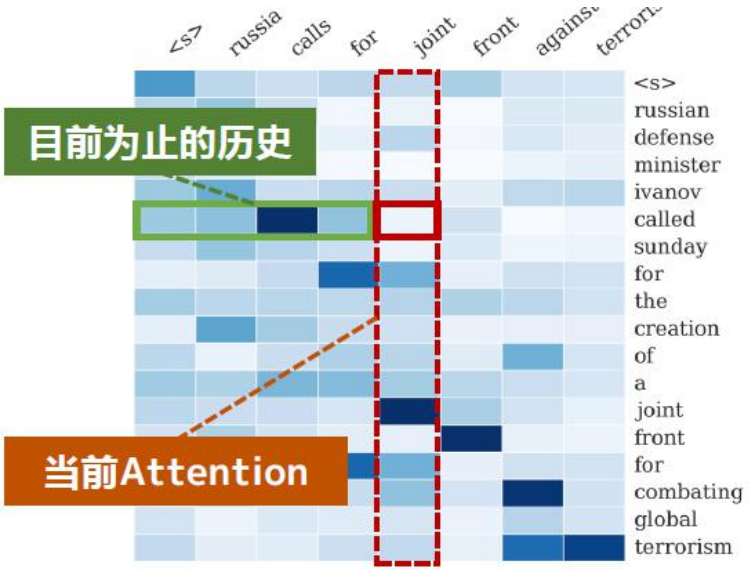


$$P^{a}(j)^{new} = \frac{P_w^a(j)\,P_s^a(s(j))}{\sum_{k=1}^{N_d} P_w^a(k)\,P_s^a(s(k))}$$

word sentence

去除重复词-- Temporal Attention

- 问题：
  - 解码端出现相同的单词（重复）： e.g. Germany beat Germany beat beat …
- 思想：
  - 解码时输出哪些词与Attention有关
  - 利用到目前为止的Attention历史信息，如果之前对某一个单词已经赋予过很高的Attention值，则适度降低这个值



<s> Russia calls for [ ]

$$\alpha_t \propto \alpha'_t \quad \beta_t$$

α'_t：词t当前的Attention

β_t：词t过去的Attention总和

$$\beta_t = \sum_{k=1}^{t-1} \alpha'_k$$

Attention Coverage Model [See+ 2017]
Intra-Attention Model [Paulus+ 2017]

Sankaran, Baskaran, et al. "Temporal attention model for neural machine translation." arXiv preprint arXiv:1608.02927 (2016).

大词表问题-- LVT

- 问题：
  - decoder词汇表过大而造成softmax层的计算瓶颈
- 思想：
  - 每个mini batch中decoder的词汇表来源于两个部分：encoder的词汇表和一定数量的高频词构成
- 优势：
  - LVT是一个针对文本摘要问题的有效方法，考虑到了摘要中的大部分词都是来源于源文之中，所以将decoder的词汇表做了约束
  - 降低了decoder词汇表规模，加速了训练过程

Sébastien Jean, Kyunghyun Cho, Roland Memisevic, and Yoshua Bengio. 2014. On using very large target vocabulary for neural machine translation. CoRR, abs/1412.2007.

大词表问题-- Vocabulary expansion

- 问题：
  - LVT技术很好地解决了decoder在生成词时的计算瓶颈问题，但却不能够生成新颖的有意义的词
- 思想：
  - 扩展LVT词汇表的技术，将原文中所有单词的最邻近单词扩充到词汇表中
- 优势：
  - 词汇表的扩展是一项非常重要的技术，word embedding在这里起到了关键作用。用原文中单词的最邻近单词来丰富词汇表，不仅仅利用LVT加速的优势，也弥补了LVT带来的问题

Nallapati, Ramesh, et al. "Abstractive Text Summarization using Sequence-to-sequence RNNs and Beyond." CoNLL 2016 (2016): 280.

生成式摘要改进方案

模型结构优化

求解目标优化

Seq2Seq目标优化存在的问题

- 问题
  - exposure bias：在训练阶段，解码端的前t-1个词是已知的，而测试阶段没有这样的监督，因此导致预测序列的累计错误
  - large number of potentially valid summaries：摘要本身可以是灵活多样的，可以有不同的句子顺序等，而最大似然估计完全忽略了这个特性
- 解决思路
  - 强化学习 -- 引入ROUGE指标。但由于ROUGE不可导，无法直接对ROUGE进行梯度计算。因此，可以考虑用强化学习将ROUGE作为reward（human 反馈）
  - 生成对抗网络 -- 引入判别器，极小化生成摘要与真实摘要的差距。（discriminator 反馈）

摘要评估：Rouge

- Rouge通过将自动生成的摘要与一组参考摘要（通常是人工生成的）进行比较计算，得出相应的分值，以衡量自动生成的摘要与参考摘要之间的“相似度”

$$\text{Rouge} - \text{N} = \frac{\sum_{s\in\{\text{ref } summ\}} \sum_{gram_n \in S} Count_{match}(gram_n)}{\sum_{s\in\{\text{ref } summ\}} \sum_{gram_n \in S} Count\,(gram_n)}$$

Chin-Yew Lin. Rouge: A package for automatic evaluation of summaries. Workshop on Text Summarization Branches Out, Post-Conference Workshop of ACL 2004.

评估摘要： Rouge

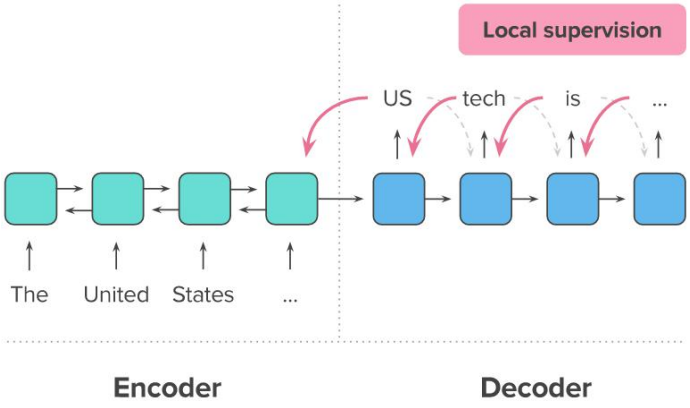
生成摘要Y ：the cat was found under the bed
参考摘要Y′ ： the cat was under the bed

#	1-gram	reference 1-gram	2-gram	reference 2-gram
1	the	the	the cat	the cat
2	cat	cat	cat was	cat was
3	was	was	was found	was under
4	found	under	found under	under the
5	under	the	under the	the bed
6	the	bed	the bed	
7	bed			
$count_{match}$	6	6	4	5

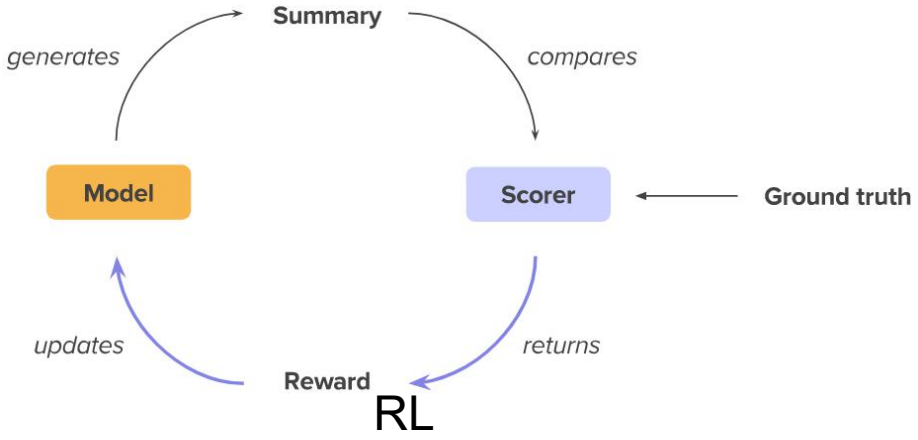
$$\text{Rouge}_1(Y, Y') = \frac{6}{6} = 1.0 \qquad \text{Rouge}_2(Y, Y') = \frac{4}{5} = 0.8$$

目标优化 -- 强化学习

- 计算ROUGE指标作为reward，根据这个reward对模型进行奖励和惩罚：
  - 如果得分较高，那么模型可以自行更新，以便将来可能出现这样的摘要；
  - 如果得分较低，则该模型将受到处罚并更改其生成过程以避免产生类似摘要
- 优点：强化学习模型非常适合提高评估整个序列的摘要分数，而不是逐个词的预测



seq2seq



RL

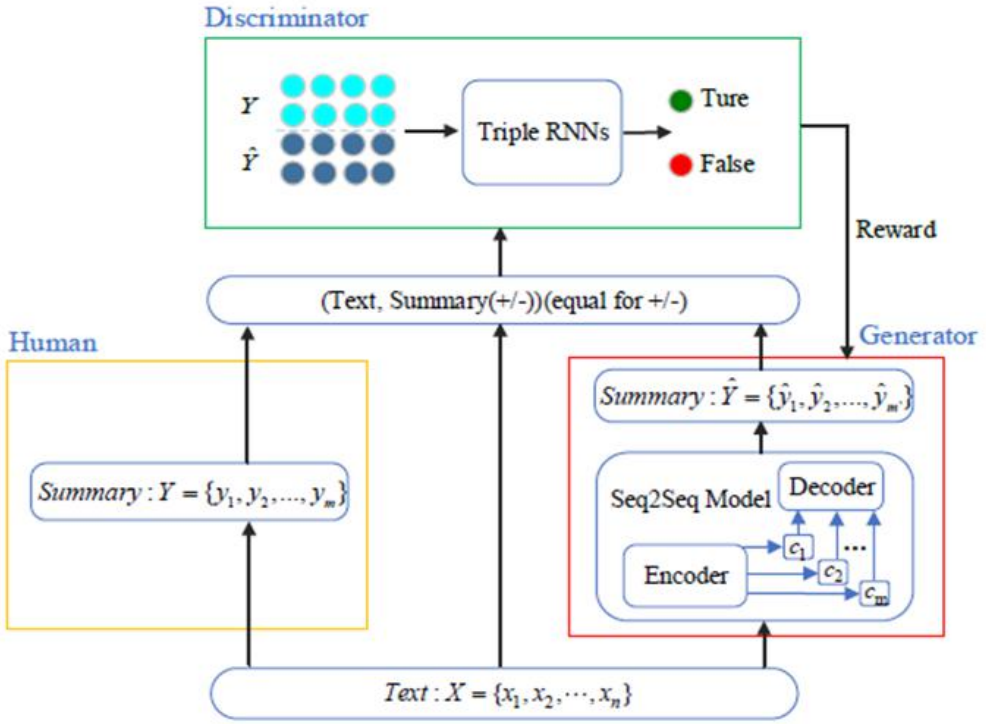
目标优化 -- 强化学习

The bottleneck is no longer access to information; now it's our ability to keep up.
AI can be trained on a variety of different types of texts and summary lengths.
A model that can generate long, coherent, and meaningful summaries remains an open research problem.

The last few decades have witnessed a fundamental change in the challenge of taking in new information. The bottleneck is no longer access to information; now it's our ability to keep up. We all have to read more and more to keep up-to-date with our jobs, the news, and social media. We've looked at how AI can improve people's work by helping with this information deluge and one potential answer is to have algorithms automatically summarize longer texts. Training a model that can generate long, coherent, and meaningful summaries remains an open research problem. In fact, generating any kind of longer text is hard for even the most advanced deep learning algorithms. In order to make summarization successful, we introduce two separate improvements: a more contextual word generation model and a new way of training summarization models via reinforcement learning (RL). The combination of the two training methods enables the system to create relevant and highly readable multi-sentence summaries of long text, such as news articles, significantly improving on previous results. Our algorithm can be trained on a variety of different types of texts and summary lengths. In this blog post, we present the main contributions of our model and an overview of the natural language challenges specific to text summarization.

目标优化 -- GAN

- 优化目标是一个极小极大博弈问题，目的是生成摘要输入给判别器时，判别器很难判断是真实还是生成摘要，以达到纳什平衡；即极大化判别器的判断能力，极小化将生成摘要判断为非人工摘要的概率



$$\min_{G}\max_{D} V(D,G) = E_{(x,y)\sim P_{data}(x,y)}[logD(x,y)] + E_{x\sim P_{data}(x),y'\sim G(\cdot|x)}[\log(1-D(x,y'))]$$

Adversarial Reinforcement Learning for Chinese Text Summarization. 2018.

基于预训练模型的文摘生成

- Encoder：使用 BERT对输入序列进行编码表示
- Decoder：包括两个阶段，先使用 Transformer 生成一个草稿，再将草稿中的每个词都做一次 mask，输入到 BERT，再通过Transformer-decoder生成最终的摘要
- 模型的损失是两个阶段损失之和

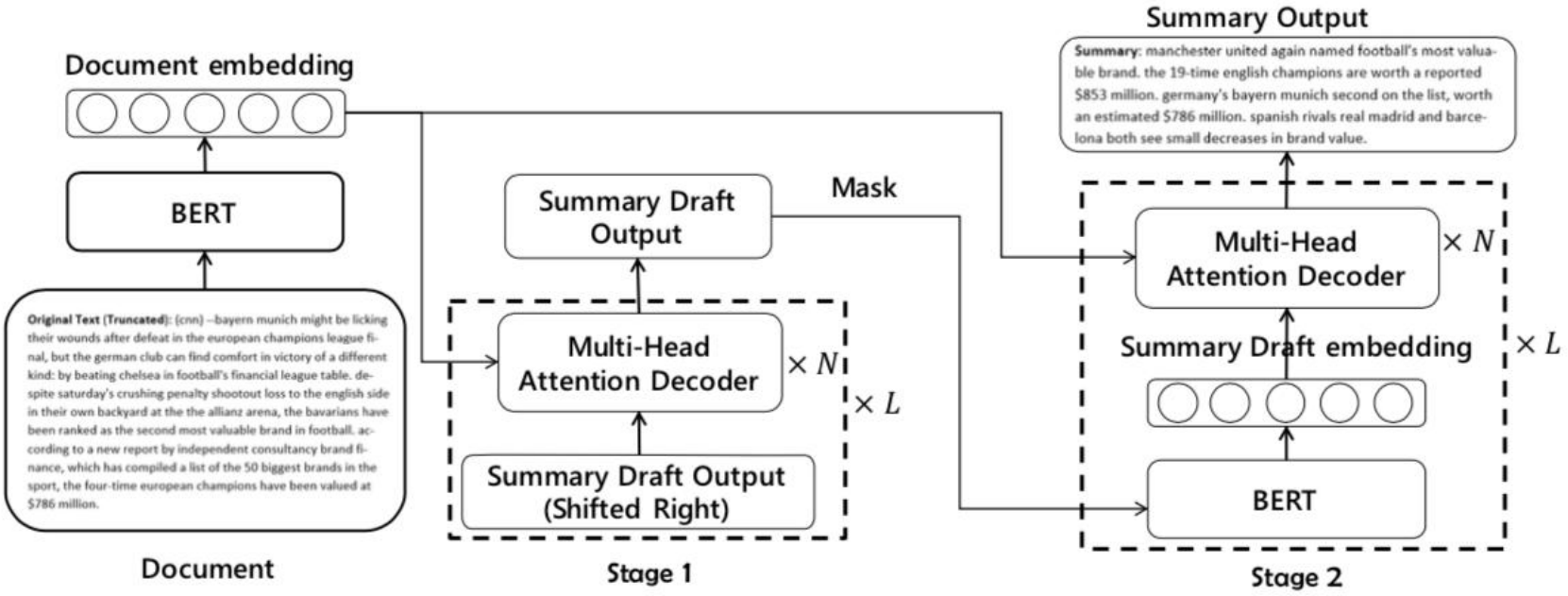


Figure 1: Model Overview, N represents decoder layer number and L represents summary length.

参考文献

- 概述
  - Mani, Inderjeet, et al. "SUMMAC: a text summarization evaluation." Natural Language Engineering 8.1 (2002): 43-68.
- 抽取式摘要
  - Hans Peter Luhn. The automatic creation of literature abstracts. IBM Journal of research and development, 2(2):159–165, 1958.
  - Rada Mihalcea and Paul Tarau. Textrank: Bringing order into text. In Proceedings of the 2004 conference on empirical methods in natural language processing, 2004.
  - G. Erkan and D. R. Radev.2004. LexRank: Graph-based Centrality as Salience in Text Summarization. Journal of Artificial Intelligence Research (JAIR).
  - Brin, Sergey, and Lawrence Page. "The anatomy of a large-scale hypertextual web search engine." Computer networks and ISDN systems 30.1-7 (1998): 107-117.
  - Neural Extractive Summarization with Hierarchical Attentive Heterogeneous Graph Network
  - Fine-tune BERT for Extractive Summarization, 2019
- 生成式摘要
  - Alexander M Rush, Sumit Chopra, and Jason Weston. A neural attention model for abstractive sentence summarization. arXiv preprint arXiv:1509.00685, 2015.
  - Sébastien Jean, Kyunghyun Cho, Roland Memisevic, and Yoshua Bengio. 2014. On using very large target vocabulary for neural machine translation. CoRR, abs/1412.2007.
  - Nallapati, Ramesh, et al. "Abstractive Text Summarization using Sequence-to-sequence RNNs and Beyond." CoNLL 2016 (2016): 280.
  - Wong, Kam-Fai, Mingli Wu, and Wenjie Li. "Extractive Summarization Using Supervised and Semi-Supervised Learning." Proceedings of the 22nd International Conference on Computational Linguistics (Coling 2008). 2008.
  - See, Abigail, Peter J. Liu, and Christopher D. Manning. "Get To The Point: Summarization with Pointer-Generator Networks." Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers). Vol. 1. 2017.

参考文献

- 生成式摘要
  - Sankaran, Baskaran, et al. "Temporal attention model for neural machine translation." arXiv preprint arXiv:1608.02927 (2016).
  - Paulus, Romain, Caiming Xiong, and Richard Socher. "A deep reinforced model for abstractive summarization." arXiv preprint arXiv:1705.04304 (2017).
  - Xu, Hao, et al. "Adversarial Reinforcement Learning for Chinese Text Summarization." International Conference on Computational Science. Springer, Cham, 2018.
  - Pretraining-Based Natural Language Generation for Text Summarization, 2019
- 摘要评估
  - Chin-Yew Lin and Eduard Hovy. Automatic Evaluation of Summaries Using N-gram Co-Occurrence Statistics. In Proceedings of the Human Technology Conference 2003 (HLT-NAACL-2003).
  - Chin-Yew Lin. Rouge: A package for automatic evaluation of summaries. Workshop on Text Summarization Branches Out, Post-Conference Workshop of ACL 2004.
  - Kishore Papineni, Salim Roukos, Todd Ward, and Wei-Jing Zhu. BLEU: a Method for Automatic Evaluation of Machine Translation.
- 更多相关论文
  - Shibhansh Dohare, Harish Karnick. Text Summarization using Abstract Meaning Representation. 2017.
  - Qingyu Zhou, Nan Yang, Furu Wei and Ming Zhou. Selective Encoding for Abstractive Sentence Summarization. ACL, 2017
  - Maxime Peyrard and Judith Eckle-Kohler. Supervised Learning of Automatic Pyramid for Optimization-Based Multi-Document Summarization. ACL, 2017.
  - Jin-ge Yao, Xiaojun Wan and Jianguo Xiao. Recent Advances in Document Summarization. KAIS, survey paper, 2017.
  - Pranay Mathur, Aman Gill and Aayush Yadav. Text Summarization in Python: Extractive vs. Abstractive techniques revisited. 2017.

欢迎加入DL4NLP！

中国科学院信息工程研究所
INSTITUTE OF INFORMATION ENGINEERING,CAS