# 文本摘要

2020.01.05 李雨承

# 目录

- 数据集：CNN/DailyMail, XSum
- Metrics: ROUGE, 人工评价
- 老方法
  - TF-IDF，LSA，LDA
- 新方法
  - 抽取式的文本摘要
  - 生成式(abstractive)的文本摘要

# 资源

- 数据集
  - CNN\DailyMail: https://github.com/abisee/cnn-dailymail
  - XSum:https://github.com/EdinburghNLP/XSum/tree/master/XSum-Dataset
  - NLPCC2017 Task3: http://tcci.ccf.org.cn/conference/2017/taskdata.php
- 博文
  - LSA： https://blog.csdn.net/u012328476/article/details/105314828
  - LDA: https://dzone.com/articles/lda-for-text-summarization-and-topic-detection

# 论文列表

- 论文
  - 抽取式
    - HIBERT: Document Level Pre-training of Hierarchical Bidirectional Transformers for Document Summarization
    - BottleSum/ Unsupervised and Self-supervised Sentence Summarization using the Information Bottleneck Principle
  - 生成式
    - Get To The Point: Summarization with Pointer-Generator Networks
    - Knowledge Graph-Augmented Abstractive Summarization with Semantic-Driven Cloze Reward
    - text summarization with pre-trained encoders
    - A Neural Attention Model for Abstractive Sentence Summarization

数据

# CNN/DailyMail

https://huggingface.co/datasets/cnn_dailymail

- CNN的文章作为Context，所有高亮句拼接后作为summary。
- Train：287113；Val：13368；Test：11490
- 平均长度：
  - 文章：781
  - 摘要：56
- 例子：https://edition.cnn.com/2021/01/04/tech/jack-ma-alibaba-ant-group-intl-hnk/index.html

# XSum

https://github.com/EdinburghNLP/XSum/tree/master/XSum-Dataset

- Extreme Summarization （XSum）。一句话概括整篇文章。
- 收集与BBC，文章和摘要都由编辑手写，质量高。
- Train：589284；Val：32736；Test：32739
- 平均长度：
  - 文章：430
  - 摘要：19
- paper：https://arxiv.org/pdf/1808.08745.pdf
- 例子：https://www.bbc.com/news/business-54838713

# NLPCC2017 Task3

- 平均长度
  - 文章：990
  - 摘要：44
- 总样本数量：50000
- 来源于报纸，新闻网页等

# 指标

# 如何评价摘要的好坏

- 流畅度&语法正确性
- 是否摘要出了原文中最重要的部分?
- 摘要文本事实是否正确?

# ROUGE

https://blog.csdn.net/qq_25222361/article/details/78694617

- ROUGE-n
  - ROUGE-1，2，3
- ROUGE-L：最长公共子串
- ROUGE-S：支持在bi-grams, or tri-grams中间插入token (一般来说至多两个)

- **ROUGE-*n*:** This metric is recall-based measure and based on comparison of *n*-grams. a series of *n*-grams (mostly two and three and rarely four) is elicited from the reference summaries and the candidate summary (automatically generated summary). Let $p$ be "the number of common *n*-grams between candidate and reference summary", and $q$ be "the number of *n*-grams extracted from the reference summary only". The score is computed as:

$$\text{ROUGE-}n = \frac{p}{q} \tag{10}$$

$$R_{lcs} = \frac{LCS(X,Y)}{m} \tag{2}$$

$$P_{lcs} = \frac{LCS(X,Y)}{n} \tag{3}$$

$$F_{lcs} = \frac{(1+\beta^2)R_{lcs}P_{lcs}}{R_{lcs}+\beta^2 P_{lcs}} \tag{4}$$

http://blog.csdn.net/qq_25222361

# 人工评测

- 设定一些指标，标注者打分：
  - 摘要覆盖了多少的原文信息
  - 是否冗余
  - 是否可读&&语法正确
- 给出原文，标注者为一系列摘要模型所生成的摘要排序

# 以前的方法

# pipeline

- 早期方法多是抽取式摘要：即在原文中寻找关键信息。
- 流程
  - 选取特征
  - 通过特征为每个句子打分
  - 选top-k作为摘要信息

# LDA

- Daume et al., BayeSum.
- 用LDA提取N个topic，和其对应的主题词。
- 为每一个topic在原文中选一句作为该topic的代表（根据主题词）
- 将所有代表句拼接作为摘要。

# LSA

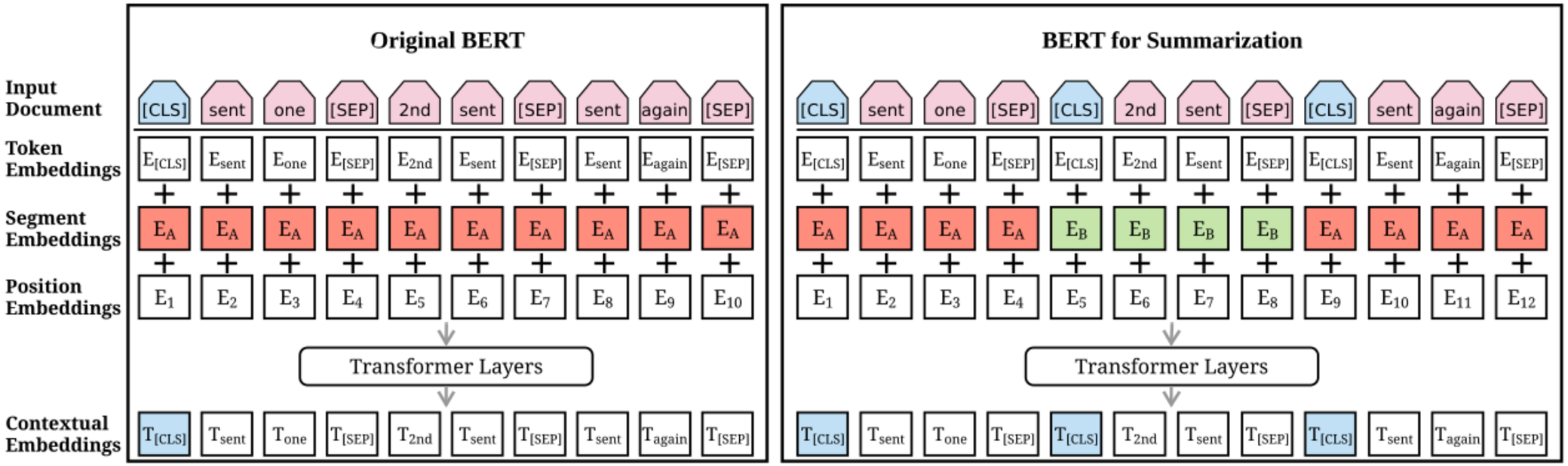
- 用$A_{m\times n}$表示一个文档，m为词表大小，n为句子的数量
- 奇异值分解：$A = U\Sigma V^T$，其中$V_{n\times n}$
- 则$V$中的列向量为句子的表示，且重要性从高到低排序。

# 神经网络方法

# BERTSUM

https://www.aclweb.org/anthology/D19-1387.pdf

- BERTSUM
  - Segment Embedding
  - 和BERT相比，通过增加额外CLS的方式取得多句的句向量
  - 使用句向量判断该句是否应该包含在摘要中。

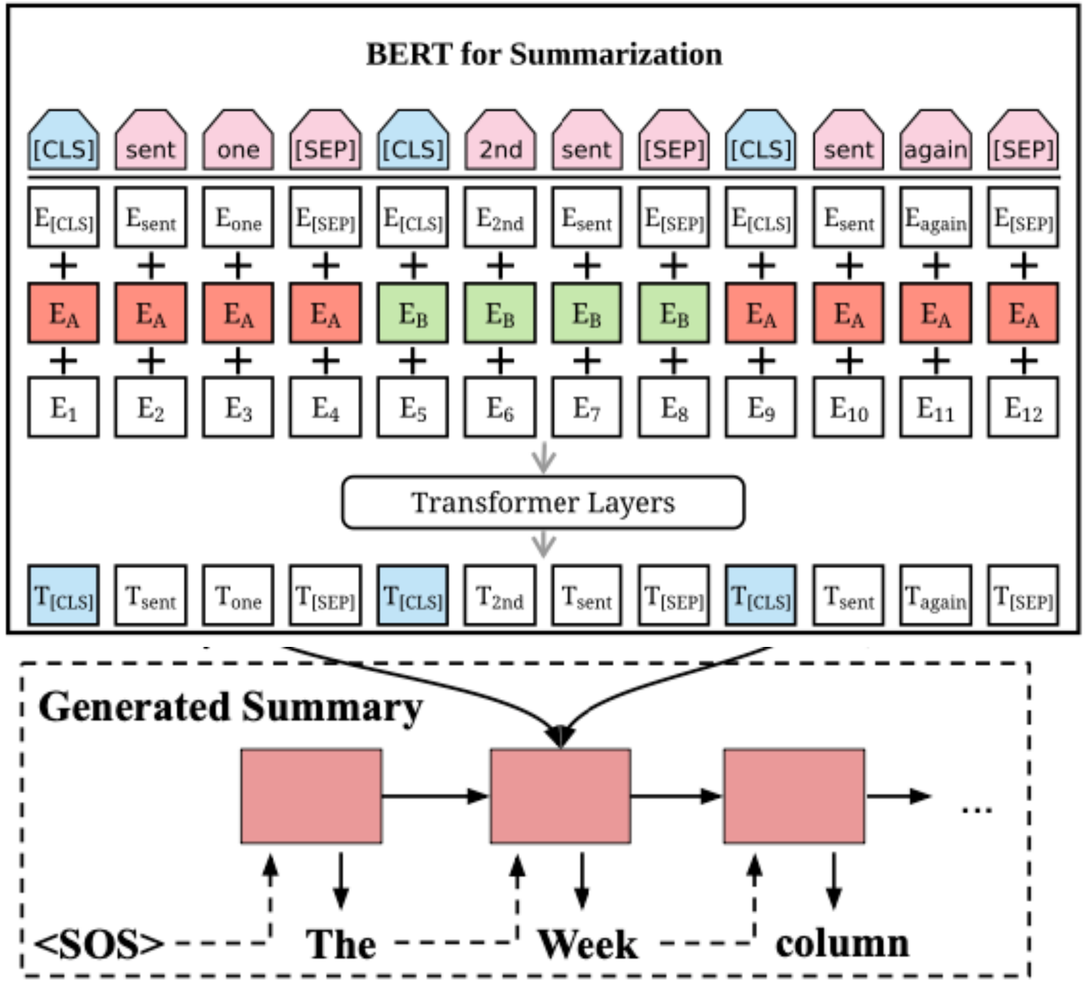


# Extractive Summarization

- Sentence classification task
  - 判断每句话是否应该包含在摘要中，最后按照probability排序，取top-k
- 问题
  - 标签是生成的，有噪音，有冗余。
    - 脚本以ROUGE值是否大于0.6为准，判断每一句是否应该被包含在摘要中。
- 优点：基本可以保证事实正确

# BERTSUM

- Abstractive Summarization
  - Seq2seq task

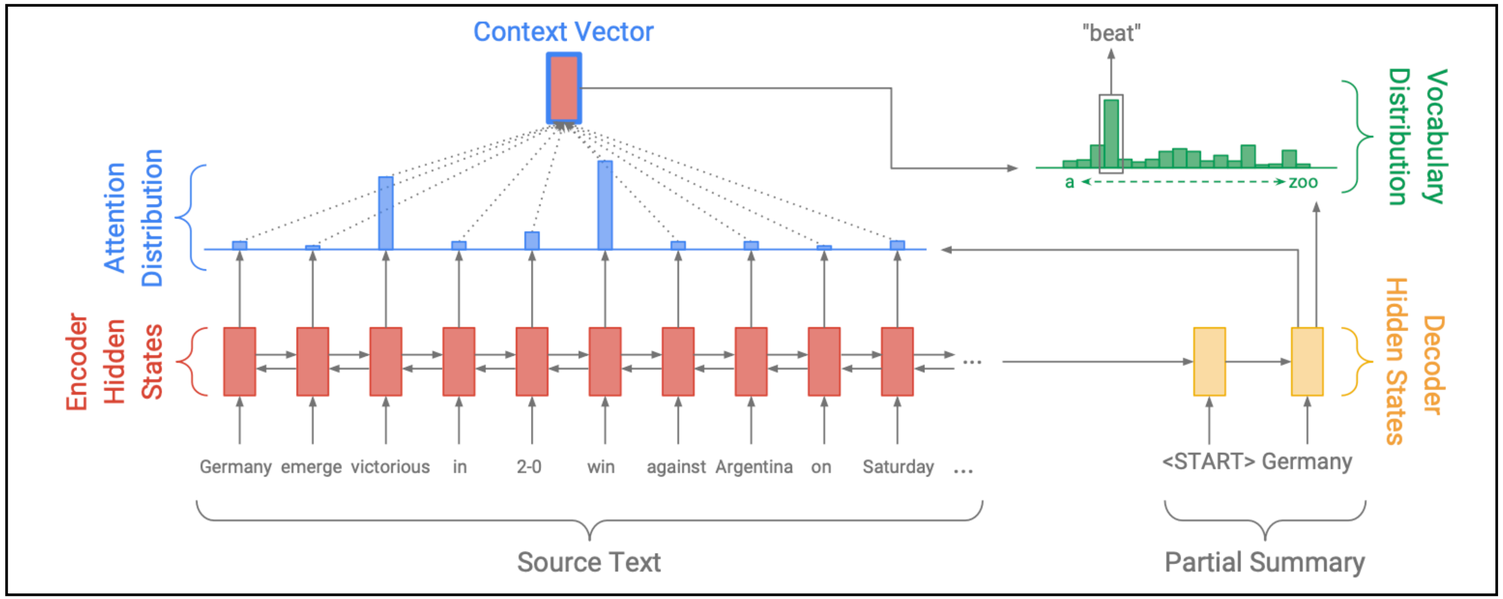


# Abstractive Summarization

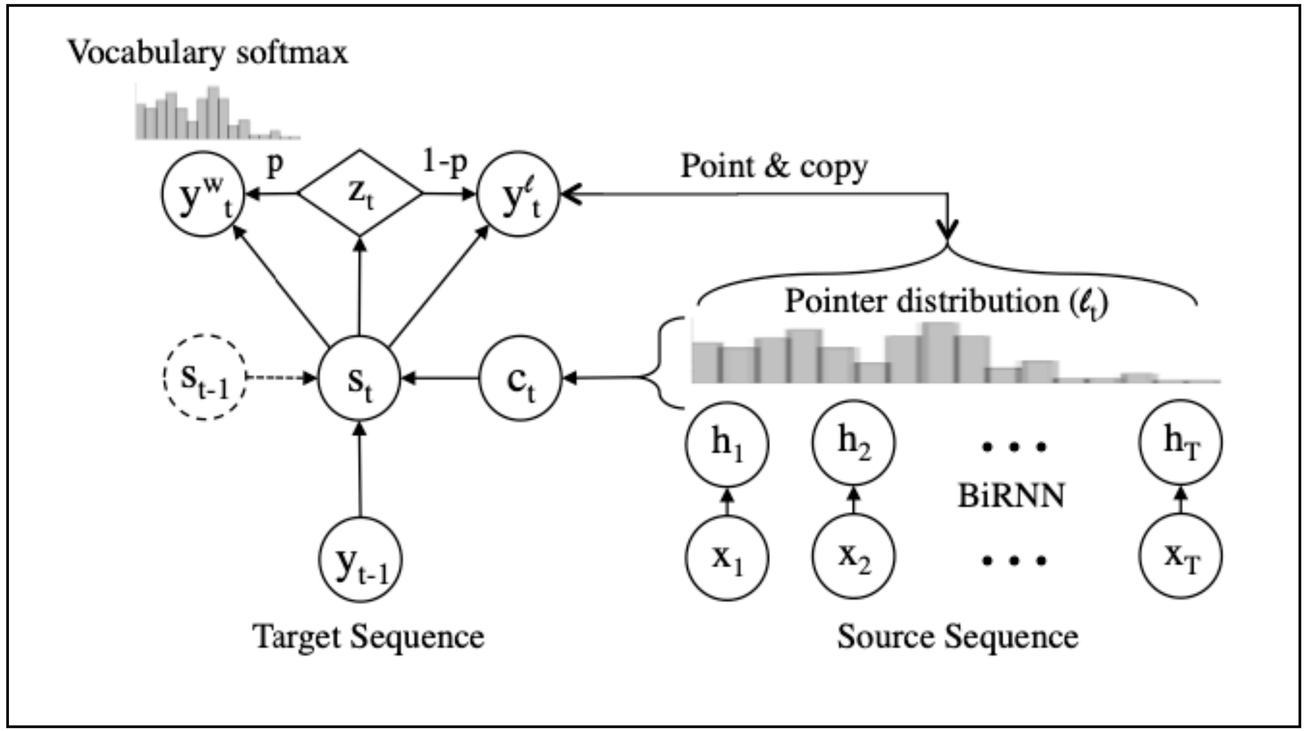
- Text generation task
- 优点
  - 更符合人类的习惯
  - 更少的限制
- 问题：无法保证事实正确

# PointerNet & CopyMechanism

- Motivation
  - 从原文中copy内容到摘要中，让内容更可信
  - 消灭OOV



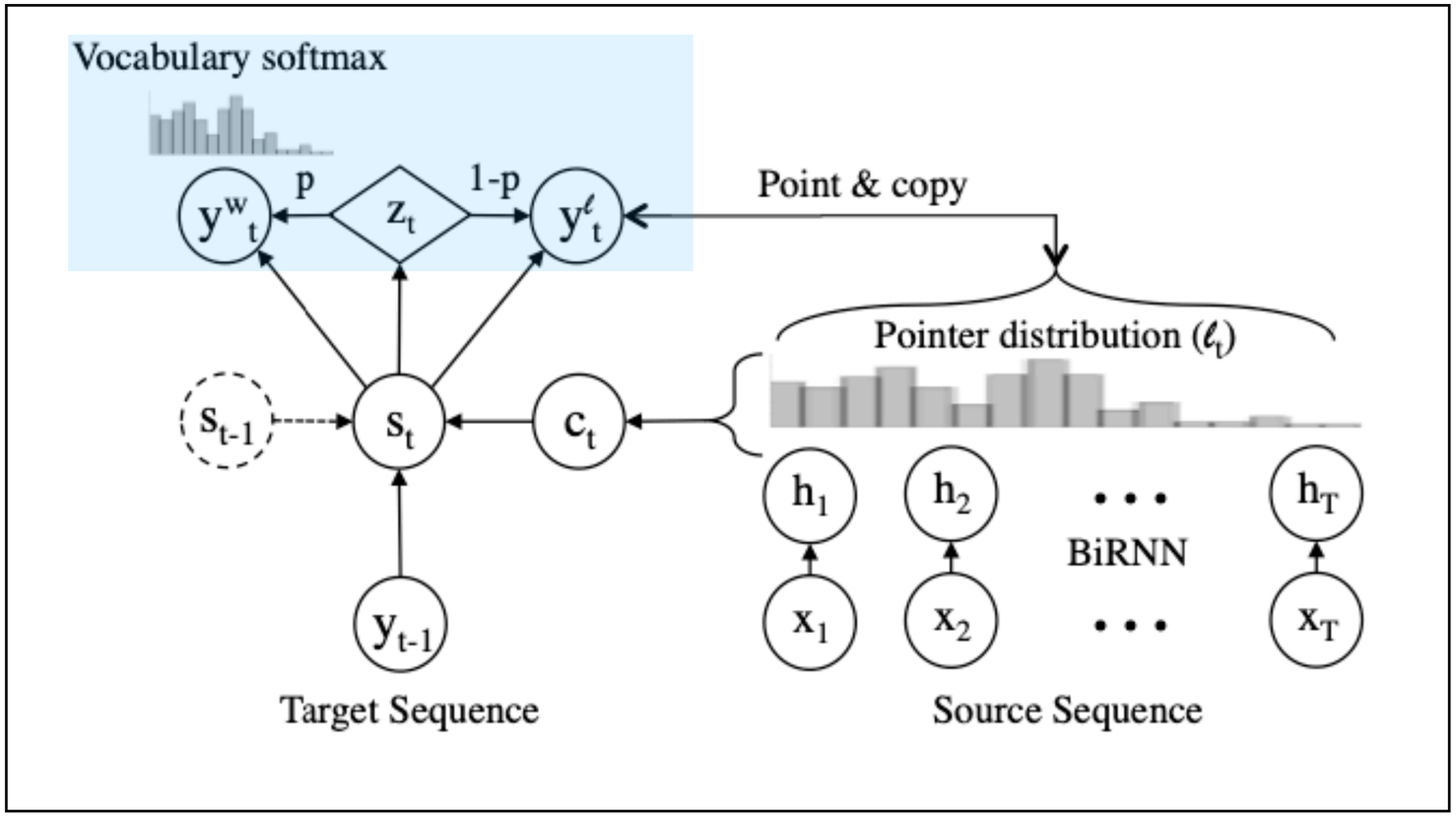

Original Seq2seq

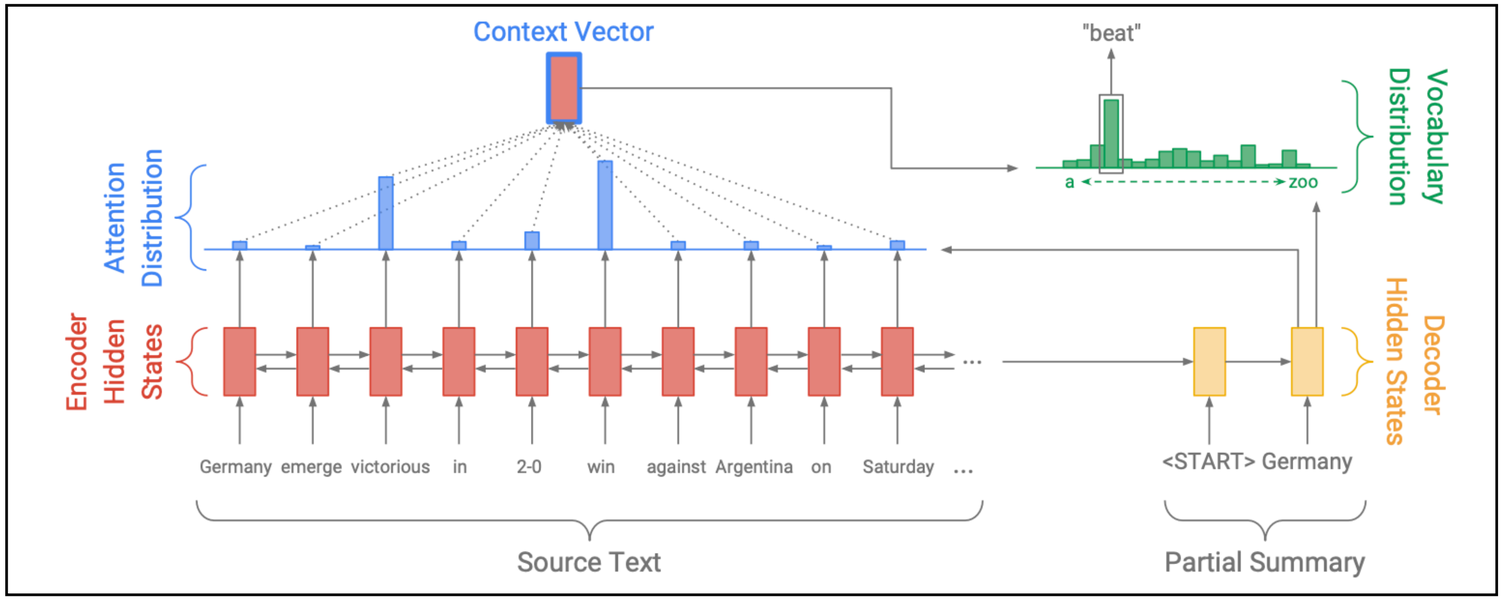



Pointer Network

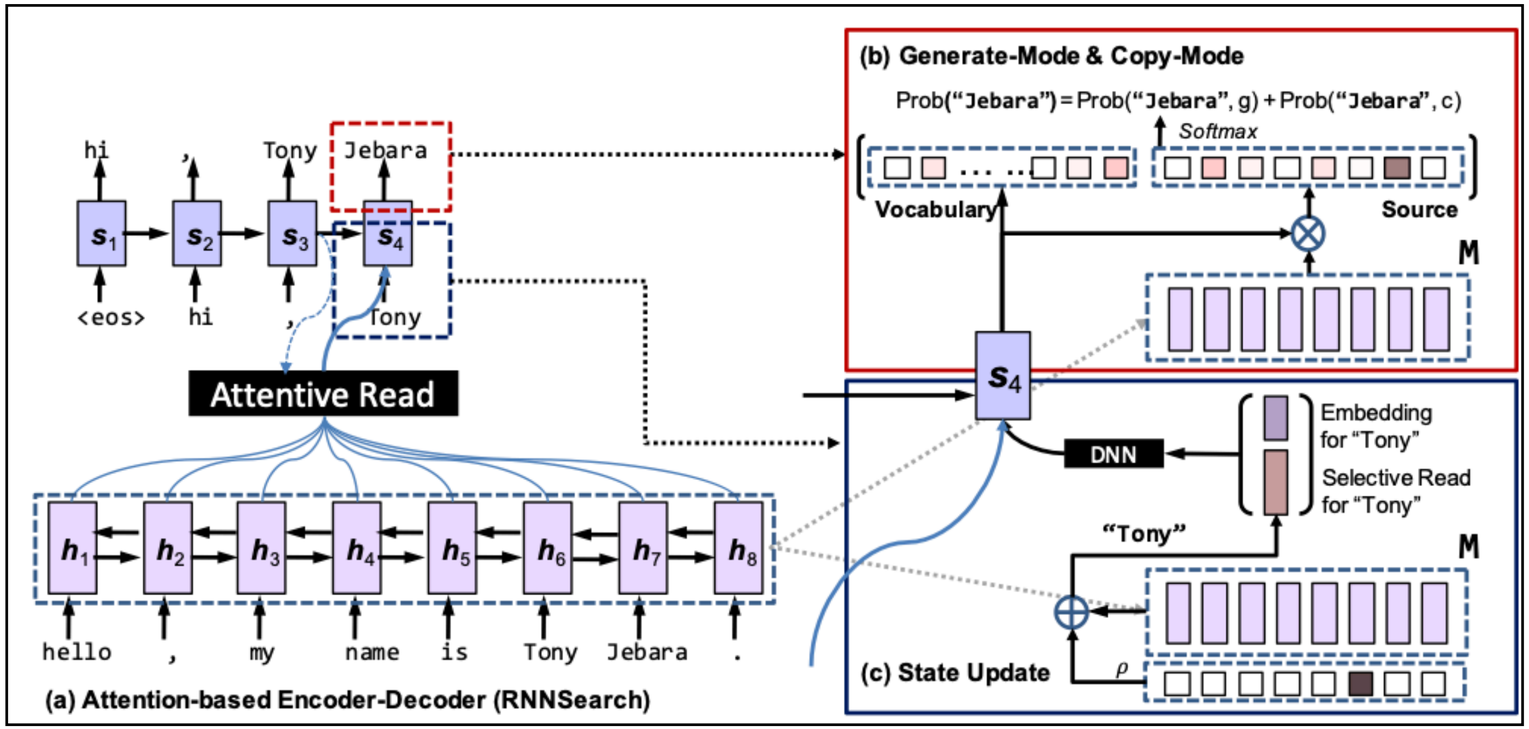
# Pointer Notwork

- $p(y \mid y_{t-1}, s_{t-1}, c)$取决于三部分
  - 门控$z$：判断采取vocabulary还是采取point&copy
  - $y_t^w$：vocabulary softmax
  - $y_t^l$：pointer softmax





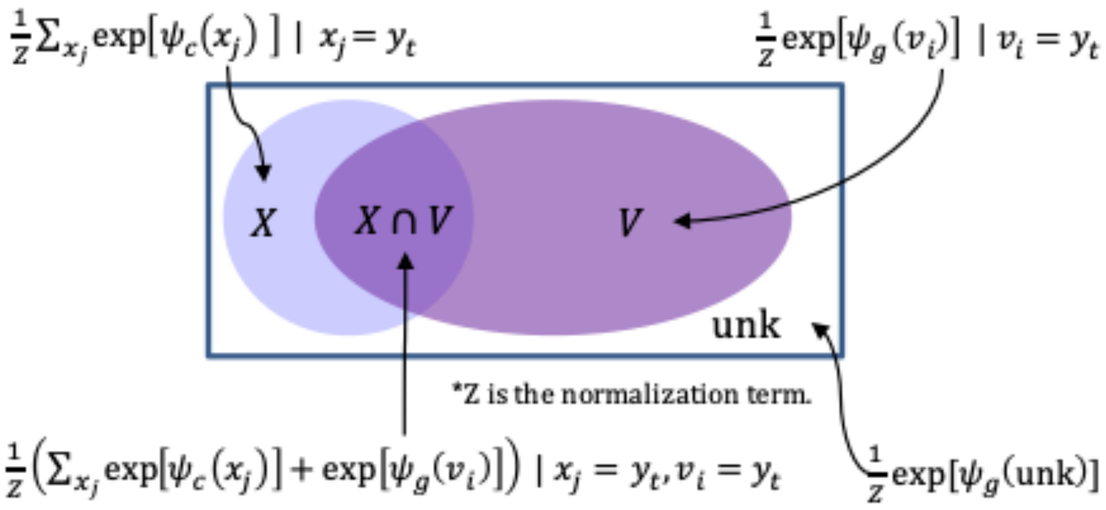

Original Seq2seq




CopyNet

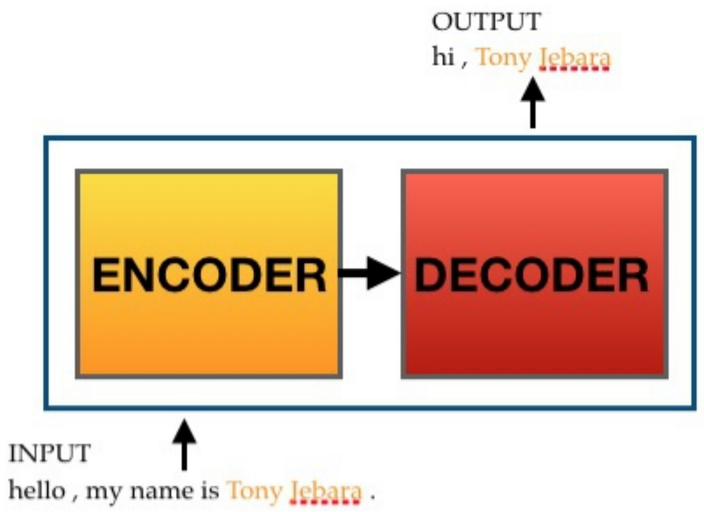
# CopyMechanism

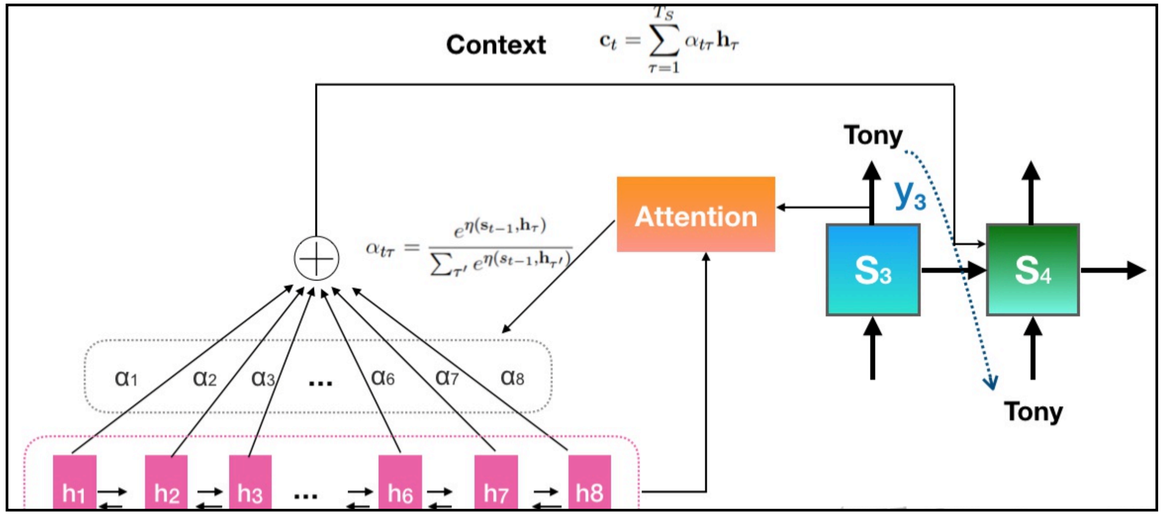
- 四种情况
  - $y_t \in X$
  - $y_t \in V$
  - $y_t \in X \cap V$
  - $y_t$ is unkown
- 例如：Tony Jebara
  - $p(jebara) = \frac{1}{z} \sum \text{softmax}(\sigma(hW)s_t)$



# State Updates

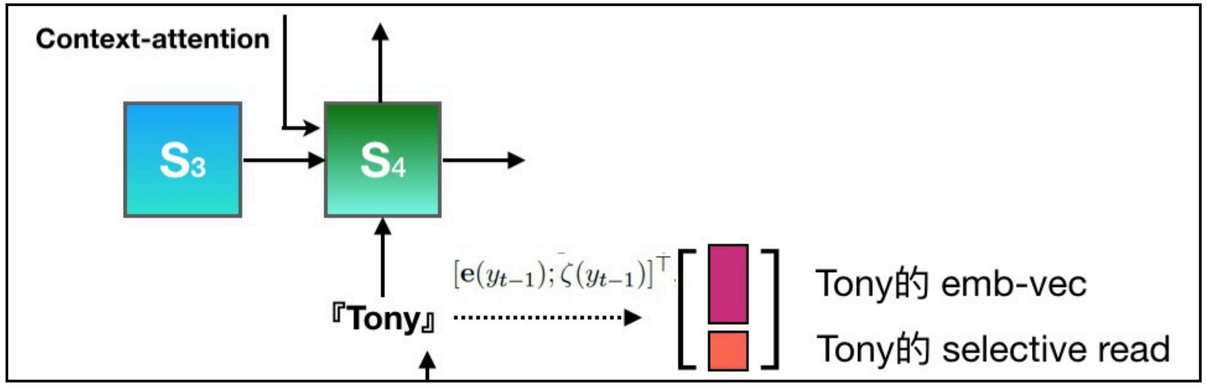
- Decoder(y)=$f(y_{t-1}, s_{t-1}, c_t)$, where $c_t = \sum ah$
- Change the representation of $y_{t-1}$
  - $y_{t-1} = emb(y_{t-1}) + sr(y_{t-1})$
  - $sr(y_{t-1}) = \sum \phi h$
  - $\phi$是可训练参数，仅当x中出现$y_{t-1}$时不为0



# State Updates

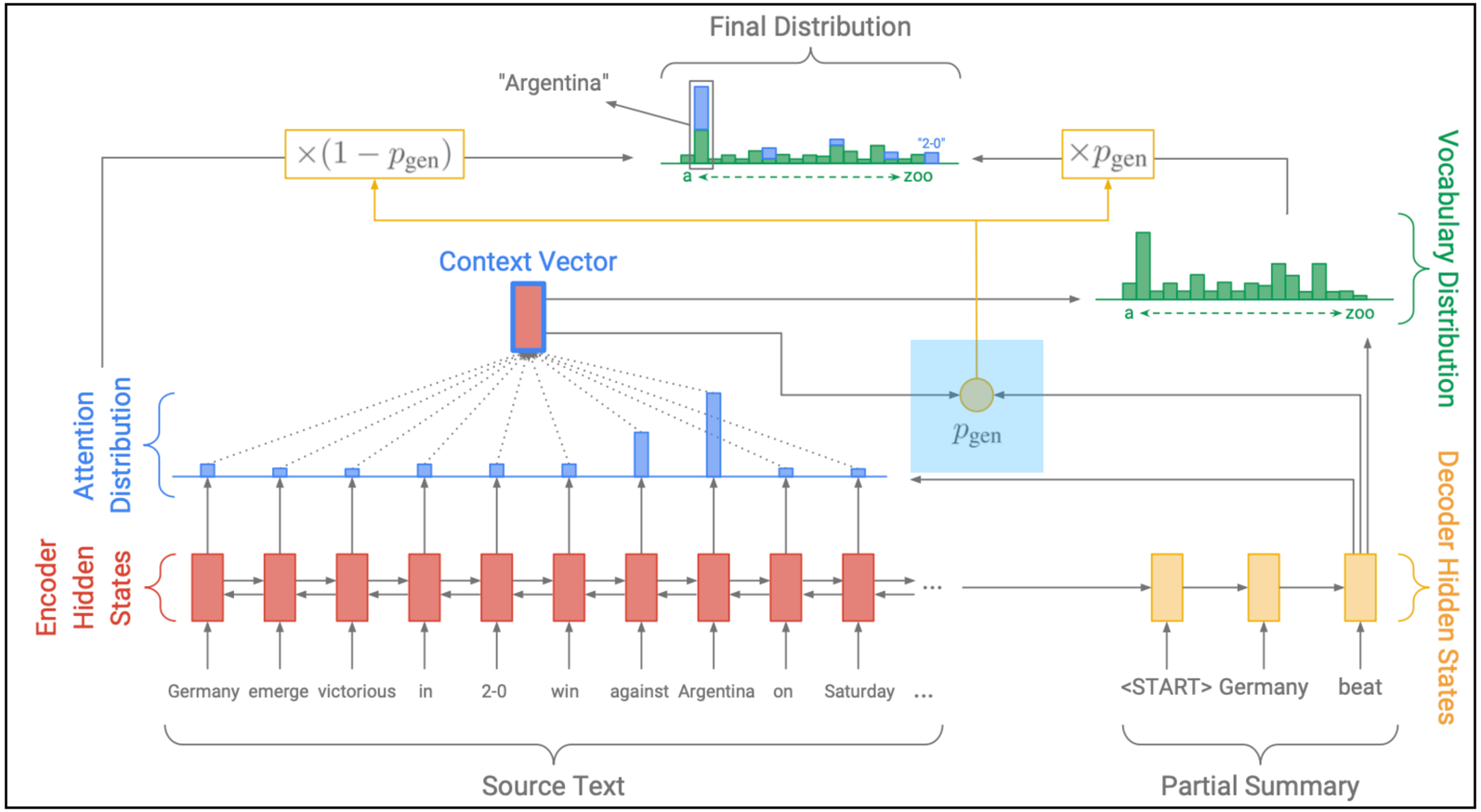
- Decoder(y)=$f(y_{t-1}, s_{t-1}, c_t)$, where $c_t = \sum ah$
- Change the representation of $y_{t-1}$
  - $y_{t-1} = emb(y_{t-1}) + sr(y_{t-1})$
  - $sr(y_{t-1}) = \sum \phi h$
  - $\phi$是可训练参数，仅当x中出现$y_{t-1}$时不为0



# POINTER-COVERAGE

- Motivation
  - 还是从原文中copy关键词到摘要
  - 额外引入了coverage mechanism：惩罚冗余的生成

# Pointer-Generator




- 和CopyMechanism唯一的区别是$p_{gen}$

# Coverage Mechanism

- 在生成的过程中，记录decoder曾经关注过哪些token。
  - 维护一个向量 $c^t = \sum_{t-1} a^i$，其中$a$是每个生成步的attention distribution

1. 增加惩罚项：$covloss = \sum \min(a^t, c^t)$
2. 在生成该步的attention distribution时，加入向量$c$

$$e_i^t = v^T \tanh(W_h h_i + W_s s_t + w_c c_i^t + b_{\text{attn}}) \quad (11)$$

**Article (truncated):** andy murray came close to giving himself some extra preparation time for his wedding next week before ensuring that he still has unfinished tennis business to attend to . the world no 4 is into the semi-finals of the miami open , but not before getting a scare from 21 year-old austrian dominic *thiem* , who pushed him to 4-4 in the second set before going down 3-6 6-4 , 6-1 in an hour and three quarters . murray was awaiting the winner from the last eight match between tomas berdych and argentina 's juan monaco . prior to this tournament *thiem* lost in the second round of a challenger event to soon-to-be new brit *aljaz* bedene . andy murray pumps his first after defeating dominic *thiem* to reach the miami open semi finals . *muray* throws his *sweatband* into the crowd after completing a 3-6 , 6-4 , 6-1 victory in florida . murray shakes hands with *thiem* who he described as a ' strong guy ' after the game . and murray has a fairly simple message for any of his fellow british tennis players who might be agitated about his imminent arrival into the home ranks : do n't complain . instead the british no 1 believes his colleagues should use the assimilation of the world number 83 , originally from slovenia , as motivation to better themselves .

---

**Pointer-Generator, No Coverage:**
andy murray came close to giving himself some extra preparation for his wedding next week before getting a scare from 21 year-old austrian dominic *thiem* .
the world no 4 is into the semi-finals of the miami open , but not before getting a scare from 21 year-old austrian dominic *thiem* .

---

**Pointer-Generator, With Coverage:**
andy murray defeated dominic *thiem* 3-6 6-4 , 6-1 in an hour and three quarters .
murray was awaiting the winner from the last eight match between tomas berdych and argentina 's juan monaco .
prior to this tournament thiem lost in the second round of a challenger event to soon-to-be new brit *aljaz* bedene .

Coverage embedding $c$

$p_{gen}$

# 一些比较

- PTGen: CopyMechanism abstractive summarization
- PTGen+Cov: Copy+Coverage
- TransformerAbs: transformer based seq2seq
- BertSumExtAbs: Bert based abstractive summarization (after pre-trained on extractive task)

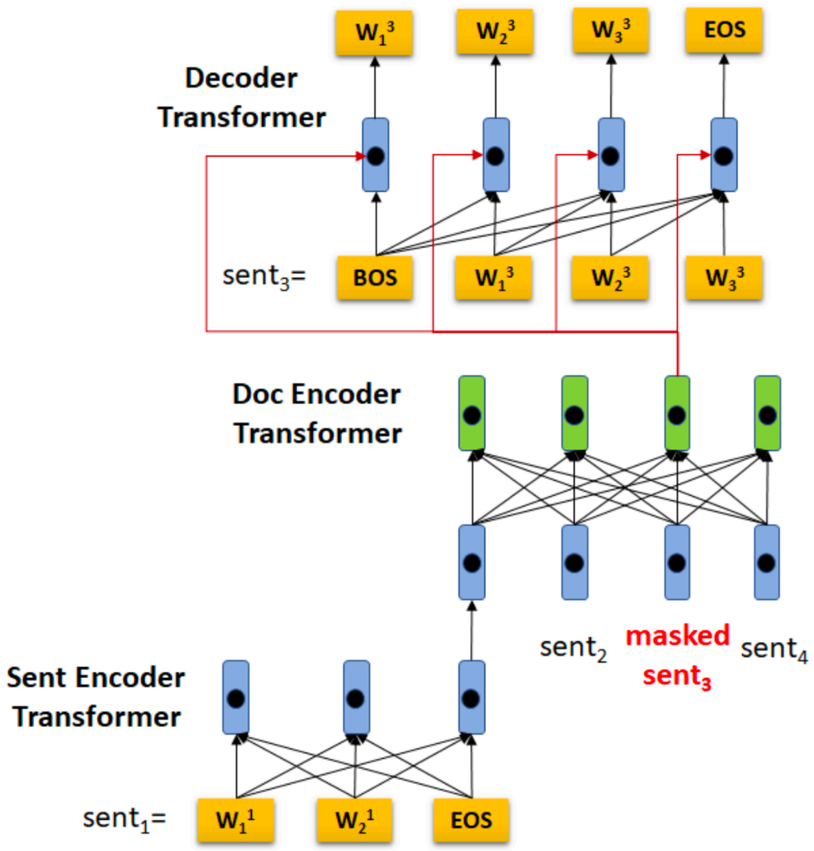
| | | | |
|---|---|---|---|
| **Abstractive** | | | |
| PTGEN (See et al., 2017) | 36.44 | 15.66 | 33.42 |
| PTGEN+COV (See et al., 2017) | 39.53 | 17.28 | 36.38 |
| DRM (Paulus et al., 2018) | 39.87 | 15.82 | 36.90 |
| BOTTOMUP (Gehrmann et al., 2018) | 41.22 | 18.68 | 38.34 |
| DCA (Celikyilmaz et al., 2018) | 41.69 | 19.47 | 37.92 |
| TransformerABS | 40.21 | 17.76 | 37.09 |
| **BERT-based** | | | |
| BERTSUMEXT | 43.25 | 20.24 | 39.63 |
| BERTSUMEXT w/o interval embeddings | 43.20 | 20.22 | 39.59 |
| BERTSUMEXT (large) | 43.85 | 20.34 | 39.90 |
| BERTSUMABS | 41.72 | 19.39 | 38.76 |
| BERTSUMEXTABS | 42.13 | 19.60 | 39.18 |

# Extractive Summarization

- Sentence classification task
- 问题
  - 标签是生成的，有噪音，有冗余。
    - 脚本以ROUGE值是否大于0.6为准，判断每一句是否应该被包含在摘要中。
- 优点：基本可以保证事实正确

# HiBERT

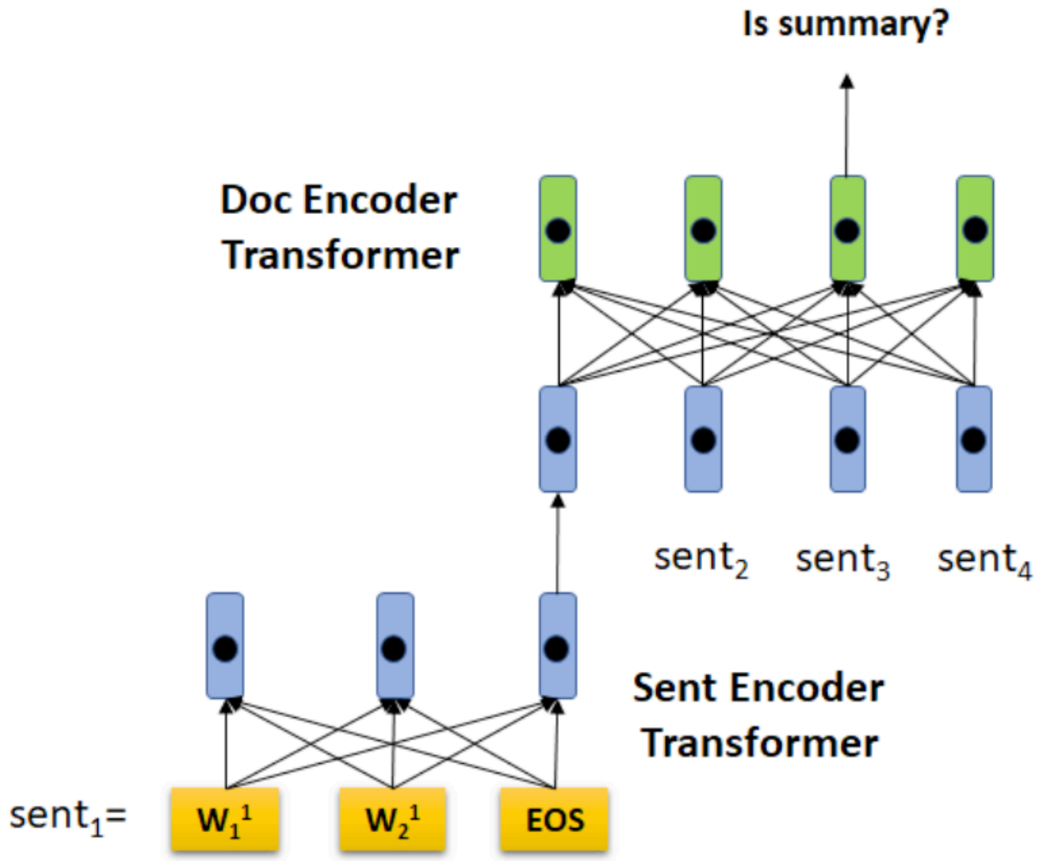
- 层级式的预训练Transformer Encoder
  - word-level，sentence-level，doc-level
- Pre-training objective:
  - 随机mask一些句子，并预测mask的句子（预测句子的每一个词）



# HiBERT

- Multi-pre-training stages
  - 首先在大规模预训练预料上的预训练
  - in-domain的预训练
  - 监督训练

| Models | R-1 | R-2 | R-L |
|---|---|---|---|
| Lead | 41.80 | 22.60 | 35.00 |
| EXTRACTION | 44.30 | 25.50 | 37.10 |
| JECS | 45.50 | 25.30 | 38.20 |
| HeriTransformer | 47.44 | 28.08 | 39.56 |
| BERT | 48.38 | 29.04 | 40.53 |
| HIBERT$_S$ (in-domain) | 48.92 | 29.58 | 41.10 |
| HIBERT$_M$ (in-domain) | 49.06 | 29.70 | 41.23 |
| HIBERT$_S$ | **49.25** | **29.92** | **41.43** |
| HIBERT$_M$ | **49.47** | **30.11** | **41.63** |



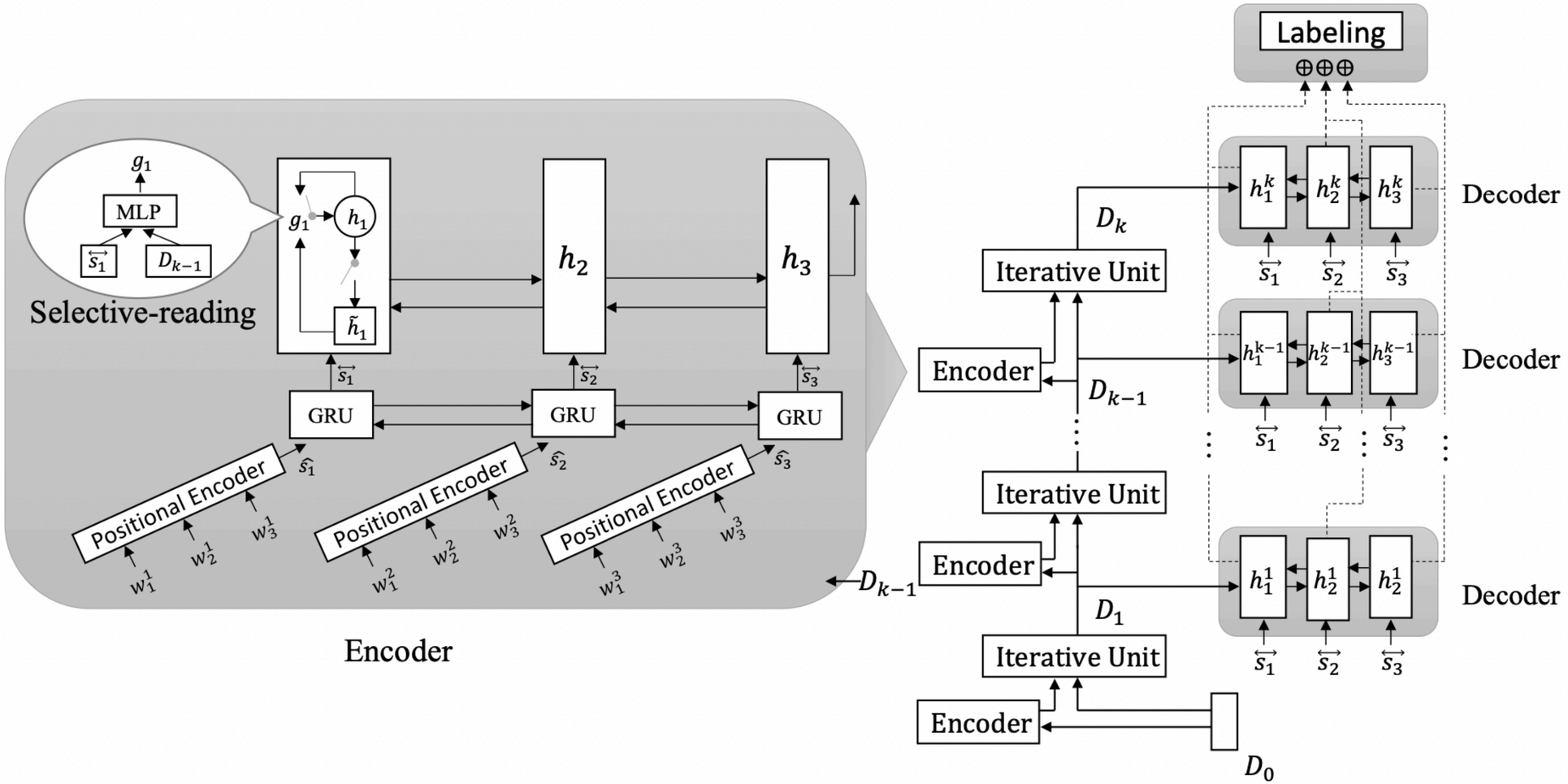
# Extractive Summarization

- 除了sentence classification外，还可以使用打分函数
  - 为每一个句子打分，选取top-k的句子，拼接起来作为摘要。
- 除了单轮抽取句子外，还可以迭代式地抽取句子作为摘要
  - 人类需要读文章多次，才能理解文章。而现有的抽取式摘要方法只通过一遍处理就生成了句向量，并据此判断是否选择该句作为摘要，是不科学的。

# Iterative Document Representation Learning Towards Summarization with Polishing

**Xiuying Chen[1], Shen Gao[1], Chongyang Tao[1], Yan Song[2], Dongyan Zhao[1] and Rui Yan[1*]**
[1]Institute of Computer Science and Technology, Peking University, Beijing, China
[2]Tencent AI Lab
{xy-chen,shengao,chongyangtao,zhaody,ruiyan}@pku.edu.cn
clksong@tencent.com

- 提出了iterative mechanism和selective reading moudle
  - 模仿人类的多次阅读，和选择性阅读




$$\boldsymbol{D_0} = \tanh(\boldsymbol{W}\frac{1}{n_s}\sum_{i=1}^{n_s}[\overrightarrow{\boldsymbol{s_i}};\overleftarrow{\boldsymbol{s_i}}] + \boldsymbol{b}) \qquad (9)$$

Document Encoder

# Neural Document Summarization by Jointly Learning to Score and Select Sentences

**Qingyu Zhou[†,*], Nan Yang[‡], Furu Wei[‡], Shaohan Huang[‡], Ming Zhou[‡], Tiejun Zhao[†]**
[†]Harbin Institute of Technology, Harbin, China
[‡]Microsoft Research, Beijing, China
{qyzhou,tjzhao}@hit.edu.cn
{nanya,fuwei,shaohanh,mingzhou}@microsoft.com

- 该模型将选择策略集成到打分模型中，解决了此前抽取式文本摘要中句子打分和句子选择这两部分割裂的问题，端到端且不再需要人为干预。
- 打分方程