|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| 文章标题 | 期刊/时间 | 类型 | 方法 | 数据集 |  |
| **《A Discourse-Aware Attention Model for Abstractive Summ arization of Long Documents》** | NAACL/2018 | 长文本  英文 | 生成式 | Arxiv、Pubmed | 1.将长文档分section（每个section由多句组成），每个section分别输入到RNN中，得到每个section的表示Hj（这个RNN是权重共享的）。  Section中的每个词也会得到相应的隐状态H(j,i)(第j个Section中的第i个词)  在算注意力时考虑到了Section对每个词的影响。（其实就是利用Hj再对每个词进行一次加权）  解决OOV和重复问题时，使用了和指针-生成器网络中相同的做法。 |
| 《Fast Abstractive Summarization with Reinforce-Selected Sentence Rewriting》 | ACL/2018 | 短文本  英文 | 两段式/强化学习  抽取：CNN+LSTM  生成：指针-生成网络  强化学习连接两阶段 | CNN/DM | **被引较多**  两段式，先抽取，再生成。使用强化学习连接两阶段，构成一个完整的端到端模型。  抽取阶段：原文与参考摘要间没有句子对的标记（就是抽取阶段没有标签，没有告诉哪句该抽哪句不该抽）。解决方案：利用Rouge分数构建抽取阶段的标签，给原文打标签0，1，训练网络，实质是一个二分类问题。  生成阶段，利用句子对标签训练生成器（单句摘要）。 |
| **《Summary Level Training of Sentence Rewriting for Abstractive Summarization**  **》** | EMNLP/2019 | 短文本  英文 | 两段式/强化学习 | CNN/Daily Mail、NYT | 这篇文章是对上一行文章中做法的改进。   1. 改进了抽取部分，用了Bert 2. 直接使用摘要级的Rouge分数作为强化学习的目标，而不是像上文中使用单独每个句子的Rouge分数。   全局优化句子选择 |
| **《Neural document summarization by jointly learning to score and select sentences》** | ACL/2018 | 短文本  英文 | 抽取式  模型：NEUSUM  使用两部分GRU  一部分编码，句子，文档  另一部分选择句子 | CNN/Daily Mail | 端到端，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  具体做法：当前权重最大的句子的隐层状态作为decode下一个时间步的输入，这样相当于同时拿文章信息和之前已经抽取句子的信息再次对其他句子进行attention操作，这样让抽取句子的信息起到一个抑制作用 |
| 《Neural Extractive Text Summarization with Syntactic Compression》 | EMNLP/2019 | 短文本  英文 | 两段式  抽取/压缩 | CNN/DM、NYT | 两阶段间无联系  抽取阶段就是常规的抽取，  Encoder:CNN+LSTM  Decoder:LSTM  压缩阶段：基于一定的语法规则，选择句子中的一部分（比如形容词，副词），打标签01，并且结合Rouge分数。（规则要删这个词，但去掉这个词Rouge分数下降，那么就不去掉了，个人的理解） |
| 《Fine-tune BERT for Extractive Summarization》 | EMNLP/2019 | 中文本 | 抽取式  基于Bert | CNN/DM、NYT | 第一篇使用Bert来做抽取式摘要的文章 |
| **《Text Summarization with Pretrained Encoders》** | EMNLP/2019 | 中文本英文  英文 | 两段式  基于BERT，修改Bert的输入格式来做抽取\生成摘要 | CNN/DM、NYT、XSum | **较为重要**  使用预训练的encoder，两次微调。  抽取阶段：在Bert上微调。  生成阶段：上阶段微调过的Encoder，接一个随机初始化的Transformer Decoder。  BertExt: 在BertSum上，再接两层Transformer，得到的向量表示过一个分类层。  BertABS: 在BertSum上，接一个随机初始化（未预训练的）Encoder，做生成式摘要  BertExtAbs: 将抽取阶段后的BertSum，后再接Encoder，做生成式摘要，相当于两次微调 |
| **《Extractive Summarization as Text Matching》** | ACL/2020 | 英文  短文本/长文本 | 抽取式  基于Bert | CNN/DM**（SOTA）**  PubMed、Multi-news、 | **效果好、同时也一定程度上解决了原先topK这种只能选固定句子的做法的缺陷**  不同于以往抽取式摘要针对的是句子级的抽取。  提出一种针对篇章级的抽取模型：针对整篇参考摘要，从原文中抽取，直接与之匹配。  考虑到了更多句子的组合策略。 |
| 《Heterogeneous Graph Neural Networks for Extractive Document Summarization》 | ACL/2020 | 短文本/长文本  英文 | 抽取式  图神经网络(构建词节点和句子组成图结构) | CNN/DailyMail  Multi-news | 引入词节点，作为句子与句子间关系的中介。构建图，若两句包含的共有词较多，则权重大。  还可以由此延申，引入段节点，篇章节点等，扩展为长文档、多文档摘要任务。 |
| 《Searching for Effective Neural Extractive Summarization: What Works and What’s Next》 | ACL/2019 |  |  |  | 主要探讨在抽取式结构中，各个网络结构对性能的影响。  如：在词级别的编码上，CNN、LSTM差不多  句子级别：Transformer强于LSTM、并且探讨了位置编码对摘要的影响。 |
| 《Controlling the Amount of Verbatim Copying in Abstractive Summarization》 | AAAI/2020 | 短文本  英文 | 生成式 | Gigaword（摘要只是一句话） | 探讨生成式摘要中复制率的影响。（**比较新颖**）  只有一个Transformer Decode，  原文和摘要拼接起来输入到Decoder中。 |
| 《Bottom-Up Abstractive Summarization》 | EMNLP/2018 | 短文本  英文 | 两段式  抽关键词/生成式 | CNN/DM、NYT | 使用数据有效的内容选择器来确定源文档中应成为摘要一部分的短语。  首先在原文中抽取重要的短语。训练时当成序列标注问题，标签构建是通过两种规则。  利用选择出来的短语生成摘要 |
| 《BottleSum: Unsupervised and Self-supervised Sentence Summarization using the Information Bottleneck Principle》 | EMNLP/2019 | 短文本  英文 | 两段式  抽取/生成  **抽取阶段：不需要额外标签(无监督/自监督)**  生成阶段：GPT2 | CNN/DM | 抽取阶段:不需要训练，只需要迭代式的选择最优的摘要句就可完成.  生成阶段:输入抽取得到的句子进入GPT2做文本生成任务 |
| 《HIBERT: Document Level Pre-training of Hierarchical BidirectionalTransformers for Document Summarization》 | ACL/2019 | 短文本  英文 | 抽取式  先用mask整句的方式预训练了一个模型，然后再在模型上微调做抽取式摘要任务 | CNN/DM、NYT | **较多引用**  使用两个预训练Transformer：第一个是用于表示句子的标记级别的标准BERT; 第二，在句子层面，并利用前者的表示来编码整个文件的句子。 |
| 《Sentence Centrality Revisited for Unsupervised Summarization》 | ACL/2019 | 短文本 | 无监督/图/相似度 | CNN/DM、NYT、 | 通过BERT获取句子的表示向量，根据句子的表示向量和彼此之间的相似度分数构建有向图，任意两个节点对于中心性（centrality）的贡献取决于它们在文档中的相对位置。  用**RST**的思想构建图，为了实现语篇单元的划分，一种方法是使用语篇解析器（discourse parser），但是它严重依赖于标注语料库的存在，同时需要额外的使用一些相关的工具，较为麻烦。另一种方法相比就很简单且直接，即直接使用句子在文档中的相对位置来逼近真实中心性的度量。因为通常来说位于文档前面的句子通常包含有更多的信息，因此通过句子之间的位置信息就可以简单的实现语篇单元的划分。 |
| 《Global Optimization under Length Constraint for Neural Text Summarization》 | ACL/2019 | 短文本 | 生成式  主要是对生成式摘要长度不可控问题的研究。 | CNN/DM | 按照论文中的观点，现有大部分模型都无法控制模型输出的摘要长度，所以很多时候都会输出超长的限制。因为在模型中都没有考虑到文本长度的信息，或者已有的一些考虑长度的模型都没有取得很好的分数。故此，论文作者提出了一个考虑到文本长度的优化目标函数（损失函数）。之前的生成类模型，大部分都是用的MLE(最大对数似然估计），并且是强制学习（Decoder中每一步的输入使用真实摘要而不是上一步的输入）所以单纯的MLE无法解决文本过长的问题（因为训练过程中，生成到原始摘要长度就停了）论文中使用了改进的MRT（Minimum Risk Training），MRT方法可以在训练过程中如同预测过程相同，下一个词的生成是根据上一个词的，直到生成结束再根据损失进行优化模型。 |
| 《Self-Supervised Learning for Contextualized Extractive Summarization》 | ACL/2019 | 短文本 | 抽取式/预训练  同一个数据集，先对模型自监督预训练，再用带标签的数据进行微调。  预训练模型：LSTM+Transformer  （采用预训练这种方式，在2020年已经有更好的工作了，采用了更大的模型参数，更多的数据集） | CNN/DM | 阶段一：用本数据集不带标签进行预训练，  阶段二：再用本数据集带标签，在预训练过的模型上进行微调  预训练任务：1.mask整个句子 2.从其他文档中随机选择的句子，然后让模型预测某个句子是否是被替换的3.随机的交换本文档中的句子位置，预测是否是交换过 |
| 《BiSET: Bi-directional Selective Encoding with Template for Abstractive Summarization》 | ACL/2019 | 短文本  英文 | 生成式  **模板方法**（需要人手工写出模板） | Gigawords | 使用模板的双向选择性编码（Biset）是在Gigawords数据集上测试的抽象摘要的新架构。基于模板的摘要依赖于手动创建模板。这种方法的优点在于它可以在不需要训练数据的情况下产生简洁和连贯的摘要。但是，它需要专家来构建这些模板。本文提出了一种从训练语料库中检索高质量模板的自动方法。给定输入文章，模型首先使用基于TF-IDF的方法检索最相似的文章。此外，通过神经网络计算相似性度量，以便对检索到的文章进行重新排序。然后选择对应于与输入最相似的文章的摘要作为模板。最后，训练序列网络的序列以生成摘要：作者提出了一种架构来学习源摘要和所选模板之间的交互。 |
| 《Scoring Sentence Singletons and Pairs for Abstractive Summarization》 | ACL/2019 | 短文本  英文 | 两段式:抽取&压缩/改写  **(创新点就在于不再针对单句抽句子，引入了句子对)** | CNN/DM、XSum | **比较有参考意义**  原先的抽取模型都是针对单句的，但是考虑这种情况：参考摘要中的一句可能是由原文中的多句组成。于是，本文考虑了单句和句子对（连续的两句）这种情况，将原文所有单句和句子对N+C(n,2)输入Bert得到表示，和参考摘要进行相似性评分。  生成阶段：直接对上阶段选择出来的单句或句子对进行压缩/改写（指针生成网络）。 |
| 《Jointly Extracting and Compressing Documents with Summary State Representations》 | NAACL/2019 | 短文本  英文 | 两段式：抽取&压缩  LSTM | CNN/DM | 抽取阶段：常规做法，序列标注问题，01标签选句子。  压缩阶段:类似于上一阶段，给句子中的每个词打01标签。  （和抽取&改写模型不同的是，第二阶段的压缩中的词全部来自原文，而改写模型使用了诸如指针-生成网络，既有原文的词，又有从词表中选择生成的新词） |
| 《Discourse-Aware Neural Extractive Text Summarization》 | ACL/2020 | 短文本（可扩展到长文本）  英文 | 抽取  Bert、图卷积网络 | CNN/DM、NYT | 作者将基本语篇单元（EDU）而不是句子作为最小选择单元，其中EDU是源自RST（修辞结构理论）的子句短语单元。通过在话语单元级别上进行操作，模型可以丢弃子句中的多余细节，因此保留了包含更多概念或事件的额外功能，从而提供了更加简洁和信息丰富的摘要。 |
| 《HipoRank: Incorporating Hierarchical and Positional Information into Graph-based Unsupervised Long Document Extractive Summarization》 | ACL/2020 | 长文本  英文 | 抽取  基于图的无监督抽取 | Arxiv/PubMed | 在基于图的排名算法中加入的边界位置函数和层次结构  值得借鉴的思想：  处于文章边界的信息比较重要；  通过建立层次结构的子图，来减少一些不必要的“弱边”；  要善于利用数据集的特点，来对自己的方法进行相对应的调整。  存在的不足：  有一定的局限性，对于长篇的新闻文章不一定适用。 |
| 《Understanding Points of Correspondence between Sentences for Abstractive Summarization》 | ACL/2020 |  | 生成式  **句子融合** |  |  |
| 《Asking and Answering Questions to Evaluate the Factual Consistency of Summaries》 | ACL/2020 |  | 生成摘要的事实性检测 |  | 提出QAGD框架，用来评测生成摘要的事实性，辅助Rouge得分。  （1） 给定一个生成的文本 T，通过一个问题生成器 QG生成与该文本相关的问题  （2） 通过一个 QA模型来分别接收源文本和生成的摘要作为输入之一回答 QG生成的问题。  （3） 最后基于相应答案的相似性来计算生成摘要的质量得分。 |
| 《Self-Attention Guided Copy Mechanism for Abstractive Summarization》 | ACL/2020 | 短文本  英文 | 生成式  对生成式摘要中复制原文中的重要词的讨论 | CDD/DM,Gigaword | 提出了基于文本摘要源句单词的中心度为指向的复制机制，中心度的获取通过入度和出度计算来度量。还有中心度感知（ centrality-aware）的注意力，并且提出了一个以损失为导向的模型，使得模型可以对源句中重要的单词有更多的关注。  使用基于Transformer的模块增强复制机制（copy mechanism），具体地说：使用自注意力（self-transformer）来计算源句中单词的重要性  中心度计算：引入了自注意力图之后，我们计算基于邻接矩阵计算每个单词的中心度。一个更直接的方法是使用TextRank算法来保证每个单词从其他单词获得更多的相关度得分。TextRank方法以入度为中心著称。这是本文的创新内容，中心度计算的结果最后作为重要信息用来确定源句单词的概率分布，进一步确定了复制分布（copy distribution）。整合中心度到注意力计算时的打分上。 |
|  |  |  |  |  |  |
| 《Unsupervised Reference-Free Summary Quality Evaluation via Contrastive Learning》 | EMNLP/2020 |  | 一种新型的对生成的文本摘要质量的评价方式  无监督/**对比学习** |  | 主要包括两部分：   1. 设计一个基于Bert的evaluator来对摘要进行打分，语义方面：semantic、语言方面:linguistic 2. 从原数据集中创建负样本，使用对比学习来训练evaluator，（对比学习的思想构建损失函数）   在测试阶段，我们不需要任何参考摘要，也就是说，对于一个没有已知参考摘要的考生摘要，我们也可以预测它的分数 |
| 《BERTScore: Evaluating text generation with BERT》 | ICLR/2020 |  | 利用Bert来评测文本生成质量的指标  （在有些文本摘要任务上也用到过） |  | 思路直接：把生成句和参考句分别用送入bert提取特征，然后对 两个句子的每一个词分别计算内积，可以得到一个相似性矩阵。基于这个矩阵，我们可以分别对参考句和生成句做一个最大相似性得分的累加然后归一化，得到 bertscore 的 precision，recall 和 F1： |
| .《[Sentence-BERT: Sentence Embeddings using Siamese BERT-Networks](https://www.cnblogs.com/shona/p/12026349.html)》 | EMNLP/2019 |  | 句子语义相似度 |  | 模型是对预训练的Bert进行修改（做法也比较简单），用于做语义相似度的任务。文章最大的亮点是在保证准确性的同时大大加快推理速度。（从 10000 个句子中找出最相似的句子对，大概需要 5000 万在 V100GPU 上耗时约 65 个小时。SBERT可将上述提到 BERT/RoBERTa 的 65 小时降低到 5 秒。） |
| 《Compressive Summarization with Plausibility and Salience Modeling》 | EMNLP/2020 | 短文本  英文 | 两段式：抽取、压缩  抽取阶段用现有的模型，压缩阶段改进 | CNN、WikiHow、XSum、Reddit | 抽取阶段：利用BertSum  压缩阶段：论文先是花了大部分来讲解这两个概念plausibility和salience，认为在进行压缩摘要时应当着重考虑这两方面。具体做法：选择出一些候选的跨度，计算这些跨度的表示，然后用于计算它们是否该删除 |
| 《Neural Extractive Summarization with Hierarchical Attentive Heterogeneous Graph Network》 | EMNLP/2020 | 短文本  英文 | 抽取式/图神经网络 | CNN/DM、NYT、Newsroom | 单词和句子都被构建成节点，它们之间的关系被构造为不同类型的边。  一定程度上解决了选topk这种只能生成固定句子数量的摘要的缺陷。 |
| 《Pre-training for Abstractive Document Summarization by Reinstating Source Text》 | EMNLP/2020 | 短文本  英文 | 生成式/预训练  先针对摘要任务预训练一个模型，再在此之上进行微调  （很类似**天马**那篇文章的思路，但是性能没有前者好） | CNN/DM、NYT、GIGA-CM | 以Transformer为基础  提出三个预训练目标：  句子重排序，下一句生成，masked文档的生成 |
| 《On Extractive and Abstractive Neural Document Summarization with Transformer Language Models》 | EMNLP/2020 | **长文本**  英文 |  | arXiv、PubMed、Newsroom、BigPatent |  |

ACL2020中，关于文本摘要事实性检测的文章数量增多，感觉这是一个新的热点？

长文档摘要的工作还是比较少，数据集目前发现的还是只有Arxiv和PubMed。有几篇都是用图神经网络来做的，或者是先在CNN这类短文本上用图做，文章提到以后可以扩充到长文本，总体感觉构建图的难度比较大。