北京大学学报(自然科学版) 第 57 卷 第 1 期 2021 年 1 月
Acta Scientiarum Naturalium Universitatis Pekinensis, Vol. 57, No. 1 (Jan. 2021)
doi: 10.13209/j.0479-8023.2020.082

基于细粒度可解释矩阵的摘要生成模型

王浩男[1] 高扬[1,3,†] 冯俊兰[2] 胡珉[2] 王惠欣[2] 柏宇[1]

1. 北京理工大学计算机学院, 北京 100081; 2. 中国移动通信研究院, 北京 100032; 3. 北京市海量语言信息处理与云计算应用工程技术研究中心, 北京 100081; † 通信作者, E-mail: gyang@bit.edu.cn

摘要 针对摘要模型中总结并解释长篇上下文信息存在的困难, 提出一种基于细粒度可解释矩阵, 先抽取再生成的摘要模型(fine-grained interpretable matrix, FGIM), 提升长文本对显著度、更新性和相关度的可解释抽取能力, 引导系统自动生成摘要。该模型通过一个句对判别(pair-wise)抽取器对文章内容进行压缩, 捕获文章中心度高的句子, 将抽取后的文本与生成器相结合, 实现摘要生成。在生成端通过可解释的掩码矩阵, 控制生成摘要的内容属性, 在编码器端分别使用多层 Transformer 和预训练语言模型 BERT 来验证其适用性。在标准文本摘要数据集(CNN/DailyMail 和 NYT50)上的实验表明, 所提模型的 ROUGE 指标和人工评估结果均优于当前最好的基准模型。实验中还构建两个测试数据集来验证摘要的更新度和相关度, 结果表明所提模型在可控生成方面取得相应的提升。

关键词 生成式摘要; 可解释抽取; 中心度; 掩码矩阵; 可控生成

Abstractive Summarization Based on Fine-Grained Interpretable Matrix

WANG Haonan[1], GAO Yang[1,3,†], FENG Junlan[2], HU Min[2], WANG Huixin[2], BAI Yu[1]

1. School of Computer Science and Technology, Beijing Institute of Technology, Beijing 100081; 2. China Mobile Research Institute, Beijing 100032; 3. Beijing Engineering Research Center of High Volume Language Information Processing and Cloud Computing Applications, Beijing 100081; † Corresponding author, E-mail: gyang@bit.edu.cn

Abstract According to the great challenge of summarizing and interpreting the information of a long article in the summary model. A summary model (Fine-Grained Interpretable Matrix, FGIM), which is retracted and then generated, is proposed to improve the interpretability of the long text on the significance, update and relevance, and then guide to automatically generate a summary. The model uses a pair-wise extractor to compress the content of the article, capture the sentence with a high degree of centrality, and uses the compressed text to combine with the generator to achieve the process of generating the summary. At the same time, the interpretable mask matrix can be used to control the direction of digest generation at the generation end. The encoder uses two methods based on Transformer and BERT respectively. This method is better than the best baseline model on the benchmark text summary data set (CNN/DailyMail and NYT50). The experiment further builds two test data sets to verify the update and relevance of the abstract, and the proposed model achieves corresponding improvements in the controllable generation of the data set.

Key words abstractive summarization; interpretable extraction; centrality; mask matrix; controllable

近年来, 神经网络在抽取式摘要和生成式摘要任务中取得显著的成功。抽取式摘要是从原文直接选择可读性好并与文章相关的句子作为整篇文章的摘要, 生成式摘要[1]是借助机器翻译衍生出来的编码–解码框架生成新的摘要序列。尽管这些方法都取得较大的成功, 但长文本的语义建模以及细粒度信息的获取仍是文本摘要领域的巨大挑战。

目前, 有两种常用方法来解决上述问题。1) 基

教育部–中国移动科研基金(MCM20170302)资助
收稿日期: 2020–06–08; 修回日期: 2020–08–07

于预训练的语言模型(如 ELMO[2], OpenAI GPT[3]和 BERT[4]), 在表示文本上下文向量的学习过程中非常出色, 并广泛应用于多个自然语言相关的子任务中(如问答系统[5]和摘要系统[6−7]); 2) 结合抽取器与生成器构成混合摘要生成框架, 首先通过抽取器来选择显著性高的句子, 然后利用这些句子, 通过生成器进一步生成最终的摘要, 称为混合摘要模型。混合摘要模型利用抽取器进一步细化信息量与摘要相关内容抽取的效果, 同时利用生成器将其汇总为符合语言表达形式的最终摘要。在训练抽取器时, 简单的隐层表示不能完整地表达句子与候选摘要之间的关系, 需要深入地探索复杂的句间关系(即识别语义, 判断句子是否与文档相关以及对摘要的贡献程度)。在做序列生成任务时, 指针–生成模型(pointer-generator)应用比较广泛, 然而, 长文档的信息具有多样性, 且重要内容具有离散的特点, 单一的指针生成模型不能有效地捕捉到文章离散多样性的特点, 导致生成的摘要局限于文章的某一部分而非整体。按照人类阅读习惯, 在对一篇文章进行总结时, 往往先根据文章的内容(如显著度、相关度和更新度)进行总结, 最后基于细粒度信息对整篇文章进行总结。因此, 对于一个可解释的文本生成模型, 能够把文章中包含的可解释的细粒度信息有效地提炼出来, 会使模型更加符合人类摘要的方式, 同时也能保证系统生成的摘要质量更高。模型具备细粒度信息后, 会引导模型在具备该信息的方向上对文章内容进行总结, 比如更新度高的细粒度信息会使系统最终生成的摘要具备多样性, 类似可控旋钮。因此, 摘要生成的可控性是文本生成领域内又一重要需求。

针对上述研究现状, 学者们提出很多方法和模型(如序列生成模型[8]), 但仅依靠序列生成模型, 难以建模长文档的上下文依赖关系。主要原因是现有模型很难仅通过向量表示准确地理解长文档的语义信息, 加上基于语言模型的生成网络是一个“黑盒”, 不能明确辨别所选内容的细粒度信息。

指针–生成模型将注意力作为指针, 以上下文作为条件, 控制选词或选句的概率。在信息选择方法中, 词级别的包括 Zhou 等[9]用软控门对原文的冗余信息进行过滤, Hsu 等[10]通过句子的重要程度更新词级别的注意力, Gehrmann 等[11]利用预训练的方法构建单词选择器来约束从源文档中获取的词级别注意力; 句级别的包括 Tan 等[12]采用基于图的注意力机制增强文章显著性内容对生成摘要的影响, Li 等[13]通过信息选择层实现对文章冗余信息的过滤, You 等[14]通过引入高斯聚焦偏差增强信息选择的能力进一步对文章显著信息建模。

我们的模型继承指针生成模型用于选择和生成的优点, 并进一步研究可解释的选择文章中的细粒度信息对摘要生成的影响。本文提出基于细粒度可解释矩阵(Fine-Grained Interpre-table Matrix, FGIM)的模型来建模丰富的句间关系, 通过该交互矩阵对文章中的句子进行决策(是否作为中心句), 通过衡量句子的丰富度和句对间的相似性来构建句子级别的抽取器, 对文章中的句子打分。依据句对的复杂关系, 获取中心度高的句子, 影响最终摘要的生成。抽取器与生成器通过端到端的方式进行训练和预测, 同时利用不同的句子特征(相关度和更新度)构建不同的可解释掩码矩阵来作用到交互矩阵上, 构造可解释旋钮。主要在 CNN/DailyMail 和 NYT50 两个数据集上对模型进行验证, 同时采用人工评估和机器评估(ROUGE)的方式辅助验证。

1 基于Transformer的编码–解码框架

编码–解码框架由编码器和解码器构成。解码器具备注意力机制, 帮助模型对输入 X 的每个部分赋予不同的权重, 抽取出更关键、更重要的上下文信息。设输入序列 $X=\{x_1, \ldots, x_j, \ldots, x_n\}$ 是一个包含 n 个词汇的序列, j 为输入序列索引。输出序列(摘要)定义为 $Y=\{y_1, \ldots, y_t, \ldots, y_m\}$, 包含 m 个词汇。

1.1 编码器

模型的基本架构基于 Transformer, 由 N 个相同的 Transformer 层堆叠构成, 每层网络含两个子层:

$$\boldsymbol{h}_1^l = \mathrm{LAYERNORM}(\boldsymbol{h}_2^{l-1} + \mathrm{MULHatt}(\boldsymbol{h}_2^{l-1})), \quad (1)$$

$$\boldsymbol{h}_2^l = \mathrm{LAYERNORM}(\boldsymbol{h}_1^l + \mathrm{FFN}(\boldsymbol{h}_1^l)) 。 \quad (2)$$

式(1)代表第一个子层(自注意(Self Attention)层), 式(2)代表前馈子层。LAYERNORM 是归一化层, 框架中多头注意力(multihead attention)的操作为

$$\mathrm{MULHatt}(\boldsymbol{h}_2^{l-1}) = \mathrm{CONCAT}(\boldsymbol{H}_1, \cdots, \boldsymbol{H}_h)\boldsymbol{W}_l, \quad (3)$$

$\boldsymbol{H}_i$ 为第 l 层在第 i 个头的自注意操作, $\boldsymbol{W}_i$ 为可训练的参数。编码器的输出定义为 $\boldsymbol{Z}_e$, 在基于 Transformer 的框架中同时采用预训练的 BERT 编码器。

1.2 解码器

对基于 Transformer 和基于 BERT 的实验设置，均采用带有注意力机制的解码器，从而可以考虑输入文档的上下文信息，解码器由 N 层 Transformer 组成。除与编码器相似的两个子层外，解码器还增加第 3 个子层，对编码器的输出以及上一个时刻解码器的输出进行自注意的操作。在每个原位置，计算解码器的位置矢量 S_t 和编码器输出 $\boldsymbol{Z}_e$ 之间的注意力分布。通过式(4)，获取解码器在 t 时刻输入 $\boldsymbol{Z}_e$ 的注意力分布:

$$\alpha_t = \text{softmax}\left(\frac{\boldsymbol{Q}\boldsymbol{K}^{\mathrm{T}}}{\sqrt{d_m}}\right)。\tag{4}$$

利用式(5)计算 t 时刻的上下文向量 $\boldsymbol{h}_t^*$:

$$\boldsymbol{h}_t^* = \alpha_t \boldsymbol{Z}_e。\tag{5}$$

解码器通过式(6)获取 t 时刻词表中单词的分布，解码当前时刻的单词:

$$\begin{aligned} P_{\text{vocab}}(w) &= P(y_t | y<t, x; \theta) \\ &= \text{softmax}(\boldsymbol{W}_2(\boldsymbol{W}_1[\boldsymbol{s}_t, \boldsymbol{h}_t^*] + b_1) + b_2)。 \end{aligned}\tag{6}$$

2 FGIM 模型

图 1 给出 FGIM 模型的整体框架，该框架结合抽取器与生成器的特点，实现端到端的混合摘要模型。模型第一部分是基于句对方法的抽取器，通过交互矩阵，对文档中的句子进行基于文档中心度的评分；第二部分是摘要生成，借助指针生成网络模型的注意力指针，利用混合连接部分，结合抽取器获得的中心度信息，影响最终的词表概率分布；第三部分利用掩码矩阵，实现对抽取器中的交互矩阵的控制，获得基于不同属性的句子中心度，影响最终摘要的生成，实现可控生成的目标。

2.1 抽取器

2.1.1 句子交互矩阵(interaction matrix)

由于文档中的句子均存在复杂的关系(如内容丰富程度、更新度及与文档的相关度等)，因此通过构建句子交互矩阵 $\boldsymbol{Q}^s$ (s 为文档中句子的数量)来获取更准确且具备可解释性的句子中心度。$\boldsymbol{Q}^s$ 可通过计算句对 i 与 j 的交互关系来构建:

$$\begin{aligned} \boldsymbol{q}_{i,j}(\boldsymbol{h}_i, \mathbf{nov}_i, \boldsymbol{h}_j, \boldsymbol{d}) = \sigma(&\underbrace{\boldsymbol{W}_c \boldsymbol{h}_i}_{\text{显著度}} + \underbrace{\boldsymbol{h}_i^{\mathrm{T}} \boldsymbol{W}_s \boldsymbol{h}_j - \boldsymbol{h}_i^{\mathrm{T}} \boldsymbol{W}_n \tanh(\text{nov}_i)}_{\text{更新度}} + \\ &\underbrace{\boldsymbol{h}_i^{\mathrm{T}} \boldsymbol{W}_r \boldsymbol{d}}_{\text{相关度}} + \boldsymbol{b}_m), \end{aligned}\tag{7}$$

其中，σ 是 sigmoid 函数，$\boldsymbol{W}_{\mathrm{c}}$，$\boldsymbol{W}_{\mathrm{s}}$，$\boldsymbol{W}_n$ 和 $\boldsymbol{W}_r$ 是模型的训练参数，$\boldsymbol{d}$ 为文档的表示，$\boldsymbol{b}_m$ 是 bias; $\boldsymbol{h}_i$ 和 $\boldsymbol{h}_j$ 分别为句子 i 和 j 的表示向量，$\mathbf{nov}_i$ 是对当前句子向量 i 所维持更新度的衰减值:

$$\mathbf{nov}_i = \frac{1}{s}\sum_{t=1}^{i-1}\sum_{k=1}^{s} \boldsymbol{h}_t \cdot \boldsymbol{q}_{tk}。\tag{8}$$

$\boldsymbol{q}_{tk}$ 为式(7)计算得到的当前 t 时刻的句对关系。

2.1.2 中心度计算

交互矩阵提供文档中句对之间相互影响程度，可以协助抽取器获取文档中句子的整体中心度。从句子级别提炼文档的中心度比从文档级别提炼的信息损失少，同时更具备细粒度属性。目前计算句子中心度均采用无监督进行摘要总结，如基于图的 TextRank[15]和 LexRank 等模型。在 FGIM 模型中，

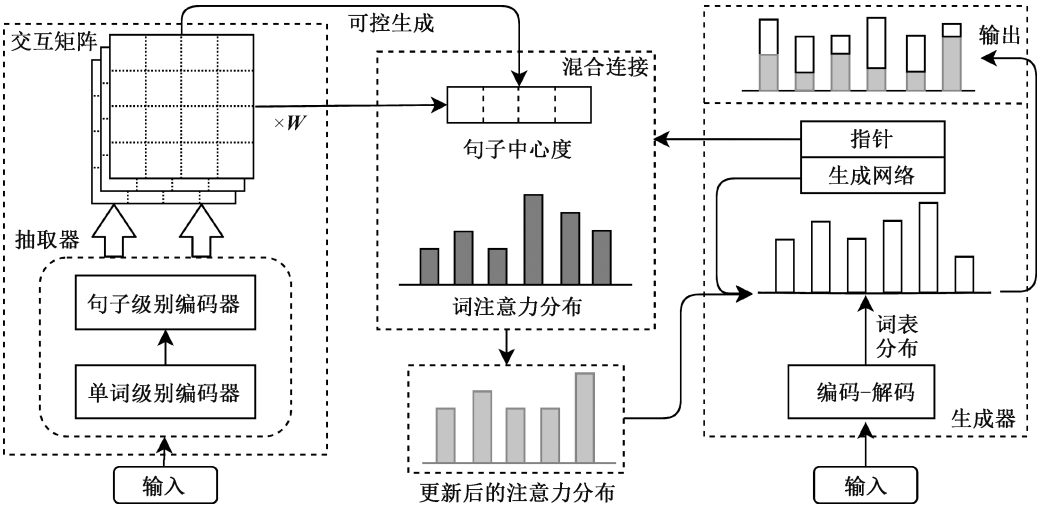

图 1　FGIM 模型结构

Fig. 1　FGIM model structure diagram

可以通过监督学习的方法, 利用可学习的参数 $\boldsymbol{W}_q$, 将交互矩阵 $\boldsymbol{Q}^s$ 转化为基于句子分布的中心度向量 $\boldsymbol{c}=[c_1, \ldots, c_s]$:

$$\boldsymbol{c}=\boldsymbol{Q}^s\boldsymbol{W}_q\text{。} \tag{9}$$

2.1.3 抽取器训练过程

抽取器的训练通常被构建为一个分类模型的训练过程, 将句子编码为隐层表示向量, 通过分类层预测这些表示是否为摘要句。与抽取的训练过程类似, 也采用单句判别(point-wise)的学习目标, 但是, 单句判别对交互矩阵的参数学习没有明显的作用。因此, 为了更好地反映句子之间的相互作用, 通过新的标签方法, 使用基于句对方法的目标函数来训练抽取器的参数, 更好地体现句子间的交互关系。句对[i, j]的标签设置见表 1。在监督学习框架下, 基于句对方法的目标函数如下:

$$L_{\text{ext}}=-\sum_{i=1}^{m}\sum_{j=1}^{m}(\hat{P}_{ij}\log r_{ij}+(1-\hat{P}_{ij})\log(1-r_{ij}))\text{,} \tag{10}$$

m 为句子的个数, r_{ij} 为句子 S_i 和句子 S_j 的共现概率:

$$r_{ij}=\sigma(c_i-c_j)\text{,} \tag{11}$$

其中, c_i 和 c_j 分别对应句对 $\{i, j\}$ 的中心度得分。

2.2 生成器

在 FGIM 模型架构中, 生成器的实现主要借助指针生成模型。基础的指针生成网络包含两个子模块: 指针网络和生成网络。这两个子模块共同确定最终生成的摘要中每个单词的概率。基础的指针生成网络采用经典的基于 Transformer 的编码–解码网络结构, 在此基础上, FGIM 集成句子中心度更新指针模块, 将抽取器获取的句子中心度信息更新到生成器中, 从而影响最终的摘要生成过程。

2.2.1 句子中心度更新模块

指针网络使用注意力机制作为指针, 选择输入语料中合适的单词作为输出。在 FGIM 模型中, 指针生成网络与抽取器中获取的句子中心度信息结合, 可以更好地协助指针生成网络, 提取文章的突出信息(原始指针生成网络不考虑句子中心度信息)。为了更好地影响序列生成过程, 句子的中心度信息需要分散到单词级别上, 影响生成器逐词的生成过程, 因此, 本文利用混合连接的方式, 结合抽取器和生成器, 实现模块的无缝连接。

表 1 Prair-wise 标签

Table 1 Label method of pair-wise

s_i	s_j	$\hat{P}_{ij}$
摘要句	非摘要句	1
非摘要句	摘要句	0
摘要句	摘要句	0.5
非摘要句	非摘要句	0.5

2.2.2 混合连接(hybrid connector)

利用句子中心度的信息, 更新指针生成网络中单词注意分布, 可以使摘要的生成过程可以向抽取器获取的重点关注的内容靠拢, 从而在单词级别上更新注意力分布:

$$\hat{\alpha}_t^n=\frac{\alpha_t^n(1+p_{\text{sen}}\boldsymbol{c}_{m_n})}{\sum\alpha_t^n(1+p_{\text{sen}}\boldsymbol{c}_{m_n})}\text{,} \tag{12}$$

P_{sen} 决定一个句子的影响程度; c_{m_n} 表示单词 n 所属句子 m 的得分, 由抽取器获取。

$$p_{\text{sen}}=\sigma(\boldsymbol{W}_{\text{sel}}\boldsymbol{E}_{\text{sel}}^t+\boldsymbol{b}_{\text{sen}})\text{,} \tag{13}$$

E_{sel}^t 代表在解码 t 时刻选取的句子 m 的隐层表示, W_{sel} 为可训练的参数。

生成概率 P_{gen} 的计算公式为

$$p_{\text{gen}}=\sigma(\boldsymbol{W}_{\boldsymbol{h}}*\boldsymbol{h}_t^*+\boldsymbol{W}_s\boldsymbol{s}_t+\boldsymbol{b}_{\text{gen}})\text{。} \tag{14}$$

PG 网络利用更新后的 $\hat{\alpha}_t$ 和生成概率 P_{gen} 来计算最终分布:

$$P_{\text{final}}(w)=p_{\text{gen}}P_{\text{vocab}}(w)+(1-P_{\text{gen}})\sum_{j:w_j=w}\hat{\alpha}_{t,j}\text{。} \tag{15}$$

2.3 可控性摘要生成

交互矩阵可以捕获文章中的句间关系, 因此文章整体的中心度 c 能够反映可解释摘要的更新度和相关度等属性。为了探索生成摘要的可解释性, 模型采用可控制的阈值方法, 对式(7)中的更新度和相关度进行调节, 构造一个包含 {0, 1} 的掩码矩阵 $\boldsymbol{M}$, 对交互矩阵 $\boldsymbol{Q}^s$ 进行更新, 从而使抽取器获取的中心度信息向更新度或相关度靠拢:

$$\hat{\boldsymbol{Q}}^s=\boldsymbol{Q}^s\odot\boldsymbol{M},\quad M_{ij}=\begin{cases}1, & \text{val}\geq\epsilon,\\ 0, & \text{val}<\epsilon,\end{cases} \tag{16}$$

其中, ⊙为元素对应相乘, val 的数值对应式(7)中的 σ (更新度)或 σ (相关度)。

利用基于不同属性的 val 值, 构建掩码矩阵 $\boldsymbol{M}_n$ (更新度)或 $\boldsymbol{M}_r$(相关度), 通过式(15)达到对 $\boldsymbol{Q}$ 矩阵可解释控制的目的, 使抽取器获取的文章中心度信息向不同的属性偏移, 从而影响单词注意力分布, 最终影响摘要的生成。

2.5 生成器训练过程

采用极大似然估计的方法对生成器进行训练，给定文档 x 和参考摘要 $y^*=\{y_1^*,y_1^*,\ldots,y_m^*\}$，生成器的训练目标是最小化目标单词序列的负对数似然:

$$L_{\text{abs}}=-\sum_{t=1}^{m}\log P_{\text{final}}(y_t^*\mid y_1^*,y_2^*,\ldots,y_{t-1}^*,x)。\quad(17)$$

在端到端的训练过程中，最终的目标函数定义为 $L=L_{\text{ext}}+L_{\text{abs}}$。

3 实验与结果分析

3.1 数据集与评价指标

FGIM的模型评估使用两个基准数据集，CNN/Dailymail[16]和New York Annotated Corpus (NYT)[17]。CNN/DailyMail数据集包含新闻文章，并由人工构建参考摘要，按照90266/1220/1093和196961/12148/10397的规模，将数据集划分为训练集/验证集/测试集。参照文献[1]进行数据预处理。NYT数据集包含110540篇英文文章和人工摘要，训练集和测试集分别含100834和9706个示列。在上述数据的预处理过程中，对测试集进行额外的预处理，删除少于50个单词的人工摘要，过滤后的测试集称为NYT50，包含3421个示例。两个数据集的分词分句均采用Stanford Core NLP分词工具。使用标准的ROUGE作为评价指标，通过计算模型生成的候选摘要与参考摘要之间的重叠词汇来衡量模型生成摘要的质量，将R-1, R-2和R-L值作为评估指标。

3.2 基准模型对比

为了比较FGIM模型的性能，选取在生成摘要中表现较好的模型作为对比：指针生成网络，基于双向GRU的序列到序列的模型框架；PG+Coverage，在指针生成网络的基础上增加Coverage覆盖机制；Select-Reinforce[18]，利用强化学习方法，以ROUGE评价指标为奖励函数，对文章中的句子进行抽取；Inconsistency-Loss，构建基于单词与句子注意力机制的损失函数；Bottom-up，使用编码器的作为内容选择器，约束生成摘要过程中用到的单词注意；ExplictSelection，在原有的序列到序列的模型框架上进行扩展，加入信息选择层，对冗余信息进行过滤；SENECA，抽取一些具有实体的句子，然后连接到基于强化学习的摘要系统进行改写；BERTSUMabs，基于BERT的抽象摘要。

3.3 参数设置

FGIM-Transformer是基于Transformer的模型，包含6层Transformer，隐层为512，前馈层维度为1024，采用多头注意力机制，包含8个头。在线性层前，dropout的概率设为0.2。基于Transformer的指针生成网络采用的学习率设为0.15，编码器的批处理大小设为32，解码器束搜索的大小设为4。模型的输入将原文档进行截取，CNN/DailyMail取文档中前400个单词的长度作为输入，NYT50取文档中前800个单词长度作为输入，在训练集和验证集上的目标摘要长度取为100个单词，在测试集上的目标摘要长度取120个单词。采用早停法和长度惩罚的方法进行模型训练。

FGIM-BERT是基于BERT的模型，在文章中每个句子的开头插入[CLS]标记，使用间隔符号[EA]和[EB]区分文档中的多个句子，通过[CLS]学习句子的嵌入式表示。在BERT模型中，位置嵌入表示的大小为512，采用“bert-base-uncased”的BERT预训练模型版本，输入文档和目标序列均采用Sub-words机制标记。Transformer层的隐层设为768，所有的前馈层设为2048。对于抽取器，使用一层Transformer获取句子的表示(式(7)中的 h_i)，该层Transformer包含8个头，dropout的概率为0.1。采用Trigram block的方法防止生成重复序列。在CNN/DailyMail和NYT50两个数据集中分别采用15 k和100 k的迭代次数，全连接层的dropout概率设为0.2。解码器包含6个Transformer层。对基于BERT的编码器和基于Transformer的解码器，分别采用0.002和0.2的学习率，解码过程与FGIM-Transformer的设置相同，在两块2080Ti GPU上进行训练。训练过程中抽取器占用24 h，生成器占用48 h，混合的FGIM模型占用24 h，模型总的参数量为1.8亿，使用交叉验证的方法选择超参数。

3.4 性能分析

表2为模型在CNN/DailyMail和NYT50数据集上的实验结果。可以看出，FGIM-BERT模型的所有指标都超过目前最好的模型。在基准模型中，均为通过先抽取再生成的框架进行摘要生成，本文的FGIM-BERT模型在相同框架的基础上，比目前最好的模型(BERTSumAbs)在两个数据集上均提高1%~6.55%。尤其在NYT50数据集上，FGIM-BERT模型在R-2指标上增幅最大，说明在生成模型中引入基于文章的可解释性细粒度信息是有效的。除使用BERT的基准模型外，FGIM-Transfor-mer的效果普遍略高于现有最优模型，说明FGIM框架具有普

表 2 CNN/DailyMail和NYT50数据集的ROUGE评价结果(%)

Table 2 ROUGE scores on CNN/DailyMail and NYT50 (%)

模型	CNN/DailyMail			NYT50		
	R-1	R-2	R-L	R-1	R-2	R-L
PG+Coverage	39.53	17.28	36.38	43.71	26.40	37.79
Select-Reinforce	40.88	17.80	38.54	–	–	–
Inconsistency-Loss	40.68	17.97	37.13	–	–	–
Bottom-Up	41.22	18.68	38.34	47.38	31.23	41.81
Explicit-Select	41.54	18.18	36.47	–	–	–
SENECA	41.52	18.36	38.09	47.94	31.77	44.34
BERTSumAbs	41.72	19.39	38.76	48.92	30.84	45.41
FGIM-Transformer	41.65	18.89	37.94	47.63	30.10	43.94
FGIM-BERT	**42.12**	**19.52**	**39.07**	**49.41**	**32.22**	**45.83**

说明: “–”表示基准模型没有使用对应数据集测试; 粗体数字表示最优结果。

遍有效性。Transformer 比 BERT 表现差, 也说明通过预训练模型可以增强模型文本表示的能力, 因此更适用于序列生成的任务。

3.5 可控性能分析

3.5.1 数据构建

为了探究系统生成的摘要性能(即是否符合预先期望的相关度和更新度), 基于原始 CNN/DailyMail 的测试数据集, 创建两个样例数据集。其中用于相关度测试的数据集, 通过添加一个对应文章的标题作为参考摘要的一部分, 测试经过相关度控制后模型生成的摘要是否与输入文档相关联。由于 CNN/DailyMail 数据集倾向于选择文章中前几句作为摘要, 因此不包含文章的整体信息。在此基础上, 通过构建更新度的测试集, 评估系统生成摘要是否具备全局信息以及鼓励生成更多样化摘要的能力。利用无监督抽取式摘要的方法 PacSum[19]对输入文档后半段的内容进行抽取, 选择最终得分排前 3 位的句子作为最终的参考摘要。考虑到 CNN/DailyMail 本身的数据特点, 输入文档去除开头前 5 句的内容, 将最终 PacSum 的输出补充到原有的参考摘要中。为了分析系统生成摘要的可解释性, 针对相关度和更新度, 设置不同的阈值 ϵ 来构造掩码矩阵。表 3 为 FGIM-BERT 模型在两个人工数据集上的实验结果。

可以看出, 在不同阈值下, 对更新度的控制可以捕捉到更多样化的摘要。由于更新度数据集中的参考摘要增加了与文章后半段内容相关联的摘要, 在最终的 ROUGE 结果中, 基于更新度的可控摘要生成的 ROUGE 得分比阈值为 0 的情况有一定程度

表 3 FGIM-BERT 可控性能比较(%)

Table 3 Controllable performance comparison of FGIM (%)

可控性	阈值	R-1	R-2	R-L
更新度(ϵ_n)	0	44.78	35.39	30.33
	0.3	45.66↑	36.28↑	43.05↑
	0.4	45.26↑	36.08↑	42.67↑
	0.5	45.28↑	35.90↑	42.71↑
相关度(ϵ_r)	0	41.35	18.50	38.57
	0.3	41.41↑	18.57↑	38.62↑
	0.5	41.52↑	18.67↑	38.55↓
	0.7	41.27↓	18.44↓	38.43↓

说明: ↑和↓表示在无控制条件下 ROUGE 分数提升或下降。

的提升, 也说明加入可控信息后, 系统生成的摘要能够向文章的全局信息靠拢。在 $\epsilon_r = 0.5$ 的情况下, 基于相关度的可控效果达到最优(除 R-L 外), 但在 $\epsilon_r = 0.7$ 时效果下降, 说明在可控性与摘要系统性能之间也存在权衡。

从体现模型可控性的示例可以看出, 加入相关性控制后(图 2(a)), 与原始 FGIM 模型相比, FGIM 模型能够生成与参考摘要中相关的内容(灰色), 同时仍能保留原始 FGIM 生成的内容(下划线); 加入更新度控制后(图 2(b)), 模型能够生成与“Talley’s longevity”(下划线)不一样主题的摘要句(灰色), 涵盖原文档中新的主题, 对文章的全局信息有更好的覆盖更新。

3.5.2 人工评价

为验证更新度和相关度可控实验的准确性, 本文还采用问答和标准排序的方法进行人工评估。

问答方法[20]: 按照问答的模式, 对系统生成摘要进行评估。首先基于参考摘要初始一组问题, 参与者阅读 FGIM 系统和其他基线模型生成的摘要, 然后按问答的模式对初始问题作答。根据标准答案进行打分(0~5 分), 与标准答案越接近, 得分越高, 说明模型生成摘要的能力越好。

标准排序方法: 为参与者提供整个文档和针对该文档的多个匿名系统(包含 FGIM)生成的摘要, 根据特定的标准(信息量、新颖度、相关度和流畅度等)选择最好和最差的摘要。计算各系统摘要被选为最好(Best, 1)和最差(Worst, –1)摘要次数差值的百分比, 作为每个系统的得分(–1~1)。

表 4 为基于问答和标准排序的人工评估结果, 其中 Gold 为数据集中给定的参考摘要, 作为不同系统之间相互比较的天花板。可以看出, FGIM-BERT

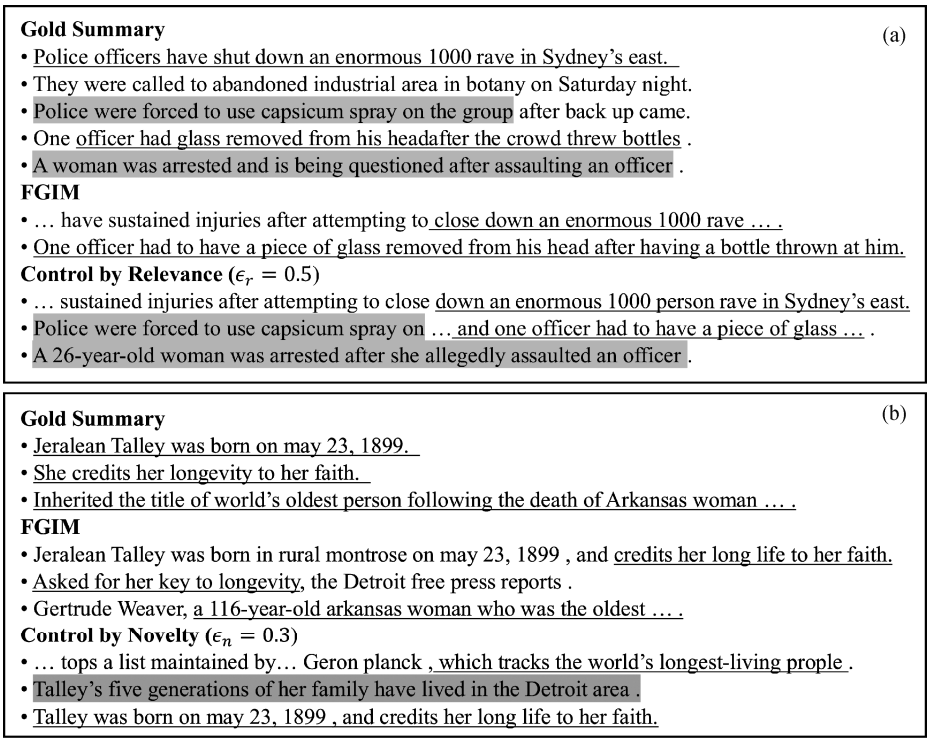

图 2　FGIM 模型的实例生成结果

Fig. 2　FGIM Model instances generated results

表 4　基于问答和标准排序的人工评估

Table 4　QA-based and criteria-based human evaluation

模型	QA	标准排序			
		信息性	多样性	相关性	流畅性
PG+coverage	26.0	−0.28	−0.43	−0.05	−0.39
Bottom-Up	31.3	−0.07	0.02	−0.08	−0.02
Inconsistency	29.8	−0.10	−0.12	−0.15	−0.14
FGIM-BERT	39.2	0.15	0.14	0.15	0.12
Gold	–	0.30	0.40	0.13	0.48
Bottom-up	–	−0.23	−0.07	−0.15	–
FGIM-BERT	–	0.10	0.03	0.05	–
FGIM (ϵ_n =0.3)	–	0.05	0.10	0.02	–
FGIM (ϵ_r =0.5)	–	0.07	−0.02	0.07	–

说明: 信息性、多样性、相关性和流畅性为人工评估的维度。

生成的摘要在问答方法中具有较高的得分, 是模型效果的上限。针对相同问题, 在所有基准模型中, FGIM-BERT 模型给出正确答案的比例最大。在标准排序的第一组排名中, 5 个系统同时进行排名, FGIM-BERT 系统生成摘要的效果更好。第二组排名中选取两个基于更新度和相关度的可控 FGIM 系统, 同时与 Bottom-up 和原始 FGIM-BERT 进行比较, 发现经过更新度控制后, 系统生成的摘要在多样性指标中表现更好, 而经过相关度控制后, 生成的摘要在与文章的相关性方面表现更好。

4 结论

本文提出一种基于细粒度可解释矩阵的模型 FGIM, 通过建立细粒度的可解释矩阵抽取重要句子, 引导摘要生成。进一步地, 模型利用可解释属性(句子更新度和句子与文章的相关性)来控制模型生成。为考虑句对的影响因素, 在训练抽取器时, 提出基于句对的优化目标。通过可解释的属性优化文章中句子分布, 并与生成器中的指针相结合。在两个通用数据集(CNN/DailyMail 和 NYT50)上的实验结果表明, 本文提出的模型均取得最优的模型效果。为了验证生成摘要所具备的新颖性和相关性的特点, 本文还人工构建两个测试集, 通过 ROUGE 值和人工评估的结果, 可以看到 FGIM 模型在可控生成能力上有显著的改进。

参考文献

[1] See A, Liu P J, Manning C D. Get to the point: Summarization with pointer-generator networks // Proceedings of the 55th Annual Meeting of the Associa-

tion for Computational Linguistics. Vancouver, 2017: 1073–1083

[2] Peters M E, Neumann M, Iyyer M, et al. Deep contextualized word representations [EB/OL]. (2018–03–22)[2020–10–10]. https://arxiv.org/pdf/1802.05365.pdf

[3] Radford A, Narasimhan K, Salimans T, et al. Improving language understanding by generative pre-training [EB/OL]. (2019–05–24)[2020–10–10]. https://s3-us-west-2.amazonaws.com/openai-assets/research-covers/language-unsupervised/language_understanding_paper.pdf

[4] Devlin J, Chang M W, Lee K, et al. BERT: pre-training of deep bidirectional transformers for language understanding // Proceedings of NAACL-HLT 2019. Minneapolis, 2019: 4171–4186

[5] Xu Hu, Liu Bing, Shu Lei, et al. BERT post-training for review reading comprehension and aspect-based sentiment analysis // Proceedings of NAACL-HLT 2019. Minneapolis, 2019: 2324–2335

[6] Liu Yang and Lapata M. Text summarization with pretrained encoders // Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing. Hong Kong, 2019: 3730–3740

[7] Zhang Xingxing, Wei Furu, Zhou Ming. HIBERT: Document level pre-training of hierarchical bidirectional transformers for document summarization // Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics. Florence, 2019: 5059–5069

[8] Nallapati R, Zhou B, Gulcehre C, et al. Abstractive text summarization using sequence-tosequence RNNs and beyond // Proceedings of the 20th SIGNLL Conference on Computational Natural Language Learning. Berlin, 2016: 280–290

[9] Zhou Qingyu, Yang Nan, Wei Furu, et al. Selective encoding for abstractive sentence summarization // Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics. Vancouver, 2017: 1095–1104

[10] Hsu W T, Lin C K, Lee M Y, et al. A unified model for extractive and abstractive summarization using inconsistency loss // Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics. Melboume, 2018: 132–141

[11] Gehrmann S, Deng Y, Rush A. Bottom-up abstractive summarization // Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing. Brussels, 2018: 4098–4109

[12] Tan Jiwei, Wan Xiaojun, Xiao Jianguo. Abstractive document summarization with a graphbased attentional neural model // Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics. Vancouver, 2017: 1171–1181

[13] Li Wei, Xiao Xinyan, Wang Yuanzhuo, et al. Improving neural abstractive document summarization with explicit information selection modeling // Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing. Brussels, 2018: 1787–1796

[14] You Yongjian, Jia Weijia, Liu Tianyi, et al. Improving abstractive document summarization with salient information modeling // Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics. Florence, 2019: 2132–2141

[15] Mihalcea R, Tarau P. Textrank: bringing order into text // Proceedings of the 2004 conference on empirical methods in natural language processing. Doha, 2014: 404–411

[16] Hermann K M, Kocisky T, Grefenstette E, et al. Teaching machines to read and comprehend // Advances in neural information processing systems. Montreal, 2015: 1693–1701

[17] Sandhaus E. The new york times annotated corpus // Linguistic Data Consortium. Philadelphia, 2008, 6(12): e26752

[18] Chen Y C, Bansal M. Fast abstractive summarization with reinforce-selected sentence rewriting // Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics. Melboume, 2018: 675–686

[19] Zheng Hao, Lapata M. Sentence centrality revisited for unsupervised summarization // Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics. Florence, 2019: 6236–6247

[20] Clarke J, Lapata M. Discourse constraints for document compression // Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics. Uppsala, 2010, 36(3): 411–441