计算机科学
COMPUTER SCIENCE

融合多粒度抽取式特征的关键词生成

甄田歌, 宋明阳, 景丽萍

引用本文

甄田歌, 宋明阳, 景丽萍.融合多粒度抽取式特征的关键词生成[J]. 计算机科学, 2023, 50(4): 181-187.

ZHEN Tiange, SONG Mingyang, JING Liping. Incorporating Multi-granularity Extractive Features for Keyphrase Generation [J]. Computer Science, 2023, 50(4): 181-187.

计算机科学 COMPUTER SCIENCE
http://www.jsjkx.com
DOI:10.11896/jsjkx.220700164

融合多粒度抽取式特征的关键词生成

甄田歌　宋明阳　景丽萍
北京交通大学计算机与信息技术学院　北京 100044
交通数据分析与挖掘北京市重点实验室(北京交通大学)　北京 100044
(20120457@bjtu.edu.cn)

摘　要　关键词是概括给定文本核心主题及关键内容的一组短语。在信息过载日益严重的今天,从给定的大量文本信息中预测出具有其中心思想的关键词至关重要。因此,关键词预测作为自然语言处理的基本任务之一,受到越来越多研究者的关注。其对应方法主要包括两类:关键词抽取和关键词生成。关键词抽取是从给定文本中快速、准确地抽取文中出现过的显著性短语作为关键词。与关键词抽取不同,关键词生成既能预测出现在给定文本中的关键词,也能预测未出现在给定文本中的关键词。总而言之,这两类方法各有优劣。然而,现有的关键词生成工作大多忽视了抽取式特征可能为关键词生成模型带来的潜在收益。抽取式特征能指明原文本的重要片段,对于模型学习原文本的深层语义表示起到重要作用。因此,结合抽取式和生成式方法的优势,提出了一种新的融合多粒度抽取式特征的关键词生成模型(incorporating Multi-Granularity Extractive features for keyphrase generation,MGE-Net)。在一系列公开数据集上的实验结果表明,和近年来的关键词生成模型相比,所提模型在大多数评价指标上取得了显著的性能提升。

关键词:自然语言处理;序列到序列;关键词生成;抽取式特征;多任务学习
中图法分类号　TP391

Incorporating Multi-granularity Extractive Features for Keyphrase Generation

ZHEN Tiange,SONG Mingyang and JING Liping
School of Computer and Information Technology,Beijing Jiaotong University,Beijing 100044,China
Beijing Key Lab of Traffic Data Analysis and Mining,Beijing Jiaotong University,Beijing 100044,China

Abstract　Keyphrase is a set of phrases that summarizes the core theme and key content of a given text. At present,information overload is becoming more and more serious,it is crucial to predict phrases with their central ideas for a given large amount of textual information. Therefore,keyphrase prediction,as one of the basic tasks of natural language processing,has received more and more attention from research scholars. Its corresponding methods mainly contain two categories,namely keyphrase extraction and keyphrase generation. Keyphrase extraction is the fast and accurate extraction of salient phrases that appear in the given text. Unlike keyphrase extraction,keyphrase generation predicts both phrases that appear in the given text and those do not appear in the given text. In summary,both have their advantages and disadvantages. However,most of the existing work on keyphrase generation has ignored the potential benefits that extractive features may bring to keyphrase generation models. Extractive features can indicate important fragments of the original text and play an important role for the model to learn the deep semantic representation of the original text. Therefore,combining the advantages of extractive and generative approaches,this paper proposes a new keyphrase generation model incorporating multi-granularity extractive features(MGE-Net). Compared with recent keyphrase generation models on a series of publicly available datasets,the proposed model achieves significant performance improvements in most evaluation metrics.

Keywords　Natural language processing,Sequence-to-Sequence,Keyphrase generation,Extractive features,Multi-task learning

关键词是一组能精确、全面地概括文本核心主题的短语。在当今的大数据时代,关键词对于快速地组织、理解文本数据具有非常重要的现实意义。关键词生成指依据给定文本,自动获取其关键词的一项任务。作为自然语言处理的基本任务之一,关键词生成被广泛应用于信息检索[1]、文本分类[2]、文本摘要[3]等领域。

到稿日期:2022-07-18　返修日期:2022-11-21
基金项目:国家自然科学基金(61822601,61773050,61632004);北京市自然科学基金(Z180006);北京市科委项目(Z181100008918012)
This work was supported by the National Natural Science Foundation of China(61822601,61773050,61632004),Natural Science Foundation of Beijing,China(Z180006) and Program of Beijing Municipal Science & Technology Commission(Z181100008918012).
通信作者:景丽萍(lpjing@bjtu.edu.cn)

如何自动预测给定文本中的关键词一直以来都受到许多学者的关注。早期研究的焦点主要在抽取式的方法上。传统的抽取式方法大多分为两步[4-5]，首先依据词性标记等特征从原文本中选择出一组候选词，然后对候选词一一进行重要性打分，在候选词中挑选出重要度分数较高的一组词语作为模型的预测结果。这些方法在预测原文本中出现过的关键词方面十分有效。然而，事实上，关键词并不绝对会出现在原文本中。如图1所示，“text summarization”等关键词在原文本中并没有出现，而这部分未出现在原文本中的关键词在数据集中并非个例，其在整个关键词集合的组成中占有较大的比重。未出现在原文本中的关键词往往代表了指定者对原文本的深层理解，而非简单地从原文本中挑选出重要的词语。因此，抽取式方法无法处理对未出现在原文本中关键词的预测。

原文本 Development, implementation, and a cognitive **evaluation** of a definitional **question answering** system for physicians. The published medical literature and online medical resources are important sources to help physicians make patient treatment decisions. Traditional sources used for **information retrieval** (e.g., pubmed) often return a list of documents in response to a users query. Frequently the number of returned documents from large knowledge repositories is large and makes information seeking practical only after hours and not in the clinical setting. This study developed novel algorithms, and designed, implemented, and evaluated a medical definitional question answering system (medqa). Medqa automatically analyzed a large number of electronic documents to generate short and coherent answers in response to definitional questions (i.e., questions with the format of what is x). Our preliminary cognitive evaluation shows that medqa out performed three other online information systems (google, onelook, and pubmed) in two important efficiency criteria namely, time spent and number of actions taken for a physician to identify a definition. It is our contention that question answering systems that aggregate pertinent information scattered across different documents have the potential to address clinical information needs within a timeframe necessary to meet the demands of clinicians.

出现在原文本中的关键词： **evaluation; question answering; information retrieval**

未出现在原文本中的关键词： **question analysis; text summarization; machine learning**

图1 数据集中的原文本及其关键词示例

Fig. 1 Example of original text and its keyphrase in dataset

为了解决上述问题，一些研究者借鉴机器翻译领域的相关工作[6-7]，提出了基于深度学习的关键词生成方法。生成式方法主要采用带有注意力的序列到序列框架，辅助以拷贝机制预测基于关键词构建的文本序列。这些方法类似于人为地概括关键词，即首先理解原文本的含义，然后依据对原文本内容的理解为其总结关键词。它们既能够预测出现在原文本中的关键词，也能够预测未出现在原文本中的关键词。而后，关键词生成方法取得了一系列进展，其有效性已经被许多研究者证实。

然而，大多数已有的生成式方法忽视了抽取式特征可能为关键词生成模型带来的潜在收益。抽取式特征能为关键词生成模型提供关于在原文本各个片段上重要性分布的信息，促进模型对原文本的理解，从而生成与原文本更相关的关键词。因此，本文提出了一种新的融合多粒度抽取式特征的关键词生成模型。该模型主要包括3部分：抽取式模块、生成式模块和特征交互层。其中，抽取式模块通过卷积神经网络获取局部语义特征，并以序列标注的方式协助模型学习抽取式特征；生成式模块利用以Transformer为主干的序列到序列框架预测基于关键词构建的目标序列；特征交互层引入互注意力机制，实现抽取式特征和生成式特征的融合，优化抽取式和生成式模块的表示学习。大量的实验证明，本文提出的模型性能优于近年来的关键词生成模型。

本文的主要贡献有以下4个方面：

(1)提出了一种端到端的多任务学习模型，联合学习关键词抽取和关键词生成。

(2)借助卷积神经网络获取多粒度局部语义特征，并利用该特征进行关键词抽取。

(3)利用特征交互层捕获抽取式特征(多粒度局部语义特征)和生成式特征之间的相互影响。

(4)在公开数据集上进行实验，评估模型原文本中出现过的关键词和未出现在原文本中的关键词两方面的预测能力。实验结果表明，本文模型优于近年来的关键词生成模型。

1 相关工作

1.1 关键词抽取

现有的关键词抽取方法可以分为两种，即两阶段方法[4-5]和序列标注法[8]。

两阶段方法指首先依据一些词性特征等启发式规则从原文本中挑选出一组候选关键词，然后再构建模型对候选关键词进行重要度打分并排序，从而确定出若干个关键词作为最终的预测结果。其中，重要度分数可以通过监督学习模型获得，如多层感知机[9]、支持向量机[10]等模型，也可以通过无监督学习模型获得，例如文献[11]提出了一种基于图的关键词排序算法。

序列标注法，也被称为关键词边界检测方法，指通过神经网络编码器对原文本进行编码获取其每个单词的上下文向量，然后依据这些上下文向量预测每个单词与关键词边界的关系。常用的序列标注模式有BIO(B-begin, I-inside, O-outside)和BIOES(B-begin, I-inside, O-outside, E-end, S-single)两种。借助这些序列标注模式，模型可以预测每个单词是关键词的开始、内部或外部等的概率，进而从原文本中抽取出关键词。

虽然抽取式方法在预测原文本中出现过的关键词方面取得了显著成效，但是这些方法无法预测未出现在原文本中的关键词。

1.2 关键词生成

在自然语言处理领域中，自动关键词预测还有很多亟待解决的问题。为了解决抽取式方法所存在的问题，文献[12]首次提出了用带有注意力机制的序列到序列框架，辅助以拷贝机制进行关键词生成。关键词生成模型既可以预测出现在原文本中的关键词，也能够预测未出现在原文本中的关键词。

自关键词生成模型被首次提出以来，研究者们沿用文献[12]的主体框架提出了许多后续工作。例如，文献[13]提出以基于关键词构建的文本序列而非单个关键词作为目标序列生成，解决了模型无法为不同原文本生成不同数量关键词的问题；有学者针对关键词生成模型中存在的预测结果冗余问题，提出了覆盖机制等方法[14-16]；还有一些学者在关键词生成模型中尝试利用标题等的额外信息[17-18]；此外，还有值得关注的一些工作[19-21]，如文献[20]提出了一种自适应奖赏函数来利用强化学习方法解决关键词生成数量不足的问题；文献[22]通过分层地解码对关键词集的分层组合性进行显式建模。

然而,这些方法忽视了抽取式特征可能为关键词生成模型带来的潜在收益。文献[23]提出了一种选择、抽取、生成一体化的模型,但该方法的关键点在于长文档的处理以及抽取任务和生成任务的简单结合。本文着眼于探索一种融合多粒度抽取式特征的关键词生成方法,以优化抽取式特征和生成式特征的学习,并显式地建模两者的相互影响。

2 研究方法

2.1 问题定义

关键词生成就是要解决为给定原文档自动生成一组关键词的问题。每篇原文档对应了若干个关键词组成的集合,单个元组〈原文档,关键词集〉可以被看作是一个数据样例。在本文中,原文档用 x 表示,与之对应的关键词集用 $K=\{k^1,k^2,\cdots,k^{|K|}\}$ 表示,$|K|$ 是原文档所拥有的关键词个数。原文档和单个关键词都是单词的序列,即 $x=x_1,x_2,\cdots,x_{|x|}$,$k^i=k^i_1,k^i_2,\cdots,k^i_{|k^i|}$。其中,$|x|$ 和 $|k^i|$ 分别代表原文档和第 i 个关键词 k^i 的单词个数。本文使用 $K^p=\{k^{p,1},k^{p,2},\cdots,k^{p,|K^p|}\}$ 和 $K^a=\{k^{a,1},k^{a,2},\cdots,k^{a,|K^a|}\}$ 分别表示出现在原文本中的关键词集和未出现在原文本中的关键词集。容易得知,$K=K^p\cup K^a$。

为了适用序列到序列框架,数据样例需要转换为〈原文本序列,目标文本序列〉的形式。因此,本文采取了文献[20]的处理方法,这也是目前大多数关键词生成工作常用的一种处理方法:以特定分隔符⟩将所有关键词连接起来构建目标文本序列。连接顺序为出现在原文本中的关键词在前,未出现在原文本中的关键词在后。对于出现在原文本中的关键词,连接会按照每个关键词在原文本中首次出现的位置先后顺序进行;对于未出现在原文本中的关键词,连接会按照数据集中原有关键词的顺序进行。此外,文献[20]还提出使用特殊符号◇标记出现在原文本中的关键词的结尾。也就是说,目标文本序列 $Y=k^{p,1}\rangle k^{p,2}\rangle\cdots\rangle k^{p,|K^p|}\diamond k^{a,1}\rangle k^{a,2}\rangle\cdots\rangle k^{a,|K^a|}$。

2.2 融合多粒度抽取式特征的关键词生成模型

序列到序列框架主要包含序列编码器和序列解码器两部分。本文采用 Transformer 作为序列到序列框架的主干,即序列编码器和序列解码器均采用 Transformer 实现。模型的整体框架如图 2 所示。

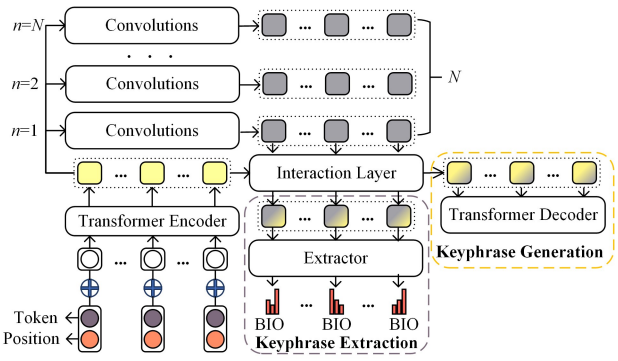

图 2 模型整体框架

Fig. 2 Overall framework of our model(MGE-Net)

模型图左侧自上而下是卷积特征抽取器和 Tansformer 编码器,分别用于获取关于原文本的抽取式特征和生成式特征;模型图右侧对应关键词抽取和关键词生成任务的抽取器和解码器;模型图中间连接两者的是特征交互层,用于建模抽取式特征和生成式特征之间的相互影响。

2.2.1 Transformer 编码器

Transformer 编码器由 L 层编码层组成,以嵌入向量序列为输入,并利用多头自注意力机制产生上下文向量。原文本序列中的每个单词通过嵌入层被映射为低维的嵌入向量,再被输入到 Transformer 编码器中。具体地,在第 t 个单词处,Transformer 编码器所输入的嵌入向量为:

$$\boldsymbol{e}_t=\boldsymbol{e}_t^{\mathrm{w}}+\boldsymbol{e}_t^{\mathrm{pos}} \tag{1}$$

其中,$\boldsymbol{e}_t^{\mathrm{w}}$ 是词嵌入向量;$\boldsymbol{e}_t^{\mathrm{pos}}$ 是位置嵌入向量。嵌入层将原文本中的每个单词及其相应的绝对位置转换为固定维度的向量表示后,再令两者按元素相加形成最终的编码器输入。编码器为原文本序列产生的上下文向量序列记为$\{[\boldsymbol{h}_1^l,\boldsymbol{h}_2^l,\cdots,\boldsymbol{h}_m^l],l=1,\cdots,L\}$,其中 m 是原文本序列的总长度。在经典序列到序列框架的注意力机制中,编码器最后一层(L 层)输出的上下文向量序列会被解码器用于参考并生成关键词,本文将其称为生成式特征,记为 $\boldsymbol{H}=[\boldsymbol{h}_1^L,\boldsymbol{h}_2^L,\cdots,\boldsymbol{h}_m^L]$。

2.2.2 多粒度卷积特征抽取器

为了强化局部语义特征,本文采用一维卷积产生初步的抽取式特征。一维卷积常被用于获取文本序列的局部特征。在本文中,编码器输出的上下文向量序列通过一维卷积得到多粒度抽取式特征。本文保留卷积核大小为 $n=1,\cdots,N$ 所得到的共 N 组抽取式特征,并辅助以适当的填充保持输入、输出长度一致。其中,N 是预定义的超参数。这些特征被记为 $\boldsymbol{S}=\{[\boldsymbol{s}_1^n,\boldsymbol{s}_2^n,\cdots,\boldsymbol{s}_m^n],n=1,\cdots,N\}$

2.2.3 特征交互层

为了显式地建模生成式特征和抽取式特征的相互影响和交互,本文采用特征交互层捕捉两者的相关性。首先,生成式特征 $\boldsymbol{H}=[\boldsymbol{h}_1^L,\boldsymbol{h}_2^L,\cdots,\boldsymbol{h}_m^L]$ 和抽取式特征 $\boldsymbol{S}=\{[\boldsymbol{s}_1^n,\boldsymbol{s}_2^n,\cdots,\boldsymbol{s}_m^n],n=1,\cdots,N\}$ 通过带有 ReLU 激活函数的线性转换,使得两者更具区分度。

$$\widetilde{\boldsymbol{H}}=LN(\boldsymbol{H}+\max(0,\boldsymbol{W}_{\boldsymbol{H}}\boldsymbol{H}+\boldsymbol{b}_{\boldsymbol{H}})) \tag{2}$$

$$\widetilde{\boldsymbol{S}}^n=LN(\boldsymbol{S}^n+\max(0,\boldsymbol{W}_{\boldsymbol{S}^n}\boldsymbol{S}^n+\boldsymbol{b}_{\boldsymbol{S}^n})),n=1,\cdots,N \tag{3}$$

其中,LN 指的是层归一化函数[24],$\boldsymbol{W}_{\boldsymbol{H}}$,$\boldsymbol{b}_{\boldsymbol{H}}$,$\boldsymbol{W}_{\boldsymbol{S}^n}$ 和 $\boldsymbol{b}_{\boldsymbol{S}^n}$ 都是可学习的参数。

然后,本文采用互注意力机制建模两种特征间的相互影响。值得注意的是,抽取式特征共 N 组,经过互注意力机制,模型最终可以得到相互学习的 N 组生成式特征和 N 组抽取式特征。

$$\boldsymbol{H}^n=LN(\widetilde{\boldsymbol{H}}+\mathrm{softmax}(\widetilde{\boldsymbol{H}}(\widetilde{\boldsymbol{S}}^n)^{\mathrm{T}})\widetilde{\boldsymbol{S}}^n),n=1,\cdots,N \tag{4}$$

$$\boldsymbol{S}^n=LN(\widetilde{\boldsymbol{S}}^n+\mathrm{softmax}(\widetilde{\boldsymbol{S}}^n(\widetilde{\boldsymbol{H}})^{\mathrm{T}})\widetilde{\boldsymbol{H}}),n=1,\cdots,N \tag{5}$$

最后,本文分别将生成式和抽取式的 N 组特征拼接起来,通过线性变换整合得到最终的等长、定维特征表示。为了方便起见,本文仍然将其记为 $\boldsymbol{H}=[\boldsymbol{h}_1,\boldsymbol{h}_2,\cdots,\boldsymbol{h}_m]$ 和 $\boldsymbol{S}=[\boldsymbol{s}_1,\boldsymbol{s}_2,\cdots,\boldsymbol{s}_m]$。

$$\boldsymbol{H}=\boldsymbol{W}_{IH}[\boldsymbol{H}^1;\boldsymbol{H}^2;\cdots;\boldsymbol{H}^N]+\boldsymbol{b}_{IH} \tag{6}$$

$$\boldsymbol{S}=\boldsymbol{W}_{IS}[\boldsymbol{S}^1;\boldsymbol{S}^2;\cdots;\boldsymbol{S}^N]+\boldsymbol{b}_{IS} \tag{7}$$

其中，; 表示张量拼接操作，$\boldsymbol{W}_{IH}$，$\boldsymbol{b}_{IH}$，$\boldsymbol{W}_{IS}$ 和 $\boldsymbol{b}_{IS}$ 都是可学习的参数。

2.2.4 抽取器

本文使用序列标注法进行关键词抽取。抽取器以抽取式特征 $\boldsymbol{S}=[\boldsymbol{s}_1,\boldsymbol{s}_2,\cdots,\boldsymbol{s}_m]$ 为输入，为每个单词预测其与出现在原文本中的关键词边界的关系及相应概率。本文选择 BIO 序列标注模式，因此，抽取器本质上相当于一个三路分类器。

$$\boldsymbol{P}_i=\text{softmax}(\boldsymbol{W}_2(\tanh(\boldsymbol{W}_1\boldsymbol{s}_i+\boldsymbol{b}_1))+\boldsymbol{b}_2) \tag{8}$$

其中，$\boldsymbol{W}_1$，$\boldsymbol{b}_1$，$\boldsymbol{W}_2$ 和 $\boldsymbol{b}_2$ 都是可学习的参数。

2.2.5 Transformer 解码器

解码器以基于关键词构建的文本序列为目标序列预测关键词。Transformer 解码器同样由 L 层解码层组成。特别地，解码器的最后一层（L 层）输出 $[\boldsymbol{c}_1^L,\boldsymbol{c}_2^L,\cdots,\boldsymbol{c}_p^L]$，其中 p 是生成序列的总长度，通过 softmax 层预测生成序列中的第 t 个单词 y_t 在词汇表 V 上的概率分布 $\boldsymbol{P}_{\text{vocab}}(y_t)$。

$$\boldsymbol{P}_{\text{vocab}}(y_t)=\text{softmax}(\boldsymbol{W}_v\boldsymbol{c}_t^L+\boldsymbol{b}_v) \tag{9}$$

其中，$\boldsymbol{W}_v$ 和 $\boldsymbol{b}_v$ 是可学习的参数。

2.2.6 拷贝机制

在实践中，词汇表 V 通常选用固定大小的、高频出现的单词表。实际上，目标序列中的单词有不在词汇表 V 中的可能性。而且，未出现在原文本中的部分关键词（短语）也有可能会选用原文本中的单词构成。为了减轻这种现象带来的影响，本文采用拷贝机制使得生成序列可以选择拷贝原文本中的单词。

具体地，在预测生成序列中的第 t 个单词时，解码器的最后一层输出 $\boldsymbol{c}_t^L$ 和生成式特征 $\boldsymbol{H}=[\boldsymbol{h}_1,\boldsymbol{h}_2,\cdots,\boldsymbol{h}_m]$ 计算注意力分数 $[\alpha_{t1},\alpha_{t2},\cdots,\alpha_{tm}]$ 和上下文语境向量 $\boldsymbol{u}_t$。

$$a(\boldsymbol{c}_t^L,\boldsymbol{h}_i)=\boldsymbol{h}_i\boldsymbol{W}_a\boldsymbol{c}_t^L \tag{10}$$

$$\alpha_{ti}=\frac{\exp(a(\boldsymbol{c}_t^L,\boldsymbol{h}_i))}{\sum_{j=1}^{m}\exp(a(\boldsymbol{c}_t^L,\boldsymbol{h}_j))} \tag{11}$$

$$\boldsymbol{u}_t=\sum_{i=1}^{m}\alpha_{ti}\boldsymbol{h}_i \tag{12}$$

在拷贝机制中，注意力分数 $[\alpha_{t1},\alpha_{t2},\cdots,\alpha_{tm}]$ 被用于作为生成序列中的第 t 个单词 y_t 在原文本词汇上的概率分布 $\boldsymbol{P}_c(y_t)$，使得模型能够从原文本中拷贝单词；而上下文语境向量 $\boldsymbol{u}_t$ 被用于计算从原文本中拷贝（p_c^t）或从词汇表 V 中生成的概率（p_g^t）。

$$\boldsymbol{P}_c(y_t)=[\alpha_{t1},\alpha_{t2},\cdots,\alpha_{tm}] \tag{13}$$

$$p_c^t=\text{sigmoid}(\boldsymbol{W}_c[\boldsymbol{c}_t^L;\boldsymbol{u}_t]+\boldsymbol{b}_c) \tag{14}$$

$$p_g^t=1-p_c^t \tag{15}$$

其中，$\boldsymbol{W}_c$ 和 $\boldsymbol{b}_c$ 是可学习的参数。最终，生成序列中的第 t 个单词 y_t 的概率分布 $\boldsymbol{P}_{\text{final}}(y_t)$ 被预测为：

$$\boldsymbol{P}_{\text{final}}(y_t)=p_c^t\boldsymbol{P}_c(y_t)+p_g^t\boldsymbol{P}_{\text{vocab}}(y_t) \tag{16}$$

2.2.7 训练

本文模型的损失函数包括抽取任务和生成任务两部分的损失。抽取器实质上相当于三路分类器，因此本文使用加权的交叉熵损失函数作为抽取部分的损失。

$$\mathcal{L}_e=-\sum_{i=1}^{|x|}\sum_{c=1}^{C}w\hat{x}_i^c\log(p_i^c) \tag{17}$$

其中，$|x|$ 是原文档的单词个数；C 是标签数量（BIO 标注模式下 $C=3$）；w 是预定义的超参数，其含义是正标签的损失权重；$\hat{x}_i^c$ 是原文档第 i 个单词的真实标签值；p_i^c 是抽取器所预测的原文档第 i 个单词是标签 c 的概率值。

生成器（解码器）以关键词序列为目标生成关键词。本文使用真实关键词（$\hat{y}$）的负对数似然函数作为生成部分的损失。

$$\mathcal{L}_g=-\sum_{t=1}^{T}\log\boldsymbol{P}_{\text{final}}(\hat{y}_t) \tag{18}$$

其中，T 是目标序列的总长度。模型的整体损失被定义为抽取部分损失和生成部分损失的加权和。

$$\mathcal{L}=\beta\mathcal{L}_e+(1-\beta)\mathcal{L}_g \tag{19}$$

其中，β 是损失平衡权重超参数。

表 1 测试数据集的统计数据

Table 1 Statistics of test dataset

数据集	样本总数	原文本平均长度	关键词平均长度	关键词平均数量	出现在原文本中的关键词占比/%	未出现在原文本中的关键词占比/%
Inspec	500	128.7	2.48	9.83	73.6	26.4
Krapivin	400	182.6	2.21	5.84	55.7	44.3
NUS	211	219.1	2.22	11.65	54.4	45.6
KP20k	20000	179.8	2.04	5.28	62.9	37.1

3 实验内容

3.1 数据集及预处理

本文选择关键词生成领域中广泛使用的 4 个科技文献数据集进行实验。这 4 个测试数据集分别是 KP20k[12]，Inspec[25]，Krapivin[26] 和 NUS[27]。测试数据集的统计数据如表 1 所列。和先前工作中的处理方式一致，本文将数据样本中的标题和摘要连接起来作为原文本；同时，本文使用最大的关键词生成数据集 KP20k 作为模型的训练集。预处理操作包括降小写、把所有数字符号替换为〈digit〉、去除重复数据等。KP20k 训练数据集共包含 530 802 个数据样本，另有验证数据集共包含 20000 个数据样本。

3.2 实验设置

本文的词汇表 V 由训练数据集中出现频率最高的 50000 个单词及一些特殊符号（如结尾符号、未知单词符号等）组成。嵌入层维度设置为 512 维。Transformer 结构参数为 6 层（即 $L=6$），8 个自注意力头，隐藏状态维度设置为 2048 维。所有可学习的训练参数（包含嵌入层在内）被随机初始化为 $[-0.1,0.1]$ 上的均匀分布。本文使用 Adam 优化算法[28]，初始学习率为 10^{-4}，批大小为 20，dropout 率设置为 0.1。为了稳定训练过程，将梯度裁剪的最大阈值设置为 1，以避免梯度爆炸。本文的模型训练采用提前停止策略：每当验证集上

的损失未下降时,学习率会降低一半;当连续若干次迭代的验证损失都未下降时,即认为模型在验证数据集上达到收敛,停止训练。若提前停止策略未生效,模型至多训练 20 个轮回。在测试过程中,本文使用贪心搜索算法作为解码生成序列时的算法。本文为正标签的损失权重超参数 w 在[0.7,0.8,0.9,1.0,1.5,2.0,2.5]中进行网格搜索,最终发现 $w=2.0$ 时能达到最佳性能。多粒度特征抽取器中的特征组数 N 设置为 3。为了平衡抽取部分和生成部分的损失,将两部分的损失分别按词数进行归一化操作,并且根据经验将损失平衡权重超参数 β 设置为 0.5。

3.3 基准方法及评价指标

本文选择一组近年来的最优关键词生成方法作为基准方法,同本文提出的模型进行比较。此外,本文还训练了以 Transformer 为主干的序列到序列模型作为基准方法之一。基准方法分别为:catSeq[13],catSeqCorr[14],catSeqTG[18],ExHiRD-s[22],ExHiRD-h[22],SEG-Net[23]和 Transformer。

这些基准方法主要采取两种解码方式,即每次生成单个关键词并使用集束搜索算法生成多个关键词,和每次以贪心的方式直接生成基于关键词构建的文本序列。文献[13]指出前者的缺点在于模型无法自动决定生成关键词的数量。为了避免不同的解码策略对实验结果造成的影响,文献[20]对上述基准方法进行了统一,使其均以基于关键词构建的文本序列作为目标输出。其中,catSeqCorr 和 catSeqTG 在其相应的原文献中分别被称为 CorrRNN 和 TG-Net。

本文按照文献[20]的处理方式,并选择宏平均的 F1@5 和 F1@M 作为评价指标。其中,F1@5 是通过比较前 5 个预测关键词和真实关键词计算出的F1分数;F1@M是通过比较所有的预测关键词和真实关键词计算出的 F1 分数。

3.4 整体性能

为了研究多粒度特征组数 N 对模型的影响,我们分别在 $N=3$,$N=2$ 和 $N=1$ 的条件下训练了本文提出的模型(MGE-Net),并列出了相应的实验结果。实验结果分为出现在原文本中的关键词和未出现在原文本中的关键词两部分。

MGE-Net 与其他基准方法在出现在原文本中的关键词上的评价指标分数如表 2 所列。其中,MGE-Net 中的最优结果用粗体表示,基准方法中的最优结果用下划线表示。可以观察到,MGE-Net 在多粒度特征组数 $N=3$,$N=2$ 和 $N=1$ 这 3 种情况下,大多数测试数据集上的评价指标分数都明显优于其他基准方法。具体而言,相比基准方法中的最优结果,MGE-Net 在 Inspec,Krapivin 和 KP20k 数据集上的 F1@5 分数分别能达到 12.5%,6.2% 和 6.1% 的性能提升。但在 NUS 数据集上,MGE-Net 的评价指标结果不如现有的基准方法 SEG-Net。我们分析造成这种结果的原因可能是不同数据集中出现在原文本中的关键词和未出现在原文本中的关键词的占比情况不同。从表 1 中的测试集统计数据可以看出,数据集中出现在原文本中的关键词占比越高,模型的提升效果越明显。这说明了多粒度抽取式特征能指明原文本中的重要片段,对于模型生成出现在原文本中的关键词起重要作用。此外,另一个可能的原因是 NUS 数据集上的原文本平均长度较大,如表 1 所列,而 SEG-Net 模型[23]设计了句子选择器来处理长文档,但这也导致了 SEG-Net 不得不采取非端到端的学习方式,使其模型变得较为复杂。

表 2 出现在原文本中的关键词上的实验结果

Table 2 Results of present keyphrase prediction

Method	Inspec		Krapivin		NUS		KP20k	
	F1@M	*F1@5*	*F1@M*	*F1@5*	*F1@M*	*F1@5*	*F1@M*	*F1@5*
catSeq	0.262	0.225	0.354	0.269	0.397	0.323	0.367	0.291
catSeqCorr	0.269	0.227	0.349	0.265	0.390	0.319	0.365	0.289
catSeqTG	0.270	0.229	0.366	0.282	0.393	0.325	0.366	0.292
ExHiRD-s	0.278	0.235	0.338	0.278	—	—	0.372	0.307
ExHiRD-h	0.291	0.253	0.347	0.286	—	—	0.374	0.311
SEG-Net	0.265	0.216	0.366	0.276	0.461	0.396	0.379	0.311
Transformer	0.296	0.255	0.356	0.290	0.416	0.355	0.370	0.310
MGE-Net($N=3$)	0.319	0.274	**0.366**	0.303	0.431	0.377	0.381	0.330
MGE-Net($N=2$)	**0.330**	**0.287**	0.355	**0.308**	**0.438**	**0.378**	0.381	0.329
MGE-Net($N=1$)	0.320	0.270	0.359	0.293	0.414	0.361	**0.383**	**0.330**

MGE-Net 与其他基准方法在未出现在原文本中的关键词上的评价指标分数如表 3 所列。

表 3 未出现在原文本中的关键词上的实验结果

Table 3 Results of absent keyphrase prediction

Method	Inspec		Krapivin		NUS		KP20k	
	F1@M	*F1@5*	*F1@M*	*F1@5*	*F1@M*	*F1@5*	*F1@M*	*F1@5*
catSeq	0.008	0.004	0.036	0.018	0.028	0.016	0.032	0.015
catSeqCorr	0.009	0.005	0.038	0.020	0.024	0.014	0.032	0.015
catSeqTG	0.011	0.005	0.034	0.018	0.018	0.011	0.032	0.015
ExHiRD-s	0.021	0.009	0.033	0.016	—	—	0.029	0.014
ExHiRD-h	0.022	0.011	0.043	0.022	—	—	0.032	0.016
SEG-Net	0.015	0.009	0.036	0.018	0.036	0.021	0.036	0.018
Transformer	0.014	0.007	0.046	0.024	0.034	0.020	0.036	0.018
MGE-Net($N=3$)	0.019	0.010	**0.068**	**0.035**	0.039	0.021	**0.047**	**0.023**
MGE-Net($N=2$)	0.020	0.010	0.055	0.029	**0.053**	**0.030**	0.043	0.020
MGE-Net($N=1$)	**0.023**	**0.012**	0.057	0.030	0.036	0.021	0.044	0.021

从表 3 可知，MGE-Net 中的最优结果用粗体表示，基准方法中的最优结果用下划线表示。从整体上看，MGE-Net 优于已有的基准方法，而且相比基准方法中的最优结果，MGE-Net 在 Krapivin 数据集上的 F1@5 分数能达到 45.8% 的性能提升。我们分析可能的原因是：模型在生成未出现在原文本中的关键词时，能够从词库中生成或从原文本中部分地拷贝一些片段。当未出现在原文本中的关键词与原文本存在重叠词语时，多粒度局部语义特征能使得相同词语的上下文语境被区分开，从而降低结果的冗余度并减少错误的拷贝。

3.5 案例分析

为了进一步研究抽取式特征所发挥的作用，我们给出了测试数据集中一个样例的预测结果，如图 3 所示。其中，出现在原文本中的关键词和未出现在原文本中的关键词在目标关键词和预测结果（经英文词干提取算法处理后进行匹配）中都分别用蓝色加粗字体和红色加粗字体表示。此外，出现在原文本中的关键词在原文本中用粗体表示；未出现在原文本中的关键词与原文本中的部分重叠片段在原文本中用下划线表示。

在图 3 中，我们可以观察到，相比 Transformer 基准方法的预测结果，MGE-Net 成功预测出了“image restoration”“stereo”和“segmentation”等出现在原文本中的关键词。而且我们可以注意到，由于融合了多粒度局部语义特征，MGE-Net 成功预测了“minimum cut”和“maximum flow”两个关键词，而非 Transformer 基准方法预测的“min cut max flow algorithms”。

输入的原文本 An experimental comparison of min cut max flow algorithms for energy minimization in vision. **Minimum cut/maximum flow** algorithms on graphs have emerged as an increasingly useful tool for exactor approximate energy minimization in low-level vision. The combinatorial optimization literature provides many min-cut/max-flow algorithms with different polynomial time complexity. Their practical efficiency, however, has to date been studied mainly outside the scope of computer vision. The goal of this paper is to provide an experimental comparison of the efficiency of min-cut/max flow algorithms for applications in vision. We compare the running times of several standard algorithms, as well as a new algorithm that we have recently developed. The algorithms we study include both Goldberg-Tarjan style "push -relabel" methods and algorithms based on Ford-Fulkerson style "augmenting paths." We benchmark these algorithms on a number of typical graphs in the contexts of **image restoration**, **stereo**, and **segmentation**. In many cases, our new algorithm works several times faster than any of the other methods, making near real-time performance possible. An implementation of our max-flow/min-cut algorithm is available upon request for research purposes.

输出的目标关键词
文中出现过的关键词：
minimum cut; **maximum flow**; **image restoration**; **stereo**; **segmentation**
文中未出现过的关键词：
graph algorithms; **multicamera scene reconstruction**; **index terms energy minimization**

Transformer: min cut max flow algorithms; energy minimization; combinatorial optimization; **graph algorithms**

MGE-Net: **min cut**; **max flow**; energy minimization; vision; **image restoration**; **stereo**; **segmentation**; **graph algorithms**

图 3 测试数据集中一个样例的预测结果（电子版为彩图）

Fig. 3 Prediction results of a sample in test dataset

结束语 本文探索了在关键词生成模型上多粒度抽取式特征的应用及融合方式，以 Transformer 为序列到序列框架的主干，融合多粒度局部语义特征，并显式地建模特征之间的相互影响。在公开数据集上的实验结果表明，在预测出现在原文本中的关键词和未出现在原文本中的关键词两方面上，本文提出的 MGE-Net 模型均优于近年来的最优基准方法。

尽管本文提出的模型在一定程度上提升了关键词生成的性能，但是关键词生成任务仍然是一个棘手的待解决问题，我们相信将来会有更多的相关研究。未来一个可能的方向是为关键词生成模型注入事实知识信息，以辅助未出现在原文本中的关键词的生成。另一个值得关注的方向是探索复杂模型的性能来源及模型的简化。

参 考 文 献

[1] JONES S, STAVELEY M S. Phrasier: a system for interactive document retrieval using keyphrases[C]// Proceedings of the 22nd Annual International ACM SIGIR Conference on Research and Development in Information Retrieval. New York: ACM, 1999: 160-167.

[2] HULTH A, MEGYESI B. A study on automatically extracted keywords in text categorization[C]// Proceedings of the 21st International Conference on Computational Linguistics and 44th Annual Meeting of the Association for Computational Linguistics. Stroudsburg, PA: ACL, 2006: 537-544.

[3] QAZVINIAN V, RADEV D, ÖZGÜR A. Citation summarization through keyphrase extraction[C]// Proceedings of the 23rd International Conference on Computational Linguistics (COLING 2010). New York: ACM, 2010: 895-903.

[4] TOMOKIYO T, HURST M. A language model approach to keyphrase extraction[C]// Proceedings of the ACL 2003 Workshop on Multiword Expressions: Analysis, Acquisition and Treatment. Stroudsburg, PA: ACL, 2003: 33-40.

[5] LIU Z Y, LI P, ZHENG Y B, et al. Clustering to find exemplar terms for keyphrase extraction[C]// Proceedings of the 2009 Conference on Empirical Methods in Natural Language Processing. Stroudsburg, PA: ACL, 2009: 257-266.

[6] CHO K, VAN M B, GULCEHRE C, et al. Learning phrase representations using RNN encoder-decoder for statistical machine translation[J]. arXiv: 1406.1078, 2014.

[7] SUTSKEVER I, VINYALS O, LE Q V. Sequence to sequence learning with neural networks[C]// Proceedings of the 27th International Conference on Neural Information Processing Systems-Volume 2. Cambridge, MA: MIT Press, 2014: 3104-3112.

[8] ALZAIDY R, CARAGEA C, GILES C L. Bi-LSTM-CRF sequence labeling for keyphrase extraction from scholarly documents[C]// The World Wide Web Conference. New York: ACM, 2019: 2551-2557.

[9] BASALDELLA M, CHIARADIA G, TASSO C. Evaluating anaphora and coreference resolution to improve automatic keyphrase extraction[C]// Proceedings of COLING 2016, the 26th International Conference on Computational Linguistics: Technical Papers. New York: ACM, 2016: 804-814.

[10] LOPEZ P, ROMARY L. HUMB: Automatickey term extraction from scientific articles in GROBID[C]// Proceedings of the 5th International Workshop on Semantic Evaluation. Stroudsburg, PA: ACL, 2010: 248-251.

[11] MIHALCEA R, TARAU P. Textrank: Bringing order into text [C]// Proceedings of the 2004 Conference on Empirical Methods in Natural Language Processing. Stroudsburg, PA: ACL, 2004:

404-411.

[12] MENG R,ZHAO S Q,HAN S G,et al. Deep keyphrase generation[C]// Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics(Volume 1:Long Papers). Stroudsburg,PA:ACL,2017:582-592.

[13] YUAN X D,WANG T,MENG R,et al. Onesize does not fit all: Generating and evaluating variable number of keyphrases[C]// Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics. Stroudsburg,PA:ACL,2020:7961-7975.

[14] CHEN J,ZHANG X M,WU Y,et al. Keyphrase generation with correlation constraints[C]// Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing. Stroudsburg,PA:ACL,2018:4057-4066.

[15] ZHAO J,ZHANG Y X. Incorporating linguistic constraints into keyphrase generation[C]// Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics. Stroudsburg,PA:ACL,2019:5224-5233.

[16] BAHULEYAN H,EL A L. Diverse keyphrase generation with neural unlikelihood training[C]// Proceedings of the 28th International Conference on Computational Linguistics. New York:ACM,2020:5271-5287.

[17] YE H,WANG L. Semi-supervised learning for neural keyphrase generation[C]// Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing. Stroudsburg,PA:ACL,2018:4142-4153.

[18] CHEN W,GAO Y F,ZHANG J N,et al. Title-guided encoding for keyphrase generation[C]// Proceedings of the AAAI Conference on Artificial Intelligence. Menlo Park,CA:AAAI,2019:6268-6275.

[19] ZHANG Y,FANG Y,XIAO W D. Deep keyphrase generation with a convolutional sequence to sequence model[C]// 2017 4th International Conference on Systems and Informatics(ICSAI). Piscataway,NJ:IEEE,2017:1477-1485.

[20] CHAN H P,CHEN W,WANG L,et al. Neural keyphrase generation via reinforcement learning with adaptive rewards[C]// Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics. Stroudsburg,PA:ACL,2019:2163-2174.

[21] WANG Y,LI J,CHAN H P,et al. Topic-aware neural keyphrase generation for social media language[C]// Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics. Stroudsburg,PA:ACL,2019:2516-2526.

[22] CHEN W,CHAN H P,LI P J,et al. Exclusive hierarchical decoding for deep keyphrase generation[C]// Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics. Stroudsburg,PA:ACL,2020:1095-1105.

[23] AHMAD W,BAI X,LEE S,et al. Select,extract and generate: Neural keyphrase generation with layer-wise coverage attention[C]// Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Confernece on Natural Language Processing(Volume 1: Long Papers). Stroudsburg,PA:ACL,2021:1389-1404.

[24] BA J L,KIROS J R,HINTON G E. Layer normalization[J]. arXiv:1607.06450,2016.

[25] HULTH A. Improved automatic keyword extraction given more linguistic knowledge[C]// Proceedings of the 2003 Conference on Empirical Methods in Natural Language Processing. Stroudsburg,PA:ACL,2003:216-223.

[26] KRAPIVIN M,AUTAEU A,MARCHESE M. Large dataset for keyphrases extraction[R]. Trento,Italy:Information Engineering and Computer Science Department of Trento University,2009.

[27] NGUYEN T D,KAN M Y. Keyphrase extraction in scientific publications[C]// Proceedings of the 10th International Conference on Asian Digital Libraries. Berlin: Springer, 2007: 317-326.

[28] KINGMA D P,BA J. Adam:A method for stochastic optimization[J]. arXiv:1412.6980,2014.

ZHEN Tiange, born in 1997, bachelor. Her main research interests include machine learning and natural language processing.

JING Liping, born in 1978, Ph.D, professor, Ph.D supervisor, is a senior member of China Computer Federation. Her main research interests include machine learning and its applications.

(责任编辑:何杨)