基于词项-句子-文档三层图模型的多文档自动摘要*

熊娇，王明文，李茂西，万剑怡

（江西师范大学 计算机信息工程学院，江西 南昌 330022）

摘要：应用图模型来研究多文档自动摘要是当前研究的一个热点，它以句子为顶点，以句子之间相似度为边的权重构造无向图结构。由于此模型没有充分考虑句子中的词项权重信息以及句子所属的文档信息，针对这个问题，本文提出了一种基于词项-句子-文档的三层图模型，该模型可充分利用句子中的词项权重信息以及句子所属的文档信息来计算句子相似度。在 DUC'2003 和 DUC'2004 数据集上的实验结果表明，基于词项-句子-文档三层图模型的方法优于 LexRank 模型和文档敏感图模型。

关键词：图模型；多文档自动摘要；句子相似度；词项-句子-文档图

中图分类号：TP391　　**文献标识码：**A

Multi-Document Summarization Based on the Term-Sentence-Document Graph Model

Xiong Jiao, Wang Mingwen, Li Maoxi,Wan Jianyi

(School of Computer Information Engineering, Jiangxi Normal University ,Nanchang, Jiangxi 330022 ,China)

Abstract: Graph model has been widely applied to document summarization by using sentence as the graph nodes, and the similarity between sentences as the weights of edge. However, the knowledge of information between terms and the information between the documents are neglected in this model. In this paper, we proposes a model called Term-Sentence-Document graph model for the problem which can make full use of knowledge when computing the similarity of sentences. The experimental results on the data sets of DUC'2003 and DUC'2004 show that the proposed model outperforms the state-of-the-art LexRank model and Document Sensitive Ranking model.

Key words: graph model; multi-document summarization; the similarity of sentences; term-sentence-document graph

1. 引言

多文档自动摘要通过给人们提供简洁全面的多文档信息来提高人们获取信息的效率。多文档自动摘要的主要方法分为两种：抽取式摘要和生成式摘要。前者主要从多篇原始文档中抽取出一些重要的句子来组成最后的摘要，后者则需要计算机在理解原文的基础上，重新组织能够表达文本主要信息的句子，作为摘要句[1~3]。本文针对抽取式多文档自动摘要方法进行研究。

近年来，许多研究方法被应用到文档摘要系统上，其中以基于质心[4~5]和基于图模型[6~9]的两种方法尤为突出。基于质心的方法主要是从文档集中选择一些比较重要的质心词（每个词的 tf*idf 值在给定的阈值范围内）构成一个能代表文档的中心句子，然后将文档中所有句子同生成的中心句子进行相似度比较，挑选出与中心句子相似度较高的句子作为文档的摘要。Radev 提出的 MEAD[10]就是一个基于质心的摘要系统，对于相关文档类中的每一个句子，MEAD 分别计算质心得分、位置信息以及同第一个句子（可能是文档的标题）的重复率这 3 个特征，然后将其线性组合起来确定哪个句子得分最高。

收稿日期：2014-6-1　　**定稿日期：**2014-7-28

基金项目：国家自然科学基金资助项目(61272212，61163006，61203313)

而基于图模型的方法则主要是将文档集构建成一个以句子为顶点，各顶点句子之间的余弦相似度构成边关系的图模型。Radev 在 2004 年提出的 LexRank[6]就是这样的一个模型，基于这个基础上再利用类似 PageRank[11]算法对这个图模型各顶点求出一个排序得分，然后在规定的摘要长度内挑选出得分排在前面的句子组成摘要，但是这种模型仅仅只是考虑了句子之间的关系。Wei 等人提出将文档信息也加入到图模型中，构建文档敏感图模型（**D**ocument-**S**ensitive **R**anking model，DsR），利用文档集的全局信息对多文档内的句子的影响，将句子与句子之间的关系分为跨文档关系与同一文档内关系，从而将文档之间的相关信息融合到句子之间的信息中，达到提高系统摘要质量的目的[12]。但是文档所包含的信息太宽泛，对于句子信息的影响不是很大，因此可以尝试融合更多的文本信息，从而使得生成的摘要更为准确。

在信息检索相关工作中，Blanco 和 Lioma 采取固定滑动窗口大小的方法得到词项间的共现关系，若两个词项同时出现在窗口内，则可以看做这两词项之间有边相连，构建词项的无向无权图，然后采用类似 PageRank 的随机游走方法根据词项顶点的入度和出度计算词项在文档中的权重[13~14]。Rousseau 等也是通过同样的方法得到词项的共现关系，同时还根据词项出现的位置关系得到词项间的顺序关系，从而构造出关于文档词项的有向无权图，不同于 Blanco 等确定边的权重方法，这里的边的权重仅仅由该词项节点的入度数来确定[15]。虽然这些研究确定词项的权重方式不同，但是它们的共同点都是首先构造出文档的词项图，然后借助词项图来确定词项的权重。这些研究都是从词项权重信息角度出发，可以看出词项权重信息对文档的自动摘要有着很大的影响。

总之，现有基于图模型的多文档自动摘要研究工作只考虑了句子层面的信息。尽管 Wei 等人提出的 DsR 模型[12]考虑了文档层面信息，但也只利用了文档和句子的信息来确定句子最后的得分，并没有充分利用文档中词项的信息。因此本文在前人工作基础上，融合词项权重信息和文档信息，构建了基于词项-句子-文档的三层图模型（**T**erm-**S**entence-**D**ocument Graph **M**odel，TSDM），进行多文档的自动摘要。

TSDM 分为 3 层，分别是词项图、句子图以及文档图。词项图是对文档集内所有词项构建一个无向带权图，顶点表示各词项，边用来刻画两词项的共现关系，通过它们的共现句子数来确定边的权重，从而构建出一个关于词项的共现矩阵，再通过马尔科夫链计算方法确定词项在当前文档集中的权重；文档图通过计算文档间的概率转移矩阵构造文档关系矩阵；而句子图则是通过结合文档关系矩阵构造句子相似度矩阵，再通过马尔科夫链预测过程确定句子权重，最后再将句子权重和其所包含的词项权重线性组合，将它作为最终的句子权重。

2. 基于多层图模型的多文档摘要

2.1 多层图模型的构建

LexRank 模型根据句子与句子之间的关系构建句子级别的关系网络，以实现文档摘要；DsR 模型在 LexRank 模型的基础上，根据文档与文档之间的相关性构建文档级别的关系图模型，再结合句子的图模型构成句子-文档双层图模型。受它们的启发，本文考虑信息粒度更小的词汇信息，在原有的句子-文档两层图模型基础上，根据词项与词项之间的关系构建词项关系网络，从而构建词项-句子-文档三层图模型 TSDM。

图 1 为词项-句子-文档三层图模型的一个简单示例，第一层为文档图，每篇文档都被看作是一个顶点，两篇文档相关时，两顶点有边相连，并且边的权重由这两篇文档之间的相关性刻画；第二层为句子图，同文档层构建方法类似，句子看作顶点，句子之间的相似度看成是边的权重，通过第一层和第二层，可以获得文档和句子之间的从属关系，借助这个关系，可以将句子之间的边分为跨文档边和同一文档内的边，然后区别处理这两种边；第三层为词项图，构建词项之间的关系图，文档中的每个词项作为该层图的顶点，顶点之间边的权重表

示词项与词项之间的共现句子数。

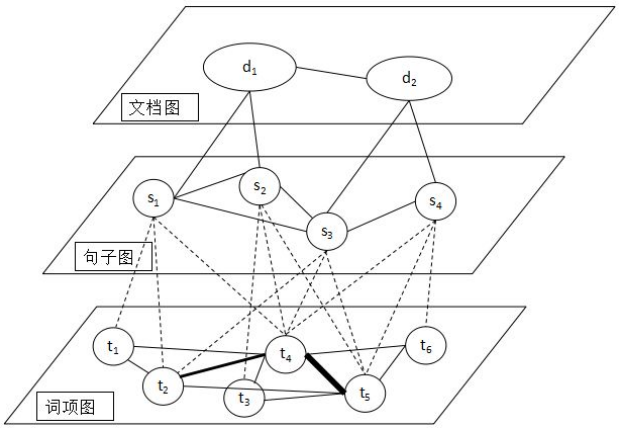


图 1 Term-Sentence-Document 三层图模型的简单示例

2.2 词项图

2.2.1 词项图构建

词项图是由文档集中所有的词项 t_i 构成顶点，设定当且仅当两个词项至少在文档集的某一句话中同时出现时，这两个词项之间才有边相连，并且边的权重为它们的共现次数即同时包含这两个词项的句子个数。不同于 Blanco 和 Lioma 工作中选择以滑动窗口为单位，本文固定以句子为度量单位，这是因为词项-句子-文档三层图模型通过句子的权重来确定当前句子是否为摘要内容，采用这种方式句子中的语义信息能完整的保存，词项之间的关系能够较好的体现。在图 1 所示的例子中共有 s_1、s_2、s_3、s_4四句话，它们的内容依次是$\{t_1t_2t_4\}$、$\{t_3t_4t_5\}$、$\{t_2t_4t_5\}$、$\{t_4t_5t_6\}$，构建出的词项图中，t_4、t_5因为在 s_2、s_3、s_4三句话中都出现，所以它们的共现次数是 3，t_2、t_4在 s_1、s_3中共现 2 次，而其他词项之间均只共现 1 次，从而得到该文档集的词项共现矩阵 $M^t=\{m_{ij}\}_{N_t\times N_t}$，$N_t$ 为词项总数，m_{ij} 为同时出现了词项 i 和词项 j 的句子个数。

2.2.2 词项权重计算

根据得到的词项共现矩阵 M^t 生成词项间的概率转移矩阵 P^t，

$$P_{ij}^t=\begin{cases} m_{ij}\Big/\sum_{j=0}^{N_t-1}m_{ij} & i\neq j \\ 0 & i=j \end{cases} \tag{1}$$

式（1）计算概率转移矩阵 P^t 时只考虑了词项在同一句话中的共现关系，它仅仅表示出局部关系，为了考虑词项在该文档集内的全局关系，采用类似 PageRank 算法加入阻尼因子的方法，将式（1）修正如下：

$$P^t=d\cdot\frac{1}{N_t}\cdot \mathrm{I}+(1-d)\cdot P^t \tag{2}$$

求得概率转移矩阵 P^t 后，通过马尔科夫链预测模型 $B_k^t=P^{t^k}\cdot B_0^t$，计算词项权重向量 $B^t=\{b_i^t\}_{N_t\times 1}$，其中 b_i^t 代表第 i 个词项的权重，权重越高，表明该词项越重要。根据马尔科夫链预测过程的特性，经过 k 迭代之后，B^t 的值最终将收敛。马尔科夫链预测过程伪代码如算法 1 所示：

算法 1 马尔科夫链预测过程

input：概率转移矩阵 P^t，词项总数 N_t，误差 μ

output：特征向量 B^t

$B_0^t = \left(\frac{1}{N_t}, \frac{1}{N_t}, \cdots \frac{1}{N_t}\right)$

$k = 0$

repeat

  $k = k + 1$

  $B_k^t = P^{tT} B_{k-1}^t$

  $\delta = \left\| B_k^t - B_{k-1}^t \right\|$

until $\delta <$ μ

return B_k^t

2.3 文档图

Wei 提出的 DsR 模型[12]在生成文档摘要时把文档信息也添加进来了，但是 DsR 模型中只对可以直接转移的文档间的关系进行处理，却忽略了可以间接转移的文档间的关系，所以本文还将通过马尔科夫随机游走算法同时捕获可以直接转移和间接转移的文档间的关系，得到更完整的文档间信息。

2.3.1 文档图构建

文档图中的顶点为每篇文档，顶点与顶点之间边的权重刻画文档之间的关系，这部分主要介绍如何定义文档间关系。DsR 模型在处理文档信息时，仅仅对文档间的相似度做了归一化处理，这样处理只能捕获可以直接转移的文档间的关系，因此，本文采用马尔科夫随机游走的方法对文档相似度进行一次随机游走用于捕获可以间接转移的文档间的关系。

首先对文档间相似度进行归一化处理，构建文档间概率转移矩阵 P^d：

$$P^d(d_i, d_j) = \frac{sim(d_i, d_j)}{\sum_{dk \in D \cap k \neq i} sim(d_i, d_k)} \tag{3}$$

其中，d_i 表示第 i 篇文档，$sim(d_i, d_j)$ 表示两篇文档的余弦相似度。再对其进行马尔科夫随机游走，$P^{dk} = P^{dk-1} \cdot P^d$，然后构建文档关系矩阵 W^d，对于同一篇文档，其自身与自身的关系看成 1，不同的文档间关系则在 1 的基础上再加上对应文档间的转移概率，W^d 具体定义如下：

$$W^d(d_i, d_j) = \begin{cases} 1 & if\ i = j \\ 1 + P^d(d_i, d_j) + \frac{1}{2} P^{d1}(d_i, d_j) & if\ i \neq j \end{cases} \tag{4}$$

通过上述处理，文档关系矩阵 W^d 便得到了文档信息，这部分信息可以为接下来度量句子间相似度时提供参考。

2.4 句子图

2.4.1 句子图构建

句子图是以文档集中每个句子为顶点，句子之间的相似度看做是边的权重。本文采用余弦相似度来计算句子相似度，构建句子相似矩阵 M^s：

$$M_{Si,Sj}^s = \frac{\sum_{w \in Si,Sj} tf_{w,Si} tf_{w,Sj} (idS_w)^2}{\sqrt{\sum_{w \in Si} (tf_{w,Si} idS_w)^2} \times \sqrt{\sum_{w \in Sj} (tf_{w,Sj} idS_w)^2}} \tag{5}$$

$$idS_w = \log\left(\frac{N_s}{N_k}\right) \tag{6}$$

tf_{w,S_i} 表示词项 w 在句子 S_i 中出现的次数，idS_w 是逆句子频率，类似于逆文档频率，N_s 表示句子总数，N_k 表示包含词项 w 的句子数。

然后同时结合文档关系矩阵 W^d 和句子相似矩阵 M^s，重新构建句子相似矩阵 $\widetilde{M}^s$，重构过程如下：

$$\widetilde{M}^s_{S_i,S_j} = W^d\left(d(S_i), d(S_j)\right) \cdot M^s_{S_i,S_j} \tag{7}$$

$d(S_i)$ 表示包含句子 S_i 的文档，依据重构后的句子相似矩阵 $\widetilde{M}^s$，构建的句子图即为最终的句子图。

2.4.2 句子权重计算

句子图建好之后，再计算各句子权重，计算句子权重过程如下：

首先，根据结合文档信息后的句子相似矩阵 $\widetilde{M}^s$，通过判断第 S_i 个句子与其它所有句子的相似度，构建句子图上的邻接矩阵 $\boldsymbol{A}$：

$$A_{S_i,S_j} = \begin{cases} 1 & sim(S_i, S_j) > \varepsilon \\ 0 & others \end{cases} \tag{8}$$

根据得到的邻接矩阵 $\boldsymbol{A}$，再求解句子的概率转移矩阵 P^s：

$$P^s_{S_i,S_j} = \frac{A_{S_i,S_j}}{\sum_{S_k} A_{S_i,S_k}} \tag{9}$$

然后再把句子概率转移矩阵 P^s 通过多次迭代求解代表各句子权重的特征向量 $B^s = \{b_i^s\}_{Ns\times 1}$，其中 b_i^s 代表第 i 个句子的权重，权重越高，表明该句子越重要。整个过程与 2.2.2 小节求解词项权重过程类似。

2.5 摘要生成

建立了 TSDM 的 3 层结构后，本小节介绍如何利用它来确定摘要句。摘要里的句子必定是最能够反映多个文档中心主题的句子，同时这些句子之间相互重复要小，即低冗余，评判依据就是句子的权重以及句子之间的相似关系。通过文档图和句子图，可以得到结合文档信息和句子信息的表示句子权重的特征向量，但是这种方法并没有考虑词项权重信息。

基于假设：一个句子若由一些比较重要的词项构成，那么它所传达的信息也应该是比较重要的，反之，若一个句子包含的词项不那么重要，那么这个句子所表达的信息也应该不重要，所以词项权重信息也将对句子重要性产生影响。借助 TSDM 模型中的词项图，根据词项权重的特征向量 B^t，求得每个句子所包含的词项权重 $B^{st} = \left\{b_i^{st} \middle| b_i^{st} = \sum_{k\in S_i} b_k^t \cdot idS_k\right\}$，然后将其与代表各句子权重的特征向量 B^s 线性结合，得出句子最终的权重 $\widetilde{B}^s$，使得这个权重包含词项、文档以及句子本身等相关信息，计算公式如下：

$$\widetilde{B}^s = \omega \cdot B^s + (1-\omega) \cdot B^{st} \tag{10}$$

式（10）得出的结果即句子权重最终结果，根据这个结果，按照权重由高到低的顺序挑选句子组成摘要，同时为保证摘要的冗余度足够小，在选择候选句子加入到摘要前，将其权重同其所有邻接的句子的权重进行对比，只有该句子的权重最大时，才能把当前句子加入摘要，直到达到规定的摘要长度。

3. 实验设计和结果

3.1 数据集

本文的实验数据采用了 DUC'2003[1]和 DUC'2004[2]任务 2 的数据集。DUC'2003 数据集包含了 30 个主题类，除去个别主题类，每个主题都含 10 篇文档，而 DUC'2004 有 50 个主题类，每个主题下包含 10 篇文档。对于每一个文档集，都给出了 4 个对应的专家摘要作为判断标准，来评价系统生成的摘要。数据集的统计信息见表 1：

表 1 实验使用数据集的统计信息

	DUC'2003	DUC'2004
类的个数	30	50
每个类中的文档数	~10	10
文档数	298	500
数据来源	TREC	TREC
摘要长度	100 words	665 Bytes

3.2 评价指标与实验设置

为了评价多文档自动摘要的结果，我们采用 DUC 评测官方评价指标 Rouge[16]来测量不同方法的优劣。Rouge 指标通过计算系统摘要同专家摘要的 N 元组（连续的 N 个单词组成）重复率来对摘要结果进行评价的。根据定义的 N 值和计算策略的不同，Rouge 指标可以进一步细化为 ROUGE-N、ROUGE-L、ROUGE-W、ROUGE-S、ROUGE-SU 等指标。ROUGE-N 计算两个摘要里的 N 元词的匹配率，计算公式如下：

$$\text{Rouge - N} = \frac{\sum_{S \in Summ_{ref}} \sum_{N-gram \in S} Count_{match}(N-gram)}{\sum_{S \in Summ_{ref}} \sum_{N-gram \in S} Count(N-gram)} \tag{11}$$

N 表示 *N-gram* 的长度，*Count*(*N-gram*)表示专家摘要中 *N-grams* 的个数。在实验中，我们采用 ROUGE 1.5.5 和 DUC 官方提供的 ROUGE 参数进行结果评估，包含 ROUGE-1、ROUGE-2、ROUGE-W 这 3 个指标，其中以 ROUGE-2 指标为主，且实验结果都是采用这 3 个指标的平均 F 值。

在文本预处理过程中，我们实验对比了许多常用的自然语言处理工具，发现德雷塞尔大学提供的开源的文本检索与挖掘工具包(Dragon Toolkit)[3]在处理英文文本分句结果上相对最优，因此在实验预处理中采用了该工具包提供的分句程序。另外，我们发现对文档集分句后进行去停用词、词干化等操作会影响词项权重结果，降低自动摘要的效果，所以，实验中未对数据集进行任何去停用词以及词干化等操作。

在词项图构建过程中求解词项概率转移矩阵 P^t 时，根据经验本文设置阻尼因子 d 为 0.15，通过马尔科夫链预测过程求解词项权重向量 B^t 时，误差 μ 取 0.00001；构建句子邻接矩阵 $\boldsymbol{A}$ 时，句子相似度阈值 ε 的值与文献[6]中的相同，均为 0.1，求解句子权重向量 B^s 时，误差 μ 取 0.001。

对比实验选择了 LexRank 模型和 DsR 模型，为了验证词项权重信息和文档信息对摘要结果的影响，实验尝试了不同的方案，实验结果见表 2 和表 3。实验中，还对比了进行 1 次随机游走后的文档关系的摘要结果和不进行游走的文档关系的摘要结果，对于公式(10)中 ω 对结果的影响将在图 2 中展示。

3.3 实验结果及分析

---

[1] http://www-nlpir.nist.gov/projects/duc/data/2003_data.html
[2] http://www-nlpir.nist.gov/projects/duc/data/2004_data.html
[3] http://dragon.ischool.drexel.edu/license.asp

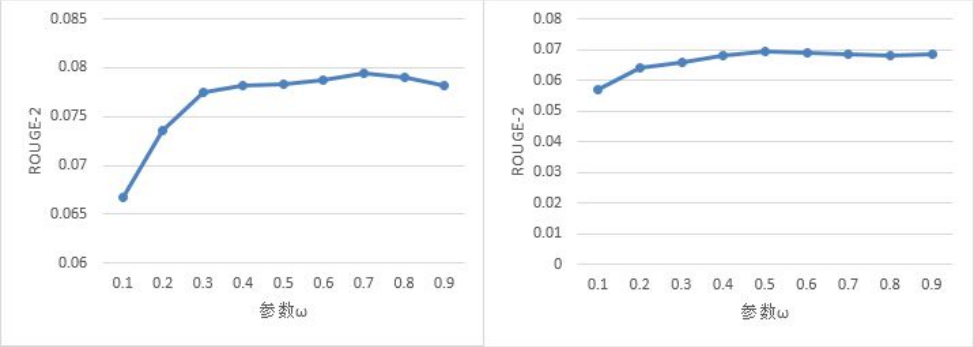


图 2 各数据集上 ω 对 ROUGE-2 的影响(左图 DUC'2003 数据集，右图 DUC'2004 数据集)

图 2 分别给出了参数 ω 在 DUC'2003 数据集和 DUC'2004 数据集上对 ROUGE-2 的影响。左图表明，在 DUC'2003 数据集上当 ω 取值为 0.7 时，ROUGE-2 取得相对最高值，而右图表明，在 DUC'2004 数据集上当 ω 取值为 0.5 时，ROUGE-2 取得相对最高值。ω 在两个数据集上的较优值不一致，这跟数据集本身的特性有关。所以实验中 ω 都是取其对应数据集上的较优值，即在 DUC'2003 数据集中 ω 的值为 0.7，在 DUC'2004 数据集中 ω 的值为 0.5。

表 2 DUC'2003 数据集对比结果

	ROUGE-1	ROUGE-2	ROUGE-W
LexRank	0.3190	0.0604	0.1212
DsR	0.3447	0.0770	0.1348
DsRM	**0.3535**	0.0794	**0.1375**
LexRankTerm	0.3288	0.0679	0.1249
TSDM	0.3518	**0.0795**	0.1366

表 3 DUC'2004 数据集对比结果

	ROUGE-1	ROUGE-2	ROUGE-W
LexRank	0.3321	0.0629	0.1253
DsR	0.3396	0.0668	0.1319
DsRM	0.3397	0.0678	**0.1321**
LexRankTerm	0.3312	0.0637	0.1246
TSDM	**0.3407**	**0.0696**	0.1304

表 2 和表 3 分别表示不同模型在 DUC'2003 和 DUC'2004 数据集上的实验结果。其中 DsRM 模型为对文档转移概率矩阵进行 1 次随机游走后的方法，LexRankTerm 模型为不考虑文档信息只把词项权重信息同句子权重进行线性结合的方法，通过实验对比发现在 DUC'2003 数据集上将 LexRank 得到的句子权重与词项权重比值设为 1:1，在 DUC'2004 数据集上该比值设为 9:1 时，效果最好。

DsR 模型较 LexRank 模型在 DUC'2003 数据集和 DUC'2004 数据集上各项指标均有较大提升。因为，如果两篇文档相似度较高，那么在这两篇不同文档中的两个句子之间的主题关联度更高。对比 DsR 模型原文的实验结果，在 DUC'2004 数据集上的 DsR 模型较 LexRank 模型的提升效果同 DsR 模型原文的比较接近，但是在 DUC'2003 数据集上却提升很多，这可能与系统生成的摘要长度有关，2003 年的系统摘要长度规定是 100 个单词左右，2004 年的系统摘要长度规定为 665 个字节左右。

实验数据表明，DsRM 模型相比 DsR 模型在 DUC'2003 数据集上 ROUGE-2 提升 3.12%，在 DUC'2004 数据集上 ROUGE-2 提升 1.50%。DsR 模型在构建文档转移概率矩阵时，只考

虑了可以直接相关的文档间关系，而 DsRM 通过一次随机游走，把文档之间的间接关系也结合进来，使得文档之间的语义关系更加完整，从而进一步提升系统自动生成的摘要的质量。

对比 LexRank 模型和 LexRankTerm 模型，LexRankTerm 模型在 DUC'2003 数据集上 ROUGE-2 提升 12.42%，在 DUC'2004 数据集上 ROUGE-2 提升 1.27%。LexRank 模型只考虑了句子之间的关系，而忽略了粒度更小的词项权重信息，本文还通过构建词项图计算词项的重要性，最后将词项的权重与句子的权重结合。从实验结果上来看，结合词项权重信息后，实验的各项指标整体上都有提升。然而，结合词项权重信息的模型没有结合文档信息的效果好，这是因为词项所包含的信息比较少，由于数据集中的文档长度都较短，词项权重信息的噪声相对文档信息要大，所以结合词项权重信息后的提升效果没有结合文档信息后的效果好。

在这两个数据集上，本文提出的融合词项和文档信息后的 TSDM 模型在各项评价指标上均有良好的表现。在 DUC'2003 数据集上相比 DsR 模型，指标 ROUGE-1 提升 2.06%，指标 ROUGE-2 提升 3.25%，指标 ROUGE-W 提升 1.34%；在 DUC'2004 数据集上相比 DsR 模型，ROUGE-1 提升 0.32%，ROUGE-2 提升 4.19%，ROUGE-W 则略有下降。这表明词项权重信息、文档信息能够显著提高多文档自动摘要的质量。然而，在 DUC'2003 数据集上，ROUGE-1、ROUGE-2 和 ROUGE-W 各项指标相比于 DsRM 算法提升并不明显，对比 DUC'2004 数据集上的实验结果，我们发现这与数据集自身特性有关。在 DUC'2003 数据集中，每个主题下的所有文章所包含的句子长度分布不均衡，这会导致部分句子长度较长但实际中相对不重要的句子的得分相应提高，因此使得部分实验结果略有下降。

4. 总结与展望

本文提出了基于词项-句子-文档三层图模型的多文档自动摘要方法。它不仅利用句子之间的相似度，而且考虑了句子所属的文档之间的关系以及句子所包含的词项权重信息来生成文档摘要。实验结果表明本文提出的模型能显著的提高自动摘要的质量。

在计算词项权重信息时，本文只利用了词项间的共现关系，未考虑词项间存在的语义关系以及句子间的句法、语义关系，因此未来的工作包括进一步研究如何深层次的利用词项间的语义关系以及句子间句法、语义关系来提高自动摘要的效果。

参 考 文 献

[1] 刘挺, 王开铸. 自动文摘的四种主要方法[J]. 情报学报, 1999, 18(1): 11-19.

[2] 秦兵, 刘挺, 李生. 多文档自动文摘综述[J]. 中文信息学报, 2005, 19(6).13-20.

[3] E.padma lahari, D.V.N Siva Kumar. A Comprehensive Survey on Feature Extraction in Text Summarization[J]. Computer Technology and Applications, 2014, 5(1): 248-256.

[4] Radev D, Winkel A, Topper M. Multi document centroid-based text summarization[C]//Proceedings of ACL'2002 Demo Session. ACL, 2002.

[5] Radev D R, Jing H, Styś M, et al. Centroid-based summarization of multiple documents[J]. Information Processing and Management, 2004, 40(6): 919-938.

[6] Erkan G, Radev D R. LexRank: Graph-based lexical centrality as salience in text summarization[J]. Journal of Artificial Intelligence Research (JAIR), 2004, 22(1): 457-479.

[7] Chen H, Jin H, Zhao F. PSG: a two-layer graph model for document summarization[J]. Frontiers of Computer Science, 2014, 8(1): 119-130.

[8] Canhasi E, Kononenko I. Weighted archetypal analysis of the multi-element graph for query-focused multi-document summarization[J]. Expert Systems with Applications, 2014, 41(2): 535-543.

[9] 纪文倩, 李舟军, 巢文涵, 等. 一种基于 LexRank 算法的改进的自动文摘系统[J]. 计算机科学, 2010, 37(5): 151-154.

[10] Radev D, Allison T, Blair-Goldensohn S, et al. MEAD-a platform for multidocument multilingual text summarization[C]//Proceedings of the 4th International Conference on Language Resources and Evaluation (LREC'2004). LREC, 2004.

[11] Page L, Brin S, Motwani R, et al. The PageRank citation ranking: Bringing order to the web[R]. California: Stanford InfoLab, 1999.

[12] Wei F, Li W, Lu Q, et al. A document-sensitive graph model for multi-document summarization[J]. Knowledge and information systems, 2010, 22(2): 245-259.

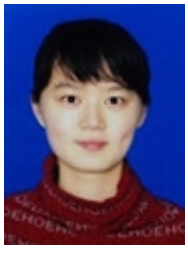
[13] Blanco R, Lioma C. Random walk term weighting for information retrieval[C]//Proceedings of the 30th annual international ACM SIGIR conference on Research and development in information retrieval. ACM, 2007: 829-830.

[14] Blanco R, Lioma C. Graph-based term weighting for information retrieval[J]. Information retrieval, 2012, 15(1): 54-92.

[15] Rousseau F, Vazirgiannis M. Graph-of-word and TW-IDF: new approach to ad hoc IR[C]//Proceedings of the 22nd ACM international conference on Conference on information & knowledge management. ACM, 2013: 59-68.

[16] Lin C Y. Rouge: a package for automatic evaluation of summaries[C]//Proceedings of the ACL Workshop on Intrinsic and Extrinsic Evaluation Measures for Machine Translation. ACL, 2005: 74-81.

作者简介：



熊娇（1990—），女，硕士研究生，主要研究领域为自然语言处理。Email:xiong_jiao@hotmail.com；

王明文（1964—），男，博士，教授，主要研究领域为信息检索、数据挖掘、自然语言处理。Email:mwwang@jxnu.edu.cn；

李茂西（1977—），男，博士，副教授，主要研究领域为自然语言处理，机器翻译。Email:mosesli@yeah.net。