Intelligent Information Management, 2009, 38-45
doi:10.4236/iim.2009.11007 Published Online July 2009 (http://www.scirp.org/journal/iim)

A New Method for Calculating Similarity between Sentences and Application on Automatic Abstracting

Wenqian JI, Zhoujun LI, Wenhan CHAO, Xiaoming CHEN
Department of Computer Science and Technology, BeiHang University，Beijing 100191*, China*

Abstract: Sentence similarity computing plays an important role in machine question-answering systems, machine-translation systems, information retrieval and automatic abstracting systems. This article firstly sums up several methods for calculating similarity between sentences, and brings out a new method which takes all factors into consideration including critical words, semantic information, sentential form and sentence length. And on this basis, a automatic abstracting system based on LexRank algorithm is implemented. We made several improvements in both sentence weight computing and redundancy resolution. The system described in this article could deal with single or multi-document summarization both in English and Chinese. With evaluations on two corpuses, our system could produce better summaries to a certain degree. We also show that our system is quite insensitive to the noise in the data that may result from an imperfect topical clustering of documents. And in the end, existing problem and the developing trend of automatic summarization technology are discussed.

Keywords: sentence similarity, automatic abstracting, lexrank, sentence-weight computing, redundancy resolution

句子相似度計算及其在自動文摘系統中的應用

紀文倩，李舟軍，巢文涵，陳小明
北京航空航天大學計算機學院 北京 100191

摘　要：計算句子的相似度在自動問答、機器翻譯、信息檢索和自動文摘等系統中有著非常重要的作用。本文首先歸納了句子相似度計算的方法，提出了一種新的句子相似度計算方法，綜合考慮了關鍵詞特徵、語義特徵、句式特徵和句子長度特徵等信息。並以此為基礎，實現了一個基於 LexRank 算法的自動文摘系統，同時對句子權重計算方法以及冗餘處理等方面進行了改進。我們實現的文摘系統，可以對中文或英文的單文本或多文本進行自動文摘。通過在哈工大和 DUC 的測評語料上進行實驗，測試結果表明該系統在一定程度上改進了文摘的質量，在多文本文摘中的抗噪聲方面也有一定的優越性。最後，文章討論了自動摘要研究存在的問題，並指出自動文摘的研究趨勢。

關鍵詞：句子相似度，自動文摘，LexRank，句子權重計算，冗餘處理

1. 引 言

在自然語言處理領域，句子相似度計算是一項基礎而核心的研究課題，長期以來一直是人們研究的一個熱點和難點。句子相似度的計算在自然語言處理的各個領域都有著非常重要的作用，在基於實例的機器翻譯中，相似度主要用於衡量文本中詞語的可替換程度[1]；在資訊檢索中，相似度更多的是反映文本與用戶查詢在意義上的符合程度；在自動問答中，相似度反映的是句子之間語義上的匹配程度；而在多文檔文摘系統中，相似度可以反映出局部主題資訊的擬合程度[2]。

自動文摘是隨著互聯網上的資訊急劇膨脹而發展起來的文本資訊處理技術，它利用電腦自動地從文本或文本集合中提煉出能準確、全面地反映文本主要內容的精簡、連貫的短文[10]，以滿足用戶快速獲取知

識的要求。

本文給出了一種計算句子相似度的新方法，並給出了該方法在自動文摘系統中的應用，設計並實現了一種基於 LexRank 的改進的自動文摘系統。

2. 句子相似度計算方法

在自然語言處理的許多領域，句子相似度計算是一項應用廣泛的技術，並發揮著重要作用。隨著這些領域的發展，句子相似度計算也誕生了很多方法。不同方法很大程度上依賴於句子的不同表示形式[3]。目前研究句子相似度的方法有基於關鍵字的方法，使用語義詞典的方法[4]，使用語義依存的方法[5]，計算編輯距離的方法[6]，基於預警框架的方法[7]，基於屬性論的方法[8]以及基於統計的方法[9]等。

2.1 基於句子不同特徵的相似度計算

通過對句子的深入分析，句子相似度計算歸結起來可以概括為三類方法：基於詞特徵的句子相似度計算，基於語義特徵的句子相似度計算以及基於句法分析特徵的句子相似度計算。

2.1.1 基於詞特徵的句子相似度計算

基於關鍵字特徵的句子相似度的計算通常採用基於向量空間模型的方法。文本中每個句子表示成一個 n 維向量 V=(d_1,d_2,…,d_n)，d_i 是對應單詞的 tf*idf 值；句子之間的相似度就等於對應向量的 Cosine 值。設兩個句子 A 和 B，它們所有有效詞構成向量空間為 $V=\{X_1,X_2,X_3,...,X_n\}$，句子 A 對應的向量為 $V_1=\{w_1,w_2,w_3,...,w_n\}$,其中 w_i 為句子 A 中有效詞 X_i 的 TF*IDF 值；同樣的句子 B 對應的向量為 $V_2=\{\varphi_1,\varphi_2,\varphi_3,...,\varphi_n\}$，其中 φ_i 為句子 B 中有效詞 X_i 的 TF*IDF 值。則兩個句子的相似度為：

$$Similarity(A,B)=\frac{\sum_{i=1}^{n} w_i * \varphi_i}{\sqrt{\sum_{i=1}^{n} {w_i}^2} * \sqrt{\sum_{i=1}^{n} {\varphi_i}^2}} \quad (1)$$

這種方法只是單獨地考慮了句子的物理特徵，沒有考慮句式特徵、句子長度以及語義特徵，因此具有一定的局限性。

2.1.2 基於語義特徵的句子相似度計算

基於語義特徵的句子相似度計算，需要一定的語義知識庫作為基礎，中文常採用 HowNet 或《同義詞詞林》，英文一般採用 WordNet。通過詞與詞之間的語義相似度，計算句子間的語義相似度。設兩個句子 A 和 B，設 A 包含關鍵字為 x_1，x_2，□，x_m, 句子 B 包含的詞為 y_1，y_2，□，y_n。詞 $x_i(1\le i\le m)$和 $y_i(1\le j\le n)$之間的相似度用 $s(x_i,y_j)$ 來表示，這樣我們得到一個語義相似度矩陣：

$$X=\begin{bmatrix} s(x_1,y_1),s(x_1,y_2),\Lambda,s(x_1,y_n) \\ s(x_2,y_1),s(x_2,y_2),\Lambda,s(x2,y_n) \\ ..... \\ s(x_m,y_1),s(x_m,y_2),\Lambda,s(x_m,y_n) \end{bmatrix} \quad (2)$$

則 A，B 句子之間的語義相似度為：

$$Similarity(A,B)=\frac{1}{2}*(\frac{\sum_{i=1}^{m} a_i}{m}+\frac{\sum_{i=1}^{n} b_i}{n}) \quad (3)$$

其中 $a_i=\max(s(x_i,y_1),s(x_i,y_2),\dots,s(x_i,y_n))$，$b_i=\max(s(y_i,x_1),s(y_i,x_2),\dots,s(y_i,x_m))$。這種方法充分考慮了句子中每個詞的深層語義資訊，使得表面不同，深層意義相同的詞被挖掘出來，而基於關鍵字特徵的相似度計算就不可識別這類資訊。但是由於詞典的不全面和一些未登錄詞的詞義代碼的缺失，也給計算帶來一定的誤差，另外這種方法也沒有考慮結構資訊。

2.1.3 基於句法分析特徵的句子相似度計算

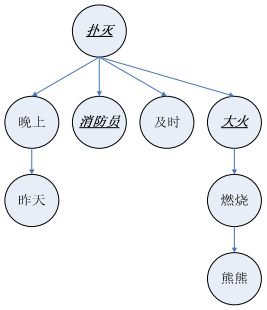
基於句子句法特徵的句子相似度計算需要考慮句子的句法結構資訊。一個完整的句子是由句子的主幹成分和修飾成分所構成，人們往往從主幹成分就可以瞭解句子的主要意思，所以在利用依存結構進行相似度計算時，只考慮那些有效搭配對之間的相似程度。有效搭配對是指全句核心詞和直接依附於它的有效片語成的搭配對，這裏有效詞定義為動詞、名詞和形容詞。全句核心詞為依存樹的根節點。例如下麵兩個句子：

例句 1　昨天晚上，消防隊員及時撲滅了熊熊燃燒的大火。

例句 2　晚上 11 點左右，熊熊燃燒的大火才徹底熄滅。

上圖中標記為斜體的詞代表了句子的主要意思，

本文工作得到國家自然科學基金項目(60573057)資助



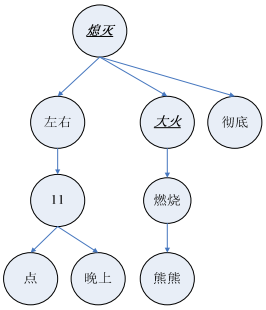


圖 1. 例句 1 的依存樹 **圖 2. 例句 2 的依存樹**

Figure 1. The dependency structure of example sentence 1

Figure 2. The dependency structure of example sentence 2

即句子 1 的有效搭配對為: 撲滅_消防員、撲滅_大火; 句子 2 的有效搭配對為: 熄滅_大火。兩個句子之間的相似度計算方法為:

$$Similarity(A,B)=\frac{Match(PairCount)}{\text{Max}\{PairCount_1,PairCount_2\}} \quad (4)$$

其中，$Match(PairCount)$ 為句子 1 和 2 有效搭配對的匹配數，$PairCount_1$ 為句子 1 的有效搭配對數，$PairCount_2$ 為句子 2 的有效搭配對數。

這種方法從句法深度進行考慮，考慮到了詞與詞之間的依存關係，對句子的理解更加充分，從而更準確的得到句子的相似度。但是，現有的句法分析技術還不夠成熟，無法將所有句法資訊特徵全部考慮進去，所以就產生了一定的誤差。

2.2 改進的句子相似度計算

由上一部分可知: 基於關鍵字特徵的方法體現了句子的表面資訊，基於語義特徵的方法考慮了組成句子的每個詞的深層的語義資訊，基於句法特徵的方法結合了詞與詞之間的依存關係。我們將關鍵字特徵和語義特徵相結合，同時引入句式特徵、句長特徵的影響，從而對句子相似度計算方法進行了改進。

向量空間模型（VSM）中，所有句子的關鍵字個數為 N，它們所有有效詞構成向量空間為 $V=\{X_1,X_2,X_3,...,X_n\}$。

首先，擴展向量空間到 N+1 維，第 N+1 維為 0 或 1 表示句子的句式特徵，當該句為肯定句時為 1，否定句為 0。

其次，考慮句子的關鍵字特徵，設句子 A 對應的向量為 $V_1=\{w_1,w_2,w_3,...,w_n,\ w_{n+1}\}$，其中 w_i (1≤i≤n) 為句子 A 中有效詞 X_i 的 TF*IDF 值；同樣的句子 B 對應的向量為 $V_2=\{\varphi_1,\varphi_2,\varphi_3,...,\varphi_n,\ \varphi_{n+1}\}$，其中 φ_i (1≤i≤n)為句子 B 中有效詞 X_i 的 TF*IDF 值。

然後考慮句子語義特徵，如 2.12 章所述，通過計算詞與詞之間的語義相似度，得到語義相似度矩陣，如公式(2)所示。根據句子的關鍵字特徵和語義特徵，我們可以得到句子 A 和 B 的語義相似度:

$$SemSim(A,B)=\frac{\sum_{i=1}^{n} w_i * a_i}{\sum_{i=1}^{n} w_i}+\frac{\sum_{i=1}^{n} \varphi_i * b_i}{\sum_{i=1}^{n} \varphi_i} \quad (5)$$

其中 $a_i=\max(s(x_i,y_1),s(x_i,y_2),\dots,s(x_i,y_n))$，$b_i=\max(s(y_i,x_1),s(y_i,x_2),\dots,s(y_i,x_n))$。

再次，我們考慮句子的長度相似性，通過公式(6)計算得到:

$$LenSim(A,B)=1-\left|\frac{Length(A)-Length(B)}{Length(A)+Length(B)}\right| \quad (6)$$

因此，最後的句子相似度計算公式為:

$$Sim(A,B)=(w_{n+1}\otimes\varphi_{n+1})*(\alpha*SemSim(A,B)+\beta*LenSim(A,B)) \quad (7)$$

其中，w_{n+1} 和 φ_{n+1} 表示句子 A 和 B 的句式特徵，當 w_{n+1} 和 φ_{n+1} 不同時 $(w_{n+1}\otimes\varphi_{n+1})$ 結果為 0，當二者相同時則等於 1。

這樣在句子相似度的計算中，不僅考慮了詞語的出現的頻率、句子長度等物理特徵，還考慮了詞語之間的語義關係。因此，計算得到的相似度更具合理性。

3. 自動文摘系統

自動文摘的研究始於 1958 年，由美國 IBM 公司的 Luhn 開創了自動摘要研究的先河[10],接著馬裏蘭州大學的 Edmundson[11][12]、美國俄亥俄州立大學的 Rush[13]、英國蘭開斯特大學的 Paice 等[14]選取字詞的不同特徵作為提取摘要的關鍵。隨後，有的學者開始引入文檔的結構特徵和語義特徵。美國耶魯大學的 Schank[15]、以及 GE 開發中心的 Rau 等[16]通過分析和推理得到文檔的摘要。Sasha Blair-Goldensohm 等提出了 SC 演算法[17]，其核心思想是: 包含越多句子的類的代表句子就越重要。因此，首先將句子聚類，根

據每個類中的句子數目決定類的重要度，抽取重要的類的代表句子作為文摘。

我國對自動文摘的研究起步較晚，中文文摘系統的研究在 20 世紀 90 年代才發展起來。上海交通大學 1997 年研製了 OA 中文文獻自動摘要系統[27]。該系統集成了位置法、指示短語法、關鍵字法和標題法等多種方法；復旦大學提出了一種基於統計的文本自動綜述方法，該方法利用文檔內和文檔之間段落的語義相關性，實現多文檔的自動綜述[25]；哈爾濱工業大學從各級文本單元的話語關係研究入手,研究跨文本單元的相似關係識別、文本時間資訊抽取以及事件的時序關係識別、文本內部修辭結構識別以及文本集合的層次主題的識別等, 並提出了基於修辭結構理論的多文檔文摘方法[26][28]。此外，還有很多大學和研究機構都取得了許多重要理論成果，實現了一批應用系統。

3.1 LexRank 演算法

自動文摘是通過選取原始文本中一組最重要的句子實現的。這裏如何定量地評定句子的重要度（本文中稱為權重）成為文摘選取的關鍵。密西根大學的 Gunes Erkan 和 Dragomir R Radev 提出的 LexRank 演算法[18]是一種在句子的圖形表示下計算句子權重的方法，他們認為如果一個句子與很多其他句子相似，那麼這個句子就是比較重要的。

首先把給定的文檔分句，並計算句子之間的相似度，如果兩個句子之間的相似度大於給定的閾值，就認為這兩個句子語義相關並將它們連接起來。按照這種方法，可以得到一個無向圖 $G=(S,E)$，圖中的每個節點 $s\in S$ 對應一個句子，而邊 $(s_i,s_j)\in E$ 表示句子 s_i 與 s_j 是相關的。節點 s 的度 d 是與 s 相連的邊的數目, 反映了其所對應句子所包含資訊的重要程度: d 越大，則對應句子所關聯的句子數目越多，那麼這個句子所包含的資訊越重要；反之亦成立。另一方面，如果一個節點的度比較大, 那麼與之相關聯的句子也相應的比較重要。這樣通過計算句子間的相似度構建圖 G，然後根據句子間的連接迭代計算句子所包含的信息量，再從中選取包含信息量最多的一組句子作為文摘。

在這個過程中我們發現，句子相似度計算的好壞對最後結果有較大的影響。原有演算法是根據句子的關鍵字特徵來計算相似度的，我們使用上文中改進的相似度計算方法，加入了語義和句子長度、句式特徵等資訊。同時，我們在計算句子權重時不僅考慮了句子通過 LexRank 得到的結果，而且加入了句子位置、指示性短語、句子長度等多個特徵；對於多文本文摘冗餘的問題也做了相關的改進工作。

3.2 改進的自動文摘系統

上一部分中已經介紹了 LexRank 演算法的核心思想，下面將介紹我們的自動文摘系統，其中將對句子權重計算方法以及去冗餘處理模組進行詳細講述。系統是在向量空間模型(Vector Space Model, VSM)上實現的，其基本流程如圖 3 所示。包含以下六個步驟：

1）預處理：將輸入的文本或文本集的內容切分成句子並標號，對不可能成為文摘句的疑問句、反問句等進行句子過濾，分詞、去除停用詞；

2）計算句子的相似度，得到句子的相似度矩陣；

3）計算句子的權重：其中將結合 LexRank 及其它特徵。

4）根據句子的權重抽取句子，並去除冗餘；

5）轉到 4）直到抽取的句子數目滿足條件；

6）結合專家文摘對文摘結果進行評測。

3.2.1 句子權重的計算

得到句子的相似度矩陣之後，通過 LexRank 演算法可以計算得到句子的權重，我們記作 LexRankSco- re。但是這樣得到的權重沒有涉及到句子位置、指示性短語等相關資訊。我們的權重計算過程如圖 4 所示。

句子位置對文摘結果有一定的影響，據統計每段首句所包含的信息量較大。我們用 PositionScore 表示句子位置的加分，如果句子 i 不是首句，PositionScore 等於 0。

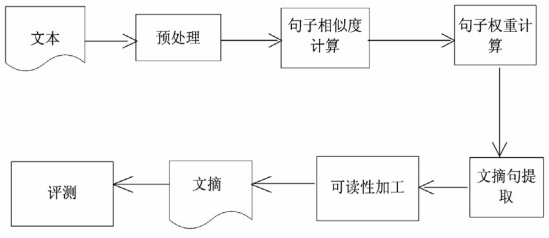


圖 3. 文摘的自動抽取流程圖

Figure 3. The workflow diagram of automatic abstracting

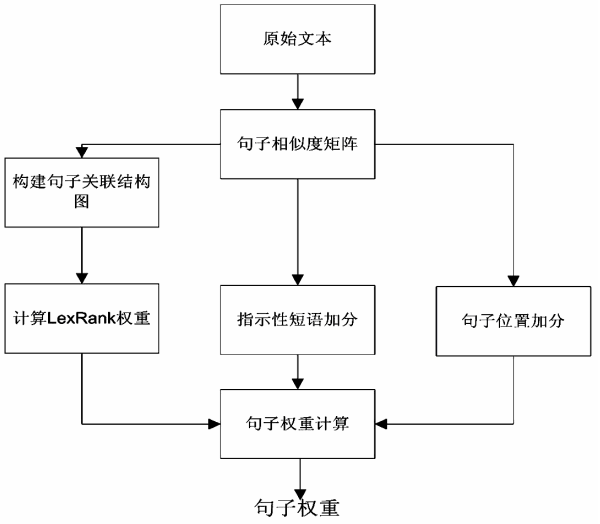


圖 4. 句子權重計算圖

Figure 4. The workflow for computing sentence-weight

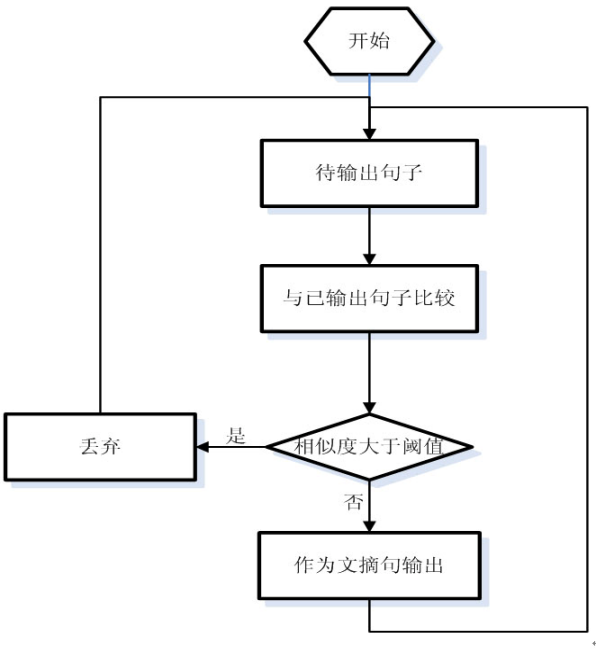


圖 5. 冗餘處理流程圖

Figure 5. The workflow for redundancy resolution

文章中常常有一些特殊的指示性短語(例如 in this paper…the purpose of the article…等), 它們對文章主題具有明顯的提示作用，可以利用它們來獲取文章的主題資訊。我們使用 InforScore 表示指示性短語所在的句子的加分，如果句子 i 中不包含指示性短語，其對應的 InforScore 等於 0。

綜上計算得到三個結果之後，我們使用使用公式(8)計算句子的權重 q_i。

$$q_i = LexRankScore + PositionScore + InforScore \quad \textbf{(8)}$$

3.2.2 冗餘處理

在自動文摘過程中，如果被抽取的句子意思相同，會影響最後的文摘所包含的信息量，這就是我們所說的冗餘問題。在多文本文摘中冗餘的消除是影響文摘結果的一個重要方面。我們消除冗餘的步驟如下:

1） 根據句子權重計算結果，對句子的重要度由高到低排序，得到候選的句子列表 $S = \{s_1,s_2,\ldots,s_n\}$，文摘集合 A 為空。

2） 權重最大的 s_i 作為文摘輸出，$A = \{s_i\}$，$S = S-\{s_i\}$；

3） 依次選舉 S 中的 s_i (i≥2), 如果 s_i 與 A 中所有句子的相似度小於等於設定的閾值 threshold，$A=A+s_i$；否則丟棄 s_i；

4） 迴圈步驟 3, 直到抽取的句子數到達一定長度。

4. 實驗

系統可以處理中文（英文）的單文本（多文本）文摘。由於測評語料的限制，我們的實驗分為三個部分，第一部分是中文單文本文摘，包括語料庫的建設和結果測評；第二部分實驗是借助於 2004 年 DUC 英文多文本文摘的測評語料進行的；第三部分主要是針對系統的抗雜訊能力進行測評。測評採用的是 2004 年 DUC 採用的 ROUGE 方法。

4.1 ROUGE 方法

ROUGE 是由 Chin-Yew Lin 等人提出的自動文摘評價方法[19]，它是通過電腦器產生的文摘和專家文摘間重疊的單詞數目來評價文摘品質的。DUC2004 採用了三種 ROUGE 評價方法，分別是 ROUGE-N，ROUGE-L，ROUGE-W[20]。ROUGE-N 計算的是機器摘要與一組人工摘要中 n-gram 的 Recall 值, 它的計算公式如下：

$$\text{ROUGE-N} = \frac{\sum_{s\subset(modelSummaries)} \sum_{gram_n\subset s} Count(Match(gram_n))}{\sum_{s\subset(modelSummaries)} \sum_{gram_n\subset s} Count(gram_n)} \quad \textbf{(9)}$$

其中，n 代表 n-gram 的長度，$Match(gram_n)$表示同時出現在系統文摘和專家文摘中的 n-gram 的數目。ROUGE-L 是根據最長公共子序列來電腦器文摘和人工文摘的相似程度，而 ROUGE-W 是帶權重的最長公共子序列。這兩個評價方法的計算都比較複雜，在此不做贅述。

4.2 中文單文本文摘測評

我們使用的是哈爾濱工業大學的《哈工大資訊檢索研究室單文檔自動文摘語料庫》。包含語料共計 211 篇，分為不同體裁。其中各體裁文章數為：奧運，57 篇；記敍文，40 篇；說明文，40 篇；議論文，46 篇；應用文，18 篇；03 年 863 評測語料，10 篇。並由 5 個人分別人工按照原文 10%以及 20%文摘句。

我們依照 2004 年 DUC 的測評方法，將這五個人人工標注的 10%句子和 20%句子分詞後分別作為 10%專家文摘和 20%專家文摘。將我們系統生成的文摘與專家文摘使用 ROUGE-N 方法測評，這裏的 n-gram 是以中文分詞後的單個詞語為單位。表 1 給出了使用詞特徵的句子相似度計算方法、使用語義特徵的句子相似度計算方法和我們改進後句子相似度計算方法對文摘結果影響的比較。結果表明改進的句子相似度計算方法使得生成的文摘品質有一定的提高。

4.3 英文多文本文摘測評

我們採用的是 DUC 2004 年的多文本測評語料。DUC 提供了五十個 TDT 英文文本集，每個文本集包含十篇同一話題的文章。要求每個文本集自動生成不超過 665 位元組的文摘。並提供了專家文摘，採用 ROUGE 方法對文摘進行評價。表 2 給出了使用詞特徵的句子相似度計算方法、使用語義特徵的句子相似度計算方法和我們改進後句子相似度計算方法對文摘結果影響的比較。

同樣，我們可以看到，使用了改進的句子相似度計算方法的系統生成的文摘品質要好於單純使用詞特徵的句子相似度計算方法或使用語義特徵的句子相似度計算方法。測評結果表明 ROUGE-1 分別提高了 9.5%、6.8%，ROUGE-2 分別提高了 16.3%、12.5%，文摘品質有了一定的提高。

表 1. 中文單文本語料測評結果

Table 1. The evaluation results on Chinese single-document corpus

	10%壓縮率		20%壓縮率	
句子相似度計算方法	ROUGE-1	ROUGE-2	ROUGE-1	ROUGE-2
TF*IDF	0.2898	0.1557	0.4011	0.2235
語義特徵	0.3184	0.1703	0.4076	0.2219
改進後的方法	0.3404	0.2057	0.4668	0.2835

4.4 抗雜訊能力測評

當處理多文本文摘時，LexRank 把所有文本看作一個整體，句子重要性的判斷是根據所有文章內容，而不是局部某一篇文章。所以 LexRank 方法對雜訊資料不敏感。我們的系統由於結合了 LexRank 演算法，在多文本文摘的抗雜訊方面有著一定的優越性。

為了觀察結果，我們對 DUC2004 年的多文本測評語料庫進行了兩種修改：第一種，每個文本集的十篇文章中我們刪除其中一篇文章，再加入一篇與該主題不相關的文章，雜訊資料為 1/10=10%；第二種，在每個文本集中加入兩篇不相關的文章，雜訊為 2/12=16.7%。實驗結果如表 3 所示，結果表明兩種情況下，文摘品質都沒有明顯下降。

5. 結束語

本文介紹了句子相似度計算的幾種方法，並根據其不足提出了一種新的句子相似度計算方法，綜合考慮了句子的關鍵字特徵、語義特徵、句式特徵和句子長度特徵等資訊。在此基礎上實現了一個基於 LexRank 演算法的自動文摘系統，並從句子權重計算

表 2. 英文多文本語料測評結果比較

Table 2. The evaluation results on English multi-document corpus

句子相似度計算方法	ROUGE-1	ROUGE-2
TF*IDF	0.3433	0.0814
語義特徵	0.3521	0.0842
改進後的方法	0.3759	0.0947

表 3. 加入噪聲數據後文摘的測評結果

Table 3. The evaluation results on English multi-document corpus with certain percent of noisy data

語料庫 噪聲數據	ROUGE-1			ROUGE-2		
	min	max	average	min	max	average
無	0.3213	0.4245	0.3759	0.0511	0.1465	0.0947
10%	0.3078	0.4164	0.3677	0.0504	0.1367	0.0903
16..7%	0.2747	0.4049	0.3465	0.0487	0.1198	0.0843

和冗餘處理等方面進行了改進。該系統可以對中文或英文的單文本或多文本進行自動文摘，通過在哈工大和 DUC 的測評語料上進行實驗，測試結果表明該系統在一定程度上改進了文摘的品質，在多文本文摘中的抗雜訊方面也有一定的優越性。

但是，對於多文本文摘，文摘句輸出順序問題仍沒有解決；而且實際上句子作為文摘的最小單元不是最理想的。這是由於有時在一個句子中還會包含冗餘資訊，有時一個句子表達的意思還不夠完整。於是有人提出了對句子進行壓縮和融合[23]，就是通過句法分析和統計的方法，對句子進行裁剪，使文摘更加精煉。

中文自動文摘由於缺乏大規模統一的測評語料以及測試平臺，不利於它的研究和發展。隨著更多的中文自然語言資源庫的健全和開放，中文句法分析和等自然語言處理技術的成熟，中文自動文摘會有更大的發展。

REFERENCES

[1] Hu Guo-Quan, Chen Jia-Jun, Dai Xin-Yu. A Examle-based Chinese-English Machine Translation Strategy [J]. Computer Engineering and Design, 2005, 26(4): 900-903. (in Chinese) (胡國全, 陳家俊, 戴新宇. 一種基於實例的漢英機器翻譯策略[J]. 電腦工程與設計, 2005, 26(4): 900-903.)

[2] Zhang Qi, Huang Xuan-Jing, Wu Li-De. A New Method for Computing Similarity between Sentences and Application on Automatic Text Summarization [J]. Journal of Chinese Information Processing. 2005, 19(2): 93-99.(in Chinese) (張奇, 黃萱菁, 吳立德. 一種新的句子相似度度量及其在文本自動摘要中的應用[J]. 中文資訊學報, 2005, 19(2): 93-99.)

[3] K. Chidananda Gowda, E. Diday. Symbolic Clustering Using a New Similarity Measure[J]. IEEE Transactions on System, Man and Cybernetic, 1992, 22(2).

[4] Li S, Zhang J. Journal of Computer Science and Technology, 2008, 17(6): 933-939.

[5] Wei Zhi-Fang, Yu Shi-Wen. A Dependency-based Model for Sentence Similarity Computing [C]. ICCIP'98, 1998. (in Chinese) (魏志方, 俞士汶. 基於骨架依存樹的句子相似度計算模型[C]. 中文資訊處理國際會議. (ICCIP'98), 1998.)

[6] Che Wan-Xiang. Similar Chinese Sentence Retrieval based on Improved Edit-Distance [J]. Chinese High Technology Letters. 2004. (In Chinese) (車萬翔等. 基於改進編輯距離的中文相似度句子檢索[J]. 高技術通訊. 2004.)

[7] Jin Yao-Hong. Text Similarity Computing Based on Context Framework Model [J]. Computer Engineering and Application. 2006(16). (In Chinese) (晉耀紅等. 基於語境框架的文本相似度計算[J]. 電腦工程與應用. 2006(16).)

[8] Pan Qian-Hong. Text Similarity Computing based on Attribute Theory [J]. Chinese Journal of Computer. 1999: 22(6). (In Chinese) (潘謙紅等. 基於屬性論的文本相似度計算[J]. 電腦學報. 1999: 22(6).)

[9] Chatterjee N. A Statistical approach for similarity measurement between sentences for EBMT. 1999.

[10] Luhn H P. The Automatic Creation of Literature Abstracts [J]. IBM Journal of Research and Development, 1958: 159-165.

[11] Edmundson, Wyllys. Automatic Abstracting and Indexing: Survey and Recommendations. Communication of the ACM, 1961, 4(5): 226-234.

[12] Edmundson. New methods in automatic abstracting [J]. Journal of the Association for Computing Machinery, 1996, 16(2): 264-285.

[13] Pollock J J, Zamora A. Automatic Abstracting Research at Chemical Abstracts Service [J]. Journal of Chemical Information and Computer Sciences, 1975, 15(4): 226-232.

[14] Paice C D. The Automatic Generation of Literature Abstracts: An Approach Based on the Identification of Self-Indicating Phrases. Information Retrieval Research.

[15] Schank C, Abelson P. Scripts, Plans, Goals, and Understanding: An Inquiry into Human Knowledge Structures [J]. Lawrence Erlbaum Associates, Hillsdale, New Jersey, 1977.

[16] Lisa F Rau, Jacobs P S. SCISOR: Extracting Information Online News[J]. Communication of the ACM, 1990, 33(11): 88-97.

[17] S Blair-Goldensohn. Columbia University at DUC 2004[C]. In DUC '04, 2004.

[18] Gunes Erkan, Dragomir R Radev. LexRank: Graph-Based Centrality as Salience in Text Summarization [J]. Journal of Artificial Intelligence Research 22(2004), 12/2004.

[19] Lin, Chin-Yew, E. H. Hovy. Automatic Evaluation of Summaries Using N-gram Co-occurrence Statistics [J]. In Proceedings of 2003 Language Technology Conference (HLT-NAACL 2003), Canada, 2003.

[20] Zajic, David, Bonnie Dorr, Richard Schwartz. BBN/UMD at DUC-2004: Topiary. In Proceedings of the Fourth Document Understanding Conference (DUC '04), 2004: 112-119.

[21] Huang Li-Qiong. Research on Chinese Automatic Summarization and Its Evaluation Method [D]. Chongqing University, 2007. (In Chinese) (黃麗瓊. 中文自動文摘及評價方法的研究[D]. 重慶大學, 2007).

[22] Chin-Yew Lin, Eduard Hovy. The Potential and Limitations of Automatic Sentence Extraction for Summarization [J]. University of Southern California, 2008: 73-80.

[23] Lin, C. Y. Improving summarization performance by sentence compression: A pilot study[C]. In Proceedings of the Sixth International Workshop on Information Retrieval with Asian Languages. 2003: 1-9.

[24] Qin Bing, Liu Ting, Li Sheng. Summarization Based on Physical Features and Logical Structure of Multi Documents[J]. High Technology Letters, 2005, 11(2): 133-136.

[25] Zheng Yi, Huang Xuan-Jing, Wu Li-De. Research and Implementation of Automatic Multi-Document Summarization System [J]. Journal of Computer Research and Development. 2003, 40(11): 107-110. (In Chinese) (鄭義，黃萱菁，吳立德. 文本自

動綜述系統的研究與實現[J]. 電腦研究與發展. 2003, 40(11): 107-110.)

[26] Xu Yong-Dong. Research on Key Technology of Multiple Documents Automatic Summarization [D]. Harbin Institute of Technology, 2007. (In Chinese) (徐永東. 多文檔自動文摘關鍵技術研究[D]. 哈爾濱工業大學, 2007.)

[27] Wang Yong-Cheng, Xu Hui-Min. OA Automatic Abstracting System on Chinese Documents [J]. Journal of the China Society for Scientific and Technical Information, 1997, 16(2): 128-132. (In Chinese) (王永成, 許慧敏. OA 中文文獻自動摘要系統[J]. 情報學報, 1997, 16(2): 128-132.)

[28] Xu Yong-Dong, Xu Zhi-Ming, Wang Xiao-Long. Multi-Document Automatic Summarization Technique based on Information Fusion [J]. Chinese Journal of Computers, 2007, 30(11): 2049-2054. (In Chinese) (徐永東, 徐志明, 王曉龍. 基於資訊融合的多文檔自動文摘技術[J]. 電腦學報, 2007, 30(11): 2049-2054.)

[29] Wang Ji-Cheng, Wu Gang-Shan, Zhou Yuan-Yuan, Zhang Yan-Fu. Research on Automatic Summarization of Web Document Guided by Discourse [J]. Journal of Computer Research and Development. 2003, 40(3): 398-405. (In Chinese) (王繼成, 武港山, 周源遠, 張福炎. 一種篇章結構指導的中文 Web 文檔自動摘要方法[J]. 電腦研究與發展. 2003, 40(3): 398-405.)

[30] Wang Meng. Research of Chinese Text Automatic Summarization Based on Conceptual Vector Space Model [D]. Department of Computer Science Central China Normal University, 2005. (In Chinese) (王萌. 基於向量空間模型的中文自動文摘研究[D]. 華中師範大學, 2005.)

[31] Xiaohua Zhou, Xiaodan Zhang, Xiaohua Hu. Dragon Toolkit: Incorporating Auto-learned Semantic Knowledge into Large-Scale Text Retrieval and Mining [C]. In proceedings of the 19th IEEE International Conference on Tools with Artificial Intelligence (ICTAI). 2007: 29-31.

