重みつき意味ネットワークを用いた複数テキスト要約手法

日大生産工 (院)　○　佐藤　裕介
日大生産工　　松田　聖

1　序論

近年,ADSL や光通信などの大容量かつ, 高速なネットワーク環境が自宅で手ごろに体験できるようになった. それにつれ, 情報の発信元が増加し, 大容量で多種多様の情報を得ることができる. しかし, 多種多様になるにつれ, そのような情報をうまく扱い, 知りたい情報を的確に得ることは大変困難な事となっている. そのような大量の情報から, 重要な情報を抽出する手法としてテキスト自動要約がある. 本研究では, 複数テキストを一つに要約する複数テキスト要約手法について述べる. 複数テキスト要約手法には様々ある [1][2] が, 本手法では, 既に著者が示した単一テキストドキュメントに対する自動要約手法『重み付き意味ネットワークによる英文テキスト要約手法』[3] を拡張させ, 複数テキストドキュメントに対し要約を行う. 複数テキストドキュメントを要約する際に様々な問題が生じる. このような問題を解決する事も本研究の目的のである.
次節において複数テキスト要約と基本概念を説明し,3 で本手法の概要を説明する.4 では研究結果を示し,5 で考察を行なう.

2　複数テキスト要約

複数テキスト要約とは, 複数のテキストドキュメントから, 特定の単語 (キーワード) に関する情報を抽出して, 最低限の情報でユーザに示す手法である. 複数のテキストドキュメントを要約する時に最初に考えられるのは, 単一テキストドキュメントの要約文を集めることである. 本研究も基本的にはこの方法で行なう. しかし, 単純な単一テキストドキュメントの要約文の集合では, 以下の問題が生じてくる.

1 冗長の文が出てくる

2 要約文の量が膨大になる

1 の問題は冗長性に関する問題であり, 複数のテキストドキュメントにキーワードに関して同じような事柄が記載されていた場合, それらを抽出すると要約文が冗長の文ばかりになってしまう. もちろんの事ながら, 要約文が冗長の文ばかりでは意味がない. この問題を解決するためには二つの文の内容がどれほど一致しているかを図る指針が必要である. 本手法では, この指針に特徴ベクトルを使用する. 特徴ベクトルは文自体の特徴を表わしたもので, この特徴ベクトルを見比べることで冗長であるかどうかを判断できる. 詳しくは 2.3 で説明する.
2 の問題は情報が大量であるほど生じる問題である. たとえば, 平均行数が 100 行の 100 個のテキストドキュメントからそれぞれ 10 %の要約率で要約したとき, 個々の要約文の集合で要約文とした場合, 複数テキストドキュメントの要約文が 1000 行になってしまう. これは平均行数の 10 倍であり, 要約文を読むのも大変である. このような要約文は, 複数テキストドキュメントの要約文として不適切である. 本手法では単語の重み付け手法の一つである tfidf 法 [4] を単語の重み付けに取り入れ, 算出された貢献度を使用し収集した要約文を削減し, 複数要約テキストドキュメントの要約文とする事でこの問題の解決法とする.tfidf 法については, 次で説明する

2.1　tfidf 法

tfidf 法は単語に対する重み付け手法の一つである. また tfidf 法は tf 法と idf 法を組み合わせた手法である.tf 法と idf 法はそれぞれ以下のようなコンセプトを持つ.

tf 法	→	同一文書で繰り返し出現する単語が重要である
idf 法	→	出現する文書数が少ない単語は文書の絞込みに役立つから重要である

Summarizing Many English Documents with Weighted Semantic Networks

Yusuke SATO† and Satoshi MATSUDA

tfidf 法で用いられる tfidf 値は文章に対してその単語が, どれほど検索に役立つかを数値化したものである. 逆に言えばキーワードの tfidf 値を見ることでその文章がどれほどキーワードを説明しているかがわかる.tfidf 値は tf 値と idf 値の二つの値から算出する.tf 値,idf 値は以下のように定義される

$$tf(t) = d(t)$$

$$idf(t) = log\frac{N}{tf(t)} + 1$$

ここで,$d(t)$ は単語 t の出現頻度であり,N は総文章数である. また,$idf(t)$ に 1 を加えるのは $log\frac{N}{tf(t)}$ が 0 である時の対処である.
tfidf 値は上記の二つの値の積で与えられる.

$$tfidf(t) = tf(t) * idf(t)$$

tfidf 値をその単語 (t) の重みとする.

2.2 特徴ベクトル

特徴ベクトルとは, その文の特徴をベクトル化したもので, 要素として文の出現順に名詞を取り, 最後にその文の貢献度をとる. このことより, 特徴ベクトル同士の差が少ない文同士ほど同じ特徴を持つといえる. 以下に特徴ベクトルの構成を示す.

$$\{ \text{名詞 } 1, \text{名詞 } 1, \ldots, \text{名詞 } n, \text{貢献度 } \}$$

例えば, 以下のような文があるとする.

$$Tom\ is\ a\ baseball\ player.$$

この文から特徴ベクトルを抽出すると, 以下のようになる (ただし, この文の貢献度を 10 とする).

$$\{Tom, basebal, player, 10\}$$

最後の要素は, 各々抽出されたテキストドキュメントにおける対象としている文の貢献度である. 貢献度とは, その文がテキストドキュメントに対してどれほど貢献しているかという値であり, 要約を行なうときにこの値を利用する. この貢献度は同じ文でもテキストドキュメントが変われば貢献度も変化するため, 最終的な要約文の抽出には利用するが, 冗長性を調べる時には考慮しない.

2.3 冗長性

このような特徴ベクトルを使用して, 文同士の冗長性を調べる. 冗長性を図るときは, 対象となる二つの特徴ベクトルの差の大きさを冗長度とする. たとえば, ある二つの特徴ベクトル $\overrightarrow{A} = \{a_1, a_2, \ldots, a_n\}$, $\overrightarrow{B} = \{b_1, b_2, \ldots, b_n\}$ が文A, Bからそれぞれ得られたとする. このとき $\overrightarrow{A}\overrightarrow{B}$ 間の冗長度 τ は次のように与えられる.

$$\tau = |\overrightarrow{A} - \overrightarrow{B}|$$

$$= \sqrt{\frac{(a_1-b_1)^2+(a_2-b_2)^2+\ldots+(a_n-b_n)^2}{n}}$$

このとき,$a_{(1\sim n)}$,$b_{(1\sim n)}$ はそれぞれ文字列であるため, 算術的な計算はできない. そのため,$a_i - b_i$ を以下のように定義する

$$(a_i - b_i) = W_{a_i b_i}$$

ここで,$W_{a_1 b_1}$ は, 重み付き意味ネットワーク上の a_1,b_1 間の重みとする. このようにして得られた冗長度 τ が閾値 (今回は 0.9) 以上になるものを冗長と見なす. また,

$$a_i = b_j$$

の時,

$$W_{a_i b_j} = 1$$

となる.
特徴ベクトルは, 文から得られるものであるので, 特徴ベクトルの次元は文の長さに依存する. そのため, 冗長性を測る際に, 対象とする特徴ベクトルの次元がばらばらになってしまう. 次元の違う特徴ベクトルの冗長性を測る場合は, 次元の小さいほうを一つづつシフトし冗長性を測る. 例えば, 以下の二つの特徴ベクトルがある.

$$\overrightarrow{A} = \{a_1, a_2\}$$
$$\overrightarrow{B} = \{b_1, b_2, b_3, b_4\}$$

この次元の異なるベクトルの冗長性を測るとすると, まず最初に $\overrightarrow{B}$ の最初の 2 要素と $\overrightarrow{A}$ の 2 要素の冗長度 ($\tau(1)$) は

$$\tau(1) = \sqrt{\frac{(a_1-b_1)^2+(a_2-b_2)^2}{n}}$$

となる. 次に $\overrightarrow{B}$ の第 2 要素, 第 3 要素と $\overrightarrow{A}$ について冗長度 ($\tau(2)$) を測る. 同様にして,$\overrightarrow{B}$ の第 3, 第 4 要素に対しても $\overrightarrow{A}$ との冗長度 ($\tau(3)$) を測る。このようにして得られたすべての冗長性 ($\tau(1\sim3)$) の最大値を $\overrightarrow{A}$ と $\overrightarrow{B}$ の冗長度とする.

2.4 重み付き意味ネットワーク

ここで, 重み付き意味ネットワークについて概略を説明する. 重み付き意味ネットワークの基本構造は知識表現手法の一つである意味ネットワークと同一だが, 異なる点はアークに対して重みが付加されているという事である. 重みを付加することにより, 物事の意味関係を表すだけではなくその関係の強さを表現することが出来る. さらに, 重みを利用し枝きり等でネットワークを整理することも可能である. 次に重み付き意味ネットワークの基本構造を示す.

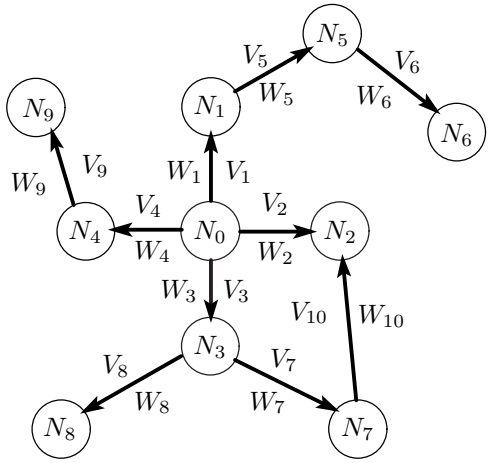

図 1：重み付き意味ネットワークの基本構造図

ここで,$N_{0\sim9}$,$V_{1\sim10}$ はそれぞれ, ノード, アークを表わし,$W_{1\sim10}$ は $V_{1\sim10}$ に対する重みである. 重みとは，アークの関係の強さを表し，重みが大きいアークで表される関係は強い関係である．重みを与えることで，同じノードにつながるノード同士であっても，その関係の強さに差が出てくる．例えば Fig.3 において，N_0 とつながっているノードは N_1, N_2, N_3, N_4 と四つある．それぞれのアークに対する重み W_1, W_2, W_3, W_4 に，

$$W_1 = 0.1, W_2 = 0.7, W_3 = 0.3, W_4 = 0.5$$

という値が与えられたとする．このとき N_0 との関係の強い順に N_2> N_4> N_3> N_1 となる．重みは様々なドキュメントを追加した結果，算出されるものであるので，その関係が他のテキストドキュメントにおいてどれくらい重要な関係であったかという指標にもなる．逆にいうと，重みの大きい関係はどのドキュメントにおいても強い関係を示していたということである．
また重みは, 以下のようにも表記できる.

$$W_{N_iN_j}$$

この表記はノード N_i とノード N_j 間のアークに対する重みを表わしている.

3 要約手法

本手法で用いる要約手法は, 単一テキスト要約手法 [3] を複数テキストドキュメントに拡張したものである. 本手法の手順は次のようになる

1 入力テキストドキュメントすべてに対し要約文を作成する.

2 得られた要約文から特徴ベクトルを作成する.

3 貢献度の順に一定数の要約文を抽出する.

4 特徴ベクトルを使用し冗長の文を削除する.

5 冗長の文がなくなるまで 3,4 を繰り返す

6 最後に残った文を複数テキストドキュメントの要約文とする

手順１において, 各々のテキストドキュメントに対して要約を行うが, このときどのテキストドキュメントに対してもキーワードを最重要単語とする. これにより, 本手法において情報の検索機能も持つことができる. なぜなら, キーワードを含まないテキストドキュメントではすべての単語の重要度が 0 になるため, 要約文が作成できず, 複数テキスト要約には関係しない. つまり, 最後のまとめを行なう時キーワードを含むテキストドキュメントのみを参照しているといえる. 手順２では, 手順１で得られた要約文に対し特徴ベクトルを作成する. 手順３では, 貢献度の高い文から順に抽出する. 今回は抽出する文を 10 文にした. また, 貢献度の最大値の１％未満の貢献度を持つ文は削除される. 手順４では手順３で得られた文に対して冗長があるかどうかを判定する. 冗長な文があれば, 貢献度の低い分を削除し, 新たに文を加えた後もう一度冗長性の判定する. 手順 3,4 を冗長な文がなくなるまで行う. 最終的に得られた文を, キーワードに関する要約文とする.

4　結果

本手法の有用性を検証するために,2 種類のキーワードを使用した. 一つは極く限られたデータにしか出現しないようなキーワード, もう一つは多数のデータに出現するようなキーワードである. 前記のようなキーワードとして”momotaro”を使用した, 後記のようなキーワードとして”dog”を使用した.”momotaro”は, 本手法で使用したテキストドキュメント群の中でもただ一つのテキストドキュメントにしか存在しないため, この要約文を単一テキスト要約と見なすことができる. つまり, 本手法では単一 (もしくは, 少量) のテキスト要約にも十分対応できることが示された. 次に”dog”を使用した場合の出力結果の一例を示す.

1 momotaro , the dog , the monkey and the pheasant were sailing but could not see the island , so the pheasant went up in the sky .

2 then momotaro and dog and monkey met a pheasant .

以上の 2 文のうち,1 は要約文として採用され,2 は冗長が高いとして削除された文である. いかに, それぞれの特徴ベクトルを示す,

1:{momotaro,dog,monkey,pheasant,island ,pheasant,sky,96}
2:{momotaro,dog,monkey,pheasant,85}

1 と 2 の文が冗長であるのは特徴ベクトルを観ると一目瞭然である. なぜなら 1 の特徴ベクトルの第１〜第４要素までが 1 の特徴ベクトルと一致しているからである. また, 文章的にも 1 の文が在れば,2 の文は規定の事実と言え, 無くても意味は通るため, 本手法の正確性が明らかにされたといえる.
次に, 要約文の取得元に着目する. 全 10 文のうち,5 文が ”AROUND THE WORLD IN EIGHTY DAYS .txt”からの抽出文であり,4 文が”THE WONDERFUL WIZARD OF OZ.txt”から,1 文が ”momotaro.txt”の抽出文である. このように, キーワード”dog”は本手法で使用したテキストドキュメント群の中でも $\frac{2}{3}$ ほどのテキストドキュメントに出現するキーワードでの要約であるが, 特定の数テキストドキュメントから抽出されていることが判る. このことは, 本手法が, 自動的に大量のテキストドキュメントの中から必要なテキストドキュメントのみを抜き出しているということが判る.

5　考察

極く限られたデータにしか出現しないようなキーワードと, 様々なデータに出現するようなキーワードに対しての本手法の要約が有効であることが示されたことで, 要約対象であるテキストドキュメント群からどのようなキーワードに対しても関連するテキストドキュメントのみを選択し, 要約文を抽出することができることができた. また, 特徴ベクトルを用いたことで, 文と文の冗長性を測ることが出来, 複数テキストドキュメントを要約する事で生じる問題の解決にもなった. さらに,tfidf 法を使用することにより, 複数テキストドキュメント間のキーワードに対する関連性をも測ることが出来た.
今後の課題として次のようなことが挙げられる. まずは, 要約文を一つの文章にすることである. これは単一テキスト要約, 複数テキスト要約関係なくいえることであるが, 文章を抽出するだけでは, 箇条書きの状態でしか要約文を作れない. そのため箇条書きではなく, 意味の通るように一つの文章に変換することが求められる. これを行なうには, 冗長性のみではなく文と文との関係を表わすような指標が必要である. また, 要約対象のテキストドキュメント群が多くなると, 処理時間も増えていく. そのため, すべてのテキストドキュメントを一気に要約するのではなく, 逐次的な要約を行なう必要がある.

参考文献

[1] 尾崎直観, 松尾豊, 石塚満. 関連する複数新聞記事からの重要文抽出法. 第 3 回 MYCOM 資料,pp80-86,2002.

[2] Inderjeet Mani 著、奥村学、難波英嗣、植田禎子 共訳 (2003),『自動要約』, 共立出版

[3] 松田聖, 佐藤裕介. 重み付き意味ネットワークによる英文要約手法. 情報処理学会　研究報告,2004-DD-42 pp1-6,2004

[4] 長尾真　編 (1996),『自然言語処理』, 岩波書店

[5] 小嶋秀樹： 単語の意味的な類似度の計算, 電子情報通信学会技術研究報告, AI92-100, pp.81-88, 1993