Support Vector Machineを用いた重要文抽出法

平尾 努[†] 磯崎秀樹[†]
前田英作[†] 松本裕治[††]

文書から重要な情報を持った文を抽出する重要文抽出技術は，文書要約技術の１つであり，より自然な文書要約を実現するための基盤技術である．重要文の抽出精度を高めるためには，複数の手がかりを統合的かつ効果的に扱うことが必要とされており，機械学習手法を取り入れた重要文抽出法が着目されつつある．本稿では，汎化能力の高い機械学習手法とされるSupport Vector Machine（SVM）を用いた重要文抽出手法を提案する．Text Summarization Challenge（TSC）のデータを用いて評価実験を行い，提案手法はLead手法などの従来手法と比較して統計的に有意な差で優れていることを実証した．また，野本らのデータを用いた評価実験でもこれに近い成績が得られた．さらに，文書のジャンルを考慮することで重要文の抽出精度が向上すること，重要文抽出に有効な素性のジャンルによる違いを明らかにした．

Important Sentence Extraction Based on Support Vector Machines

Tsutomu Hirao,[†] Hideki Isozaki,[†] Eisaku Maeda[†]
and Yuji Matsumoto[††]

Extracting from a text the sentences that contain important information is a form of text summarization. If done accurately, it supports the automatic generation of summaries similar to those written by humans. To achieve this, the algorithm must be able to handle heterogeneous information. Therefore, parameter tuning by machine learning techniques have received attention. In this paper, we propose a method of sentence extraction based on Support Vector Machines (SVMs). To confirm the performance of our method, we conduct experiments on the Text Summarization Challenge (TSC) corpus and Nomoto's corpus. Results on the former show that our method is better (statistically significant) than the Lead-based method. Moreover, we discover that document genre is important with regard to extraction performance; the effective features of each genre are clarified.

1. はじめに

重要文抽出は，文書を構成する文集合から重要な文のみを抽出する技術である．重要文抽出によって得られる文の集まりは結束性を欠く場合があるものの，それ自体を一種の要約と見なすことも可能である．したがって，この重要文抽出は，計算機を用いて自然な要約を実現する自動要約技術の重要な基本技術の１つである．

重要文抽出に関する研究は1950年代後半[13]に始まる．以後，数多く行われた研究の多くでは，何らかの手がかりに基づき文の重要度を決定し，あらかじめ与えられた要約率を満たすように重要度の高い順に文を抽出する．文の重要度を決定する手がかりとしては，文を構成する単語の重み，文の出現位置，文書構造，手がかり表現などがあり，これらを統合して扱うことが効果的である[3]．たとえば，Edmundson[3]，Nobataら[15]は，各手がかりを定量化して数値によって表現し，その重み付き線形和を文の重要度とする手法を提案した．しかし，重みを人手によって決定しているため，手がかりの数が増えるにつれ，適切な重みの値を見つけることが難しくなるという問題があった．

一方，大量の学習データが利用できる場合には機械学習手法を用いて複数の手がかりを効果的に扱うことが可能であり，近年，重要文抽出においても機械学習の利用が注目されている．Kupiecら[10]やAoneら[1]はベイズ分類器を用いた手法，Maniら[14]，野本ら[25]，奥村ら[23]，Lin[12]は決定木学習を用いた手法を提案し

† 日本電信電話株式会社NTTコミュニケーション科学基礎研究所
NTT Communication Science Laboratories, NTT Corporation
†† 奈良先端科学技術大学院大学情報科学研究科
Graduate School of Information Science, Nara Institute of Science and Technology

ている．しかし，十分な量の学習データを集めることができない場合や，計算量の観点から十分な量の学習データを扱うことができない場合には，用いる機械学習手法の種類と用いる手がかりの数や種類を適切に選択しなければならない．ところが，重要文抽出のための学習データセットは文書分類などの他のタスクと比べデータ数が少なく，重要文抽出に有効な手がかりの種類と数は必ずしも明確ではない．したがって，重要文抽出のようなタスクに対しては汎化能力の高い機械学習手法を用いることが望ましい．

機械学習手法の 1 つである Support Vector Machine [18]（以下，SVM）は，従来の機械学習手法と比較して汎化能力が高いとされ，重要文抽出に適している．文書分類[7),27)]，チャンキング[9)]，係り受け解析[8)]などの自然言語処理タスクに適用されており，その有効性が報告されている．

そこで本稿では，SVM を用いた重要文抽出手法を提案する．国立情報学研究所主催の評価型ワークショップである NII-NACSIS Test Collection for IR Systems Workshop（NTCIR）の一環として開催された Text Summarization Challenge（TSC）[4)]の重要文抽出タスクのデータと野本のデータ[25)]とを用いて評価実験を行い，従来手法に対する提案手法の有効性を検証する．

以下，2 章では SVM の概略と SVM を利用した重要文抽出手法について述べ，3 章で評価実験の概要を説明する．4 章では実験結果を示し，提案手法の有効性を統計的に検証する．さらに，重要文抽出のジャンル依存性を明らかにするとともに，文書のジャンルごとに有効な素性を解析した結果について述べる．また，提案手法で導入した文のランキング法に関する考察を行い，提案手法と TSC の本試験に参加したシステムとの比較も行う．

2. Support Vector Machine に基づく重要文抽出手法

2.1 Support Vector Machine（SVM）

SVM は，Vapnik によって提案された二値分類のための教師あり学習アルゴリズムである[18)]．その概念図を**図 1** に示す．

学習データセットは次のベクトルとして表すことができる．

$$(\mathbf{x}_1, y_1), \cdots, (\mathbf{x}_u, y_u),\ \mathbf{x}_j \in \mathbf{R}^n,\ y_j \in \{+1, -1\}$$

$\mathbf{x}_j$ を事例 j を表す n 次元の特徴ベクトル，y_j を，事例 j が正例であるときに $+1$，負例であるときに -1 となる教師信号とすると，SVM は，$\mathbf{x}_j$ を以下の分離平面で正例，負例に分類する．

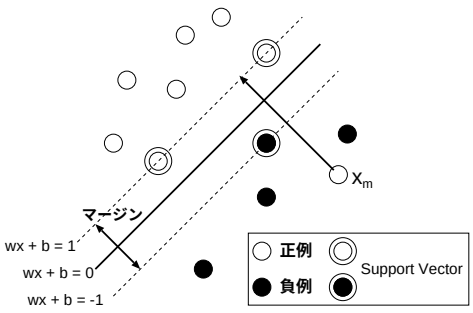

図 1 SVM の概念図
Fig. 1 Support Vector Machines.

$$\mathbf{w} \cdot \mathbf{x} + b = 0,\ \mathbf{w} \in \mathbf{R}^n, b \in \mathbf{R} \tag{1}$$

まず，学習データセットが線形分離可能である場合を考える．学習データを正しく分離する平面は一般的に多数存在するが，SVM ではマージン最大化に基づき最適な分離平面を決定する．マージンを最大化するために $\mathbf{x}_j$ が満たす条件は以下の式となる．

$$y_j\,(\mathbf{w} \cdot \mathbf{x}_j + b) - 1 \geq 0 \tag{2}$$

ここで，マージンは $2/||\mathbf{w}||$ であり，これを最大化するには，以下の 2 次計画問題を考えればよい．

$$\begin{aligned} &\underset{\mathbf{w},b}{\text{Minimize}} \quad J(\mathbf{w}) = \frac{1}{2}||\mathbf{w}||^2 \\ &\text{s.t.} \quad y_j(\mathbf{w} \cdot \mathbf{x}_j + b) - 1 \geq 0 \end{aligned} \tag{3}$$

この 2 次計画問題をラグランジュの未定乗数法を用いて解くことによって最終的な判別関数 $f(\mathbf{x}) = \mathrm{sgn}(g(\mathbf{x}))$ が求まる．$g(\mathbf{x})$ は以下の式となる．ここで，λ_j は正のラグランジュ乗数である．

$$g(\mathbf{x}) = \sum_{j=1}^{u} \lambda_j y_j (\mathbf{x}_j \cdot \mathbf{x}) + b \tag{4}$$

さらに，学習事例が線形分離不可能な場合には，各学習事例に関して非負の ξ_j を導入して式 (2) の条件を緩めることができる．このとき，以下の最小化問題を解くことによって最適な $\mathbf{w}$，b が求まる．

$$\begin{aligned} &\underset{\mathbf{w},b,\xi}{\text{Minimize}} \quad J(\mathbf{w},\xi) = \frac{1}{2}||\mathbf{w}||^2 + C\sum_{j=1}^{u}\xi_j \\ &\text{s.t.} \quad y_j(\mathbf{w} \cdot \mathbf{x}_j + b) - (1 - \xi_j) \geq 0 \end{aligned} \tag{5}$$

式 (5) 右辺の第 1 項はマージンの大きさに関する項，第 2 項は正しく分離できなかった学習事例（図 1 の $\mathbf{x}_m$）に対するペナルティ項である．C はこれら 2 つの項のトレードオフを決めるパラメータである．SVM の最小化問題は，凸 2 次計画法の問題に帰着する．このとき，$g(\mathbf{x})$ は式 (4) と同様となる．$\lambda_j \neq 0$ となる $\mathbf{x}_j$ は support vector と呼ばれ，判別関数は support

vector のみによって記述される．

さらに，ベクトルの内積によって定義される Kernel 関数を用いることによって，線形 SVM のアルゴリズムを容易に非線形に拡張することができる．この非線形 SVM を使うと少ない計算量で複雑な分離平面を記述することができる．これは式 (4) の内積を Kernel 関数（$K(\mathbf{x}\cdot\mathbf{x}_i)$）で置き換えることで実現される．したがって，$g(\mathbf{x})$ は以下の式となる．

$$g(\mathbf{x}) = \sum_{i=1}^{u} \lambda_i y_i K(\mathbf{x}_i, \mathbf{x}) + b \qquad (6)$$

本稿では，学習結果の可読性に優れている polynomial 関数（式 (7)）を Kernel 関数として用いる．

$$K(\mathbf{x}, \mathbf{y}) = (\mathbf{x}\cdot\mathbf{y} + 1)^d \qquad (7)$$

2.2 SVM による文のランキング

重要文抽出は，文書中の各文に対して重要（正例），非重要（負例）のラベルを付与する二値分類問題としてとらえることができる．したがって，学習事例が与えられれば，前節で説明した SVM を用いて重要文，非重要文の特徴を学習し，未知事例として入力された文書中の各文を重要文，非重要文に分類することができる．

ただし，SVM を含む機械学習手法を用いた場合，ある要約率を設定して学習を行ったとしても，テストにおいて，文書中の何割の文が重要文として分類されるかは分からない．一般的に重要文抽出タスクでは，重要文の数が要約率で指定されるため，与えられた要約率に対応できるように文をランキングすることが必要となる．そこで本稿では，$g(\mathbf{x})$ の値の大きいものから順に与えられた要約率まで重要文として採用する．なお，$g(\mathbf{x})$ によるランキングの有効性については，考察で詳しく述べる．

2.3 素　　性

任意の文 S_i を表す特徴として，以下に述べる素性を用いた．利用した素性は，過去の研究報告を参考にするだけでなく，本稿独自の新しい 2 種類の素性として，係り受け構造を考慮した TF·IDF と文に含まれる固有表現の種類を導入した．

また，素性ベクトル $\mathbf{x}_j$ の各要素は 0 か 1 の二値となるように $\mathbf{x}_j$ を定義した．たとえば，文の長さ $\mathrm{Len}(S_i)$ のように二値とならない特徴は，まず文書内での $\mathrm{Len}(S_i)$ の最大値で割ることによって $[0,1]$ の値に正規化し，その後，正規化した値が $[0,1]$ を 10 分割した区間 $[0.0,0.1)$,$[0.1,0.2)$,$\cdots$,$[0.9,1.0]$ のどこに属するかを表す 10 次元の二値ベクトルに変換した．たとえば，$\mathrm{Len}(S_i) = 0.75$ であれば，これがベクトル (0000000100) に変換され，素性ベクトル $\mathbf{x}_j$ の要素のうちの 10 個となる．こうして，最終的に，各文の素性ベクトル $\mathbf{x}_j$ の次元 n は 546 となる．以下に用いた各素性の定義を示す．

文の位置[3)]

文 S_i の位置を表す特徴として，文書全体における位置 $\mathrm{Posd}(S_i)(1 \le r \le 10)$ とパラグラフ内における位置 $\mathrm{Posp}(S_i)(11 \le r \le 20)$ を以下の式で定義した．ただし，r はベクトル $\mathbf{x}_j$ の要素番号を表す．

$$\mathrm{Posd}(S_i) = 1 - \mathrm{BD}(S_i)/|D|$$
$$\mathrm{Posp}(S_i) = 1 - \mathrm{BP}(S_i)/|P|.$$

ここで，$|D|$ は S_i を含む文書 D の文字数，$\mathrm{BD}(S_i)$ は，文書の先頭から S_i までの文字数である．$|P|$ は S_i を含むパラグラフ P の文字数であり，$\mathrm{BP}(S_i)$ はそのパラグラフの先頭から S_i までの文字数である．したがって，$\mathrm{Posd}(S_i)$，$\mathrm{Posp}(S_i)$ が大きいほど文 S_i が文書およびパラグラフの先頭に近いことを表す．

文の長さ[10)]

文 S_i の長さを表す特徴として，$\mathrm{Len}(S_i)(21 \le r \le 30)$ を以下の式で定義した．

$$\mathrm{Len}(S_i) = |S_i|$$

$|S_i|$ は文 S_i の全文字数を表す．実際には，本節の冒頭で述べたように $\mathrm{Len}(S_i)$ は，$[0,1]$ に正規化した後，10 次元のベクトルに変換している．

TF·IDF[20)]

文 S_i に含まれる単語の重み（TF·IDF 値）に基づいた特徴として，$\mathrm{TI}(S_i)(31 \le r \le 40)$ を以下の式で定義した．

$$\mathrm{TI}(S_i) = \sum_{t\in S_i} tf(t, S_i)\cdot w(t, D)$$

ここで，$tf(t, S_i)$ は S_i における単語 t の出現頻度であり，$w(t, D)$ は文書 D における単語 t の TF·IDF 値である．$w(t, D)$ は SMART で用いられている以下の式で定義する．

$$w(t,D) = 0.5\left(1 + \frac{tf(t,D)}{tf_{max}(D)}\right)\cdot\log\left(\frac{|DB|}{df(t)}\right)$$

$tf(t, D)$ は，単語 t の文書 D における出現頻度，$tf_{max}(D)$ は文書 D に含まれる単語の最大頻度，$df(t)$ は，単語 t を含む文書の数である．$|DB|$ は対象とする文書集合に含まれる文書数である．なお，本稿では，形態素解析器「茶筌」[29)]を用いて解析した結果，名詞および未知語と判定されたものを単語とした．以下の記述において，単語とは名詞および未知語を指すものとする．対象文書集合は毎日新聞 94 年，95 年，98 年の記事である．

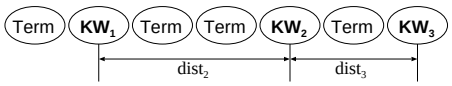

図 2 キーワード密度の定義
Fig. 2 Definition of key words density.

キーワード密度[5),13)]

文 S_i に出現する重要語（キーワード）の出現密度 $\mathrm{Den}(S_i)(41 \leq r \leq 50)$ を計算する式として，文献 11) で提案された以下の式を用いる．

$$\mathrm{Den}(S_i) = \frac{\sum_{t \in KW(S_i)} w(t, D)}{d(S_i)}$$

ここで，$KW(S_i)$ は S_i に出現するキーワードの集合である．その数を $|KW(S_i)|$，$k(\geq 2)$ 番目に出現したキーワードと $k-1$ 番目に出現したキーワードの距離（何単語離れているか）を $dist_k$ として，$d(S_i)$ は，以下の式で定義される．

$$d(S_i) = \frac{\sqrt{\sum_{k=2}^{|KW(S_i)|} (dist_k)^2}}{|KW(S_i)| - 1}$$

$d(S_i)$ は S_i に出現するキーワード間の 2 乗平均距離を表しており，これが小さいことはキーワードが密集して出現していることを意味する．**図 2** の例では，$|KW(S_i)|$ は 3，$dist_2 = 2$，$dist_3 = 1$ である．本稿では，文書 D に含まれるすべての単語の $w(t, D)$ を求め，それらの平均と標準偏差を μ，σ として $\mu + 0.5\sigma \leq w(t, D)$ を満たす t をキーワードとした．

タイトルとの類似度[3)]

文 S_i と S_i を含む文書 D のタイトル T との類似度 $\mathrm{Sim}(S_i)(51 \leq r \leq 60)$ を以下の cosine measure を用いて定義した．

$$\mathrm{Sim}(S_i) = \frac{\vec{v}(S_i) \cdot \vec{v}(T)}{\|\vec{v}(S_i)\| \, \|\vec{v}(T)\|}$$

ここで，$\vec{v}(S_i)$，$\vec{v}(T)$ は，特定の単語が出現したときに 1，出現しなかったときに 0 を要素に持つベクトルである．

係り受け関係を考慮した TF·IDF

係り受け関係を考慮した TF·IDF，$\mathrm{TI}_{dep}(61 \leq r \leq 70)$，$\mathrm{TI}_{wid}(71 \leq r \leq 80)$ を係り受け構造木の最長パスを形成する文節に含まれる単語の集合 t_d と最終文節に直接係る文節に含まれる単語の集合 t_w を用いて以下のように定義した．なお，最長パスが複数ある場合には，文の先頭に近い文節から始まるパスを処理対象とする．

$$\mathrm{TI}_{dep} = \sum_{t \in t_d} w(t, S_i)$$

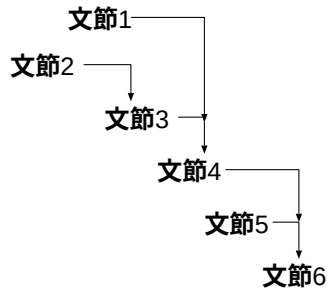

図 3 係り受け解析結果の例
Fig. 3 An example of a dependency structure tree.

$$\mathrm{TI}_{wid} = \sum_{t \in t_w} w(t, S_i)$$

TI_{dep} は最終文節（述語）を修飾する文節の中で最も多くの情報を持っていると考えられる文節集合の重要度，TI_{wid} は述語を直接修飾する文節集合の重要度である．ただし，最終文節を修飾する文節数が重要度に比例することを仮定している．**図 3** の例では，「文節 2，3，4，6」が最長パスを形成しているので「文節 2，3，4，6」に含まれる単語 t の重みの和が TI_{dep} となり，最終文節 6 には，「文節 4，5」が直接係っているのでこれらの「文節 4，5」に含まれる単語 t の重みの和が TI_{wid} となる．係り受け解析には Cabocha☆を用いた．

固有表現

Nobata ら[15)]は，文書のタイトルに出現する固有表現に着目した重要文抽出手法を提案している☆☆．しかし，文書中に出現するすべての固有表現が文の重要度に影響を与えると予想されるので，本稿では S_i に特定の種類の固有表現が存在する場合に 1 をとる素性 $x[r]$ $(81 \leq r \leq 88)$ を定義した．ここでの固有表現とは Information Retrieval and Extraction Exercise（IREX）[17)] の固有表現基準による固有表現および数値表現を指し，以下の 8 種に分類される．

PERSON, LOCATION, ORGANIZATION, ARTIFACT, DATE, MONEY, PERCENT, TIME

また，固有表現の抽出には磯崎のアルゴリズム[6)]を用いた．

接続詞[23)]

S_i の文頭に特定の接続詞が出現した場合に，$x[r] = 1$ $(89 \leq r \leq 138)$ とする．接続詞は 50 種である．

☆ http://cl.aist-nara.ac.jp/~taku-ku/software/cabocha
☆☆ 文献 1) でも固有表現を素性としているが，本稿のように詳細なものではない．

助　詞[23)]

S_i に特定の助詞が出現した場合に，$x[r] = 1$ $(139 \leq r \leq 151)$ とする．助詞は，“格助詞 − 一般”（11 種）とトピックマーカとされる係助詞（「は」,「も」）の計 13 種である．

文末表現（小分類および大分類）[23),25)]

S_i に特定の小分類に属する文末表現が出現した場合に，$x[r] = 1$ $(152 \leq r \leq 172)$ とし，大分類についても同様に $x[r] = 1$ $(173 \leq r \leq 176)$ とする．

文末表現の分類については，福本らの分類[28)]に加え「特殊」,「署名」,「その他」を加えた 21 種を用いた.「特殊」は会話文などのかぎ括弧で終わる文,「署名」は文書の著者を示す文を指す．大分類については，福本らの分類[28)]，田村らの分類[26)]に「その他」を加えた 4 種を用いた．分類ルールについては，文献 26)，28) を参照されたい．分類の詳細を以下に示す．

大分類：意見
　小分類：意見，問掛，要望
大分類：断定
　小分類：断定，推量，理由，判断，義務
大分類：叙述
　小分類：叙述，可能，伝聞，様態，存在，継続，状態，使役，現在，過去
大分類：その他
　小分類：特殊，署名，その他

修辞関係[19)]

田村らの手法[26)]を用いて S_i の修辞関係（計 4 種）を決定し，S_i が S_{i-1} に対して特定の修辞関係である場合に $x[r] = 1$ $(177 \leq r \leq 180)$ として定義した．修辞関係は,「順接」,「転換」,「結論」,「説明」の 4 種である．修辞関係の決定ルールについては文献 26)，28) を参照されたい．

用　言

S_i に出現する用言を日本語語彙大系[24)]を用いて分類し，S_i に特定のクラスの用言が出現したときに $x[r] = 1$ $(181 \leq r \leq 546)$ として定義した．日本語語彙大系における用言の基本分類は 36 であるが，多義語のように複数の基本分類に属する場合があり，こうした「基本分類の組」も 1 つの分類と見なし，合計 366 分類とした．

上述した文の各素性は，文の重要度を表すものと手がかり表現に基づくものとに大別される．文の位置，長さ，TF·IDF，キーワードの密度，タイトルとの類似度，係り受け関係を考慮した TF·IDF は文の重要度を様々な観点から表したものであり，一方，固有表現，接続詞，助詞，文末表現，修辞関係，用言はそれぞれ手がかり表現の 1 種である．

表 1　TSC のデータセット
Table 1　Details of TSC's data sets.

ジャンル	報道	社会	社説	解説	合計
文書数	16	76	41	47	180
文数	342	1,721	1,362	1,096	4,521
重要文数（約 10%）	34	172	143	112	461
重要文数（約 30%）	103	523	414	330	1,370
重要文数（約 50%）	174	899	693	555	2,321

表 2　野本のデータセット
Table 2　Details of Nomoto's data sets.

ジャンル	報道	社説	春秋	合計
文書数	25	25	25	75
文数	440	558	426	1424
重要文数（約 15%）	62	82	60	204

3. 評価実験の設定

3.1 コーパス

評価実験には，重要文抽出の研究のために公開されている 2 種のデータセット，TSC[4)]のデータと野本らのデータ[25)]を利用した．

TSC のデータは，毎日新聞 94 年，95 年，98 年からとられた 180 文書からなる．各文書はあらかじめ，印刷紙面の情報を基に報道，社説，社会，解説の 4 つのジャンルに分類され，各文書に属する文の中から約 10%，30%，50%の要約率☆に応じて重要文が指定されている．その詳細を**表 1** に示す．重要文の決定は，1 文書あたり 1 名の熟練要約筆記者が行い，のべ 3 名の要約筆記者が 2 カ月程度の時間をかけて作成した．

野本らのデータは日本経済新聞 95 年より抽出された 75 文書で構成される．その分野の内訳は，報道，社説，春秋であり，各々25 文書である．1 文書あたり 4 名から 9 名の被験者（大学院生）が約 10%の要約率を目安として，文書中の各文が重要であるか否かの判定をしている．各文書には文書の属するジャンルとともに重要文の判定を行った人数，各文を重要文であると判定した人数が与えられている．本稿では，各文書において半数以上の被験者が重要であると判定した文を正しい重要文と見なし，実験を行った．このとき，平均要約率は約 15%であった．詳細を**表 2** に示す．

3.2 評価指標

TSC の重要文抽出タスクでは各文書の各要約率に対して抽出すべき文数があらかじめ指定されている．したがって，システムがその数だけ重要文を抽出した

☆ 要約率は，文字数ではなく文数に応じて設定されている．たとえば，10 文からなる文書の場合，10%の要約率では 1 文，30%の要約率は 3 文，50%の要約率で 5 文が重要文となる．

場合，Precision，Recall およびそれらの調和平均である F-measure は同じ値となる．システムによって抽出された重要文の数を a，抽出した文集合に含まれる正解重要文の数を b とすると，F-measure は，a/b となるので，この値で評価を行う．なお，野本らのデータに関しても TSC のタスクと同様に抽出すべき重要文の数が既知であるとして同じく F-measure で評価する．

さらに，TSC のデータには，各文書に対して 10%，30%，50%の複数の要約率が設定されている．ここで，人間の作成した正解データに対し，10%の要約率で重要文として選ばれた文が，30%，50%の要約率でも選ばれ，30%の要約率で選ばれた文が 50%の要約率でも選ばれたとする．この場合，10%の要約率で選ばれた文が最も重要な文で，続いて 30%，50%の要約率で新たに選ばれた文が順に重要であると考えることができる．すなわち，文書中の文に粗いランクが付与されてると考えることができる．そこで，こうした文の重要度ランクを考慮してシステムを評価するために難波ら[21]によって提案された pseudo-utility 値を用いた評価も行う．以下にその定義を示す．

$$\begin{aligned}
\text{pseudo-utility}(10) &= \frac{\frac{sys_1}{10} + \frac{sys_2}{30} + \frac{sys_3}{50}}{\frac{rnk_1}{10}} \\
\text{pseudo-utility}(30) &= \frac{\frac{sys_1}{10} + \frac{sys_2}{30} + \frac{sys_3}{50}}{\frac{rnk_1}{10} + \frac{rnk_2}{30}} \\
\text{pseudo-utility}(50) &= \frac{\frac{sys_1}{10} + \frac{sys_2}{30} + \frac{sys_3}{50}}{\frac{rnk_1}{10} + \frac{rnk_2}{30} + \frac{rnk_3}{50}}
\end{aligned} \quad (8)$$

ここで，rnk_1，rnk_2，rnk_3 は，それぞれ，10%，30%，50%の要約率にのみ含まれる文の数，sys_1，sys_2，sys_3 は，システムが出力した重要文のうち，10%，30%，50%のそれぞれの要約率にのみ含まれる正解文の数である．たとえば，ある文書で 10%の要約率での重要文が S_a，30%の要約率での重要文が S_a，S_b，S_c，50%の要約率での重要文が S_a，S_b，S_c，S_d，S_e であったとする．ここで，10%の要約率における重要文としてシステムが S_b を抽出した場合，F-measure では，0/1 となるが，pseudo-utility では $\frac{0/10+1/30+0/50}{1/10}$ となる．

4. 実験結果と考察

提案手法の有効性を検証するために，TSC のデータ[4]と野本らのデータ[25]を用いて評価実験を行った．提案手法，Lead 手法，決定木学習[16]を用いた手法の重要文抽出精度の比較をした．Lead 手法とは文書の先頭から順に与えられた要約率を満たす文までを重要文として抽出する手法であり，新聞記事を対象とした場合には，重要文の抽出精度が高いことが知られている[2]．また，決定木学習のプログラムには，C4.5 を利用した．2.3 節で述べた素性をすべて用い，C4.5 の確信度（certainty factor）によって文のランキングを行った．以下，この手法を C4.5 と略記する．

表 3 各手法の性能評価（TSC データ）
Table 3 Performance of each method (TSC data).

要約率	手法	F-measure	pseudo-utility
10%	Lead	0.374	0.436
	C4.5	0.383	0.501
	SVM(Lin)	0.399	0.519*
	SVM(Poly)	**0.462****	**0.574****
30%	Lead	0.442	0.590
	C4.5	0.403	0.539
	SVM(Lin)	0.469	0.620
	SVM(Poly)	**0.483***	**0.639**
50%	Lead	0.561	0.642
	C4.5	0.576	0.622
	SVM(Lin)	0.621**	0.680
	SVM(Poly)	**0.628****	**0.705**

SVM の Kernel 関数については polynomial 関数を用い，次数 d の値は予備実験により，全般的に良好な結果を得た $d = 2$ とした．以下，この手法を SVM(Poly) と略記する．また，線形 SVM も比較対象とした．これを SVM(Lin) と略記する．式 (5) におけるパラメータ C は，0.0001〜1 まで変化させて得た最適値 0.001 を用いた．SVM のプログラムには TinySVM☆を用いた．

4.1 TSC のデータによる評価結果

TSC の全データ（180 文書）を各ジャンルの文書が均等に配分されるように 5 等分し，学習 4，テスト 1 の比率にわけて交差検定を行った．各要約率ごとに SVM，C4.5 を用いた学習を行い性能評価を行った．F-measure，pseudo-utility による評価結果を**表 3** に示す．なお，pseudo-utility を適用するには人間が作成した正解データにおいて，低い要約率で選ばれた重要文が必ず高い要約率においても選ばれていなければならない．よって，この条件を満たさない 41 文書を pseudo-utility による評価対象から除外した．また，Lead 手法と SVM の抽出精度について Wilcoxon 符合付き順位和検定を行い，Lead 手法に対して有意水準 1%，5%で優れていたものについて，それぞれ **，* で示した．

表 3 より，どの要約率においても SVM(Poly) の成績が最も高く，F-measure では，Lead 手法に対しても統計的に有意な差で優れている．続いて SVM(Lin) の成

☆ http:/cl.aist-nara.ac.jp/~taku-ku/software/TinySVM/

表 4 各ジャンルでの評価結果（F-measure）
Table 4 F-measure of each genre (TSC data).

	SVM (Lin)			SVM (Poly)			Lead		
要約率	10%	30%	50%	10%	30%	50%	10%	30%	50%
報道	0.479	**0.545**	0.612	**0.531**	0.537	**0.654**	0.479	0.505	0.604
社会	0.518	0.539	**0.643**	**0.616**	**0.543**	0.636	0.518	**0.543**	0.615
社説	0.341	0.430	0.578^{**}	**0.364**	$\mathbf{0.432}^{*}$	$\mathbf{0.583}^{**}$	0.316	0.376	0.510
解説	0.229	0.365	0.626^{**}	$\mathbf{0.274}^{*}$	$\mathbf{0.410}^{*}*$	$\mathbf{0.645}^{**}$	0.159	0.324	0.504

表 5 各ジャンルでの評価結果（pseudo-utility）
Table 5 Pseudo-utility of each genre (TSC data).

	SVM (Lin)			SVM (Poly)			Lead		
要約率	10%	30%	50%	10%	30%	50%	10%	30%	50%
報道	0.590	0.689	0.704	**0.656**	0.695	**0.762**	0.594	**0.699**	0.711
社会	0.643	0.701	0.709	**0.711**	**0.709**	0.720	0.589	0.708	**0.724**
社説	0.412	$\mathbf{0.543}^{**}$	0.630^{*}	**0.433**	0.533^{*}	$\mathbf{0.674}^{**}$	0.346	0.431	0.521
解説	0.346^{*}	0.499	0.651^{*}	$\mathbf{0.388}^{**}$	$\mathbf{0.564}^{*}$	$\mathbf{0.681}^{**}$	0.166	0.453	0.557

績が高く，Lead 手法と C4.5 が最も低い．SVM(Poly) と SVM(Lin) の差は，要約率が低くなるにつれて大きくなっている．C4.5 は 30%の要約率では，Lead 手法よりも成績が低く，それ以外の要約率では，Lead 手法とほぼ同等の成績である．一般に，学習データ数を固定したまま素性の数，すなわち特徴空間の次元を高くすると汎化能力が低下する．決定木学習は SVM などに比べ特にその傾向が強い（たとえば，文献 27)）．今回の評価実験では学習に用いた文書数は 144 であり，素性数 546 に対して非常に少ない．この点が C4.5 を用いた場合の抽出精度が Lead 手法とほぼ同様であった一因と考えられる．したがって，決定木学習のための適切な素性選択を行えば C4.5 の抽出精度は改善できる可能性がある．逆に，素性数を減らすことなく高い抽出精度を実現できる SVM に大きな利点があるといえる．

また，F-measure と pseudo-utility を比較すると，10%の要約率で C4.5，SVM ともに値が大きく向上しているが，Lead 手法では値の向上は小さい．つまり，SVM や C4.5 のように位置以外の様々な情報を考慮して文を抽出することが有効であることを示している．

次に，SVM と Lead 手法に関して，表 3 の内訳である各ジャンルごとの F-measure による評価結果を**表 4**，pseudo-utility による評価結果を**表 5** に示す．

表 4 より,「報道」と「社会」では Lead 手法でも抽出精度が高いため，有意差はないが，10%要約率の「社説」を除き,「社説」,「解説」では，有意な差で SVM が優れている.「報道」,「社会」というジャンルでは客観的事実を報道する文書が多く，文書の先頭にそれらの概要など多くの重要事項が書かれている．これに対し,「社説」,「解説」というジャンルは,「報道」,「社会」とは文書構造が異なっており，文書の先頭に重要な情報が偏っていない．このため,「報道」,「社説」と比較すると Lead 手法のような単純な手法では十分な成績が得られない．

表 6 野本らのデータによる評価結果（F-measure）
Table 6 F-measure (Nomoto's data).

	SVM (Lin)	SVM (Poly)	Lead
全平均	0.378	**0.405**	0.373
報道	0.543	**0.593**	0.563
社説	0.327	**0.370**	0.294
春秋	**0.263**	0.253	0.261

表 5 より，各手法の全体的な成績の傾向は表 4 と同様だが，Lead 手法の「社説」,「解説」は抽出精度の向上が小さいことが分かる．特に要約率 10%の場合には顕著である．これは，10%要約率に含まれる文が文書の先頭付近に少ないことに加え，30%要約率，50%要約率に含まれる文も文書の先頭付近には少ないことによる．また，SVM(Lin) と SVM(Poly) の差は表 4 の場合よりもさらに広がっている．

なお，Pentium III 1.0 GHz，メモリ 1 GByte の Linux PC で，TSC の全 180 文書（4,521 文）の学習とテストにかかった時間は 10%要約率では 11.02 秒と 0.54 秒，30%要約率では 23.37 秒と 2.20 秒，50%要約率では 25.6 秒と 3.29 秒であった．

4.2 野本らのデータによる評価結果

TSC のデータの場合と同様に，野本らの全データ（75 文書）を 5 等分し，交差検定により性能評価を行った．F-measure による評価結果を**表 6** に示す．

表 4，表 5 と同様に SVM(Poly) が最も良い成績で，次いで SVM(Lin)，Lead 手法の順となる．文書のジャンルごとの成績に着目すると,「報道」は SVM，Lead

手法ともに良い成績であり，その差は小さい.「社説」は SVM が好成績で，Lead 手法との差が大きい.「報道」と「社説」に関しては，表 4 と同様の傾向であり，その成績も近い．これに対して,「春秋」では，すべての手法で成績が低い．SVM(Lin) が最も高成績ではあるが，他の手法との差はわずかである．これは,「春秋」のようなコラムではわずかな文で文書全体の内容を表すことが困難であり，被験者による重要文の判定の揺れが大きいことが影響していると考えられる.「報道」と「社説」においては，SVM(Poly) が SVM(Lin) より精度が高い.

野本らのデータを用いた評価実験では SVM と Lead 手法の間に統計的な有意差はみられなかった．交差検定を行った 5 組のデータセット間で SVM，Lead 手法ともに抽出精度のばらつきが大きく，これは低い要約率における重要文抽出タスクの特徴でもある．したがって，これら 2 手法間の抽出精度に統計的な有意差があるかどうかの判定を行うにはより多くのデータが必要である．ただし，TSC のデータの評価結果と野本らのデータの評価結果を比較すると「報道」「社説」における成績やジャンル間の成績の差において同様の傾向を示した.

なお，TSC のデータと同様の実験環境での解析時間は，全 75 文書（1,424 文）で，学習に 1.15 秒，テストに 0.1 秒であった.

4.3 重要文のジャンル依存性

表 4，表 6 より，文書のジャンルによって，その抽出精度は大きく異なることが分かる．特に,「社説」,「解説」の成績は「報道」,「社会」と比較すると低い．この原因として，重要文を判定するための有効な素性が文書のジャンルによって異なるということが考えられる．そこで，文書のジャンルごとに学習を行うことによって高い抽出精度が実現できるのではないかと考えられる．このことを確かめるため，以下の手順で実験を行った.

手順 1 ある特定のジャンル i より 36 文書を無作為に抽出し，これで学習を行う.

手順 2 ある特定のジャンル j より学習データと重ならないように 4 文書を無作為に抽出し，これをテストデータとして評価する.

手順 3 すべての組 (i, j) について手順 1〜手順 2 を 10 回繰り返し，その平均値を求める.

ここで，ジャンルの集合は，{ 社会，社説，解説 } にしぼった．TSC のデータからこれらのジャンルを選んだ理由はデータ数が多いこと，複数の要約率が設定されていることである．2 次の polynomial 関数を用いた SVM による実験の結果より，F-measure を**表 7** に pseudo-utility を**表 8** に示す.

どのジャンルの組合せでも，学習データとテストデータのジャンルが同一の場合に最も高い抽出精度が得られている．表 5，表 6 と表 4，表 7 を比較すると若干の例外はあるものの，全体的には，表 4，表 5 よりも学習データが少ないにもかかわらず，成績が向上している．このことから，文書のジャンルを考慮して学習することで，抽出精度が向上することが分かる．そして，文書のジャンルごとに有効な素性が異なるということを示している.

さらに詳しく，ジャンルごとに有効な素性を調べる．2 次の polynomial 関数を用いた場合の $g(\mathbf{x})$ は，$\vec{x} = (x[1], \cdots, x[n])$ として，式 (6) の内積を展開して計算すると以下の式となる.

$$g(\mathbf{x}) = b + \sum_{i=1}^{u} w_i + 2 \sum_{i=1}^{u} w_i \sum_{k=1}^{n} x_i[k]x[k] +$$

表 7 F-measure によるジャンルごとの評価結果（TSC データ）
Table 7 F-measure for three genres (TSC data).

学習 \ テスト	社会			社説			解説		
	10%	30%	50%	10%	30%	50%	10%	30%	50%
社会	**0.603**	**0.572**	**0.656**	0.348	0.389	0.546	0.282	0.372	0.622
社説	0.512	0.489	0.594	**0.375**	**0.516**	**0.634**	0.221	0.395	0.611
解説	0.371	0.449	0.623	0.220	0.405	0.573	**0.303**	**0.446**	**0.649**

表 8 Pseudo-utility によるジャンルごとの評価結果（TSC データ）
Table 8 Pseudo-utility for three genres (TSC data).

学習 \ テスト	社会			社説			解説		
	10%	30%	50%	10%	30%	50%	10%	30%	50%
社会	**0.702**	**0.726**	**0.734**	0.380	0.461	0.574	0.369	0.516	0.651
社説	0.614	0.643	0.675	**0.459**	**0.635**	**0.715**	0.313	0.484	0.645
解説	0.481	0.582	0.660	0.301	0.485	0.606	**0.424**	**0.563**	**0.657**

$$\sum_{i=1}^{u} w_i \sum_{h=1}^{n} \sum_{k=1}^{n} x_i[h]x_i[k]x[h]x[k] \quad (9)$$

ここで，$w_i = \lambda_i y_i$ である．さらに，ベクトルは二値ベクトルであることを利用すると，上の式は以下のように書き換えられる．

$$g(\mathbf{x}) = W_0 + \sum_{k=1}^{n} W_1[k]x[k] + \sum_{h=1}^{n-1} \sum_{k=h+1}^{n} W_2[k,h]x[h]x[k] \quad (10)$$

ここで，$W_0 = b + \sum_{i=1}^{u} w_i$，$W_1[k] = 3\sum_{i=1}^{u} w_i x_i[k]$，$W_2[h,k] = 2\sum_{i=1}^{u} w_i x_i[h]x_i[k]$ である．$W_1[k]$ は個々の単独素性 ($x[k]$) に対する重み，$W_2[h,k]$ は 2 つの素性の組 ($x[h]x[k]$) に対する重みである．このとき，判別関数 $g(\mathbf{x})$ は，$x[k]$，$x[k]x[h](1 \le k; h \le n)$ を新たな素性と見なした 1 種のスコア関数ととらえることができる．したがって，$W_1[k]$，$W_2[h,k]$ の絶対値が大きいということはそれに対応する素性がスコアに大きく影響を与えると解釈することができる．

図 4 に，各ジャンルの全文書を用いて学習したときの素性の重みの分布を示し，**表 9** に重みが 0.5 以上の素性の数を示す．ただし，正の最も大きい重みで割ることによって正規化している．図 4，表 9 の素性は単独のものと組合せのものがあることに注意されたい．「社会」では，重みの高い素性はわずかで，多くは，ほぼ 0 の重みである．これに対して,「社説」,「解説」では，ある程度の大きさの重みを持つ素性が多いことが分かる．これは,「社説」,「解説」では,「社会」よりも多くの素性を考慮すべきことを示している．

ここで，$W_1[k]$，$W_2[h,k]$ の値が正で大きい 10 位までの素性を**表 10** に示し，負で大きい 10 位までを**表 11** に示す．これらの素性は図 4 から分かるように特に高い重みを持った素性である．表 10，表 11 を概観する．正の重みを持つ特徴を見ると文書内の先頭付近に重みが与えられていることが分かる．また，負の重みを持つ特徴を見ると，TF·IDF およびそのバリエーションの値が低いものが多い．しかし，詳細に見ていくと，各ジャンルによって高い重みが与えられた素性やその組合せが異なっていることが分かる．表 8，表 10 で学習とテストが同じジャンルである場合に最も良い成績が得られたことを裏付けている．また，要約率ごとの特徴は，10%，30%の要約率では素性の組合せに対して重みが与えられているが，50%の要約率では単独で重みが与えられている点である．これは，

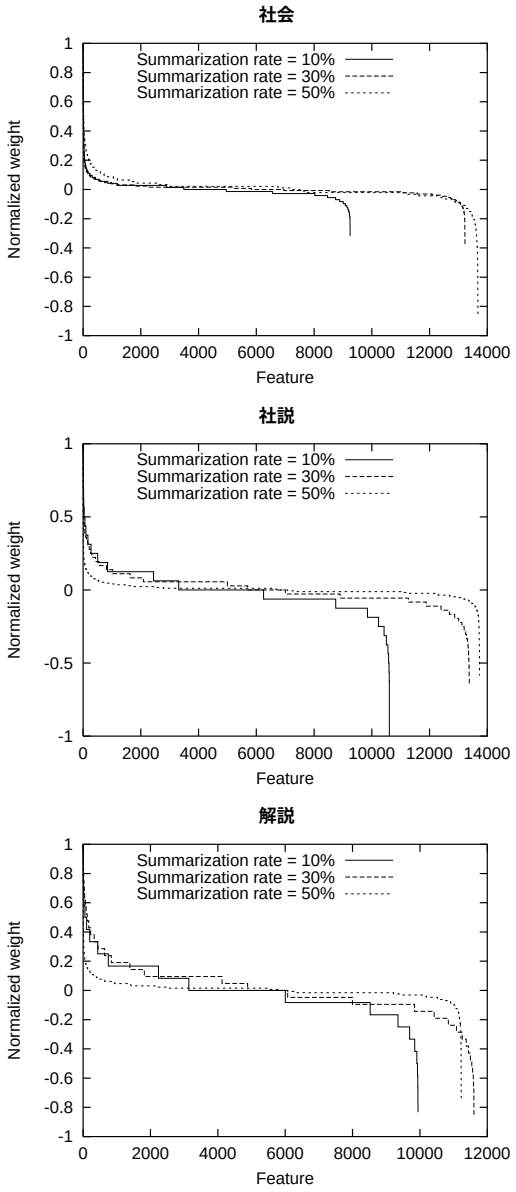

図 4 素性の重みの分布
Fig. 4 Distribution of feature's weight.

表 9 高い重みを持つ素性の数
Table 9 The number of features with high weight.

ジャンル \ 要約率	10%	30%	50%
社会	14	9	21
社説	63	33	4
解説	103	121	7

表 3 において，10%，30%の要約率では SVM(Poly) が SVM(Lin) よりも抽出精度が高いが，50%の要約率では抽出精度がほぼ同等であることも裏付けており，特に低い要約率では素性の組合せに着目することが重要であることが分かる．

表 10 正の重みを持つ素性
Table 10 Effective features and their pairs (positive).

Summarization rate 10%

社会	社説	解説
0.9≤Posd≤ 1.0 ∧ 助詞:「が」	0.9≤Posd≤ 1.0 ∧ 助詞:「が」	0.8≤Posd< 0.9 ∧ 0.6≤TI_{dep}<0.7
0.9≤Posd≤ 1.0 ∧ NE:DAT	0.9≤Posd≤ 1.0	NE:ART
0.9≤Posd≤ 1.0 ∧ NE:ORG	0.9≤Sim≤ 1.0 ∧ 助詞:「を」	0.2≤Den<0.3 ∧ 文末 (大):叙述
0.9≤Posd≤ 1.0	助詞:「で」∧ NE:ORG	0.8≤Posd< 0.9 ∧ 助詞:「は」
0.9≤Posd≤ 1.0 ∧ 0.9≤Posp≤ 1.0	0.9≤Sim≤ 1.0 ∧ 助詞:「が」	0.9≤Posd≤ 1.0 ∧ 助詞:「に」
0.9≤Posd≤ 1.0 ∧ 0.9≤Sim≤ 1.0	0.9≤Posp≤ 1.0 ∧ 助詞:「が」	0.9≤Posd≤ 1.0 ∧ 文末 (小):現在
0.9≤Posd≤ 1.0 ∧ 文末 (大):叙述	0.9≤Posp≤ 1.0 ∧ 0.9≤Sim≤1.0	0.9≤Posd≤ 1.0 ∧ NE:DAT
0.9≤Posd≤ 1.0 ∧ 文末 (小):過去	0.9≤Posd≤ 1.0 ∧ NE:ORG	0.0≤Posd< 0.1 ∧ 助詞:「が」
0.9≤Posd≤ 1.0 ∧ 助詞:「に」	0.9≤Sim≤ 1.0	0.7≤Len< 0.8 ∧ 0.1≤TI_{wid}<0.2
0.9≤Posp≤ 1.0 ∧ 0.9≤Sim≤ 1.0	0.9≤TI_{wid}≤ 1.0 ∧ 文末 (大):意見	0.6≤Len< 0.7 ∧ 0.8≤TI_{dep}<0.9

Summarization rate 30%

社会	社説	解説
0.9≤Posd≤ 1.0	0.9≤Posp≤ 1.0 ∧ 助詞:「が」	0.9≤Posp≤ 1.0 ∧ 助詞:「は」
0.9≤Posd≤ 1.0 ∧ 助詞:「が」	0.9≤Posd≤ 1.0	0.0≤TI_{wid}<0.1 ∧ 助詞:「は」
0.9≤Posd≤ 1.0 ∧ 文末 (大):叙述	0.9≤Posd≤ 1.0 ∧ NE:ORG	NE:ART
0.9≤Sim≤ 1.0	0.9≤Posp≤ 1.0 ∧ NE:ORG	助詞:「から」∧ 助詞:「に」
0.9≤Posd≤ 1.0 ∧ 0.9≤Sim≤ 1.0	0.9≤Posd≤ 1.0 ∧ 助詞:「に」	0.4≤TI_{dep}< 0.5 ∧ 0.4≤Sim<0.5
0.9≤Posd≤ 1.0 ∧ NE:DAT	0.0≤Posd< 0.1 ∧ 助詞:「を」	用言:5
0.9≤Posd≤ 1.0 ∧ NE:ORG	0.9≤Posp≤ 1.0 ∧ 0.9≤Sim≤1.0	0.2≤Posd< 0.3 ∧ 0.9≤Posp≤1.0
0.9≤Posd≤ 1.0 ∧ 助詞:「を」	0.9≤Posd≤ 1.0 ∧ 助詞:「が」	助詞:「を」∧ NE:ART
0.9≤Posd≤ 1.0 ∧ NE:LOC	0.9≤Posd≤ 1.0 ∧ 0.9≤Posp≤1.0	0.9≤Posp≤ 1.0 ∧ 0.9≤TI≤1.0
0.9≤Posd≤ 1.0 ∧ 助詞:「は」	0.9≤Posp≤ 1.0	0.9≤Sim≤ 1.0 ∧ 助詞:「に」

Summarization rate 50%

社会	社説	解説
助詞:「が」	0.0≤Posp< 0.1	助詞:「を」
0.8≤Posd< 0.9	0.0≤Posd< 0.1	助詞:「が」
助詞:「を」	助詞:「を」	0.9≤Posp≤1.0
0.9≤Sim≤ 1.0	0.9≤Posp≤ 1.0 ∧ 助詞:「が」	助詞:「は」
助詞:「を」∧ 助詞:「は」	0.9≤Posp≤ 1.0 ∧ 助詞:「に」	0.9≤Posp≤ 1.0 ∧ 助詞:「は」
0.9≤Posd≤ 1.0	0.0≤Posp< 0.1 ∧ 文末 (大):叙述	0.9≤Posp≤1.0 ∧ 文末 (大):叙述
0.9≤Posp≤ 1.0 ∧ 助詞:「を」	助詞:「に」	0.4≤Len< 0.5
0.2≤TI_{dep}< 0.3 ∧ 助詞:「を」	0.9≤Posp≤ 1.0 ∧ 助詞:「を」	0.2≤Len<0.3
助詞:「に」	0.6≤Len< 0.7	0.9≤Posp≤ 1.0 ∧ 助詞:「が」
助詞:「も」	文末 (大):意見	助詞:「を」∧ 文末 (大):叙述

「社会」の特徴は，固有表現（DATE，ORGANIZATION），助詞（「が」），タイトルとの類似，文末表現（「叙述」）の重みが大きいことである．これは，客観的な報道記事である場合が多いので，文末表現が「叙述」であり，DATE，ORGRANIZATION を含む文が文書の先頭付近に重要文として多く出現するからだと考える．また，主格の省略はしばしば行われるので，主格を含む文は重要と考えられる．さらに，タイトルは一種の要約なので，タイトルとの類似性の高いものには正，低いものには負の重みが与えられている．

「社説」の特徴は，助詞，タイトルとの類似の重みが大きいが，段落内の位置，文末表現（「意見」），係り受け関係を考慮した TF·IDF（TI_{wid}），文の長さの重みも大きいことである．文書の位置に関しては，50%の要約率では文書の末尾付近も高い重みを得ている．また，「社説」は論説文の構造を持った文書であるので，文書を構成するサブトピックの導入部に重みが与えられている．さらに，報道とは異なり，筆者の意見を表す文末表現にも重みが与えられている．

「解説」の特徴は，固有表現（ARTIFACT），助詞（「は」），文の長さ，バリエーションも含めた TF·IDF，キーワードの密度である．文書の位置では末尾付近の重要度が高い．これは，社説と同様に論説文的な文書構造を持っているので，文書の末尾に近い文は話題のまとめを表す文が多く出現するからであると考えられる．また，「解説」には，特定の技術や時事問題について解説する文書が多く，文書の主題（解説の対象）に関する固有表現（たとえば，「デビットカード」のような ARTIFACT）に高い重みが与えられたと考える．さらに，トピックマーカである助詞（「は」）の重みも高い．

また，本稿で新たに導入した固有表現と係り受け構造を考慮した TF·IDF は，絶対値の高い重みが与えられており，その有効性を確認することができた．

4.4 判別関数を用いたランキングの有効性

SVM の判別関数（$g(\mathbf{x})$）の値を用いて文のランキングを行うことの有効性について議論する．

表 12，**表 13** に 4.1 節での実験設定で，学習時とテス

表 11 負の重みを持つ素性
Table 11 Effective features and their pairs (negative).

Summarization rate 10%

社会	社説	解説
0.0≦Sim< 0.1	0.0≦Den< 0.1 ∧ 文末 (大):叙述	0.2≦TI_{wid}<0.3 ∧ 助詞:「を」
0.9≦Posd≦ 1.0 ∧ 0.0≦Sim<0.1	0.2≦TI< 0.3 ∧ 助詞:「は」	0.1≦TI_{wid}<0.2 ∧ NE:ORG
0.0≦TI_{wid}< 0.1 ∧ 格助詞:「を」	0.3≦TI≦ 0.4 ∧ NE:LOC	助詞:「は」∧ NE:LOC
0.6≦Posp< 0.7	0.3≦TI_{dep}<0, 4 ∧ NE:ORG	NE:DAT ∧ 文末 (小):過去
0.7≦Posd< 0.8 ∧ 0.9≦Posp≦ 1.0	0.0≦Posd<0.1 ∧ 助詞:「も」	0.9≦Posp≦1.0 ∧ 0.7≦Sim<0.8
0.9≦Posd≦ 1.0 ∧ 0.6≦Posp< 0.7	0.4≦TI< 0.5 ∧ 0.3≦TI_{wid}<0.4	0.5≦Posd<0.6 ∧ 助詞:「が」
0.5≦Posd< 0.6	0.1≦TI_{wid}< 0.2 ∧ 文末 (小):過去	0.3≦TI<0.4 ∧ 0.2≦TI_{dep}<0.3
0.9≦TI_{wid}≦ 1.0 ∧ 修辞:順接	0.4≦TI_{wid}< 0.5 ∧ 助詞:「を」	0.3≦TI<0.4 ∧ 0.9≦Sim≦1.0
0.9≦Posp≦ 1.0 ∧ 0.0≦Sim<0.1	助詞:「と」∧ NE:LOC	0.5≦TI≦0.6 ∧ 格助詞:が
0.9≦Posd≦ 1.0 ∧ 0.0≦Den< 0.1	助詞:「は」∧ 文末 (小):その他	0.9≦Posp≦1.0 ∧ 0.2≦TI<0.3

Summarization rate 30%

社会	社説	解説
0.0≦Sim<0.1	助詞:「を」∧ 助詞:「も」	0.1≦TI_{wid}<0.2 ∧ NE:ORG
0.9≦Posd≦1.0 ∧ 0.0≦Sim<0.1	0.0≦Den<0.1 ∧ 文末 (大):叙述	0.2≦Den<0.3 ∧ 助詞:「で」
0.8≦Posd<0.9 ∧ 0.0≦Den<0.1	0.1≦Len<0.2	0.5≦TI<0.6 ∧ 助詞:「は」
0.5≦Posd<0.6	0.0≦TI_{wid}<0.1 ∧ NE:DAT	0.3≦TI_{dep}<0.4 ∧ 助詞:「を」
0.3≦Posd<0.4	助詞:「が」∧ NE:DAT	0.5≦Sim<0.6 ∧ 修辞:順接
0.8≦Posd<0.9 ∧ 0.1≦TI<0.2	助詞:「が」∧ 助詞:「は」	0.9≦Posp≦1.0 ∧ 0.1≦TI_{wid}<0.2
0.0≦Sim<0.1 ∧ 文末 (大):叙述	0.9≦Posd≦1.0 ∧ 0.3≦TI_{dep}<0.4	助詞:「が」∧ 用言:29
0.5≦Posd<0.6 ∧ 助詞:「が」	0.1≦Den<0.2 ∧ NE:LOC	助詞:「で」∧ 助詞:「は」
0.6≦Sim<0.7 ∧ 助詞:「で」	0.4≦TI_{wid}<0.5 ∧ 0.7≦Sim<0.8	助詞:「に」∧ 用言:29
助詞:「で」∧ 助詞:「を」	助詞:「が」∧ 修辞:順接	0.1≦Posd<0.2 ∧ 修辞:順接

Summarization rate 50%

社会	社説	解説
0.0≦Len<0.1	0.6≦Posp< 0.7	0.1≦Len<0.2
0.0≦Posd<0.1	0.1≦Len< 0.2	文末 (小):その他
0.1≦Len<0.2	文末 (小):過去	文末 (小):過去
0.0≦TI<0.1	0.0≦Den< 0.1 ∧ 文末 (小):過去	文末 (小):その他
0.0≦Sim<0.1	0.0≦Den< 0.1 ∧ 文末 (大):叙述	NE:PERSON
0.2≦TI<0.3	助詞:「も」	0.0≦Den<0.1
文末 (小):その他	助詞:「も」∧ 文末 (大):叙述	0.6≦Posp< 0.7
文末 (大):その他	0.0≦Den< 0.1	0.0≦Len<0.1
0.3≦TI_{dep}<0.4	0.3≦Posd< 0.4	0.4≦Posp< 0.5
0.9≦Posd≦1.0 ∧ 0.0≦Sim<0.1	0.1≦TI<0.2	文末 (小):過去 ∧ 文末 (大):叙述

表 12 学習とテストの要約率を変化させた場合の F-measure
Table 12 F-measure of each combination.

学習 \ テスト	10%	30%	50%
10%	**0.462**	0.479	0.585
30%	0.448	**0.483**	0.619
50%	0.373	0.468	**0.628**

表 13 学習とテストの要約率を変化させた場合の pseudo-utility
Table 13 Pseudo-utility of each combination.

学習 \ テスト	10%	30%	50%
10%	**0.574**	0.631	0.667
30%	0.550	**0.639**	0.702
50%	0.491	0.615	**0.705**

ト時の要約率の組合せを変化させた場合の F-measure と pseudo-utility を示す．SVM には 2 次の Polynomial kernel を用いた．F-measure，pseudo-utility ともに学習時とテスト時の要約率が同じ場合に最も高い抽出精度が得られている．だが，10%要約率で学習した場合には，30%要約率でテストした場合でも好成績であり，30%要約率で学習した場合には，10%，50%要約率のテストでも好成績，50%要約率で学習した場合には，30%要約率のテストでも好成績である．しかし，学習時とテスト時の要約率が大きく異なる場合，10%要約率で学習し，50%要約率でテストする場合と 50%要約率で学習し，10%要約率でテストする場合では，成績は大きく下がる．これは，表 10，11 より，10%要約率では素性の組合せが重要であり，50%の要約率では単独の素性が重要であるという違いが影響していると考えられる．学習時のテスト時の要約率の差が小さい場合には，学習とテストの要約率が一致しなくても，良い成績を残す傾向があることが分かる．つまり，重要度が上位 n%に入る文を正例，それ以外を負例として学習した場合，$g(\mathbf{x})$ による上位 n%付近までのランキング性能は良いと考える．

また，ランキングを行わずに二値のラベル付けのみの性能を評価すると，10%要約率では，Precision=0.039，Recall=0.031，30%要約率では，Precision=0.453，Recall=0.097 という非常に低い値となった．SVM は正例と負例をあわせた全体での正解率を

表 14 TSC 上位システムとの性能比較
Table 14 Performance comparison with TSC top two systems.

	SYSTEM1		SYSTEM2		提案手法	
要約率	F-measure	Pseudo-utility	F-measure	Pseudo-utility	F-measure	Pseudo-utility
10%	**0.363**	**0.518**	0.337	0.450	0.356	0.488
30%	0.435	0.559	0.452	0.603	**0.493**	**0.613**
50%	0.589	0.664	**0.612**	0.673	0.590	**0.681**

重視する傾向があるため，負例の数が正例の数よりも多い場合には，負例側に偏った分類結果となりやすいことが原因であると考える．ここで，ランキングにより抽出する文の数を調整することで成績が大きく向上していることを考えると，ランキングが有効に働いていることが分かる．

さらに，文献 22) では，ある順位より上位に事例が出現する確率が順位づけ関数となることを証明したうえで，$g(\mathbf{x})$ がその確率の近似と見なせることを示している．

以上より，本研究での $g(\mathbf{x})$ によるランキング手法は妥当であったと考える．

4.5 TSC 上位のシステムとの比較

TSC の重要文抽出タスクにおいて好成績を残したシステムと本稿で提案した手法との性能の比較を試みた．文献 21) より上位 2 システムの評価結果を抜粋し，その F-measure，pseudo-utility を算出した．また，SVM(Poly) を用いて TSC の本試験と同じ設定（予備試験の 30 文書で学習を行い，本試験の 30 文書でテストを行う）で提案手法の性能を算出した．両者を比較した結果を**表 14** に示す．

提案手法は TSC 上位 2 システムにほぼ匹敵する抽出精度を達成していることが分かる．30 文書という少ない学習データしか用いていないが，高い抽出精度である．SYSTEM1，SYSTEM2 ともに複数の手がかりを用いた重要文抽出手法であるが，人手により各手がかりの重みを決定する手法をとっており，扱っている素性数は提案手法と比較して 1/10 以下である．このように人手によるチューニングでは扱える素性の数は少ない．重要文抽出のための素性は SYSTEM1，SYSTEM2 で用いられたもので十分とはいえず，より多くの素性を扱えることが望ましい．よって，大量の素性を扱うことができ，少ない学習データでも抽出精度の高い提案手法は重要文抽出に有効であるといえる．

5. まとめと今後の課題

本稿では，Support Vector Machine（SVM）を用いた重要文抽出手法を提案し，重要文抽出研究のために公開されている TSC のデータと野本らのデータを用いて従来手法との比較評価を行い，提案手法の有効性を示した．また，重要文抽出のための素性として，固有表現，係り受け構造を考慮した TF·IDF を提案し，これらの素性の有効性を示した．さらに，文書のジャンルごとにわけて学習することによって重要文の抽出の精度が向上することを実証し，各ジャンルにおける重要文抽出に有効な素性の相違を明らかにした．

本稿で用いたデータセットでは，各文書ごとにあらかじめ文書の属するジャンルが定義されており，その情報を用いてジャンルごとの評価実験を行った．しかし，一般に重要文抽出の対象となる文書に対し属するジャンルがあらかじめ定義されているとは限らないことを考えると，自動的に文書のジャンルを判定することが必要となる．これは今後の課題である．さらに，「社会面」に書かれた記事であっても文書の表現スタイルとしては「社説」に近い場合もある．このような場合に対処するための重要文抽出を前提とした文書のジャンル分類も今後の課題である．

謝辞 研究を進めるにあたって，TinySVM，Cabocha を提供していただくとともに有益なコメントをいただいた奈良先端科学技術大学院大学の工藤拓氏に感謝いたします．また，NTCIR，TSC の運営に関わられたすべての皆様に感謝いたします．さらに，貴重な重要文データをご提供くださった国文学研究資料館の野本忠司氏に感謝いたします．

参 考 文 献

1) Aone, C., Okurowski, M. and Gorlinsky, J.: Trainable Scalable Summarization Using Robust NLP and Machine Learning, *Proc. 17th International Conference on Computational Linguistics and 36th Annual Meeting of the Association for Computational Linguistics*, pp.62–66 (1998).
2) Brandow, R., Mitze, K. and Rau, L.F.: Automatic Condensation of Electronic Publications by Sentence Selection, *Information Processing & Management*, Vol.31, No.5, pp.675–685 (1995).
3) Edmundson, H.: New Methods in Automatic Abstracting, *J. ACM*, Vol.16, No.2, pp.246–285

(1969).
4) Fukushima, T. and Okumura, M.: Text Summarization Challenge: Text Summarization Evaluation in Japan, *Proc. NAACL2001 Workshop on Automatic Summarization*, pp.51–59 (2001).
5) Hirao, T., Hatayama, M., Yamada, S. and Takeuchi, K.: Text Summarization based on Hanning Window and Dependency Structure Analysis, *Proc. 2nd NTCIR Workshop on Research in Chinese & Japanese Text Retrieval and Text Summarization*, pp.349–354 (2001).
6) Isozaki, H.: Japanese Named Entity Recognition based on Simple Rule Generator and Decision Tree Learning, *Proc. 39th Annual Meeting of the Association for Computational Linguistics*, pp.306–313 (2001).
7) Joachims, T.: Text Categorization with Support Vector Machines: Learning with Many Relevant Features, *Proc. European Conference on Machine Learning*, pp.137–142 (1998).
8) Kudo, T. and Matsumoto, Y.: Japanese Dependency Structure Analysis Based on Support Vector Machines, *Proc. Empirical Methods in Natural Language Processing and Very Large Corpora*, pp.18–25 (2000).
9) Kudo, T. and Matsumoto, Y.: Chunking with Support Vector Machine, *Proc. 2nd Meeting of the North American Chapter of the Association for Computational Linguistics*, pp.192–199 (2001).
10) Kupiec, J., Pedersen, J. and Chen, F.: A Trainable Document Summarizer, *Proc. 18th ACM-SIGIR*, pp.68–73 (1995).
11) Kwok, C., Etzioni, O. and Weld, D.: Scaling Question Answering to the Web, *Proc. 10th ACM-WWW*, pp.150–161 (2001).
12) Lin, C.: Training a Selection Function for Extraction, *Proc. 18th ACM-CIKM*, pp. 55–62 (1999).
13) Luhn, H.: The Automatic Creation of Literature Abstracts., *IBM Journal of Research and Development*, Vol.2, No.2, pp.159–165 (1958).
14) Mani, I. and Bloedorn, E.: Machine Learning of General and User-Focused Summarization, *Proc.15th National Conference on Artificial Intelligence*, pp.821–826 (1998).
15) Nobata, C., Sekine, S., Murata, M., Uchimoto, K., Utiyama, M. and Isahara, H.: Sentence Extraction System Assembling Multiple Evidence, *Proc. 2nd NTCIR Workshop on Research in Chinese & Japanese Text Retrieval and Text Summarization*, pp.319–324 (2001).
16) Quinlan, J.: *C4.5: Programs for Machine Learning*, Morgan Kaufmann (1993).
17) Sekine, S. and Eriguchi, Y.: Japanese Named Entity Extraction — Analysis of Results, *Proc. 18th International Conference on Computational Linguistics*, pp.1106–1110 (2000).
18) Vapnik, V.: *The Nature of Statistical Learning Theory*, Springer, New York (1995).
19) Watanabe, H.: A Method for Abstracting Newspaper Articles by Using Surface Clues, *Proc. 16th International Conference on Computational Linguistics*, pp.974–979 (1996).
20) Zechner, K.: Fast Generation of Abstracts from General Domain Text Corpora by Extracting Relevant Sentences, *Proc. 16th International Conference on Computational Linguistics*, pp.986–989 (1996).
21) 難波英嗣，奥村　学：複数の要約率の重要文データを用いた要約評価方法，言語処理学会第 8 回全国大会講演論文集，pp.559–562 (2002).
22) 賀沢秀人，平尾　努，前田英作：Ranking SVM による順位付け学習と重要文抽出への応用，第 5 回情報論的学習理論ワークショップ，pp.37–42 (2002).
23) 奥村　学，原口良胤，望月　源：決定木学習を用いたテキスト自動要約に関するいくつかの考察，情報処理学会第 59 回全国大会講演論文集（分冊 5），pp.393–394. 5N–2 (1999).
24) 池原　悟，宮崎正弘，横尾昭男，中岩浩己，小倉健太郎，大山芳史，林　良彦：日本語語彙大系，岩波書店 (1999).
25) 野本忠司，松本裕治：人間の重要度判定に基づいた自動要約の試み，情報処理学会研究報告 NL-120-11，pp.71–76 (1997).
26) 田村直良，和田啓二：セグメントの分割と統合による文章の構造解析，自然言語処理，Vol.5, No.1, pp.59–78 (1998).
27) 平　博順，春野雅彦：Support Vector Machine によるテキスト分類における属性選択，情報処理学会論文誌，Vol.41, No.4, pp.1113–1123 (2000).
28) 福本淳一，安原　宏：日本語文章の構造解析，情報処理学会研究報告 NL-85-11，pp.81–88 (1991).
29) 松本裕治，北内　啓，山下達雄，平野善隆，浅原正幸，松田　寛：日本語形態素解析システム『茶筌』version 2.0 使用説明書第二版，Information Science Technical Report NAIST-IS-TR99012, Nara Institute of Science and Technology (1999).

(平成 14 年 5 月 20 日受付)
(平成 15 年 6 月 3 日採録)

平尾　努（正会員）

1995 年関西大学工学部電気工学科卒業．1997 年奈良先端科学技術大学院大学情報科学研究科博士前期課程修了．同年，NTT データ通信株式会社（現，株式会社 NTT データ）入社．2000 年より日本電信電話株式会社 NTT コミュニケーション科学基礎研究所に所属．博士（工学）．自然言語処理の研究に従事．言語処理学会，ACL 各会員．

磯崎　秀樹（正会員）

1983 年東京大学工学部計数工学科卒業．1986 年同大学工学系大学院修士課程修了．同年，日本電信電話株式会社入社．1990〜91 年スタンフォード大学ロボティクス研究所客員研究員．現在，NTT コミュニケーション科学基礎研究所特別研究員．博士（工学）．人工知能・自然言語処理の研究に従事．電子情報通信学会，人工知能学会，言語処理学会，AAAI，ACL 各会員．

前田　英作（正会員）

1984 年東京大学理学部動物学科卒業．1986 年同大学院理学系研究科修士課程修了．同年，日本電信電話（株）入社．現在，NTT コミュニケーション科学基礎研究所知能情報研究部知識処理研究グループリーダ．工学博士．1995 年〜1996 年ケンブリッジ大学（英国）客員研究員．主としてパターン認識，統計的機械学習，生物情報処理の研究に従事．IEEE，日本バイオインフォマティックス学会各会員．

松本　裕治（正会員）

1977 年京都大学工学部情報工学科卒業．1979 年同大学大学院工学研究科修士課程情報工学専攻修了．同年電子技術総合研究所入所．1984〜85 年英国インペリアルカレッジ客員研究員．1985〜87 年（財）新世代コンピュータ技術開発機構に出向．京都大学助教授を経て，1993 年より奈良先端科学技術大学院大学教授，現在に至る．工学博士．言語処理学会，人工知能学会，日本ソフトウェア科学会，認知科学会，AAAI，ACL，ACM 各会員．