不完全なニュース集合からのタイムスタンプ推定

上嶋　宏† 三浦　孝夫† 塩谷　勇††

† 法政大学 工学研究科 電気工学専攻　〒 184–8584 東京都小金井市梶野町 3-7-2
†† 産能大学 経営情報学部　〒 259–1197 神奈川県伊勢原市上粕屋 1573
E-mail: †{i03r3207,miurat}@k.hosei.ac.jp, ††shioya@mi.sanno.ac.jp

あらまし　近年，オンラインニュース等の時系列文書の要約や，新しい話題の検出や追跡を行う研究が盛んに行われている．しかし，それらのほとんどでは，各記事のタイムスタンプ（発行時間）が取得可能であることを前提としており，タイムスタンプが不明な記事はこれらのタスクには貢献できない．また複数のソースを扱う場合，ソース間に速報性の差による割り当ての矛盾が生じる．本稿では，ニュース記事のタイムスタンプを少数の不完全なデータから，効果的に推定する手法を提案する．EM アルゴリズムや逐次的なクラスタリング手法を用いることにより，記事が述べている事象に基づきタイムスタンプを推定する．TDT2 コーパスを用いた実験により本手法の有効性と考察を示す．
キーワード　タイムスタンプ推定,TDT, 時系列データ, EM アルゴリズム, 文書クラスタリング

Estimating Timestamp From Incomplete News Corpus

Hiroshi UEJIMA†, Takao MIURA†, and Isamu SHIOYA††

† Dept.of Elect.& Elect. Engr., HOSEI University　3-7-2, KajinoCho, Koganei, Tokyo, 184–8584 Japan
†† Department of Management and Information Science, SANNO University　1573, Kamikasuya, Isehara city, Kanagawa 259–1197 Japan
E-mail: †{i03r3207,miurat}@k.hosei.ac.jp, ††shioya@mi.sanno.ac.jp

Abstract　Recently there have been a lot of researches to summarize news stream and to extract *edge* (beginning) of new events in the news stream. But, in these tasks, most of data are assumed to carry *timestamp* (temporal information). In the other words, news article that doesn't have timestamp can't make contribution to these tasks. In this investigation, we propose a novel technique to give timestamp to a collection of news corpus with no explicit timestamp by the small number of incomplete data. Here we learn temporal information and topic information by means of EM algorithm and incremental clustering, then we estimate timestamp of the news text based on events discussed in the news text. In this work, using TDT2 corpus, we show how well our approach works by some experiments.
Key words　Timestamp estimation, TDT, Stream data, EM algorithm, Document Clustering

1. 前　書　き

近年，インターネット等を通じて利用可能な電子文書やメールマガジン，オンラインニュースが増加している．これまで，次々と配信される大量の文書データから，有用な情報を抽出する為のクラスタリングや要約手法が提案されている．その中でも，ニュースストリームから話題構造を自動的に獲得し，動向を容易に把握することを目的とした話題抽出と追跡（Topic Detection and Tracking, TDT）プロジェクトがある [7]．TDT2004 では，(1) 事象発見 (New event detection)，(2) リンク発見 (Story link detection), (3) 話題発見 (Topic detection)，(4) 話題追跡 (Topic Tracking) の 4 つのタスクが設定されている．

一般に，ニュースストリームを含め，オンラインで配信される文書データは，配信される話題が時刻に対応するものが多い．そのため，これらは時系列文書と呼ぶことができる．配信される話題が時刻に対応しているため，時系列文書のクラスタリングや，TDT の各タスクにおいて，各文書はタイムスタンプ（発行時間）あるいは発行順序にしたがって処理される．また，タイムスタンプを考慮することにより，精度の高いクラスタリングが可能になることが知られている [1]．

しかし，これらのタスクでは，データのタイムスタンプあるいは発行順序が取得可能である状態を想定しているため，タイムスタンプを持たない場合，そのデータはこれらのタスクに貢献することができない．また，複数のソースから時系列文書が

配信されている場合, 各ソースにより速報性に差がある場合があり, 必ずしも同じタイムスタンプを持つとは限らない.

本稿では，ストリーム以外から配信されている文書や，内容時間と発行時間が大きく異なる，過去の事象について述べた記事など，タイムスタンプを持たない文書のタイムスタンプを推定する手法を提案する．これにより，今までタスクに貢献できなかったデータが使用可能になることや，複数のソースを一つの時系列に統一することが可能となり，よりスムーズに話題の発見や追跡が可能になる.

筆者らは，事象に基づいた教師学習データからのタイムスタンプ推定が有効に働くことを示した [9]．しかし，ここでは記事に含まれている話題情報がすべて学習可能で，かつ限定されていることを想定している．また，タイムスタンプの推定に，ある一定量の，話題とタイムスタンプの両方が取得可能である完全なデータを必要とした.

しかし，実際のニュースソースの状況においては，限られた話題や事象のみが配信される状態はまれであり，話題としての集合をなさない単発的な記事も非常に多数あることが考えられる.

そのため，学習できる訓練データが十分でないことがある．このような状態は，逐次的な処理においてた完全に対応することは難しい．TDT タスク等において，逐次的な処理が要求されないタスクに対しては，一括処理 (batch) により，より正確なタイムスタンプの推定手法が適用可能と考えられる．そこで本稿では，訓練データが非常に少数である場合や，訓練データ（話題が既知のデータ）内に存在しない話題が出現する場合に対応したタイムスタンプ推定手法を提案する．ここでは，EM アルゴリズムを用いることにより，訓練データが不完全な状況に対応する.

本稿では，話題とタイムスタンプの両方が取得可能であるデータ集合 ($D_{topic,time}$) に加え，話題のみが既知のデータ集合 (D_{topic}) や，発行時間のみが取得可能なデータ集合 (D_{time})，またそれらの両方が不明であるデータ集合 (D_u) などを総合的に利用して，タイムスタンプが不明であるデータ集合 (D_{topic}, D_u) のタイムスタンプを推定する.

本稿で提案するタイムスタンプ推定のアルゴリズムの概略は以下のとおりである.

(1) 話題が既知のデータ集合 ($D_{topic,time}$，D_{topic}) から，EM アルゴリズムを用いたベイズ規則により，各データ集合 (D_{time}，D_u) を話題へと分類する.
(2) 1) の結果に基づき，発行時間が取得可能なデータ集合 ($D_{topic,time}$，D_{time}) を用いて，単一パス法により，各話題を時間に依存した事象へとクラスタリングする.
(3) 1),2) の結果に基づき，k-NN によりタイムスタンプが未知のデータ集合 (D_{topic}，D_u) を 1 の話題情報を用いて 2 で生成された各事象へと割り当てる.
(4) 3) で割り当てられた事象からデータ D_{topic}，D_u のタイムスタンプを推定する.

EM アルゴリズムとは Expectation(期待値)，Maximization (最大化) を意味し，観測したデータからは直接観測できない確率変数があるような，直接，最尤推定法を適用できない場合に有効な推定方法である [15]．文書分類において，EM アルゴリズムは，訓練データが少数であるときに効果的に働くことが知られている.

本稿で用いる手法は，EM アルゴリズムを用いたベイズ規則による分類に基づく．これにより，訓練データが非常に少数である場合や，訓練データ（話題が既知のデータ）内に存在しない話題が出現する場合においても，対応することができ，その分類結果に基づいた，時間が既知のデータを用いたクラスタリングにより，文書集合を時間依存の事象に分割することができる．さらに，この結果に基づきタイムスタンプを推定することで，不完全なニュース集合からでも，効果的にタイムスタンプの推定を行うことができる.

次章では関連研究について述べ，3 章では EM アルゴリズムを用いたベイズ分類について，4 章で単一パス法による逐次的なクラスタリングについて述べる．5 章でタイムスタンプの推定手法を提案し，6 章において TDT2 コーパスを用いた提案手法の実験と結果，考察について示す，最後に 7 章で結びとする.

2. 関 連 研 究

文書に時間を割り当てる方法としては，Mani らの研究がある [5]. Mani らは，文書内の時制表現を抽出する手法を提案しており，例えば，文章内の “yesterday”などの索引的な表現からその文書の発行時間が 4 月 20 日であった場合，その時制表現は 4 月 19 日について述べていると推定する．このように，文書から文章内の時制表現を抽出する．しかし，これらの研究は文書のタイムスタンプや，他の文章や文書との関係から時制表現が示す時間を推定するもので，根本的な文書のタイムスタンプを推定する本手法とは異なる.

Papka らは Inquery による単一パス法に基づいた話題検出法を提案している [8]．また，Yang らも単一パス法を用い，逐次，バッチ双方における話題検出を行っている [11]．逐次的な処理において，Papka らは，文書ベクトル表現に，補助コーパスから求めた IDF 値を使用しており，Yang らは補助コーパスによる IDF 値に加え，逐次 IDF 値を更新する手法を提案している．しかし，本稿では，すでにタイムスタンプを推定するデータはすべて揃っていることを想定しているため，従来の IDF 値を用いる.

石川らは，忘却の概念を利用した文書クラスタリング手法を利用している [4]．ここでは，C^2ICM を用いることにより文書データを逐次クラスタリングしており，少ない計算量での差分によるクラスタ更新を行っている．ここでは，統計的な確率に基づき文書間の類似度を求めている．また，Papka らや，Yang らも類似の忘却の概念を利用している.

これらのように，時間を考慮した時系列文書のクラスタリング手法は多く提案されている，しかし，これらの手法のほとんどは，文書のタイムスタンプが利用可能であること，あるいはリアルタイムで処理することを想定している．そのため，タイ

ムスタンプ自体を予測する手法は，筆者らの知る限り提案されていない．

3. EM アルゴリズムを用いた分類

本稿では，話題が取得可能であるデータ ($D_{topic,time}$,D_{topic}) を教師データとして，話題が未知であるデータ (D_{time},D_u) を各話題へと分類する．ここで，EM アルゴリズムを用いたベイズ規則に基づき分類を行う．

EM アルゴリズムでは，混合正規分布を仮定し，不完全な観測データ $x_1, ..., x_N$ が，モデル $P_\theta(x)$ から得られたとき，この不完全な観測データ $x_1, ..., x_N$ から，未知のパラメータ θ の値を推定する．現時点での θ を使用して，条件付確率モデルから，完全データにおいてのサンプル数，期待値を求める（E ステップ)，続いて，ここで求めたサンプルから期待値を最大化するパラメタ $\hat{\theta}$ を求める（M ステップ)．この E ステップと M ステップを繰り返すことにより，モデルの対数尤度を最大化するパラメータを求める．

一般的に，文書分類においては，分類規則の訓練に，少数のラベル付の訓練データだけでなく，話題情報を持たない，大量の不完全なデータを使用し，分類規則の精度を高める手法として使用される [6]．そのため，一般に EM アルゴリズムにより精度を高める場合は，大量のラベルなしデータが述べている話題のすべてが，ラベルを持つデータに含まれていることを想定している．しかし，ラベルを持たないデータは，ラベルを持つ訓練データからは学習できない話題について述べている可能性が高い．本稿では通常の精度を高める目的ではなく，EM アルゴリズムにより，訓練データには含まれない話題について述べているデータに対しても，既存の話題のうち近いものに割り当て，EM アルゴリズムを適用することにより，既存の話題を汎化させ，より意味的に大きな話題を生成することを目的としている．

3.1 単純ベイズ法にる文書分類

今，文書 d をカテゴリ集合 $C = \{c_1, ..., c_n\}$ に分類することを考える [12]．ベイズ規則による分類は，文書 d がカテゴリ c に属する確率 $P(c|d)$ の確率分布を求めることである．また，排他的な分類の場合，最大事後確率をとるカテゴリ $c_k(c_k = argmax_{c \in C} P(c|d))$ へ文書 d 分類することで誤分類を抑えると考える．

ここでベイズ規則は以下のように与えられる．

$$P(c_k|d) = P(c_k) \times \frac{P(d|c_k)}{P(d)}$$

すなわち，ベイズ規則での分類規則生成（訓練）は訓練データ集合から，確率分布 $P(c_k)$，$P(d)$，$P(d|c_k)$ を推定することである．

しかし，文書ベクトル $d = (w_1, ..., w_m)$ はほぼすべての文書で異なり，$P(d|c_k)$ や $P(d)$ の推定が問題であるため，一般に，文書内での単語 w_j の出現は，統計的に他の単語の出現とは独立であるという仮定をおき，各文書を単語の集合と考る，単純ベイズが使われる．単純ベイズでは，$P(d|c_k)$ を以下の形式に分解して考える [2]．

$$P(d|c_k) = \prod_{i=1}^{|d|} P(w_j|c_k)$$

これにより，文書主導の排他的分類の場合，ベイズ規則は以下のように書くことができる．

$$P(c_k|d) = P(c_k) \times \prod_{i=1}^{|d|} P(w_j|c_k) \quad (1)$$

また，ここでは文書内での単語の出現回数は考慮せず，単語が出現したか否かのみを考えるバイナリ独立モデルを用いる．

3.2 EM アルゴリズム

EM アルゴリズムを文書分類適用する場合，アルゴリズムは以下のように定義される．

(1) 入力：ラベル付文書，ラベルなし文書
(2) ラベル付文書のみから単純ベイズ分類規則 $\hat{\theta}$ を生成
(3) 以下のステップを分類規則のパラメタが収束するまで繰り返す
  ・(E-step) 現在の分類規則 $\hat{\theta}$ を使用してラベルなし文書を各カテゴリへと分類する $(P(c_j|d_i;\hat{\theta}))$
  ・(M-step) 推定された事後確率（分類結果）を利用して，分類規則 $\hat{\theta} = P(D|\theta)P(\theta)$ を再度生成する．
(4) 出力：分類規則 $\hat{\theta}$

具体的に，本稿では $P(w_i|c_k)$(分類規則) を以下の式で求める．

$$P(w_i|c_k) = \frac{1 + \sum_{j=1}^{|D|} N(w_i, d_j) P(c_k|d_j)}{|V| + \sum_{i=1}^{|V|} \sum_{j=1}^{|D|} N(w_i, d_j) P(c_k|d_j)} \quad (2)$$

ここで D は文書データ全体を表し，w_i はデータ内の各単語を表す．また $N(w_i, d_j)$ は文章 d_j における単語 w_i の発生回数であるが，前述のとおり，本稿では出現の有無により 0 か 1 の値をとる．さらに，$P(c_k|d_j)$ は前述の文書 d_j がカテゴリ c_k に属する確率であり，ラベル付けされたデータに関しては，そのラベル付けられたカテゴリ c_m においては，$P(c_m|d_j) = 1$ であり，それ以外のカテゴリに対しては 0 をとる．対して，ラベルなしデータに関しては，最初は全カテゴリに対して 0 であるが，最初は通常のベイズ分類により，その後は EM アルゴリズムの E-step により，徐々に適切な値へと更新される．式 1 と式 2 により EM アルゴリズム内で分類規則を生成する．同様に $P(c_j)$ は以下のように与えられる．

$$P(c_j) = \frac{1 + \sum_{j=1}^{|D|} P(c_k|d_j)}{|C| + |D|} \quad (3)$$

式 (2),(3) は，それぞれ $P(w_i|c_k), P(c_j)$ のスムージングを行っている．

4. 時間距離を考慮したクラスタリング

本章では，EM アルゴリズムによる話題への分類結果に基づき，各話題を事象へと分割するステップを論じる．

TDT や時系列文書クラスタリングにおいて，時間を考慮したクラスタリングは良い性能を示すことが知られている [1], [11]. これはニュースの話題や事象の出現は時間に大きく依存することを意味する．そのため，文書がどのような事象について述べているかを求めることで，より正確なタイムスタンプの推定が行えると考えられる．そこで本稿では，文書とクラスタ間の時間距離を考慮してタイムスタンプを持つ文書 ($D_{topic,time}$,D_{time}) のクラスタリングを行う．これは各クラスタが事象を示し，ニュース的意味を持つことを示す．事象は，各話題に基づいたクラスタリング結果から生成されるので，時間を考慮した話題の部分集合である．

4.1 文 書 表 現

文書とクラスタの表現は，“Bag of words”と呼ばれる，文書を単語の集合考える従来のベクトルスペースモデルを利用する．また，ベクトルを構成する単語として本稿では不要語 (stop-word) を削除した後，BrillTagger [3] により, 名詞と固有名詞のみを抽出して使用する．さらにステミングを行うことにより語幹を取り除く．ここで文書 d_i における単語 t_j の重み $w(t_j, d_i)$ は，従来の tf*idf 法を拡張した ltc で表す．これは SMART システムにより提供される重みで，TDT タスクにおいてよい性能を示すことが知られている [11].

$$w(t_j, d_i) = (1 + log_2 TF_{t_j,d_i}) \times IDF_{(t_j)} / \|\vec{d_i}\|$$

ここで TF_{ij} は文書 d_i における単語 t_j の出現頻度 (Term Frequency) を表し，$IDF_{(t_j)}$ は文書集合での単語 t_j を含む文書の割合の逆数である．これにより，文書ベクトル $\vec{d_i}$ は以下のように表すことができる．

$$\vec{d_i} = (w(t_1, d_i), ..., w(t_n, d_i))$$

クラスタはクラスタに含まれるベクトルの重心で表現する．

4.2 単一パスクラスタリング

本稿では, タイムスタンプを持つデータを単一パス法によりクラスタリングする [11]．単一パス法は非常に単純な方法で、文書とクラスタの類似度がしきい値以上ならばクラスタに追加し, 超えない場合は文書を新しいクラスタとする方法である．通常は，リアルタイムの学習に使われる手法であるが，本稿ではタイムスタンプを持つデータの発行順序を考慮したクラスタリングを行うために，この手法を用いる．

この単一パスクラスタリングは, 以下の手順で実行される．

(1) しきい値 th を設定
(2) 最初は空の集合 C から始め，1 つ目の文書 d_1 自身をクラスタの重心とする
(3) 次の文書 d_i を読み込み，既存の全クラスタ C のそれぞれとの類似度 $sim(\vec{d_i}, \vec{C})$ を計算する
(4) 最も類似したクラスタ c_{d_i} との類似度 sim_{max} をが，しきい値より大きい ($sim_{max} > th$) なら，d_i をクラスタ C_{d_i} に追加し，クラスタ C_{d_i} の重心を再計算する．もし $sim_{max} < th$ なら，d_i を新しいクラスタの重心とする
(5) 3-4 をデータがなくなるまで繰り返す

ここで，sim_{max} は以下のように定義される．

$$sim_{max} = MAX(sim(\vec{d_i}, \vec{C}))$$

文書とクラスタ間の類似度はコサイン尺度と呼ばれる方法を用い，以下の式で与えられる．ここで V_C はクラスタ C の重心を表す．

$$sim(\vec{d}, \vec{C}) = \frac{\vec{d} \cdot \vec{V_C}}{|\vec{d}||\vec{V_C}|}$$

4.3 忘却関数とタイムウィンドウ

本稿では，文書とクラスタ間の時間距離を考慮した類似度計算を行うために，忘却関数を適用する [4], [11].

ニュースの話題は，ある一定の期間に集中して発生することが多く，話題の出現が記事のタイムスタンプに大きく依存する. [11]．そのため，例えばある 2 つの文書の文書ベクトル間の類似度が高くても，発行時間に大きな差がある場合，その 2 つの文書は同じ話題について述べている可能性は低くなる．逆に，発行時間が非常に近い場合，この 2 つの文書が同じ話題について述べている可能性は非常に高くなる

本稿では，忘却関数により文書が古くなればなるほど分類への重要度を減少 (忘却) させる．すなわち，時間的に近いものほど重要と考えクラスタリングを行う．クラスタと文書間の類似度に忘却関数を適用するということは，クラスタが時間により小さくなることに対応する [図 1]．また，タイムウィンドウ [11] を導入し，ウィンドウ内にあるクラスタのみと比較する．類似度比較を行う期間を限定することで，完全に忘却する期間を決定する．これによりクラスタリングや話題発見を行う期間の長期化に対応でき，ノイズや計算量の減少が考えられる．また，本稿においてはタイムウィンドウのサイズを 90 日とした．

本稿では以下のように忘却関数を定義する．

$$\omega_\lambda(t) = \lambda^t (0 \leqq \lambda \leqq 1.0)$$

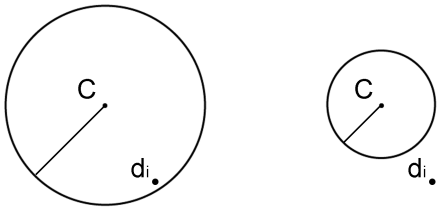
ここで t は時間距離を示し，その単位は日数である．本稿では，この忘却関数と，コサイン尺度から以下のように時間距離を考慮した新しい距離基準 sim' を定義する．

$$sim'(\vec{d_i}, \vec{C}) = \omega_\lambda(|time_{d_i} - timeC) \times sim'(\vec{d_i}, \vec{C})$$

ここで $time_{d_i}$ と $time_C$ はそれぞれ文書 d_i のタイムスタンプ，クラスタのタイムスタンプを示す．クラスタのタイムスタンプは，クラスタに含まれる文書集合中最新の文書のタイムスタンプとする．

図 1 は，$\lambda = 0.97$ の時の，時間距離が 0 日 ($\omega_\lambda(0)$) の場合と，30 日 ($\omega_\lambda(30)$) の場合のクラスタの状態を示している．文書ベクトル d_i とクラスタ C 間のベクトルの類似度が同じでも，時間距離が離れている場合，クラスタには追加されない．

この類似度を用いて, 上記の単一パスクラスタリングを行う．



(a)$\omega_\lambda(0) = 0$　　(b)$\omega_\lambda(30) = 0.5454$

図 1　忘却関数によるクラスタと文書関係の変化

5. タイムスタンプ推定

本章では，タイムスタンプをもたない D_{topic} や D_u 内の文書 n のタイムスタンプを推定するために，n の事象を決定するステップについて論じる．タイムスタンプを持つ文書のクラスタリング結果と文書 n を比較することで，n が述べている事象を推定し，この事象に基づいて文書 n のタイムスタンプ推定を行う．

5.1　k 近傍法によるクラスタの決定

n のクラスタ (事象) の決定は，k 近傍法による投票で決定する．ここで，全文書集合 D は，前述のベイズ規則に基づき，いずれかの話題に割り当てられている．各事象は，時間を考慮したクラスタリングによる，話題の部分集合なので，各事象は話題に属すると考えることができる．そのため，文書 d が述べる話題の事象だけと比較，投票することにより決定する．

例えば図 2 において，k=10 の場合，文書 n がベイズ規則により話題 $topic_1$ に割り当てられた時，タイムスタンプを持つ文書中，話題 $topic_1$ について述べてている文書集合 D_{topic_1} だけから投票を行い，n に近い 10 個の文章が属するクラスタのうち，もっとも多いものに属するとする．この場合，n が属するクラスタは C_1 となる．

また，最近傍文書のみを使う場合 ($k = 1$) の場合は，事象へのクラスタリング結果は反映されず，ベイズ規則による話題の分類結果だけが反映される．

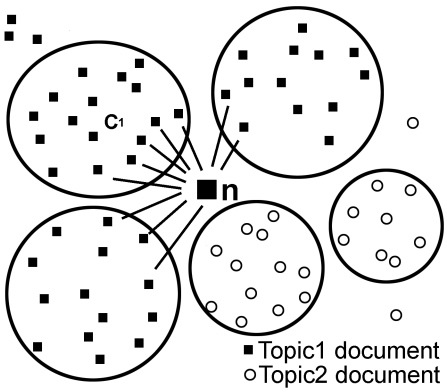


図 2　文書ベクトル $\vec{n}$ の所属クラスタ決定

5.2　タイムスタンプ推定

クラスタ (事象) 割り当て結果に基づき，文書 n の示す事象を考慮したタイムスタンプ推定を行う．文書 n のタイムスタンプの割り当てには, もっとも近い k 個のデータのうち, 文書 n が属すると判断されたクラスタ (事象) に属する文書のみを使用する．図 2 の場合，n に近い上位 10 個のうち，n が属するクラスタ C_1 に属する 5 つの文書を使用して n のタイムスタンプを推定する．

5.2.1　タイムスタンプ予想曲線

ここで，本稿では，タイムスタンプ予想曲線を用いることで文書 n のタイムスタンプを推定する．図 2 における 5 つの文書のうちの 1 つの文書 d_{C_11} から，文書 n のタイムスタンプ予想曲線を以下の式により与える．

$$TS_{n,d_{C_11},\lambda}(day) =$$
$$sim(d_{C_11}, n) \times dist\ r_{C1}(time_{d_{C_11}}) \times \omega_\lambda(|time_{d_{C_11}} - day|)$$

$TS_{n,d_{C_11},\lambda}(day)$ は，文書 n のタイムスタンプが時間 day である推定の度合いであり，文書 d_{C_11} と n，それらが属するクラスタ C_1 から与えられる．

クラスタリング同様，忘却関数を使用し，教師文書 d_{C_11} の発行時間から離れるほど，文書 d_{C_11} の影響力は小さくなる．

ここで，$distr_{C1}(time_{d_{C_11}})$ は，クラスタ C_1 に属する文書集合における，時間 $time_{d_{C_11}}$ においてのタイムスタンプの分布（例：図 4) である．ニュース記事において，大抵の話題や事象は，ある期間に集中して起こる特性があるため，事象のタイムスタンプ分布を考慮することは重要であり，クラスタ内のタイムスタンプ分布は，そのクラスタが示す事象について述べている文書のタイムスタンプの発生確率と考えることができる．

これにより，文書 d_{C_11} が文書 n に与えるタイムスタンプ予想曲線は，文書 d_{C_11} と文書 n の類似度と，クラスタ C_1 の時間 $time_{d_{C_11}}$ におけるタイムスタンプ分布の積を頂点とし，時間距離により減衰する図 3 のような曲線をあたえる．

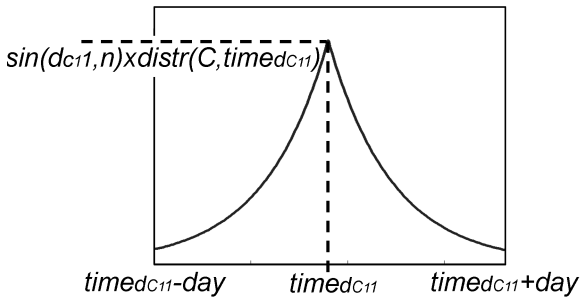


図 3　タイムスタンプ予想曲線の例 ($\lambda = 0.97$)

このタイムスタンプ予想曲線を，文書 n のタイムスタンプ推定に使用する教師文書のそれぞれに対し求め，各曲線の総和をとる．すなわち，以下の式になる．

$$TS_{n,T_C,\lambda}(date) = \sum_{t_{ci} \in T_C} TS_{n,t_{ci},\lambda}(date)$$

例として図 2 の場合，文書 n に近い 10 個の文書のうち，n と同じクラスタ C_1 に属する 5 つの文書から求まる値の合計を取り，図 5 のようなタイムスタンプ予想曲線を得る．

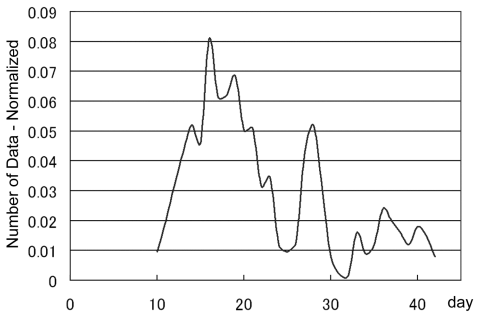


図 4　あるクラスタ内のタイムスタンプの分布例

5.2.2　誤差許容範囲

本節では，タイムスタンプ割り当ての精度を評価する為に誤差許容範囲を設定する．これは，実際のタイムスタンプと予測したタイムスタンプの差の許容範囲であり，この許容範囲に基づき，予想曲線からタイムスタンプを求める．すなわち，これは求めるタイムスタンプの細かさの基準を表す．

本節では，予想曲線の誤差許容範囲内の総和が最大になる日付を文書 n のタイムスタンプと推定する．例えば，図 5 において，誤差許容範囲が m 日の場合，前後 m 日のタイムスタンプ予想曲線の総和が最大になる日付 day_n を文書 n のタイムスタンプとする．

形式的には，文書 n のタイムスタンプ day_n は以下の式で推定する．

$$day_n = MaxArg_{day} \int_{day-m}^{day+m} TS_{n,T_C,\lambda}(day)$$

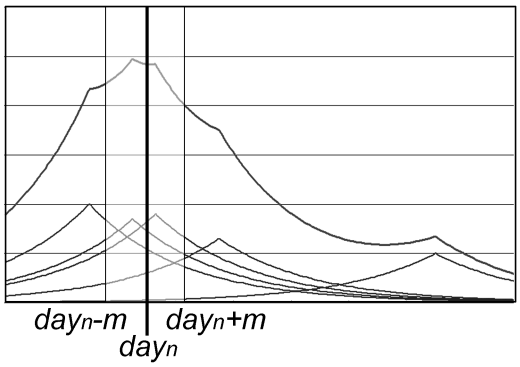


図 5　タイムスタンプ予想曲線

6. 実　　験

6.1　TDT2 コーパス

本稿では, 実験に TDT2 コーパスを用いる [7].

TDT2 コーパスは，放送されたニュースをテキストへ書き写したものと，ニュース通信の 2 種類のニュースソースからなり，1998 年 1 月から 6 月までの 6ヶ月間分のデータを含む. また,TDT コーパスには英語と中国語のデータが含まれるが, 今回は英語のソースである 4 つの放送データ (ABC,CNN,VOA,PRI) と 2 つのニュース通信 (APW,NYT) の記事の計 6 つのソースを使用する.

ここで，本稿では上記ニュースソースはいずれも速報性が高く，それぞれのソースにおいて，事象や話題の発生や変化に差はないものと想定している.

また TDT2 コーパスには，100 個の話題が定義されており，代表的なものとして,「冬季長野オリンピック」や「モニカルインスキ事件」等がある．この各話題について述べている記事には，話題との適合の度合いから “YES”（完全に適合している），“BRIEF”（一部で関連している）の 2 種類のタグが人手によって付与されている．本稿では，“BRIEF” タグは無視し，“YES” タグを持つ，あるどれかの話題についてのラベルを持つ 8040 件の記事と TDT2 コーパスで定義されている話題について述べていない 45580 件の記事の計 53620 件の記事を利用する.

また，話題の大きさは，10 件に満たない記事しかないものから，1000 件以上の記事が含まれるものまで様々である.

図 6 に “ Yes” タグが付与された記事の時間毎の分布と，それ以外の記事の分布を示す．記事の分布は全体としては一定であるが，話題を持つ記事の分布には偏りがあることがわかる.

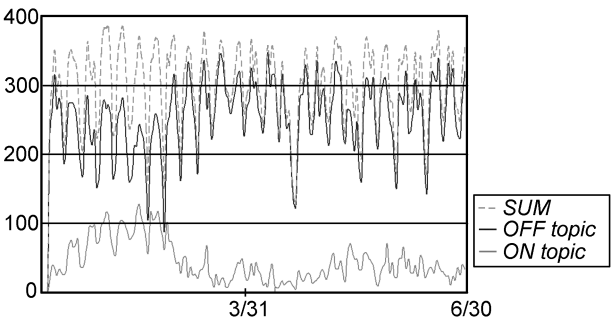


図 6　TDT2 コーパスのタイムスタンプ分布

6.2　実 験 手 順

．本提案手法の評価を，以下 3 点において行う.,

- 不完全な状態での有効性
- EM アルゴリズムの精度への影響
- 話題が限られている状態での有効性

まず最初に，本手法の，不完全な状態においての有効性を評価する．不完全な状態，すなわち TDT2 コーパスの約 8 割を占める，どの話題にも属さない，話題情報を持たない状態のデータを使用して実験を行う．これは，教師データにより，学習できない話題が多数ある状態である．実験は，TDT2 コーパス全データ（約 53000 件) を使用して行い，訓練データ，テストデータの分割は，以下のように行う.

- $D_{topic,time}$:話題、タイムスタンプの双方が取得可能なデータ: 話題が取得可能な記事集合全体の 1%(約 80 件)
- D_{topic}:話題のみ取得可能なデータ: 話題が取得可能な記事集合全体の 1%(約 80 件)
- D_{time}:タイムスタンプのみ取得可能なデータ:全体の 18%(約 10000 件)

- D_u:すべてが未知のデータ:全体の 80%

上記データを話題が割り振られているデータ集合からランダムに抽出し，D_{topic},D_u のタイムスタンプ推定を行い，精度を評価する．

ここでは，5 回の抽出作業を行い，それぞれの実験の精度の平均値を本手法の有効性とする．また，誤差許容範囲については，1 週間 (7 日)，2 週間 (14 日)，1ヶ月（30 日）の 3 パターンを評価する．例えば，誤差許容範囲が 1 週間の場合，予測したタイムスタンプと実際のタイムスタンプの差が 1 週間以内なら正解とする．また，k 近傍法の k の値は，1,2,5,10,30 の計 5 パターンと，最近点法について比較を行う．ここで，1-NN はベイズ規則による話題の分類結果に基づいて推定するのに対し，最近点法は，分類，クラスタリングの結果に関係なく，最も近いものを割り当てる為，NN と 1-NN を区別する．

続いて，EM アルゴリズムがどれほどタイムスタンプ推定に影響しているかを評価する．ここで，EM アルゴリズムによる収束回数は 20 回を最大とし，前述の実験と同じ条件において，EM アルゴリズムを適用しない，通常のベイズに基づいたタイムスタンプ推定 (収束回数 0 回) と，EM アルゴリズムによる収束回数 5 回，10 回，20 回の計 4 パターンを評価する．

最後に，全データの話題が限定されている状態での本手法の有効性について評価を行う．これは [9] において行われたように，タイムスタンプを推定する記事の話題が学習可能である状態についての本手法の評価を行う．そのため，実験には話題が割り当てられている約 8000 件の記事のみを使用し，以下のような状態でデータを用いる．

- $D_{topic,time}$:話題、タイムスタンプの双方が取得可能なデータ:全体の 1%
- D_{topic}:話題のみ取得可能なデータ:全体の 1%
- D_{time}:タイムスタンプのみ取得可能なデータ:全体の 18%
- D_u:すべてが未知のデータ:全体の 80%

ここで，EM アルゴリズムを使用した場合と使用しなかった場合について比較を行う．

6.3 実験結果

表 1, 図 7 に不完全な状態においての有効性を示す．

(EM=10)

(%)	1week	2weeks	1month
NN	31.80	40.64	54.55
k=1	36.83	45.40	58.46
k=3	36.98	46.05	60.39
k=5	35.17	44.57	59.50
k=10	32.33	42.33	58.31
k=20	14.85	30.37	57.88

表 1　不完全な状態においてのタイムスタンプ推定精度

続いて，表 2, 図 8 に EM アルゴリズムの精度への影響を評価する．

最後に，表 3, 図 9 に話題が限られている状態での精度を評価する．

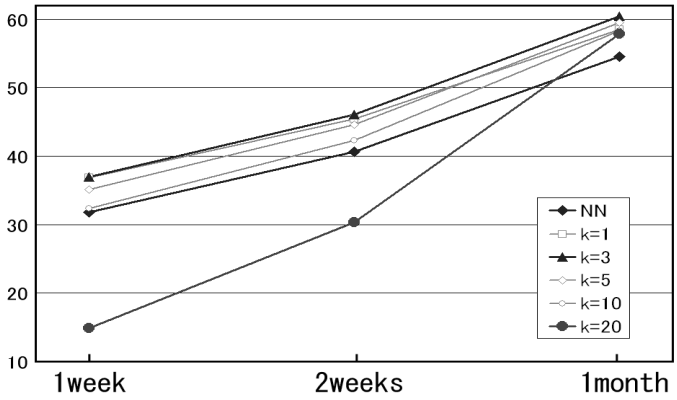


図 7　不完全な状態のいてのタイムスタンプ推定精度 (グラフ)

(K=3)

(%)	1week	2weeks	1month
Non-EM	35.33	43.97	57.90
EM5	36.07	44.90	58.92
EM10	36.98	46.05	60.39
EM20	36.07	44.84	58.74

表 2　EM の収束回数に対する精度の変化

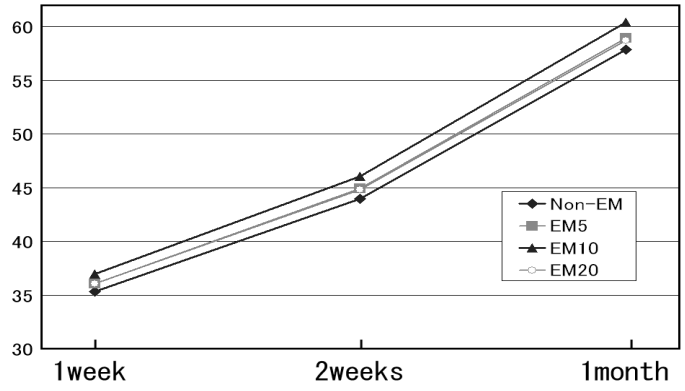


図 8　EM の収束回数に対する精度の変化

(K=5,EM=10)

	1week	2weeks	1month
OnTopic+EM	56.75	69.17	81.61
OnTopic(NonEM)	50.05	60.87	73.55

表 3　話題が限られている状態での精度

7. 考　　察

表 1, 図 7 より，教師データが非常に少数で，かつ教師データから学習できない話題が多数存在する状態においても，誤差許容範囲 7 日で，約 37%，誤差許容範囲 30 日では約 60%と，高い精度でタイムスタンプを割り当てることができた．

ここで，事象を割り当てた場合，すなわち NN 以外の場合は，一般的に NN よりも優れており，話題，事象に基づきタイムスタンプを割り当てることで，高い精度でのタイムスタンプ推定が可能であることがわかる．

EM アルゴリズムの収束回数と精度の関係を，表 2, 図 8 に示す．本実験においては，収束回数が 10 回の時において，最

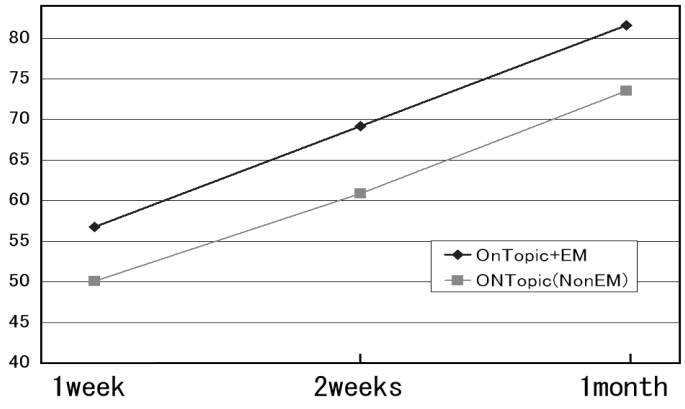


図 9　話題が限られている状態での精度

も良い成績を得た，逆に 20 回になると精度が悪化した．

文書集合においては，話題はその内容から人手により割り当てられているため，単語の出現の確率分布と話題の分類は必ずしも一致しないことが原因と考えられる．文書分類においては，EM アルゴリズムの収束回数に伴い，精度が上がるとは一概には言えず，EM アルゴリズムの最適な収束回数を推定する手法も提案されている [13]．また，EM アルゴリズムには、結果が初期値の与え方に強く依存するという問題がある．今後，これらの問題を扱った EM アルゴリズムの改良手法を適用するなど，これらの問題に対応していく必要があると考えられる．

表 3, 図 9 より，与えられている文書の話題が限定されている場合も，EM アルゴリズムが有効に働いていることがわかる．また，EM アルゴリズムを用いた場合，誤差許容範囲 7 日において，約 60%，誤差許容範囲 30 日においては，約 83%と非常に高い精度にてタイムスタンプの推定が可能となっている．

いずれの場合においても，EM アルゴリズムを用いた場合のほうが優れた精度を得ており，話題への分類において EM アルゴリズムを用いることは有効であると考えられる．また，最初の実験においては，TDT で定義されている話題に対して，EM アルゴリズムの繰り返しにより，対応できたと考えられる．

本実験においては，単一パス法におけるしきい値 $th = 0.1$, 忘却関数 $\lambda = 0.97$ の時に最も良い結果を示した．

次に，表 4 に，各期間毎のタイムスタンプ推定の精度を示す．全体的に EM アルゴリズムの使用により各期間で精度は向上しているが，最初の 1 月の期間において，EM アルゴリズムの使用により精度が落ちていることがわかる．これはそれ以前の期間の文書が存在しないことなどから，EM アルゴリズムにより，各話題や事象の重みがデータの多いほうへと収束していったと考えられる．また，話題を持つデータの量が非常に少ない期間においても，他の期間と同程度のタイムスタンプ推定精度を得ていることがわかる．これにより，EM アルゴリズムの有効性と，本手法が，不完全な状態へも対応可能であることがわかる．

8. 結　　び

本稿では, 不十分で不完全な訓練データを用いた，タイムスタンプ推定手法を提案した．TDT2 コーパスを用いた実験により，本手法の有効性を証明した．今後，この手法を応用することにより，今まで抽出できなかった情報の抽出．例えば，ソース毎に，発行時間に大きな差があるために，違う話題と判断されたものの抽出や，そのような情報の話題追跡への応用，また，より時間情報の欠落や，誤差の大きい Web ページや技術文書等，他のソースについての本手法の応用を考えている．

(K=5,EM=10,Arrowable 2weeks)

(%)	Jan	feb	Mar	Apr	May	Jun
EM	34.40	50.53	43.63	40.94	43.11	48.42
NonEM	39.10	48.92	41.44	39.37	40.90	44.21

表 4　期間毎のタイムスタンプ推定精度

謝辞

本研究の一部は文部科学省科学研究費補助金 (課題番号 16500070) の支援をいただいた．

文　　献

[1] Allan,J., Carbonell,J., Doddington,G., Yamron,J. and Yang,Y.: Topic Detection and Tracking Pilot Study: Final Report, Proc. DARPA Broadcast News Transcription and Understanding Workshop (1998).

[2] Lewis.D D.: Naive (Bayes) at forty; The independence assumption in information retrieval, In Proceedings of ECML-98, 10th European Conference on Machine Learning ,1998

[3] Grossman,D. and Frieder,O.: Information Retrieval - Algorithms and Heuristics, Kluwer Academic Press,1998

[4] Ishikawa,Y. and Kitagawa,H.: An Improved Approach to the Clustering Method Based on Forgetting Factors, in proc. 5th European Conference on Research and Advanced Technology for Digital Libraries (ECDL'01),2001

[5] Mani,I. Wilson,G: Robust temporal processing of news. In Proceedings of the 38th Annual Meeting of the Association for Computational Linguistics (ACL 2000), pages 69・6, New Brunswick, New Jersey, 2000. Association for Computational Linguistics.

[6] Nigam,K. Mccallum,A. Thrun,I. Mitchell,T.: Text Classification from Labeled and Unlabeled Documents using EM, Kluwer Academic Publishers, Boston. Manufactured in The Netherlands.

[7] National Institute of Standards and Technology(NIST); http://www.nist.gov/speech/tests/tdt/

[8] Papka,R. and Allan,J.: On-line new event detection using single-pass clustering, Technical Report UMASS Computer Science Technical Report 98-21, Department of Computer Science, University of Massachusetts,1998

[9] Uejima,H., Miura,T. SHioya,I.: Giving Temporal Order to News Corpus, The 16th IEEE International Conference on Tools with Artificial Intelligence,2004

[10] Wayne,C., Doddington,G. et al.: TDT2 Maltilanguage Text Version 4.0 LDC2001T57, Philadelphia:Linguistic Data Consortium (LDC),2001

[11] Yang,Y., Pierce,T. and Carbonell,J.: A Study on Retrospective and On-Line Event Detection, Proceedings of SIGIR-98, 21st ACM International Conference on Research and Development in Information Retrieval

[12] 上嶋 宏, 三浦 孝夫, 塩谷 勇.: 同義語，多義語の考慮による文書分類の精度向上, 電子情報通信学会誌 Vol.J87-D-I No.2,2004

[13] 新納 浩幸, 佐々木 捻.: EM アルゴリズムの最適ループ回数の予測を用いた, 語義判別規則の教師なし学習，情報処理学会論文誌 Vol.44,No.12,2003

[14] 福本 文代, 鈴木 良弥, 山田 寛康.: 話題の推移に基づく続報記事の自動抽出, 情報処理学会誌,Vol.44, No.7,2003

[15] 岩崎 学.: 不完全データの統計処理, エコノミスト社,2002