イベント開催通知メールに基づく関連コンテンツのプリフェッチ

玉野浩嗣[†1]　中村匡伸[†1]　片山　透[†1]

イベント開催通知メールから利用者が関心を持ちそうなイベントを推定することで関連コンテンツをプリフェッチするシステムを提案する．利用者の関心の推定には半構造化文書分類器を用いる．複数のイベント情報が記述された電子メールでは，イベントごとの記述範囲が明確でないため文書分類器への入力の作成が問題となる．そこで本研究では，文書中に現れる語が複数のイベントのそれぞれに関与している度合いを文書構造に従って計算することによりこの問題の解決を試みた．これにより文書分類に基づく高い精度で利用者の関心を推定しプリフェッチするシステムを構築した．

Contents Pre-Fetching based on Events in E-mail Messages

Hiroshi Tamano,[†2] Masanobu Nakamura[†2]
and Toru Katayama[†2]

We propose a system which pre-fetches web contents based on events in e-mails. Our system is capable of predicting a user's interests for each event so as only to pre-fetch user's interested contents. User interest prediction is based on semi-structured text classifier. In case of applying text classifier, we have to identify the part for each event in the multi-event e-mail. We solved this problem by using layout information. Our experiment shows our system can pre-fetch contents by predicting user's interests accurately.

1. はじめに

近年，モバイル通信環境の整備により大部分のエリアでネットワークの利用が可能となった．しかし，通信インフラの整備とは関係なく，利用者の通信キャリアとの契約状況などによりネットワークにアクセスできない状況は依然として存在する．そのため，オフライン環境下での利用者のウェブコンテンツへのアクセス要求を解決しようとする動きがある．その代表例が，Gears や HTML5 である．これらにより，ウェブアプリケーションごとのオフライン化が進んだ．例えば，ウェブメーラの Gmail やオフィススイートの Zoho などである．しかし一方で，利用者の一般のウェブコンテンツへのアクセスについては未だ解決されていない．

利用者のウェブコンテンツへのアクセスを予測し，オフラインになる前にコンテンツをプリフェッチしておく方法に，スケジューラに記載された予定に基づく方法[3] が提案されている．提案方法では 19 時から渋谷の○○で飲み会という予定がある場合，渋谷の地図や，渋谷の他の飲み屋の検索結果をプリフェッチすることができる．利用者のデータアクセスはその時のコンテキストに依存しておこるため，予定に基づく方法は有効である．しかし，この方法では利用者がスケジューラを使用している必要があった．

そこで，我々は電子メールでやりとりされるイベント開催通知に注目した．現在，会議，セミナー，展示会などのイベントは電子メールを通してやりとりされることが多い．電子メールの中から利用者が関心を持ちそうなイベント情報を抽出することで今後使用する可能性の高いコンテンツのプリフェッチが可能である．そこで，本稿ではイベント開催情報の自動抽出と利用者の関心の推定に基づき，関連コンテンツをプリフェッチするシステムを提案する．

システムはオフィス用途を念頭におき関連コンテンツとして，会議やセミナーの通知メールに記載される資料などの URL を考える．また，電子メールからのイベント開催情報の抽出にはパターンマッチを用い，利用者が関心を持ちそうなイベントの推定に半構造化文書分類器を用いた．利用者のイベントへの関心を文書分類で推定する場合，複数イベントが記述された電子メールについてイベントごとの記述範囲を特定する必要がある．本研究ではイベント開催メールのレイアウトからわかる文書の木構造を利用することによりこの問題を解決を試みた．

2. 関連研究

本提案のシステムは，電子メールからイベント情報を抽出し，これに基づきプリフェッチを行う．電子メールとイベント情報に関する研究とウェブコンテンツのプリフェッチに関する研究について述べる．

†1 NEC サービスプラットフォーム研究所
†2 Service Platforms Research Laboratories, NEC Corporation

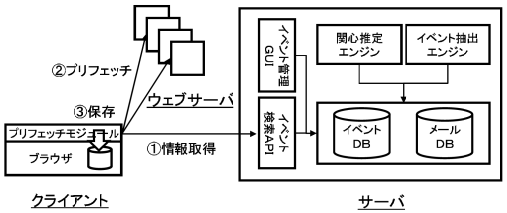

図 1　システムの概略図
Fig. 1　System overview

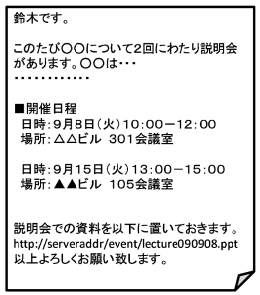

図 2　箇条書きされたイベント通知
Fig. 2　Itemized event description

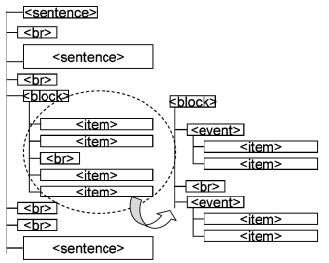

図 3　図 2 の文書木
Fig. 3　Tree structure of the document in Fig.2

電子メールで会議やセミナーなどの情報が頻繁にやりとりされることから，イベント情報の自動抽出が行われた．レイアウト情報とパターンマッチングによる方法[1]により，スケジューラへの自動入力に耐えうる精度での抽出が示された．また，電子メールとスケジューラの予定を関連付ける方法，関連付けて提示するシステム[2]も提案されている．

次に，ウェブコンテンツのプリフェッチに関しては，近い将来アクセスが期待されるコンテンツを予測しプリフェッチする方法と，利用者が現在閲覧中のコンテンツの次のアクセスを予測しプリフェッチする方法がある．前者は，利用者のスケジューラの予定に基づいてウェブコンテンツをプリフェッチする．[3]予定から抽出したキーワードを基に検索クエリを生成して検索結果をプリフェッチする．また，プリフェッチされたコンテンツを利用者が評価することで，検索クエリ生成をさらに良いものへと改善する．後者は履歴に基づく方法とコンテンツに基づく方法がある．履歴に基づく方法[6]では，URL アクセスパターンをマルコフ連鎖モデルで学習して次のアクセスを予測する．また，コンテンツに基づく方法[7]では，過去に閲覧したいくつかのページからキーワードを抽出しておき，現在閲覧中のページのリンクに書かれている文字列と比較することにより次のアクセスを予測する．

3. システムの概要

本提案のシステムの概略を図 1 に示す．システムはサーバと，クライアントで構成される．それぞれを構成する要素について以下で説明する．

3.1 イベント抽出エンジン

イベント抽出エンジンは，メール DB から電子メールを取得し，イベントを抽出してイベント DB へ格納する．イベントとは，会議，セミナー，展示会などを指す．本研究では，特に図２のような箇条書きで記述されたイベントのみを抽出の対象とする．これは，多くのイベント通知メールは，読み手に見やすいように箇条書きで記述されているためである．イベント抽出エンジンは，箇条書きを利用し日時，場所などの項目ごとの情報を取得する．また，イベントの周りに記述されている URL についてもあわせて取得する．

3.2 関心推定エンジン

関心推定エンジンは，利用者に関心のあるイベントをいくつか指定してもらうことにより利用者の関心を学習し，未知のイベントに対して推定を行なう．初めに利用者は後述のイベント管理 GUI により抽出されたイベントの一覧から，関心のあるもの，ないものにラベル付けを行う．ある程度ラベルを付けると関心推定エンジンが利用者の関心を学習し，推定を行えるようになる．推定精度により適宜，ラベル付けを行い再学習を行うことができる．

3.3 イベント管理 GUI

イベント管理 GUI は，利用者向けにイベントを管理するための GUI を提供する．（図 4 参照）（1）イベント一覧表示：抽出されたすべてのイベントを日程が新しい順にソートして表示する．（2）教師信号入力ボックス：イベントに対し関心が有るか，無いか利用者の教師信号を受け付ける．これにより，機械が利用者の関心を学習する．ある程度のイベントに入力すれば，あとは入力しなくてよい．（3）推定結果表示：イベントが利用者の関心にマッチするか推定した結果を表示する．関心のある確率を棒グラフで表現する．もし推定が間違っていれば，教師信号を入力することにより推定を訂正することができる．

また図には示さないが，イベント情報の詳細表示，イベントが抽出されたメールの閲覧，

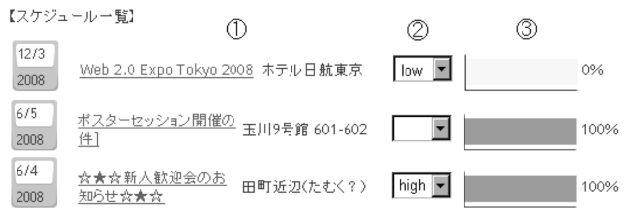

図 4　イベント管理 GUI の画面
Fig. 4　Screenshot of event manager

プリフェッチ対象の確認などの機能を有する．

3.4　イベント検索 API

イベント検索 API は日付を検索条件とし，検索範囲内のイベントの詳細情報を XML 形式で提供する．詳細情報には，イベントのタイトル，開始日時，終了日時，場所，関連 URL，推定結果などの情報が含まれている．

3.5　プリフェッチモジュール

プリフェッチモジュールはサーバからの情報をもとにデータのプリフェッチを行う．イベント検索 API にアクセスし利用者の明日のイベント情報を取得する．それらのイベントの中から利用者が関心があると推定されたイベントの関連 URL からファイルをダウンロードし，ローカルキャッシュへ保存する

4.　イベント情報の抽出

前述のイベント抽出エンジンでは，ラベル付き箇条書きと文章ブロックをパターンマッチで認識し，イベント情報を抽出する．

4.1　ラベル付き箇条書きと文章ブロックの認識

ラベル付き箇条書きは” ラベル：内容” のようなラベルと内容がセパレータにより分けられた記法である．長谷川らの研究[1] により，ラベル付き箇条書きはレイアウト情報とパターンマッチを使い認識できることから，彼らの方法を用いてラベル付き箇条書きの認識を行う．

文章ブロックとは文書中の一つの内容の単位であり，インデントや改行を用いて視覚的にひとつの塊のように記述されているものを指す．文章ブロックは自身の中にさらに文章ブロックを持つことができるため，木構造を形成することができる．図 2 の” 開催日程” の部分は文章ブロックである．

文章ブロックは開始行と終了行を特定することにより認識する．開始行は図 2 の” 開催日程” のように記号などのプレフィクスを伴うことが多い．そのため以下のようなパターンで特定する．(以下はパターンの一部である)

<Block>　:=　<Prefix> <Title> | <LParenthes> <Title> <RParenthes>
<Title>　:=　<.. 以外の文字 >+
<Prefix>　:=　○ | ● | ◎ | □ | ■ | ◆ | ◇ . . .
<LParenthes>　:=　【| < | . . .
<RParenthes>　:=　】| > | . . .

文章ブロックの終了は開始行から下に行を見て行き，インデントの数が開始行のそれよりも少なくなった場合か，改行が一定数連続した場合か，同じプレフィクスの文章ブロックの開始行を見つけた場合である．文章ブロックの終了行を見つけている時に，別のプレフィクスを持つ文章ブロック開始行を見つけた場合，再帰的に文章ブロックの認識をする．

4.2　イベントの抽出

文章ブロックを認識することにより文書を木構造に変換し，ラベル付き箇条書きからイベントを認識する．図 2 のイベント開催通知メールを木構造へ変換したものを図 3 の左に示す．右はイベントを認識後の構造である．<item> は箇条書き，<block> は文章ブロック，<sentence> はその他の文，
 は改行を意味する．

いくつかの <item> 要素をまとめてイベントとして認識する．イベントとして認識する条件は，少なくとも日時と場所を表わす <item> 要素があり，それらは木構造上の兄弟関係であるとする．認識手順としては，木構造上の兄弟関係にある要素について上から順番に <item> 要素をバケットに入れていく．次の <item> 要素まである行数以上離れてしまった時，またすでに場所（または日時）を表わす <item> 要素がバケットにある状態で場所（日時）をバケットに入れようとした時，また最後の <item> 要素の時に，バケットの中に場所と時間を表わす <item> 要素があれば，それらの <item> 要素をまとめてイベントとして認識し，バケットを空にして続ける．そうでなければバケットを空にして続ける．

以上のようにしてイベントを認識した後，イベントをタイトル，日時，場所，その他箇条書き，通知者，関連 URL の６つ組として保存する．タイトルが箇条書きで記述されない場合，電子メールの Subject をタイトルとする．その他の箇条書きは日時，場所，タイトル以

外の箇条書きはその他として一つにまとめる．通知者は電子メールの From に記載されたメールアドレスとする．最後に関連 URL はイベント抽出位置から木構造を上にたどりながら他のイベント要素を含む要素をさけて巡回して取得する．

5. 利用者の関心の推定

利用者が関心を持ちそうなイベントを推定する方法について述べる．

5.1 あらまし

前述の関心推定エンジンでは，利用者が関心を持ちそうなイベントを文書分類により推定する．文書分類とは文書をその内容に基づきいくつかに分類するもので，スパムフィルターは文書分類の応用例である．利用者の関心のあるイベントの推定は，スパムフィルターと類似点が多いことから文書分類を基本とする．

文書分類を使いイベントを利用者の関心の有無で分類する場合，複数のイベントを含む電子メールが問題となる．文書分類は，文書中に現れる語によって分類を行う．そのため，一通の電子メールに複数のイベントが記述してあると，それぞれのイベントに関係する範囲を切り出してから文書分類する必要がある．以降イベントに関係する範囲をイベント文書と呼ぶことにする．本提案のシステムでは，イベント開催メールのレイアウトからわかる構造に着目してイベント文書を切り出す．

5.2 関心推定エンジンの構成

関心推定エンジンは，前処理と分類器で構成される（図 5 参照）．前処理は，抽出されたイベントを分類器が受けとれる表現へ変換する．分類器はスプリッティングを用いた半構造化文書分類器を用い，1. タイトル，2. 場所，3. 通知者，4. その他の箇条書き，5. イベント文書の５つの項目を受け取り分類を行う．スプリッティングは，文書をいくつかの項目に分け，それぞれ独立の分類器で学習し，最終的な結果をすべての分類器の判定により決定する方法であり，フラットな分類器より精度が良い．[4] イベントのそれぞれの項目は，前処理で形態素解析により語に分解され，次の表現で半構造化文書分類器に入力される．

$$\{(f_1(w_1,e), f_1(w_2,e),\ldots),\ldots,(f_5(w_1,e), f_5(w_2,e),\ldots)\} \quad (1)$$

ここで，$f_i(w_j,e)$ はイベント e の i 番目の項目に語 w_j が出現する回数を表わす．例えば，イベント e の 2 番目の項目である場所に Tokyo という語が 1 回出現していた場合 $f_2(\texttt{Tokyo},e)=1$ である．また以降，式 1 の $(f_5(w_1,e), f_5(w_2,e),\ldots)$ 部分をイベント文書表現と呼ぶとする．イベント文書表現は，メールの中でそのイベントに関連する部分を表現するものであり，分類に最も重要な情報である．複数のイベントがメールに記載されている場合，イベント文書表現の作成が問題となる．節 5.3 でこれを解決する．

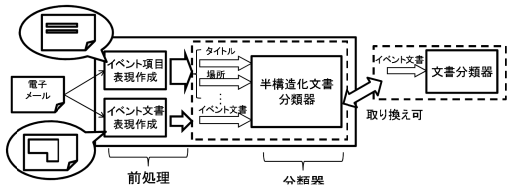

図 5 関心推定エンジンの構成
Fig. 5 Architecture of user interest prediction engine

5.3 イベント文書表現の作成

一般の文書からあるトピックのかたまりを特定することは難しい．一方，イベント開催メールは読み手を考え，一目でトピックの範囲がわかるようにインデントなど用いて見やすくレイアウトされていることが多い．そこで，イベント抽出時に導入した文章ブロックによる木構造を基にイベント文書表現を作成する．

電子メールに複数のイベントを記述する場合，基本となる記述パターンは図 6 の独立型と従属型の２パターンである．独立型はそれぞれのイベントが別々の内容で，記述もイベントごとに一箇所にまとまっている．イベント文書としては，このかたまりが認識できればよい．それに対し従属型は，イベントの内容は一つでそれらが複数の開催日をもつ．連続開催のセミナー案内などがよく従属型で記述される．従属型の場合，日時，場所から離れた場所に内容が記載されている．イベント開催通知メールは，適宜，独立型と従属型を組み合わせ記述されている．（例えば図 6 の複合型）

以上の独立型，従属型をふまえてイベント文書表現を作成する．まず，抽出対象のイベント要素から兄弟要素，親要素をたどりながら要素に出現する語に対して重みを与えていく．その後，それぞれの語ごとに重みを加算したものが，イベント文書表現の各語の値となる．

重み付けのルールを図 7 に示す．灰色の要素はイベント要素を表わし，すべての図はイベント要素 D に関してイベント文書表現を求める場合の重み付けルールを表わしている．ターゲットのイベント要素の重みは 1 である．（A）親方向について．一つ親方向をたどるに従い重みを $\alpha(<1)$ 倍する．これは，一段上の要素に出現する語は要素 D だけでなく他の要素にもかかるため，関係が $\alpha(<1)$ 倍弱くなると考えられるためである．（B）兄弟方向について．兄弟要素に他のイベントが出現していない（B 1）と兄弟要素に他のイベント要

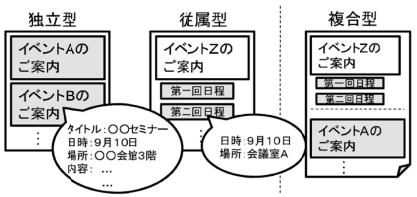

図 6　イベントの記述パターン
Fig. 6　Basic event cescription and application

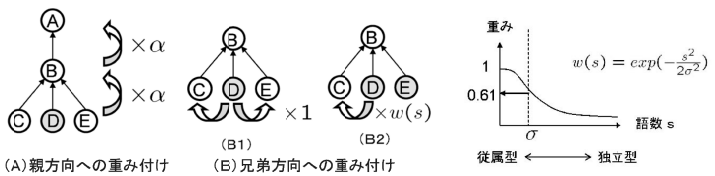

図 7　重み付けルール
Fig. 7　Weighting rules

図 8　重み関数のグラフ
Fig. 8　Decay curve of weighting

素が出現している（B２）の二つの場合に分けられる．（B１）の場合，要素 C，E について要素 D と同じ重みを与える．兄弟要素に他のイベントがなければ，それら兄弟要素は関連があると考える．（B２）の場合，他のイベント要素の語は抽出しないため要素 C への重みだけを考える．ここで，独立型と従属型を考える必要がある．もし，要素 D と要素 E が図 6 の独立型であった場合，イベント要素Dはその中で内容が閉じている可能性が高い．そのため要素 C へは重みを与えるべきではない．一方要素 D と要素 E が従属型だった場合，要素 C は関係する内容が書かれている可能性が高く重みを与えるべきである．ここで，独立型か従属型かの指標にイベントが油出された部分の語数を用いる．従属型の場合はイベントの内容は別に記述されているためイベントとして抽出できる語数が少なく，独立型の場合はイベントが抽出された場所にまとまっているため語数が多い．そこで，要素 D に出現する語数を s として式 2 で重みを与える．

$$w(s) = exp(-\frac{s^2}{2\sigma^2}) \tag{2}$$

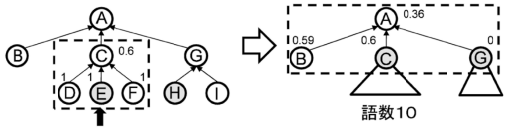

図 9　重み付けの例
Fig. 9　Example of weighting

式 2 は語数 s が増えるに従い単調減少する．（図 8 参照）語数が多ければ，独立型よりに解釈し重みが減り，語数が少なければ従属型よりに解釈し重みが増える．また，σ の値を調節することにより，減少のスピードを調節できる．（α と σ の値はそれぞれ独立に交差確認法により適切な値を設定する．）

次に図 9 を使い，少し大きい文書木にて重みが再帰的に計算される様子を見る．ただし $\alpha = 0.6, \sigma = 50$ とする．図左のイベント要素Eについてイベント文書表現を作成する．まず，図左の点線枠内について，図 7 のルールにしたがい重みを与える．要素Eは重み 1，要素 D，F については（B１）を適用し重み 1，要素 C については（A）を適用し重み 0.6．次に図右のように一段上で重み付けする．要素 G は子要素にイベント要素を含むため，子孫も含め全体をイベント要素と見なし重みは与えない．また，要素 B は（B２）の適用により要素 C 以下（つまり要素 CDEF）の語数により式 2 で重みが計算され 0.59 となる．

以上，各要素に重み付けする方法を見たが，最後に各語 w_j ごとに，すべての出現箇所で，定義した重みを加算して $f_5(w_j, e)$ を求めてイベント文書表現を得る．

5.4　イベントの分類器

本提案のシステムの分類器は，スプリッティングを用いた半構造文書分類器が利用できる．例えば，スプリッティングを用いたナイーブベイズ[5] やサポートベクターマシン[5] などである．今回はシンプルで比較的精度が良いナイーブベイズ（多項モデル）で半構造化文書分類器を実装した．スプリッティングを用いたナイーブベイズは式 3 を最大にするクラス c に文書 d を分類する．

$$p(c|d) \propto p(c) \prod_{s \in d} p_s(d_s|c) \tag{3}$$

$$p_s(d_s|c) = \prod_{w \in s} p_s(w|c)^{f_s(w,d)} \tag{4}$$

ただし，クラスを c，文書を d，文書における部分を s，語を w，文書 d の項目 s に出現

表 1 本提案システムの推定精度 $(\alpha, \sigma) = (0.6, 40)$
Table 1 Predictive accuracy

	正解率	再現率	適合率	F 値
被験者A	0.908	0.908	0.947	0.927
被験者B	0.907	0.933	0.856	0.893
被験者C	0.951	0.976	0.924	0.949

表 2 人手でイベント文書を切り分けた場合の推定精度
Table 2 Predictive accuracy given human help

	正解率	再現率	適合率	F 値
被験者A	0.905	0.913	0.942	0.927
被験者B	0.906	0.962	0.853	0.904
被験者C	0.933	0.964	0.903	0.933

する語 w の数を $f_s(w, d)$ とする.

6. 実　　験

本研究において，プリフェッチすべき対象は利用者が関心のあるイベントの関連コンテンツである．そこでシステムが関心のあるイベントをどれくらい正しく推定することができるか，またある水準の推定精度を達成するためにどれほど学習させればよいのかを調べた．

実験は被験者 ABC の 3 名で行なった．それぞれの電子メール 2,472 通，2,198 通，10,294 通を解析し，イベントが 184　161，464 ずつ抽出された．イベントの内容は，グループ会議，セミナー，展示会，飲み会などさまざまである．これに各自がイベント管理ウェブアプリから関心の有る無しを入力し，10 分割交差確認により精度を求めた（表 1 参照）．3 名の結果より，再現率は 90%，適合率は 85%より高い精度で推定することができた．また，電子メールに複数のイベントが記載されていた場合，それを一通に 1 イベントになるように手で編集して同様の実験も行なった（表 2 参照）．

表 1 と表 2 の差は 3%以内であることから，一通の電子メールに複数のイベントがある場合でも，本提案のシステムは正しくイベント文書を抽出して判定できていることがわかる．

次に，学習数を変化させた場合の正解率の変化を実験により求めた（図 10 参照）．図より，約 80 のサンプルについて学習を行えば正解率が約 90%程度に達することがわかる．

7. ま　と　め

本提案のシステムは，電子メールからイベント開催情報を抽出し，利用者が関心を持ちそうなイベントについて関連コンテンツのプリフェッチを行なう．イベント開催情報の抽出にはパターンマッチングを用い，利用者が関心を持ちそうなイベントの推定には半構造化文書分類器を用いた．利用者のイベントへの関心を文書分類で推定する場合，複数記述されたイベントについてそれぞれのイベントに関連する範囲を特定する必要があった．本研究では，レイアウトからわかる文書の木構造に着目し，重みを付けて語を抽出することによりこれを解決した．これにより，利用者が関心を持ちそうなイベントを文書分類器で推定することができるようになった．また，3 人の被験者の実験では再現率は 90%，適合率は 85%より高い精度で推定が可能であった．

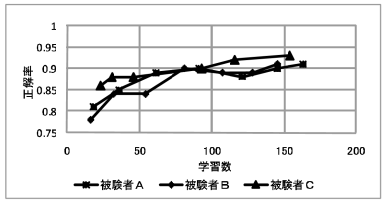

図 10 正解率の推移
Fig. 10 Improvement in predictive accuracy as training size grows

参　考　文　献

1) 長谷川隆明，高木伸一郎：文書構造の認識と言語の特徴の利用に基づく電子メールからのスケジュールとＴＯＤＯの抽出，情報処理学会論文誌，Vol.40, no.10, pp.3694-3705, Oct. 1999.
2) 高橋悟史，宮前雅一，寺田努，西尾章治郎：電子メールとスケジュールの関連性を考慮した情報閲覧システム，DEWS2007.
3) Komninos, A., Dunlop, M.D: A calendar based Internet content pre-caching agent for small computing devices,Journal of Personal and Ubiquitous Computing (online First), Springer, 2007
4) J. Yi, N. Sundaresan.: A classifier for semi-structured documents. In Proceedings 6th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, pages 340.344. ACM Press, 2000.
5) Bratko, A., Filipic, B.: A Study of Approaches to Semi-structured Document Classification. Technical Report IJS-DP-9015, Department of Intelligent Systems, Jozef Stefan Institute, November 2004.
6) R. R. Sarukkai: Link prediction and path analysis using Markov chains, Proceedings of the Ninth International World Wide Web Conference, Amsterdam, May 2000.
7) Davison B.: Predicting Web Actions from HTML Content. ,Proceedings of the The Thirteenth ACM Conference on Hypertext and Hypermedia (HT'02),College Park, MD, ACM, June 2002, pp. 159-168.