Blog 空間探索のための Blog データベースの設計

黒田 晋矢† 福田 直樹† 石川 博†

†静岡大学情報学部情報科学科 〒432-8011 静岡県浜松市城北 3-5-1

E-mail: †cs2038@s.inf.shizuoka.ac.jp {fukuta, ishikawa}@inf.shizuoka.ac.jp

あらまし 近年の Blog 人口の爆発的な増加に伴い，Blog 全体がもつ情報量が膨大になり，Blog を対象とする研究も進められている．しかしながら，研究に用いるデータの収集には研究者それぞれが Web をクローリングし，収集したデータを加工している現状がある．ここで研究者がクローリングを行うことなく必要なデータを収集できれば，クローリングにかかる手間を省き研究を効率化できる．本稿では，Blog 上のすべての種類のデータを収集し，Blog に関するどのようなデータでも柔軟に得ることができる Blog データベースを設計する．Blog データベースでは，Web 上の Blog の集合がもつ性質をデータベース上で再現することを基本理念とする．さらに複数の研究者とクローラが同時アクセスをする運用を想定して，Blog データベースに要求される機能要件を洗い出し，その要件に基づくスキーマ設計を示す．そして Blog データベースを利用した研究の一例として Blog 記事の推薦を紹介し，提案方式の有用性を考察する．

キーワード 情報検索，Web とインターネット，マルチメディア DB，ブログデータベース

Designing of Blog database for searching in Blog space

Shinya KURODA† Naoki FUKUTA† Hiroshi ISHIKAWA†

†Department of Computer Science, Faculty of Informatics, Shizuoka University 3-5-1

Jouhoku, Hamamatsu-shi, Shizuoka, 432-8011 Japan

E-mail: †cs2038@s.inf.shizuoka.ac.jp {fukuta, ishikawa}@inf.shizuoka.ac.jp

Abstract Recently, the number of blog users is increasing explosively. A huge amount of blog entries are published frequently. Blog researchers crawl their needed blog data and work on them individually. Providing blog data as common dataset will be a good idea to save researchers time and concentrate their efforts for important parts. In this paper, we propose a blog database which maintains all data contained by blogs and provides flexible, shared access to them. The basic idea of blog databases is to rebuild all properties of blogs on relational databases. First, we investigate the requirements necessary for blog databases thoroughly and describe the design of schemas based on them. Then, we introduce example researches that are using the proposed blog databases (i.e., recommending useful blog entries by analysis) and show how the functions of our blog databases run.

Keyword Information Retrieval, Web and Internet, Multimedia DB, Blog Database

1. はじめに

ここ数年，手軽に利用できる Blog を提供するホスティングサービスが増え，予想を超える爆発的な勢いで Blog 利用者が増えている[1][2]．多くの場合，Blog は個人が記事を投稿し，Blog の記事には個人の意思が反映されるため，Blog 空間の調査を通じて世論的な関心を知ることができる．Blog 空間とは，本稿では Web 上にある Blog の全体集合のことを指すとする．

Blog の爆発的な増加に伴い，Blog 検索サービスにも通常のキーワード検索以外に，さまざまな機能が付加されている．“blogWatcher”[3]や“もぶろげっと”[4][5]には，それぞれに特徴的な検索機能がある．これら検索サイトは，それぞれにデータベースをバックグラウンドに持ち，独自のデータ収集を行っている．Blog の研究者は，研究用のデータを自ら収集することがある．データの収集にはクローリングが行われるが，Blog の増加を背景に，網羅的な収集には時間がかかる上，プログラムによる自動化はネットワーク等への負荷が大きい．

検索サービスにはバックグラウンドにデータベースが用いられているが，データの収集は，それぞれのサービスがそれぞれのサービスに合うような，独自の形式で行っている．

データを収集する際には，主にクローリングが行われる．検索サービスとは別に，Blog を対象とした研究を行う者は，Blog を独自に収集する場合が多い．多数

の研究者が同じようなデータを収集するという冗長性を排除できれば，研究活動の効率向上になる．

今後，Blog に関する研究が多様に進められることが考えられる．Blog 上のすべてのデータを保持するデータベースが存在すれば，Web 上をクローリングする代わりにデータベースを検索することで研究に必要なデータを得ることができる．すなわち研究を効率よく進めることができる(図 1)．そこで，本研究では Blog 上のすべてのデータを管理する統合的なデータベースの設計を行うことを目的とする．

本研究における Blog とは，機械的に HTML ソースが生成されるものを指すこととする．広義の Blog は，個人の意見を表明する記事全般を指すため，個人がそれぞれの作成方法で公開している Web 日記なども含む．しかし，Web 日記のソースは個人の作成方法に依存し，一定の形式ではないため，正確な解析が難しい．また，ソースだけでなく URL も定型がないため，収集方法を工夫する必要がある．本研究では，基礎的な段階として解析や収集を容易にするため，機械的なコード生成を行うもののみを対象とする．具体的には，ホスティングサービスが提供する Blog サービスが対象となる．

2. 関連研究

2.1. e-Science におけるデータベースの利用

天文学やライフサイエンスの分野において，世界的に統合データベースを作る研究が行われている[6][7]．統合データベースはさまざまな観測で得られた膨大なデータをすべて管理する．研究者はデータベースの中から必要なデータを，ネットワーク経由でいつでもどこでも自由に利用できる．これまではそれぞれの研究者が独自にデータを収集し，研究を行ってきた．統合データベースではデータを収集する生産者がデータベースを管理し，利用者である研究者はデータを利用するという形を取る(図 1)．データを収集する必要がなくなる上，必要なデータはすべて手に入るようになり，研究の効率は大きく上昇した．このような研究方法を，一般的に e-Science と呼ぶ．Blog についても e-Science の考え方を導入することで，多くの研究者のデータ収集の手間を省くことができると考えられる．本来の e-Science で用いられる統合データベースとは，既に存在する複数のデータベースを，統一されたひとつのインターフェイスにより利用できるようにするものである．本研究では，この考え方を Blog 研究に適用する．開発の規模を考慮し，第一段階として，単一のデータベースとしての設計を行う．

Web 上のデータは，時間とともに変化・増大している．ある時刻における Web 全体，又は一部を保存し，任意時刻の状態を再現できる Web アーカイブがある．

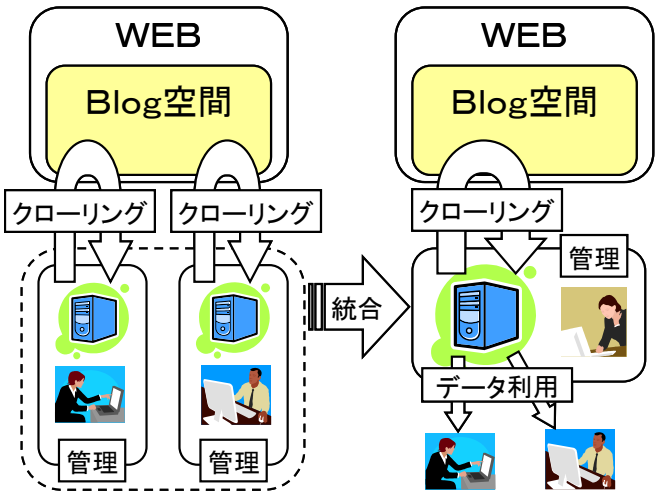

図 1. Blog データベースのモデル

Web アーカイブでは，クローリングにより定期的に Web のスクリーンショットが保存され，画像などのマルチメディアファイルも同時に保存されている．最も大規模な Web アーカイブとして，Internet Archive[8]が挙げられる．Internet Archive は Web 全体のアーカイブを作成している．その他には，各国の国立図書館では，文化的に重要な Web コンテンツの保存を行っている．Web アーカイブにより提供されるデータは，基本的にすべての要素を含むものの，検索に利用できるような特定要素の抽出は行われていない．本研究では，Blog のみを対象とする代わりに，すべての Blog に共通する要素は収集とともに抽出し，検索条件として用いられるようにする．必要なデータを独自に選別する無駄を省くことができる．

NTCIR[9]や TREC[10]では，効率的な文書検索に関する研究成果が報告されている．これら研究会においては，テストコレクションと呼ばれる研究用のデータセットが用意されている．テストコレクションは，研究用に作為的に構成されたデータセットであり，模範解答データも存在する．本研究では，テストコレクションの目的と同じく，Blog 研究のためのデータセットを提供する．データの提供のみではなく，条件により程度データを絞り込み，必要なデータのみを提供できるシステムとして設計する．

2.2. Blog データベース

テキストマイニングにより，特殊な検索サービスを提供する blogWatcher[3]がある．blogWatcher では HTML の解析によって，古くから存在する Web 日記などの Blog ツールによらないものも Blog として扱い，エントリを抽出する．奥村らによると，エントリだけに注目した場合，Blog と Web 日記に本質的な違いがないためである．本研究におけるデータベースでは，簡単のため Web 日記は取り扱わない．

画像付きのエントリに特化したもぶろげっと[4][5]がある．2007 年 2 月現在，もぶろげっとと同様の機能をもつ Blog 検索サービスは類を見ない．もぶろげっとでは，画像や動画を含むエントリのみを収集しており，検索結果にサムネイルを併せて表示する機能がある．動画については，内部モジュールにより，動画の内容を適切に伝えられる 10 枚のサムネイルを表示する．もぶろげっとは，イメージファイルへのリンクをもつエントリのみを収集する．直接 URL が記述され，論理的なリンクが張られていない場合には，これを収集しない．本研究では，網羅的なデータを提供するために，記事中に出現する URL がマルチメディアファイルである，と判断できる（例えば拡張子が.jpg 等である）場合は収集対象とする．記事内で紹介されているということは，そのファイルは記事の解析に重要な役割を担っていると考えられるからである．

3. Blog データベース

3.1. Blog データベースの機能要件

3.1.1. データベースとしての要件

Blog データベースとして必要となる，一般的なデータベースの要件を挙げる．

・データ検索
・データ復旧
・同時実効制御
・アクセス制御

(i)　データ検索

Blog は，個人の意見が記事として掲示される．さらに，コメントやトラックバックによる意見の相互交換が行われる．Blog 研究では，それらを総合的に見る場合もあれば，個別に見る場合もある．一般的な手法を用いる研究の例として，文書ベクトルを用いたテキストの解析を用いる研究[11][12]がある．また，個別の要素を用いる例として，トラックバックに注目したクローリングによる Blog 記事の推薦[13]研究がなされている．これらの研究では，すべて記事のテキストを必要とする．しかし，同じく記事を対象とする場合でも，前者の例[11]では，テキストだけでなく HTML タグも必要としたり，後者の例[13]では，トラックバックリンクが張られている記事のみを必要としたり，研究により必要なデータが違う(図 2)．また，同じデータであっても，重要度の違いや扱い方により，独自の評価に置き換えることある．このような多角的なデータ利用のために，データベースの検索機能が有効である．

(ii)　データ復旧

格納されたデータは，ブログクローラや解析プログラムのバグ，機器の故障などの不測の事態により失われる可能性がある．Blog データは膨大であり，データの復旧にも膨大な手間がかかる．そのため，機械的にデータを復旧できることが必要である．

(iii)　同時実行制御

データを一元的に管理する方法は，データを収集するクローラや，データを利用する研究者など，複数の利用者が存在する．データに対して同時にアクセスすることによるデータの破損を防ぐため，同時実行制御が必要となる．

(iv)　アクセス制御

Blog は，特定の個人や団体が記事を書くが，それらの著作権は著者に帰属し，無計画なデータの公開は著作権の侵害になる可能性がある．また，解析の仕方によっては，著者のプライバシーを侵害する可能性がある．そこで，収集したデータを一般に公開する場合，問題となる情報を隠匿するためアクセス制御を行う必要がある．

3.1.2. Blog 空間を再現するための要件

Blog 空間を再現するために必要なデータは，Blog がもつ機能や要素である．Blog がもつ要素をおおまかに挙げる．

・全体情報
・エントリ
・コメント
・トラックバック
・マルチメディアファイル

各項目について詳しい要件を考察する．

(i)　全体情報

本研究において，全体情報とは，ひとつの Blog 全体に共通する基本的な要素を指す．具体的には，Blog のトップページのみを解析すれば得られる情報である．それらの情報は，ページのトップやサイドバーに表示

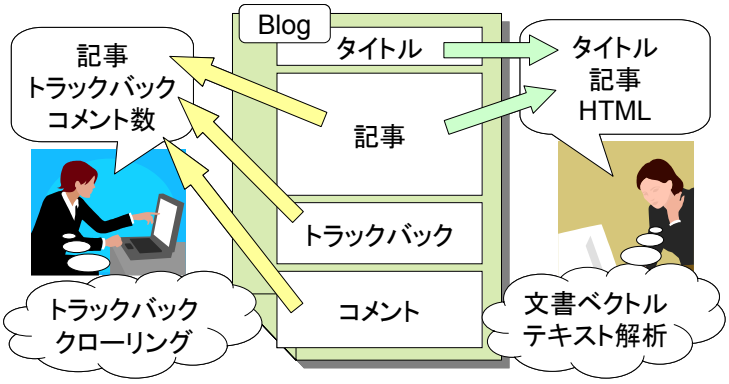

図 2. 必要なデータの違い

される．サイドバーに表示される情報は，ホスティングサービスによりさまざまで，統一的に扱うことは難しい．そこで，本研究では各 Blog サービスに共通する部分は抽出し，それ以外は，HTML ソースを収集することで対応し，統一的な処理を行えるようにする．

また，Blog には更新情報を配信するサービスがある．一般的に RSS や ATOM 等の XML 形式であることが多い．また，XML を用いない RSS2.0 等も少数ながら存在する．これら配信サービスも全体情報として取り扱う．なお，RSS や ATOM 等は，更新情報を配信するサービスであり，配信される情報のすべてが，Blog 内を解析することで得られる．よって，RSS や ATOM のファイルを収集することはしない．

(ii) エントリ

エントリとは記事のことであり，Blog において最も重要なパートである．著者は記事を載せることで意見を表明し，読者は記事を読み，コメントやトラックバックを利用して記事に対する意見を述べる．なお，本研究においてエントリと呼ぶ場合は，記事本文以外にコメントやトラックバックを含むものとする．

Blog では，記事を投稿する際，記事を分類するための単語を設定できる．サービスによっていくつかの種類があるが，カテゴリやジャンル，タグ，テーマ等で呼ばれる．基本的には任意の単語を設定できる．また，複数語を指定できるものもある．Blog では分類語によるカテゴライズが行われている．分類語は複数種類があるが，概念的には同一のものであり，本研究ではまとめてひとつの要素として取り扱う．

エントリ固有の URL をブラウザで開くと，トップページと共通する部分（サイドバー等）や，そのエントリ内の，コメントやトラックバックが同時に表示される．トップページと共通する部分は全体情報にて扱う．また，コメントやトラックバックについては，記事毎に任意数つけられるため，別途の要件として扱う．トップページと同じく，各エントリ固有の URL 毎にソースを収集し，二次的な解析に用いられるようにする．

Blog のソースにおいて，記事の本文に対するタグの係り方はホスティングサービス毎に違うため，本文の抽出には工夫が必要であり，研究効率を下げる要因になると考えられる．Blog データベースでは，記事本文を抽出し，エントリの要素として扱うことで，抽出にかかる手間を省き，効率の向上を図る．HTML タグを研究対象とする場合があるため，本文の HTML タグは残したままにしておく．HTML タグの排除は容易に行えるため，テキストのみを利用する場合は，利用者が適宜処理を行うこととする．

(iii) コメント

コメントは，エントリのうち読者が記事に対する意見を投稿したパートである．コメント本文と併せて，投稿者に対する連絡先も記載できる．連絡先としてはメールアドレスや URL（ホームページ等）が指定できるが，これらは投稿者のプロパティと考えられるため，重要である．

また，コメント本文に HTML を記述することができる場合や，URL を自動リンクする Blog サービスもあり，エントリの記事本文と同様に，HTML タグを含む状態で保存する．

(iv) トラックバック

コメントと同じく，エントリに従属するパートである．コメントは記事に対して，同じエントリ内で意見を返すのに対して，トラックバックは別に記事を書くことで，記事に対する意見を表明する．トラックバックしてもらうための URL はエントリに固有であるため，エントリの要件として扱う．

トラックバックすると，元の記事にトラックバックした記事へのリンクが張られる．その際，トラックバック先の記事に関する概要が同時に掲載される．これらリンクや概要が，トラックバックにおける重要な要件である．トラックバックの記事の概要にも HTML タグを含む可能性があり，エントリの記事本文やコメントと同じく，HTML タグを含んだ状態で保存する．

トラックバックは機械的に自動で行われるため，まったく関係のない記事からもリンクを張ることができる．この点を悪用したスパムが問題となっている．スパムは，本来はノイズであるが，スパム排除のために，スパムの特性を研究することが考えられるため，本研究では区別なく収集する．

(v) マルチメディアデータ

マルチメディアデータには，画像や動画などが挙げられる．Blog の中には，写真や著者が描いた絵を紹介しているものがある．最近では，YouTube[14]等，膨大な投稿動画を取り扱うサイトがあり，その中で著者が興味をもった動画を紹介する Blog がある．画像や動画といった，マルチメディアデータを紹介することが主目的の Blog において，ただエントリのテキストを収集するだけでは正確な解析を行うことはできない．そこで，本研究においては，Blog の記事内で紹介される画像や動画も必要要件と考える．

ほとんどのホスティングサービスにおいて，画像については容易にアップロードでき，記事中に機械的に取り入れることができる．画像の場合は IMG タグにより挿入されるが，タグの属性はサービスにより流動的である．7 種類のホスティングサービスについて IMG タグの属性を調査したところ，画像の概要となる ALT

属性があるものは 5 種類で，うち 3 種類はファイル名が指定されている．残り 2 種類は画像の概要が指定されている．TITLE 属性はなく，WIDTH と HEIGHT 属性については，4 種類のサービスにおいて指定されている．うち 1 種類は，サムネイル画像のサイズが指定されている．以上から，IMG タグの属性抽出は効果的ではないため，ファイルのプロパティから得られる情報を取り扱う．

また，動画や音声ファイルの場合，HTML タグによる直接的な掲示は稀である．特に投稿動画サイトの紹介などでは，記事中に提示される URL は，動画が閲覧できる HTML ファイルへのリンクである場合が多く，直接マルチメディアファイルを指すとは限らない．これら複数の形式に，統一的に対応できることがマルチメディアデータの要件となる．

尚，マルチメディアデータは著作物であり，著作権を侵害する恐れがあるため，一般にデータを提供する場合には十分な配慮が必要である．

3.2. Blog データベースのテーブル設計

3.1 で挙げた要件に従い，スキーマを設計する．Blog では，個人の意見をログとして公開する．ホスティングサービスごとにソースの記述方式が異なるため，ログ部分の抽出には工夫が必要であり，データセットの準備に手間がかかる．そのため，ひとまとまりのテキスト（記事本文やコメント文，トラックバックの概要等）を抽出し，保存しておく．抽出方法は，ホスティングサービス毎に処理を分ける必要があるが，本研究で設計するテーブルは，処理方法を問わず，統一的にデータを取り扱える点を考慮し，Blog 毎に共通する要素をテーブルの属性とする．

その他の設計方針としては，簡単のため，複雑な解析を必要とせず，タグの係り方を調べる程度の解析で抽出できる情報を，テーブルの属性として取り扱う．属性はテキストやソースの内容に関する属性でもあり，必要なデータを検索するための条件として利用できることが求められる．

3.2.1. 全体情報テーブル（Blog テーブル）

全体情報テーブルを表 1 に示す．

全体情報として扱う情報は，ホスティングサービス毎や Blog 毎の設定により様々である．よって，Blog 空間全体に共通する要素として，ソースか，RSS 等から抽出できる情報のうち，表 1 に定義したものを全体情報テーブルの要素とする．TITLE，AUTHOR は完全一致，ABSTRACT は部分一致による検索を行うことを想定している．ソースの解析方法はホスティングサービス毎に異なるので，ホスティングサービスの識別番号として SERVICE_ID を振る．

表 1. 全体情報テーブル

要素名	データ型	備考
ID	INT	識別番号
SERVICE_ID	INT	サービス識別番号
URL	VARCHAR(255)	トップページのURL
RSS	VARCHAR(255)	RSSのURL
RSS2	VARCHAR(255)	RSS2.0のURL
ATOM	VARCHAR(255)	ATOMのURL
TITLE	VARCHAR(64)	Blogのタイトル
AUTHOR	VARCHAR(64)	Blogの著者
ABSTRACT	VARCHAR(128)	Blogの説明
SOURCE	BLOB	トップページのソース
LAST_UPDATE	TIMESTAMP	最新更新時刻
CHECK_TIME	TIMESTAMP	最新確認時刻

表 2. エントリテーブル

要素名	データ型	備考
ID	INT	識別番号
BLOG_ID	INT	所属Blogの識別番号
URL	VARCHAR(255)	記事のURL
TB_URL	VARCHAR(255)	記事にトラックバックするためのURL
TITLE	VARCHAR(64)	記事のタイトル
AUTHOR	VARCHAR(64)	記事の著者
GROUP_TAG	VARCHAR(64)	記事の属性
NUM_COMMENT	INT	コメント数
NUM_TB	INT	トラックバック数
TEXT_ARTICLE	VARCHAR(65535)	記事本文（テキストのみ）
HTML_ARTICLE	VARCHAR(65535)	記事本文（タグを含む）
SOURCE	BLOB	記事のソース
SUBMIT_TIME	TIMESTAMP	記事投稿時刻
CHECK_TIME	TIMESTAMP	最新確認時刻

更新情報の配信形式には，一般的に RSS1.0，ATOM が知られている．それぞれ Blog 毎に固有の URL をもつので，抽出する．それ以外に，あまり用いられない形式として，RSS0.9 や RSS2.0 などがある．これらはまとめて RSS2 として扱うこととする．

3.2.2. エントリテーブル

エントリテーブルを表 2 に示す．

Blog の管理者が複数いる場合，投稿者が記事によって違う場合があるため，AUTHOR に投稿者名を保存す

る．また，記事の分類属性は，ホスティングサービスにより様々な呼び方があるが，統一して GROUP_TAG で扱う．通常，GROUP_TAG に保存される語は複数になる．そのため，GROUP_TAG は部分一致による検索の条件として利用することを想定する．

エントリのソースは時系列で変化する．例えばコメントやトラックバックがつくことが挙げられる．さらに，記事本文が修正されることがある．ソースを収集した時刻を CHECK_TIME で保存し，コメント数やトラックバック数の変化，記事本文の修正があった場合にはソースを再取得し，CHECK_TIME を更新する．LAST_UPDATE は，投稿時刻を保存する要素であり，投稿時刻が変更された場合は更新する．

3.2.3. コメントテーブル

コメントテーブルを表 3 に示す．

Blog ではコメントをつける際，メールアドレスや URL を記述する欄がある．これらを要素として抽出する．コメント本文を，HTML タグを含んだ状態で保存する．

WRITE_TIME は，コメント欄に表示される時間を抽出し，保存する．コメント文は，Blog 管理者により修正される可能性がある．修正された場合は，新しいコメント文を保存し，修正された時間を CHECK_TIME として保存する．

3.2.4. トラックバックテーブル

トラックバックテーブルを表 4 に示す．

トラックバックは，エントリに対してリンクが張られ，リンク先の記事の内容を示す概要を載せることができる．リンク先の URL やタイトル，概要を要素として抽出し，保存する．コメント同様，概要部分は HTML タグを含んだまま保存する．

トラックバックもまた，管理者による修正等，内容を変更されることがある．他のテーブル同様，再収集時刻を管理する．

3.2.5. マルチメディアテーブル

マルチメディアテーブルを表 5 に示す．

ファイルの URL では，リンク先にあるファイルの URL も収集対象とする．例えばサムネイル画像の場合，IMG タグで表示された画像に対して，アンカータグによりリンクが張ってあるとする．この場合，サムネイル画像とは別に，アンカータグ内の HREF 属性のファイルも収集する．ただし，リンク先が HTML 形式等ページを指す場合，マルチメディアファイルが複数存在する等，参照対象を特定することが難しいため，ページファイルを収集し，TAG に HEAD タグを保存する．

表 3. コメントテーブル

要素名	データ型	備考
ID	INT	識別番号
BLOG_ID	INT	所属Blogの識別番号
ENTRY_ID	INT	所属記事の識別番号
TITLE	VARCHAR(64)	コメントのタイトル
AUTHOR	VARCHAR(64)	コメントの投稿者
URL	VARCHAR(255)	投稿者が指定したURL
ADDRESS	VARCHAR(64)	投稿者メールアドレス
COMMENT	VARCHAR(4096)	コメント本文
WRITE_TIME	TIMESTAMP	コメント投稿時刻
CHECK_TIME	TIMESTAMP	コメント確認時刻

表 4. トラックバックテーブル

要素名	データ型	備考
ID	INT	識別番号
BLOG_ID	INT	所属Blogの識別番号
ENTRY_ID	INT	所属記事の識別番号
TB_BLOG_TITLE	VARCHAR(64)	リンク先Blog名
TB_BLOG_URL	CARCHAR(255)	リンク先BlogURL
TB_ENTRY_TITLE	VARCHAR(64)	リンク先タイトル
TB_AUTHOR	VARCHAR(64)	リンク先Blog著者
TB_ENTRY_URL	VARCHAR(255)	リンク先記事URL
TEXT_SUMMARY	VARCHAR(2048)	トラックバック概要（テキストのみ）
HTML_SUMMARY	VARCHAR(4096)	トラックバック概要（HTML含む）
TB_TIME	TIMESTAMP	トラックバック時刻
CHECK_TIME	TIMESTAMP	最終確認時刻

表 5. マルチメディアテーブル

要素名s	データ型	備考
ID	INT	識別番号
URL	VARCHAR(255)	ファイルのURL
TAG	VARCHAR(4096)	ファイルを出力するHTMLタグ
HEIGHT	INT	映像の場合，縦幅
WIDTH	INT	映像の場合，横幅
FILE_SIZE	INT	ファイルの容量
KIND_OF_FILE	VARCHAR(16)	ファイルの種類
FILETYPE	VARCHAR(16)	ファイルの形式
FILE	BLOB	ファイル本体

その他の属性は，特定が難しいため抽出しない．また，

記事中に URL が直接記述してあることが考えられるが，拡張子がマルチメディアファイルに属するものである等，明らかな場合は収集する．

TAG では，ソースのうち，マルチメディアファイルを記事中に埋め込む際に記述される一連の HTML タグや JavaScript 等を抽出し，保存する．3.1.2 (v)で述べた通り，IMG タグ等はあまり実用的ではないが，前後のタグも含めて保存することで解析に利用できる可能性がある．

HEIGHT，WIDTH，FILE_SIZE，FILE_TYPE は，マルチメディアファイルのプロパティから抽出する．FILE_SIZE は容量，FILE_TYPE は拡張子である．KIND_OF_FILE は拡張子を画像や動画，音声等に分類する要素である．これらは，マルチメディアファイルを検索する際の条件として機能する．

CHECK_TIME では，ファイルを収集した時刻を保存し，FILE_SIZE が変化していた場合に再収集し，FILE と CHECK_TIME を更新する．

4. Blog データベースの評価

ブログデータベースの有用性評価のひとつの方法は，ブログデータベースがブログ空間解析に利用できることを示すことである．そこで以下に示す事例を用いる．

まず事例を説明し，その事例で必要となる問い合わせを作成し，それが妥当な時間で実行可能なことを示す．さらに，事例そのものの有効性も確認したい．

4.1. 利用事例(記事の推薦)

本項では，設計した Blog データベースを用いた研究の例を挙げることで，有用性の一端を示す．例として，知名度の低い，有用な Blog の推薦を行う．

4.1.1. 目的

知名度が低いことは，参照される回数が少ないと言える．その中には，現在はあまり知られていないが，世間的な興味を集める情報が掲載されている可能性がある．そのような情報を扱う Blog を探し出すことを目的とする．

4.1.2. 推薦方法

まず，通常の検索と同じく，検索したいワードを決める．次に，一般的な分野として考えられる言葉を選び，グループ検索ワードとする．「スポーツ」や「映画」など，ジャンルとして分けられる語が相当する．

エントリテーブルの TEXT_ARTICLE に検索ワードを含み，エントリテーブルの GROUP_TAG にグループ検索ワードを含むエントリ集合を考える(図 3)．その集合を対象に，エントリテーブルの HTML_ARTICLE，コメントテーブルの HTML_COMMENT，トラックバックテーブルの HTML_SUMMARY から URL を抜き出すことで参照関係を解析し，PageRank[15]を用いる．PageRank とは，他のサイトからリンクされた数と，他のサイトにリンクをしている数を基に各サイトの人気度のようなものを算出する手法である．検索ワードを含むエントリが所属する Blog について，PageRank によりランキングする．

```
SELECT  BLOG.ID, BLOG.URL, ENTRY.BLOG_ID, ENTRY.URL,
        ENTRY.GROUP_TAG, ENTRY.ARTICLE,
        COMMENT.COMMENT, TRACKBACK.SUMMARY
FROM    BLOG, ENTRY, COMMENT, TRACKBACK
WHERE   ENTRY.GROUP_TAG LIKE '%グループ検索ワード%'
        AND ENTRY.ARTICLE LIKE '%検索ワード%'
        AND BLOG.ID = ENTRY.BLOG_ID
        AND BLOG.ID = COMMENT.BLOG_ID
        AND BLOG.ID = TRACKBACK.BLOG_ID
        AND ENTRY.ID = COMMENT.ENTRY_ID
        AND ENTRY.ID = TRACKBACK.ENTRY_ID
        AND ENTRY.BLOG_ID = COMMENT.BLOG_ID
        AND ENTRY.BLOG_ID = TRACKBACK.BLOG_ID;
```

図 3. PageRank のための集合を求める SQL の例

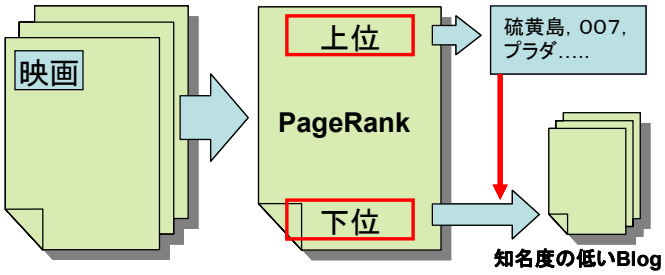

図 4. 知名度の低い Blog 発見

算出された PageRank のうち，ある閾値以上の上位 Blog 集合と，別の閾値以下の下位 Blog 集合を取る．上位 Blog 集合における検索ワードを含むエントリと，下位 Blog 集合における検索ワードを含むエントリの投稿時刻 LAST_UPDATE をチェックし，上位 Blog 集合のエントリより十分早く投稿された下位 Blog 集合のエントリを求める．このエントリを含む Blog が，有用性が高い可能性がある．

4.1.3. 評価

利用事例そのものの評価方針について説明する．

PageRank は単純に考えれば，リンクされる回数が多いほど上位になる．求められた Blog は下位集合であり，参照された回数が少ない．そのため，知名度はそれほど高くないと考えられる．

記事の有用性についての検証は，実際に中身を人間が判断する必要がある．他の検索サービスを用いて，検索した上位の結果と，内容を比較することとする．

例えば映画の場合，公開日や配給会社が載っていることは普通だが，メインキャスト以外のキャストについても記述がある場合は有用であると言える．

4.2. 評価方針

4.1 の事例により，良好な結果が得られれば，ある側面において Blog データベースを利用する価値があると判断できる．また，Blog を対象としたいくつかの研究[11][12][13]に対してデータセットを提供し，様々な研究に対応できることが証明できれば，Blog データベースの有用性を示すことができると考える．

4.3. 収集されたデータ

2007 年 2 月現在において，約 26 万件の Blog トップページのソースが収集されている．また，簡易的な解析により，全体情報のうち，SERVICE_ID，URL，RSS が抽出済みである．

5. おわりに

本稿では，Blog 空間解析に利用できる Blog データベースの必要性を論じ，その要求に沿ったデータベーススキーマの設計を示した．さらに，設計したスキーマの有効性に対する評価方針を提示した．すなわち，上記事例を実現し，それを実行することにより，Blog データベースの，ある点での有効性を評価できる見通しを得た．しかしながら，Blog データベースに関しては，上述の評価を実行することがまず必要である．

関連した課題としては，評価実験に基づくスキーマの再設計に加えて，効率的なデータ収集の実現と，正確なメタデータの抽出方法を確立することが挙げられる．また，利便性を高めるため，二次的データを抽出しておくことが挙げられる．例として，Blog 間やエントリ間の参照・被参照関係の抽出が挙げられる．

Blog は時系列に沿って更新され，変化していく．同じ記事であっても，追記や変更が容易である．現在の仕様は，最新の状態をデータとして提供することを想定いるが，これでは時系列による変化を捉えることができない．収集方法の変更により，任意時刻における Blog やエントリの状態を再現できるように，Web アーカイブの形式を取ることが考えられる．

参 考 文 献

[1] 総務省(報道資料), “ブログ・SNS の現状分析及び将来予測”, (2006.12)

http://www.soumu.go.jp/s-news/2005/pdf/050517_3_1.pdf

[2] 総務省(報道資料), “ブログ及び SNS の登録者数(平成 18 年 3 月末現在)”, (2006.12)

http://www.soumu.go.jp/s-news/2006/060413_2.html

[3] 奥村 学, 南野 朋之, 藤木 稔明, 鈴木 泰裕, “Blog ページの自動収集と監視に基づくテキストマイニング”, 人工知能学会, セマンティックウェブとオントロジー研究会, SIG-SWO-A401-01, 2004.

[4] 伊原 伸介, 林 貴宏, 尾内 理紀夫, “もぶろげっと: 画像情報を含む Blog 記事検索システム”, インタラクティブシステムとソフトウェアに関するワークショップ(WISS2005)論文集, pp.69-74, 2005.12

[5] 高地 利幸, 林 貴宏, 尾内 理紀夫, “Blog 検索エンジンもぶろげっと™への動画検索機能の導入”, 日本ソフトウェア科学会第 23 回大会講演論文集, 3A-2, 2006.9

[6] 本田 敏志, 大石 雅寿, 白崎 裕治, 田中 昌宏, 川野元 聡, 水本 好彦, “天文学連携データベースシステム(ヴァーチャル天文台)の開発・計算機資源の国際連携機構”, 日本データベース学会 Letters Vol.4, No.1, pp.173-176, (2005)

[7] 文部科学省 ライフサイエンスの広場 研究事業, “統合データベースプロジェクト”, (2007.2)

http://www.lifescience-mext.jp/download/30th/30-1-2.pdf

[8] Internet Archive, http://www.archive.org/index.php, (2007,2)

[9] NTCIR, http://research.nii.ac.jp/ntcir/, (2007,2)

[10] TREC, http://trec.nist.gov/, (2007,2)

[11] 神林 真美, 福田 直樹, 石川 博, “Blog 空間における拡大アンカーテキストと明示的リンク解析に基づくクラスタリング手法”, データ工学ワークショップ 2007 年(発表予定)

[12] 戸田 智子, 福田 直樹, 石川 博, “Blog 記事のクラスタリングに基づいたカテゴリ別話題変遷パタンの抽出”, データ工学ワークショップ 2007 年(発表予定)

[13] 鎌田 基之, 福田 直樹, 石川 博, “TrackBack と特徴語に基づく Blog クローリングと Blog 記事の推薦”, データ工学ワークショップ 2007 年(発表予定)

[14] YouTube, http://www.youtube.com/, (2007, 2)

[15] PAGE L., BRIN S., MOTWANI R., AND WINO-GRAD T. The Pagerank citation algorithm: bring-ingorder to the web. Tech. rep., Stanford Digital Library Technologies Project, 1998.