

ビッグデータ処理技術を用いた Wikipedia マイニング

プロジェクトマネジメントコース

ソフトウェア開発管理グループ

矢吹研究室

1242005

石井康之

謝辞

本研究を進めるにあたり，矢吹研究室矢吹太郎准教授には，多くの時間をご指導にさいて頂きました．また矢吹研究室の皆様には，多くの知識や示唆を頂きました．協力していただいた皆様に感謝の気持ちと御礼を申し上げます．

目次

第 1 章	序論	7
第 2 章	背景	9
2.1	研究背景	9
2.2	Wikipedia とは	9
第 3 章	目的	19
第 4 章	手法	21
4.1	VirtualBox のインストール	22
4.2	Ubuntu のインストール	23
4.3	Wikipedia の編集履歴データの取得	25
第 5 章	結果	27
第 6 章	考察	29
第 7 章	結論	31
	参考文献	33

第 1 章

序論

当研究はビッグデータ処理技術を用いた Wikipedia マイニングを行う。

Wikipedia とは、非営利の Wikimedia 財団がインターネット上で運営する、無料のオンライン百科事典プロジェクトである。記事の内容はボランティアの人々の協力によって、信頼のおける品質が保たれている。

ビッグデータとは、市販されているツールや従来のデータ処理で行うことが困難なほど巨大なデータ集合の集積物のことである。

先行研究では、Google BigQuery という、Google Cloud Platform が提供するビッグデータ解析サービスを扱った。Google BigQuery のサイトの中に、サンプルデータとして、Wikipedia の編集履歴データを提供していたので、それを用いて研究を行った。しかし、提供されていたデータは英語版だけのものであり、多言語の研究を行うことができなかったため、別の研究方法を検討する必要がある。

そこで当研究は、多言語版 Wikipedia もデータ解析できる技術を得るために、ローカルで研究が行える環境を用意し、データマイニングを行う。

Wikipedia の全編集データをマイニングすることによって、Wikipedia の品質が保たれている理由を見つけ出す。

第 2 章

背景

2.1 研究背景

Wikipedia は、多くのボランティアにより、始まってから 10 年足らずの間に、大きな成長を見せたオンライン百科事典プロジェクトである。総記事数の文字数は 10 億文字を超え、ブリタニカ国際大百科事典とエンカルタ総合大百科の合計と比較しても上回る。Wikipedia は、さまざまな言語が参加しているグローバルなプロジェクトでもある。2015 年 9 月までには、291 個もの言語が参加している。

このオープンなプロジェクトの百科事典は、制限無く誰でも自由に使用でき編集することもできる。

誰でも自由に編集できるからこそ、ボランティアの人々は気軽に参加でき、特定の企業や個人のお金を稼ぐのに力を貸していると感じることなく、時間と労力を注ぐことができる。

記事の内容はボランティアの人々の協力によって、信頼のおける品質が保たれている。しかし、中には協力的では無く、悪意のある編集をするものがある。悪意のある編集者はその記事の内容とは関係ないことを書き込んだり、記事の破壊行為を繰り返している。Wikipedia では、悪意のある編集をする人とわかっているにもかかわらず規制などをしたりはしない。記事は完成・確定されることはなく、新しい情報にいつでも改変することができる。

本研究では、Wikipedia の全編集データをマイニングすることによって、Wikipedia の品質が保たれている理由を見つけ出す。

2.2 Wikipedia とは

フリー・ライセンスの百科事典である。フリーには 2 つの意味がある。無料という意味と、自由という意味だ。Wikipedia のフリーは後者の自由という意味であり、四つの自由が与えられている。著作物を複製する自由、改変する自由、再頒布する自由、そして、改変版を再頒布する自由だ。そして、営利目的に使っても、非営利に使ってもかまわない。というものがある。Wikipedia がフリーの百科事典であるというのは、無料でアクセスできるということではなくて、自由に複製、改変、利用してかまわないということである。

Wikipedia という名前は、ウェブブラウザ上でウェブページを編集することができる Wiki というシステムを使用した百科事典であることに由来する造語である。設立者の 1 人であるラリー・サンガーにより命名された。

Wikipedia は 2015 年 9 月までには、291 個もの言語が参加している。この百科事典は多くの言語のボランティアたちによって書かれたグローバルなプロジェクトでもある。

2.2.1 記事の編集の仕方

一部の保護されているページを除いて、全てのページには「編集」と書かれたリンクがあり、このリンクを使って、あなたが閲覧しているページを編集することができます。編集ができることはウィキペディアの大きな特徴で、この機能を使って、あなたが記事を修正したり、記事に加筆することができるのです。記事に情

報を加筆する時には、情報の出典を明記してください。出典が不明な記述は、除去の対象となります。

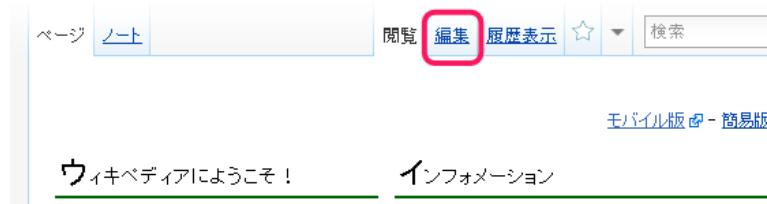


図 2.1 図の挿入例

これから常に使ってほしい大切な機能が「プレビューを表示」ボタンです。サンドボックスでなにか編集をして、それから「以上の記述を完全に理解し同意した上で投稿する」ボタンではなく、「プレビューを表示」ボタンを押してみましょう。そうすると、あなたがページに加えた変更の結果を、実際に保存する前に確認することができます。間違いは誰にでもあります。この機能は、間違いがないか自分で確認するためのものです。また、「プレビューを表示」ボタンを使えば、試しにページの体裁や表現をいろいろと変えてみても、ページの変更の記録にいちいち記録されずにすみますし、他にもいろいろと利点があるのです。でも、プレビューをした後、最後には保存するのを忘れないでください。

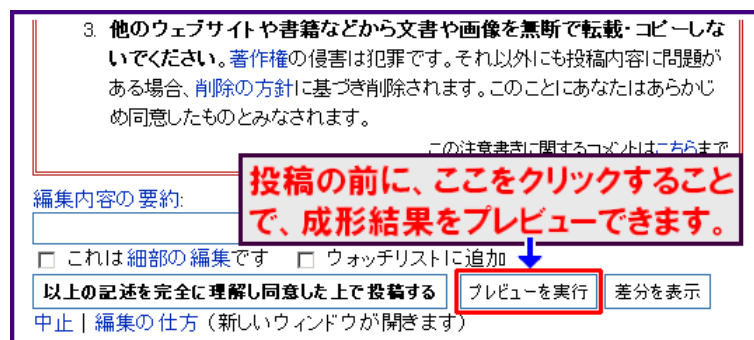


図 2.2 図の挿入例

「以上の記述を完全に理解し同意した上で投稿する」ボタンを押す前に、あなたが行った編集の説明を、編集用のテキストボックスと保存ボタンの間にある要約欄に書き込むようにしましょう。ウィキペディアでは、ここに編集の説明を書き込むことが大切なエチケットと考えられています。ただ単に誤字を直したような時には「誤字修正」と書けば充分です。文章の意味に影響を及ぼさないような、小さな修正のときには、要約欄の下にある「これは細部の編集です（説明）」のチェックボックスにチェックをいれておいてください（この機能はログイン時にのみ有効です）。



図 2.3 図の挿入例

2.2.2 Wikipedia の編集履歴データ

データ Creative Commons Attribution-ShareAlike 3.0 Unported License (CC-BY-SA) および GNU Free Documentation License (GFDL) の下にライセンスされており（Wikipedia:著作権および利用規約を参照）、再配布や再利用のためにデータベース・データの提供が行われています。データの生成は不定期に行われている。

Wikipedia ではクローリング行為のデータダウンロードは禁止されている．強引なクローリングは，Wikipedia が劇的に遅くなる原因となってしまいますためである．データベースから自動的にデータ収集している行為が発券された場合、システムの管理者から自身のサイトから Wikipedia のアクセスを禁止されてしまう措置が起こってしまうこともある．また，ウィキペディア財団が法的措置を検討する場合もあるので，注意が必要．

Index of /jawiki/

../		
20150118/	21-Jan-2015 04:39	-
20150227/	24-Feb-2015 17:51	-
20150313/	16-Mar-2015 14:37	-
20150409/	06-Apr-2015 06:19	-
20150422/	25-Apr-2015 13:52	-
20150512/	15-May-2015 08:17	-
20150602/	16-Jun-2015 01:34	-
20150703/	08-Jul-2015 14:44	-
20150805/	13-Aug-2015 21:29	-
20150826/	29-Aug-2015 04:03	-
20150901/	10-Sep-2015 23:02	-
latest/	10-Sep-2015 23:02	-

図 2.4 図の挿入例

ここに日本語版 Wikipedia の履歴データが記録されている．

URL は <https://dumps.wikimedia.org/jawiki/>

他の言語もこのような形式で履歴データが残されている．他の言語のデータを取得したい場合は URL の <https://dumps.wikimedia.org/> wiki/ の部分を変更すればよい．言語は英語のスペルで頭文字 2 文字でよい．

例：英語の場合はスペルは English なので，<https://dumps.wikimedia.org/enwiki/> とすればよい．

Index of /jawiki/latest/

../		
jawiki-latest-abstract.xml	08-Sep-2015 18:41	1823775581
jawiki-latest-abstract.xml-rss.xml	08-Sep-2015 18:41	751
jawiki-latest-abstract1.xml	08-Sep-2015 18:36	643971794
jawiki-latest-abstract1.xml-rss.xml	08-Sep-2015 18:40	754
jawiki-latest-abstract2.xml	08-Sep-2015 18:24	416000876
jawiki-latest-abstract2.xml-rss.xml	08-Sep-2015 18:40	754
jawiki-latest-abstract3.xml	08-Sep-2015 18:40	397852002
jawiki-latest-abstract3.xml-rss.xml	08-Sep-2015 18:40	754
jawiki-latest-abstract4.xml	08-Sep-2015 18:24	365950954
jawiki-latest-abstract4.xml-rss.xml	08-Sep-2015 18:40	754
jawiki-latest-all-titles-in-ns0.gz	08-Sep-2015 16:33	9871154
jawiki-latest-all-titles-in-ns0.gz-rss.xml	08-Sep-2015 16:33	775
jawiki-latest-all-titles-in-ns0.gz-rss.xml	08-Sep-2015 16:33	17646032
jawiki-latest-all-titles-in-ns0.gz-rss.xml	08-Sep-2015 16:33	754
jawiki-latest-category.sql.gz	02-Sep-2015 13:42	3115854
jawiki-latest-category.sql.gz-rss.xml	08-Sep-2015 16:33	760
jawiki-latest-categorylinks.sql.gz	02-Sep-2015 13:35	139172740
jawiki-latest-categorylinks.sql.gz-rss.xml	08-Sep-2015 16:33	775
jawiki-latest-externalinks.sql.gz	02-Sep-2015 13:41	186863400
jawiki-latest-externalinks.sql.gz-rss.xml	08-Sep-2015 16:33	775
jawiki-latest-geo-tags.sql.gz	02-Sep-2015 13:43	1324387
jawiki-latest-geo-tags.sql.gz-rss.xml	08-Sep-2015 16:33	760
jawiki-latest-image.sql.gz	02-Sep-2015 13:08	11887582
jawiki-latest-image.sql.gz-rss.xml	08-Sep-2015 16:33	751
jawiki-latest-image.sql.gz-rss.xml	08-Sep-2015 16:33	27830530
jawiki-latest-imagelinks.sql.gz	08-Sep-2015 16:33	766
jawiki-latest-imagelinks.sql.gz-rss.xml	02-Sep-2015 13:42	732
jawiki-latest-interwiki.sql.gz	08-Sep-2015 16:33	763
jawiki-latest-interwiki.sql.gz-rss.xml	02-Sep-2015 13:43	21926248
jawiki-latest-lwlinks.sql.gz	08-Sep-2015 16:33	757
jawiki-latest-lwlinks.sql.gz-rss.xml	02-Sep-2015 13:42	100477638
jawiki-latest-langlinks.sql.gz	08-Sep-2015 16:33	763
jawiki-latest-langlinks.sql.gz-rss.xml	10-Sep-2015 23:02	4420
jawiki-latest-md5sums.txt	02-Sep-2015 13:43	108858170
jawiki-latest-page.sql.gz	08-Sep-2015 16:33	748
jawiki-latest-page.props.sql.gz	02-Sep-2015 13:43	31465947
jawiki-latest-page.props.sql.gz-rss.xml	08-Sep-2015 16:33	766
jawiki-latest-page-restrictions.sql.gz	02-Sep-2015 13:43	82481
jawiki-latest-page-restrictions.sql.gz-rss.xml	08-Sep-2015 16:33	787
jawiki-latest-pagelinks.sql.gz	02-Sep-2015 13:33	523757143
jawiki-latest-pagelinks.sql.gz-rss.xml	08-Sep-2015 16:33	763
jawiki-latest-pages-articles-multistream-index...	03-Sep-2015 08:46	19678416

図 2.5 図の挿入例

どれか開くと上記のような画面になる．

ウィキページのデータは SQL のテーブルではなく、XML で提供されている。XML ファイルの文字エン

コーディングは UTF-8 である。非常にファイルサイズが大きいので、通常のエディタやブラウザで、解凍してはいけない。

データの詳細は下記のとおり

- pages-articles.xml.bz2 - ノートページ、利用者ページを除く最新版のダンプ
- pages-meta-current.xml.bz2 - 全ページの最新版のダンプ
- pages-meta-history.xml.7z - 全ページの全ての版のダンプ
- all-titles-in-ns0.gz - 全項目のページ名一覧 (標準名前空間)

2.2.3 Wikipedia:編集回数の多いページの一覧

期間: 2014-07-01 2014-07-31 のランキング。

順位 [↕]	ページ [↕]	編集回数 [↕]	総編集回数 [↕]
1 [↕]	利用者:タベストリー/sandbox [↕]	651 [↕]	917 [↕]
2 [↕]	Wikipedia:管理者伝言板/投稿ブロック/ソックパペット [↕]	379 [↕]	3098 [↕]
3 [↕]	利用者:ワナー成増/sandbox [↕]	376 [↕]	2475 [↕]
4 [↕]	Wikipedia:管理者伝言板/投稿ブロック/history20140727 [↕]	368 [↕]	4090 [↕]
5 [↕]	Wikipedia:メインページ新着投票所/新しい項目候補 [↕]	328 [↕]	10594 [↕]
5 [↕]	ハピネスチャージプリキュア! [↕]	328 [↕]	1878 [↕]
7 [↕]	FNS27 時間テレビ (2014 年) [↕]	306 [↕]	306 [↕]
8 [↕]	Wikipedia:改名提案/history20140727 [↕]	283 [↕]	6760 [↕]
9 [↕]	利用者:Tribot/log [↕]	272 [↕]	1720 [↕]
9 [↕]	利用者:ワナー成増/下書き [↕]	272 [↕]	363 [↕]
11 [↕]	2014 年のテレビ (日本) [↕]	237 [↕]	2321 [↕]
12 [↕]	インテリビレッジの座敷童 [↕]	229 [↕]	233 [↕]
13 [↕]	洪門 [↕]	209 [↕]	243 [↕]
14 [↕]	義経=ジンギスカン説 [↕]	200 [↕]	716 [↕]
15 [↕]	妖怪ウォッチ [↕]	198 [↕]	618 [↕]
16 [↕]	スカッとゴルフ パンヤ [↕]	197 [↕]	1514 [↕]
17 [↕]	笑福亭べ瓶 [↕]	192 [↕]	394 [↕]
18 [↕]	博士と助手〜細かすぎて伝わらないモノマネ選手権〜 [↕]	178 [↕]	1503 [↕]
19 [↕]	マレーシア航空 17 便 [↕]	168 [↕]	168 [↕]
20 [↕]	小保方晴子 [↕]	161 [↕]	862 [↕]
20 [↕]	Wikipedia:リダイレクトの削除依頼/2014 年 7 月 [↕]	161 [↕]	161 [↕]
22 [↕]	大相撲力士一覧 [↕]	151 [↕]	1499 [↕]
22 [↕]	花子とアン [↕]	151 [↕]	1147 [↕]
22 [↕]	利用者:チンドレ・マンドレ/sandbox [↕]	151 [↕]	516 [↕]
22 [↕]	ノート:集団的自衛権 [↕]	151 [↕]	280 [↕]
26 [↕]	ALDNOAH.ZERO [↕]	149 [↕]	190 [↕]
27 [↕]	ノート:野々村竜太郎 [↕]	143 [↕]	143 [↕]
28 [↕]	赤穂市 [↕]	142 [↕]	707 [↕]
29 [↕]	STAP 研究と騒動の経過 [↕]	140 [↕]	180 [↕]
30 [↕]	Wikipedia:コメント依頼/みしまるもも 20140528 [↕]	139 [↕]	209 [↕]
31 [↕]	GENEZ [↕]	136 [↕]	259 [↕]

32	ジェンパクト・ヘッドストロング・ビジネスコンサルティング	134	134
33	Wikipedia:保護依頼:history20140727	129	4587
34	静岡市	122	3252
35	利用者:ワーナー成増	119	419
36	利用者:ワーナー成増/下書き 2	117	176
37	計報 2014 年	116	877
38	利用者:Gowithitjam/sandbox	115	115
39	仮面ライダー鎧武/ガイム	113	2793
39	帝京大学	113	2743
41	2014 FIFA ワールドカップ	112	607
42	ドラえもん (1979 年のテレビアニメ) の帯番組時エピソード一覧	111	150
43	パワーパフガールズ	109	3655
43	利用者:Psychotic Blue/下書き 2	109	896
43	利用者:南北円上王	109	285
43	Wikipedia- ノート:管理者への立候補	109	265
47	烈車戦隊トッキュウジャー	108	958
48	Wikipedia:コメント依頼/history20140727	106	3119
48	利用者:Tamrono157/サンドボックス	106	144
50	パナソニックショップ	104	851
50	2014 FIFA ワールドカップ・決勝トーナメント	104	140
52	利用者:会話:Enyokoyama/sandbox	101	251
53	金田一少年の事件簿 (テレビドラマ)	100	1208
53	東海中学校・高等学校	100	1028
53	刺激惹規性多能性獲得細胞	100	766
56	さばげぶっ!	99	171
56	ドラえもん (1979 年のテレビアニメ) のエピソード一覧 (2001 年 - 2005 年)	99	119
58	Wikipedia統合提案/history20140727	97	4536
59	Wikipedia削除の復帰以来	96	1833
59	HERO (テレビドラマ)	96	841
59	RAIL WARS! -日本国鉄鉄道公安隊-	96	199
62	利用者:舍利弗/アンコール・ワット	95	95
63	SASUKE	94	4100

64	Wikipedia議論が盛んなノート	93	646
65	2014年の日本競馬	90	1063
66	うえのやまさおり	88	88
67	金田一少年の事件簿の犯罪者	86	907
67	利用者:Ajikoube-828/sandbox	86	308
69	国際プロレス	85	663
69	GODZILLA ゴジラ	85	525
69	実況パワフルプロ野球 2013	85	488
69	利用者:会話:南北円上王	85	145
73	2014年のオールスター（日本プロ野球）	83	83
74	ハマトラ（アニメ）	81	340
75	利用者:Quark Logo/sandbox3 文禄・慶長の役	80	91
75	星亮一	80	80
75	俺の屍を超えて行け 2	80	217
75	入江仁之	80	215
79	家族狩り	79	132
79	山下達郎	79	2486
79	ヘイトスピーチ	79	686
79	ノート:ゼーロン	79	79
79	ノート:橋本環奈	79	79
84	森川智之	78	2599
84	白雪姫	78	416
84	バイナリーオプション	78	89
87	Wikipedia:分割提案	77	1645
88	橋本環奈	76	161
88	利用者:やまさきなつこ/sandbox	76	128
88	牛丸謙吾	76	76
91	DDT プロレスリング	75	1596
91	利用者:K s/sandbox	75	1375
91	利用者:Iso10970/sandbox	75	205
94	ウルトラマンギンガ S	74	112
95	ガールズ&パンツァー	73	1271
95	チェルシーFC	73	1176
95	まじもじるも	73	140
98	Wikipedia:Bot 作業依頼	72	2161

98	札幌競馬場	72	688
100	浪曲	71	731
100	集団的自衛権	71	131
100	利用者:会話:B side of the moon	71	112

ビッグデータとは

パソコン、スマートフォンが普及し「ビッグデータ」という言葉が流行することからもわかるように、私たちは膨大な情報を日々生み出しながら生活している。Google や Yahoo! に寄せられる大量の検索クエリや、Twitter、Facebook などの SNS に投稿される文章や画像、動画、スマートフォンを利用するサービスなどで収集される位置情報データ、防犯カメラで記録される人間の表情や動きのデータなどの膨大な量のデータを

指す。

ビッグデータとは、一般的にペタ (1,000 兆)、バイト級のデータ量といわれている。このような数値的定義もあるが、ペタバイト以下であればビッグデータではないという訳でもなく、本質的には、「従来の手段では管理しきれない規模のデータ」を指す。

10^n	接頭辞	記号	漢数字表記	十進数表記	分類
10^{21}	ゼタ (zetta)	Z	十垓	1,000,000,000,000,000,000,000	ビッグデータ
10^{18}	エクサ (exa)	E	百京	1,000,000,000,000,000,000	ビッグデータ
10^{15}	ペタ (peta)	P	千兆	1,000,000,000,000,000	ビッグデータ
10^{12}	テラ (tera)	T	一兆	1,000,000,000,000	
10^9	ギガ (giga)	G	十億	1,000,000,000	
10^6	メガ (mega)	M	百万	1,000,000	
10^3	キロ (kilo)	K	千	1,000	

4V による BigData の定義

IBM による BigData の定義で 4V というものがある。4V とは、容量 (Volume)、種類 (Variety)、頻度・スピード (Velocity)、正確さ (Veracity) から構成されている。

容量 (Volume)

ビッグデータの特徴である容量の巨大さを指す。企業内外にはデータが溢れており、数テラバイトから数ペタバイトにもおよぶ。またデータが増大することによる計算量も非常に膨大となる。

種類 (Variety)

ビッグデータは企業システムで通常扱っているような顧客情報や販売データ、経理データ、在庫データなどの構造化データであるとは限らない。テキスト、音声、ビデオ、ページ遷移、ログファイルなどのさまざまな種類の非構造化データも存在する。

頻度・スピード (Velocity)

今この瞬間にも、ものすごい頻度で RFID などの IC タグやセンサーなどからデータが生成されている。昨今の変化の著しい市場環境では、これらのデータによりリアルタイムに対応したものを求められている。

正確さ (Veracity)

データの矛盾、曖昧さによる不確実性、近似値を積み重ねた不正確などを排除して、本当に信頼できるデータが意思決定には重要である。

以上が IBM による 4V の定義であるが、容量 (Volume) については最初に記述したように、必ずしもペタバイト以上でなければならないとは考えていない。

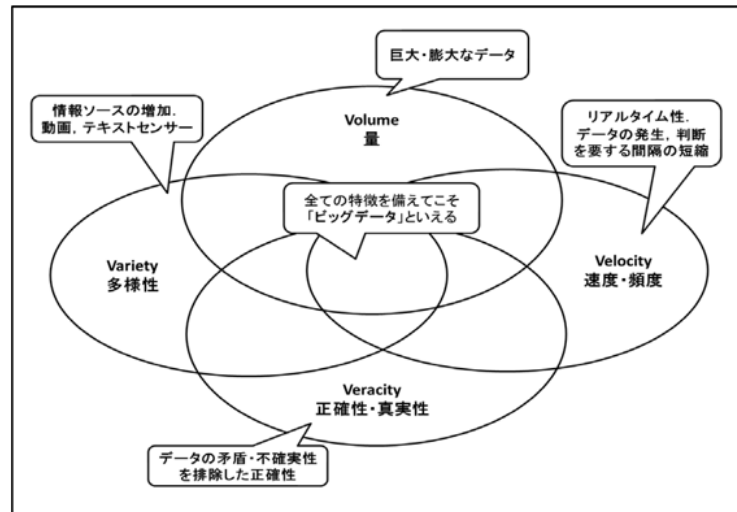


図 2.6 ビッグデータの 4V

3V でのビッグデータの定義

3V で表す場合は、容量 (Volume)、種類 (Variety)、頻度・スピード (Velocity) の 3 つになり、正確さ (Veracity) は含まれない。確かにビッグデータには正確でないデータが混在することもあり、例えば Twitter などの SNS データには、冗談やデマ情報の書き込みなども混じっており、センサーなどでも故障によるノイズが混じることもあります。データ量が少ない場合には、外れ値として手作業で除去することも可能だがいわゆるビッグデータと言われている大量のデータの場合は、手作業によるデマ情報やノイズなどの除去はほとんどふかのうである。

しかし、ビッグデータで収集するデータは殆どが生データであるという特徴がある。正確さをどう定義するにもよるが、センサーからの入力データなどは生データそのものだが、SNS からの入力データなども生データであり、その意味では正確なデータといえる。つまり、編集などで手が加わっていないデータであり、またそこで発せられるメッセージはその人が、その人の環境により制約などを感じるデータではないからだ。これは、勤務する企業・組織内で作成する報告書など対比して考えるとわかりやすいだろう。

ビッグデータ処理のパターン

下記の表に、3 つの処理パターンの特性を簡潔にまとめたものを記述する。

	バッチ処理	インタラクティブクエリー処理	ストリームデータ処理
実行タイミング	ユーザー指定と定期的実行	ユーザー指定と定期的実行	常時連続実行
処理単位	蓄積データをバッチで一括処理	蓄積データをバッチで一括処理	少数のフローデータ処理
実行時間	分～時間	秒～分	ミリ秒～秒
処理モデル	MapReduce	クエリ・OLTP	ストリーム処理

ビッグデータの処理パターンには、「バッチ処理」、「インタラクティブクエリー処理」、「ストリームデータ処理」の 3 種類のパターンがある。

バッチ処理では蓄積データをバッチで一括処理だが、これは Google 検索用に開発された MapReduce 処理を利用した Hadoop が代表的である。しかし、Hadoop はビッグデータ処理用として開発されたものではないので、処理結果作成に時間がかかるという欠点がある。

インタラクティブクエリー処理は、蓄積された大容量データをオンラインクエリなど使用して一括解析処理するものである。インタラクティブクエリー処理では蓄積されたビッグデータを数秒から数分で実行する。

ストリームデータ処理は大量発生する実世界データを逐次に時系列処理する技術である。データ発生時にあ

あらかじめ登録したシナリオにしたがって集計・分析に必要なデータを抽出し、データ処理を行う。このように逐次時系列でデータ処理できることから、最新の情報、その中での特異な値の発生などに対してリアルタイムに対応するシステムを構築できることが特徴で IoT への応用に最適な処理方法といえる。

バッチ処理

最初に MapReduce で代表される、バッチ処理の特性を見る。

数十年前のメインフレームは、主記憶が数百キロバイト、価格は数千万程度だった。これを現在の PC（主記憶数ギガバイト、価格は 10 万円程度）と比べた場合、価格性能比では約 100 万倍にもなる。また、CPU 処理スピードと、ネットワークの帯域幅についてはそれぞれ、主記憶と類似の性能向上を遂げてきている。

このようなことは、コンピューター関係以外の業種ではまったく例を見ない群を抜く性能向上である。この急激なプラットフォームの真価がクラウドコンピューティングやそのうえで実行されるビッグデータ処理などを可能にしている。

ただし、これはクラウドなどに限ったことではない。IT の世界ではこれまでも短いタイムスパンで新しいテクノロジー・ブレークスルーやビジネスモデルが出現してきており、これはプラットフォームやネットワークの新派に依存している部分が多くある。言葉を変えれば、これらの新しい発想はその時点でのプラットフォーム性能で初めて成り立つものであり、これをわずかも前の世代に思いついたとしても、実現不可能である場合が多い。

このようにクラウドなどの先端 IT システムは、現在のそしてこれからも進化を続けるはずのプラットフォームやネットワークに依存したものである。

インタラクティブクエリー処理

インタラクティブクエリー処理は、BigQuery による説明を記述する。Google がリリースするソフトウェアツールには、もともと Google が社内使用の目的で開発していたものも多く、BigQuery もそれに当てはまる。Google も当初は社内使用でも MapReduce を使用してビッグデータ処理を行っていたが、バッチ処理による結果生成の遅延や処理を行うための準備の煩雑さなどから、それに代わるツールとして開発されたのが BigQuery である。BigQuery はデータの入力または JSON フォーマットのファイルから直接行うことができる。また、Cloud Storage からのデータロードも可能である。ビッグデータの解析や絞り込みは RDB (Relational Database) の SQL に類似したクエリ言語を使用し、UI 画面や PC のコマンドラインから容易にデータ検索を行うことができる。他にも Excel を使用し検索・表示を行うことができる。

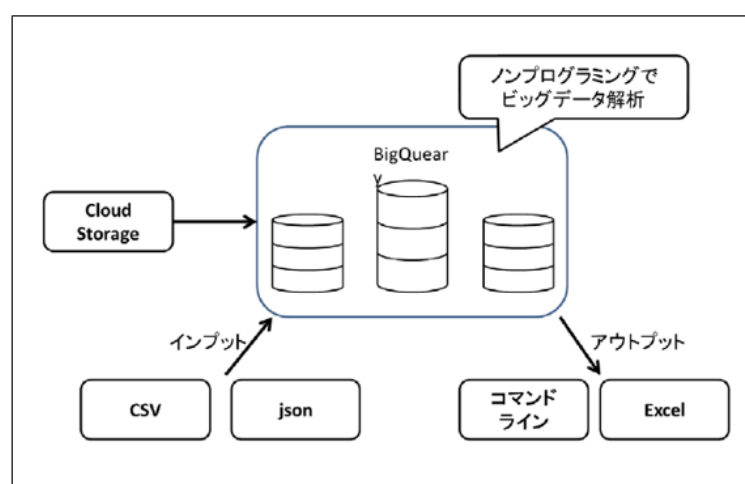


図 2.7 BigQuery の入出力

BigQuery を使用したインタラクティブクエリー処理では、マウス操作と簡単なキー入力によって全ての操作を行うことができる。また結果出力も数秒から数分以内で得ることができる。したがって、何かこのデータを

解析したいと考えたとき、その場で気軽に行えるという点が一番の特徴として挙げられる。

BigQuery でのビッグデータ解析は大量のデータを超高速で行えるのも大きな特徴である。例として、15 億行のデータに対する比較的複雑な集計問い合わせが 20 秒から 25 秒で返ってきたというユーザの実行結果もある。BigQuery では、インデスクを作成する必要がなくデータをロードするだけでこのような高速クエリが実行できる。キャッシュは戸鵠ボタンから有効無効を切り替えられるが、キャッシュを使っていなくても、使っているばあ地と同様の結果が得られる。

ストリームデータ処理

ストリームデータ処理は、大量発生する時系列のデータ（ストリームデータ）をリアルタイムに逐次処理する技術。

ストリームデータ処理は、データ発生時に、あらかじめ登録したシナリオにしたがって集計・分析に必要なデータを抽出し、データ処理を行う。その際、分析対象データをメモリー上で処理する「インメモリデータ処理技術」により、高速なデータ処理を実現している。これらの技術によって、大量データを高速に、かつリアルタイムに処理できる。例えば、株価のテクニカル指標やランキング情報から売買をリアルタイムに自動判定する。といったシステムに大変有効である。他にも、リアルタイムの在庫管理や、不正操作の監視を行うシステムなど、多くの利用目的が考えられる。

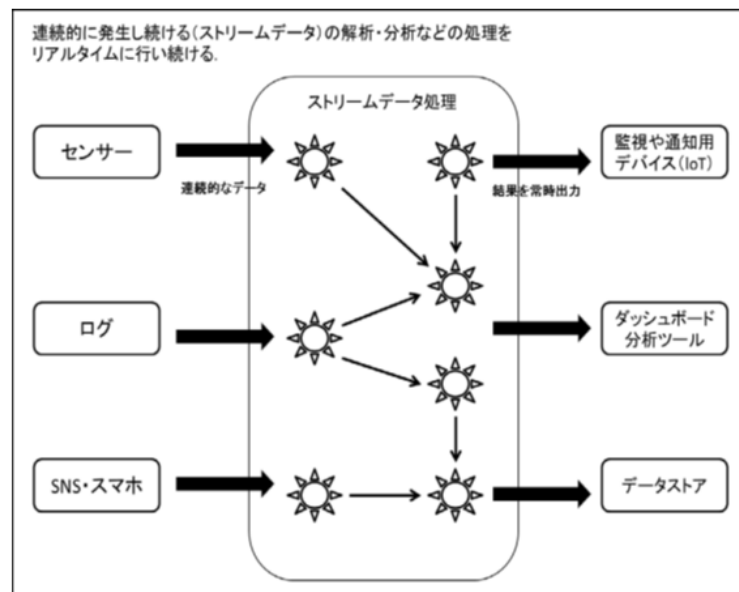


図 2.8 ストリームデータ処理

第 3 章

目的

研究目的

Wikipedia を一つのプロジェクトとみなし、このオンライン百科事典で品質管理がどのように行われているか調査する。この調査により、オープンな共同作業プロジェクトにおける、品質管理マネジメントのあり方についての知見を得たい。

プロジェクトマネジメントとの関連

本研究は、プロジェクトマネジメントを学ぶことを目的としているため、プロジェクトマネジメントとの関連が全面的にある。

考案するゲームは、プロジェクトマネジメント知識体系ガイド (PMBOK®ガイド) の第 4 版(以下、PMBOK)を参考にし、プロジェクトマネジメントとの一連の活動や、PMBOK に記載されている 9 つの知識

エリアについての内容を活用する。

PMBOK とは、プロジェクトマネジメントに関する知識体系である。

現在は、PMBOK に従ってプロジェクトマネジメントを実施することが、デファクトスタンダードになっている。

PMBOK に記載されている 9 つの知識エリアとは、何をやるべきかという観点、何を管理するべきかという観点からみたものである。

9 つの知識エリアは、以下 9 つのマネジメントについての内容となっている。

1. プロジェクト統合マネジメント
2. プロジェクト・スコープ・マネジメント
3. プロジェクト・タイム・マネジメント
4. プロジェクト・コスト・マネジメント
5. プロジェクト品質マネジメント
6. プロジェクト人的資源マネジメント
7. プロジェクト・コミュニケーション・マネジメント
8. プロジェクト・リスク・マネジメント
9. プロジェクト調達マネジメント

本研究では、上記の PMBOK の中の品質マネジメントが関連性が最もあるものであるといえる。このオープンなプロジェクトの百科事典は記事の作成や、編集が主に行われて創り上げられており、その成果物がこの百科事典である。

成果物のイメージ 差し戻しに関するデータを収集し、編集回数や頻度などの要素を洗い出す。そして、いくつかの要素から条件を決めクラスター分析を行う。その結果から悪意のある編集がされている記事に共通する点を見つけ、Wikipedia のオープンなプロジェクトでの品質マネジメントの知見を得る。

第 4 章

手法

研究方法

1. Wikipedia 日本語版の編集履歴まで含んだファイルをダウンロードし、ローカルでデータマイニングを行う。
2. どのような品質管理が行われているか、分析結果から調査を行う。
3. オープンなプロジェクトにおける品質管理マネジメントのあり方を提案する。

研究を行うための用意

開発環境として Linux を扱う。そのために、VirtualBox と CentOS を用意する。

VirtualBox とは

使用しているパソコン上に仮想的なパソコンを作成し、別の OS をインストール・実行できるフリーのパソコン仮想化ソフトのことである。本研究では、LinuxOS を扱いたい、パソコン本体は WindowsOS の為、このソフトを利用する。

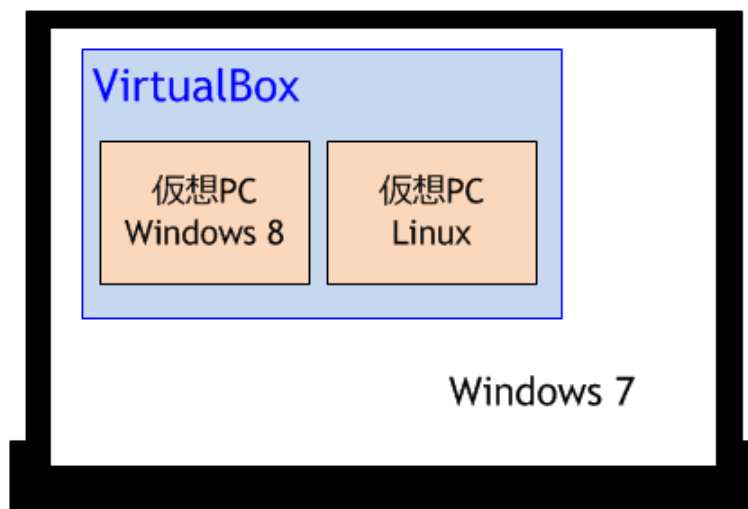


図 4.1 図の挿入例

VirtualBox はコンピュータ上で直接動作している通常の OS にとってはアプリケーションの一つであり、他のソフトと同じように起動することができる。起動すると仮想的なコンピューターが構築され、元の OS とは独立に別の OS を起動することができる。VirtualBox が実行されている OS をホスト OS、VirtualBox 上で実行されている OS をゲスト OS という。

元は独立系のソフトウェア企業が開発・販売していた製品だった。しかし、開発元が Sun Microsystems 社に買収され、その後同社が Oracle 社に買収されたため、Oracle 社が開発元となり、正式名称も「Oracle VM

VirtualBox」となった。また、VirtualBox 本体は GPL に基づいたオープンソースソフトウェアとして公開され、誰でも自由に入手・利用・改変・再配布などが行える。

VirtualBox を使う上での注意点

現時点での VirtualBox は仮想メモリをサポートしていないため、実メモリ以上のメモリを仮想 PC が使用することはできない。仮想メモリを使うと動作が遅くなるため、仮想 PC には実メモリ以内のサイズを割り当てる。そのため、仮想 PC を 1 台だけ起動するのであれば問題ないが、複数の仮想 PC を同時に起動させる場合これがネックになってしまう。同時起動させる全ての仮想 PC のメモリサイズの合計が実メモリのサイズを超えないようにする。そのため、VirtualBox をインストールする PC には多くのメモリが必要で、最低 4GB 以上の PC を使うようにするべきである。

4.1 VirtualBox のインストール

4.1.1 ダウンロード

Oracle が提供している OracleVirtualBox というのを下記のサイトからダウンロードする。本研究では WindowsOS を使用しているので、VirtualBox platform packages の中にある「VirtualBox 5.0.4 for Windows hosts」というのを選択する。



図 4.2 図の挿入例

ダウンロードサイト

<https://www.virtualbox.org/wiki/Downloads>

4.1.2 インストーラーの実行

下記のサイトから VirtualBox の導入から OS インストールまでの解説が載ってるので参考にする。



図 4.3 図の挿入例

参考サイト

<http://success.tracpath.com/blog/2013/10/15/virtualbox>

4.2 Ubuntu のインストール

4.2.1 ISO イメージをダウンロードする

まず, <https://www.ubuntulinux.jp/download/ja-remix> より Ubuntu 14.04 の ISO イメージをダウンロードする。本研究では 64bit 版を選択した。



図 4.4 Ubuntu の ISO イメージダウンロードページ

4.2.2 インストールを開始する

次にインストールした VirtualBox を立ち上げ、ウインドウ左上にある「新規 (N)」のボタンを押してゲストマシンの作成を行う。ゲストマシンの名前、メモリサイズ、HDD の設定をするとウインドウが閉じられる。

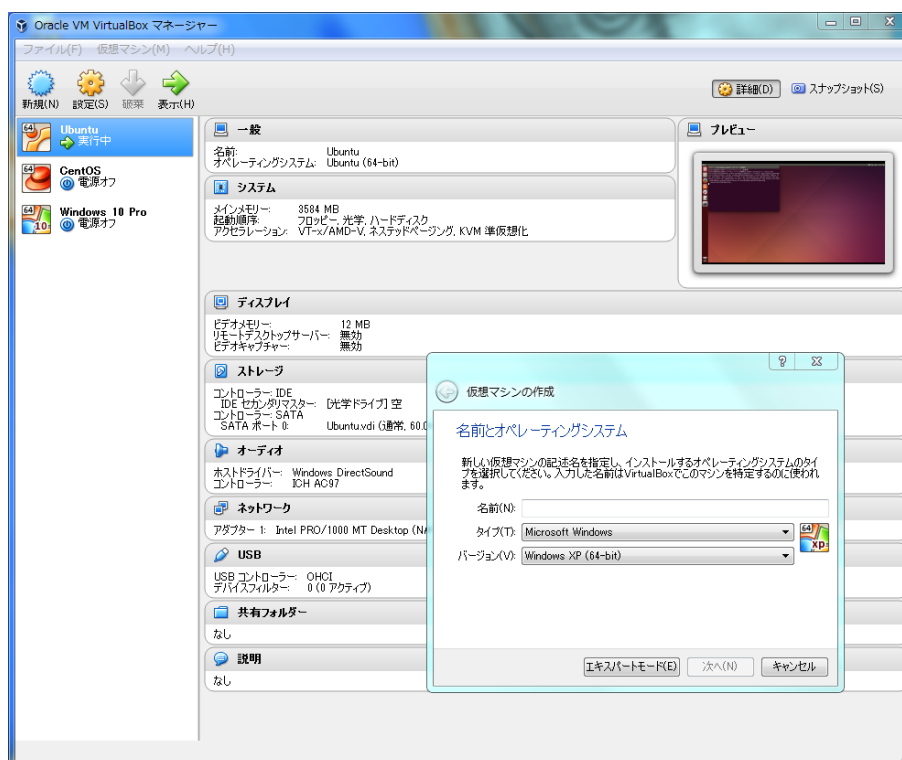


図 4.5 VirtualBox のウインドウとゲストマシン作成ウインドウ

続いて「設定 (S)」を開き、「ストレージ」の項目からダウンロードした ISO ファイルをセットして「OK」をクリックする。

「起動 (T)」を押すとゲストマシンが起動し、Ubuntu のインストールウィザードが表示される。表示内容に従ってウィザードを進めていくと、ゲストマシン内に Ubuntu がインストールされる。

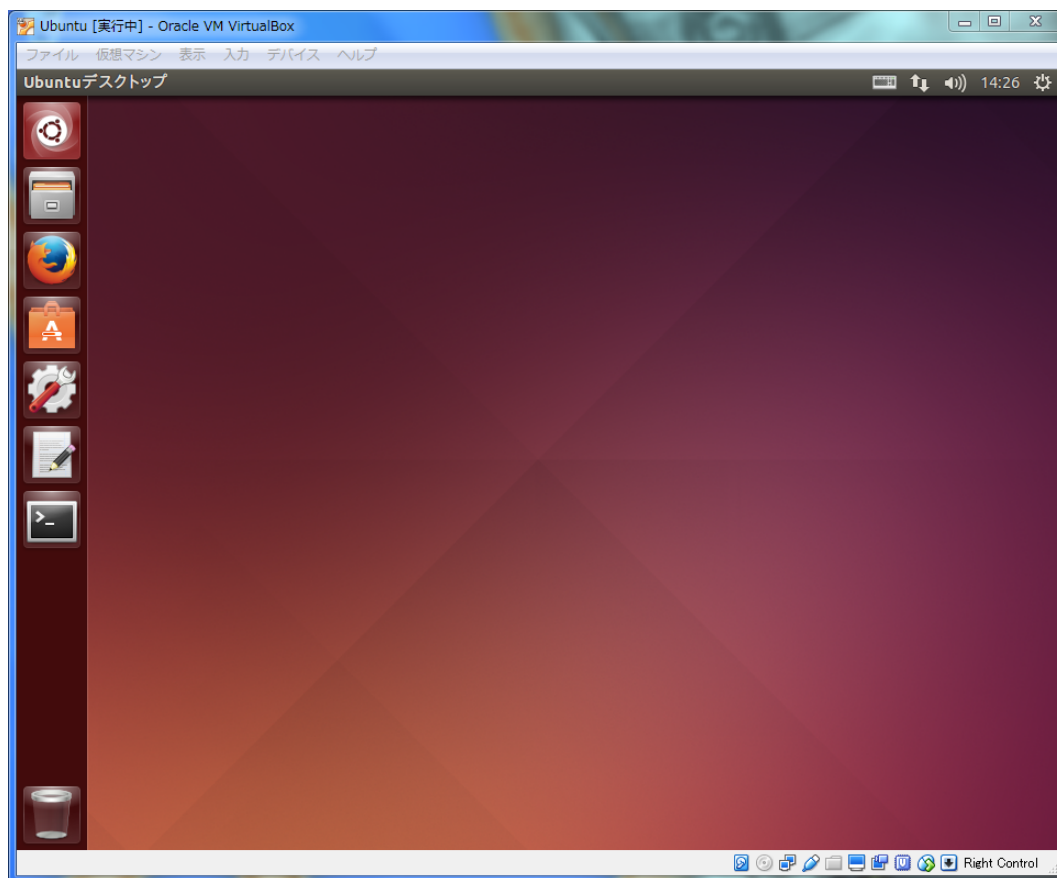


図 4.6 VirtualBox 上で動作する Ubuntu

4.3 Wikipedia の編集履歴データの取得

4.3.1 ファイルのダウンロード

日本語版 Wikipedia のデータは、<https://dumps.wikimedia.org/jawiki/>からダウンロードできる。

Index of /jawiki/

../		
20150221/	24-Feb-2015 17:51	-
20150313/	18-Mar-2015 14:37	-
20150402/	05-Apr-2015 06:19	-
20150422/	25-Apr-2015 13:52	-
20150512/	15-May-2015 08:17	-
20150602/	18-Jun-2015 01:34	-
20150703/	08-Jul-2015 14:44	-
20150805/	13-Aug-2015 21:29	-
20150826/	29-Aug-2015 04:03	-
20150901/	10-Sep-2015 23:02	-
20151002/	06-Oct-2015 03:14	-
latest/	06-Oct-2015 03:14	-

図 4.7 日本語版 Wikipedia データダウンロードサイト

再現性を確保するため、20150901 番を扱う。latest だと、最新だが、日々更新されて変更されてしまう可能性があるため今回は利用しない。

http://www.mwsoft.jp/programming/munou/wikipedia_data_list.html によると、stub-meta-history.xml が使えるので、これを利用する。

2015-09-03 20:46:56	done	All pages, current versions only. jawiki-20150901-pages-meta-current1.xml.bz2 318.6 MB jawiki-20150901-pages-meta-current2.xml.bz2 687.5 MB jawiki-20150901-pages-meta-current3.xml.bz2 263.3 MB jawiki-20150901-pages-meta-current4.xml.bz2 1.2 GB
2015-09-03 04:50:23	done	Recombine articles, templates, media/file descriptions, and primary meta-pages. jawiki-20150901-pages-articles.xml.bz2 2.0 GB
2015-09-03 01:08:10	done	Articles, templates, media/file descriptions, and primary meta-pages. jawiki-20150901-pages-articles1.xml.bz2 292.6 MB jawiki-20150901-pages-articles2.xml.bz2 587.7 MB jawiki-20150901-pages-articles3.xml.bz2 216.1 MB jawiki-20150901-pages-articles4.xml.bz2 974.8 MB
2015-09-02 10:10:48	done	Recombine first-pass for page XML data dumps <i>These files contain no page text, only revision metadata.</i> jawiki-20150901-stub-meta-history.xml.gz 3.9 GB jawiki-20150901-stub-meta-history1.xml.gz 894.5 MB jawiki-20150901-stub-meta-history2.xml.gz 1.4 GB jawiki-20150901-stub-meta-history3.xml.gz 416.8 MB jawiki-20150901-stub-meta-history4.xml.gz 12.4 GB jawiki-20150901-stub-meta-current1.xml.gz 12.4 MB jawiki-20150901-stub-meta-current2.xml.gz 64.9 MB jawiki-20150901-stub-meta-current3.xml.gz 33.6 MB jawiki-20150901-stub-meta-current4.xml.gz 160.2 MB jawiki-20150901-stub-articles1.xml.gz 10.6 MB jawiki-20150901-stub-articles2.xml.gz 50.5 MB jawiki-20150901-stub-articles3.xml.gz 25.1 MB jawiki-20150901-stub-articles4.xml.gz 110.9 MB
2015-09-02 04:54:26	done	First-pass for page XML data dumps <i>These files contain no page text, only revision metadata.</i> jawiki-20150901-stub-meta-history1.xml.gz 894.5 MB jawiki-20150901-stub-meta-history2.xml.gz 1.4 GB jawiki-20150901-stub-meta-history3.xml.gz 416.8 MB jawiki-20150901-stub-meta-history4.xml.gz 12.4 GB jawiki-20150901-stub-meta-current1.xml.gz 12.4 MB jawiki-20150901-stub-meta-current2.xml.gz 64.9 MB jawiki-20150901-stub-meta-current3.xml.gz 33.6 MB jawiki-20150901-stub-meta-current4.xml.gz 160.2 MB jawiki-20150901-stub-articles1.xml.gz 10.6 MB jawiki-20150901-stub-articles2.xml.gz 50.5 MB jawiki-20150901-stub-articles3.xml.gz 25.1 MB jawiki-20150901-stub-articles4.xml.gz 110.9 MB
2015-09-08 18:41:15	done	Recombine extracted page abstracts for Yahoo jawiki-20150901-abstract.xml 1.7 GB
2015-09-08 18:40:39	done	Extracted page abstracts for Yahoo 2015-09-08 18:40:34: jawiki (10 7400) 8393 pages (91.7 2258.3/sec all curr), 8393 revs (91.7 105.7/sec all curr), ETA 2015-09-09 04:34:51 [max 3277733] jawiki-20150901-abstract1.xml 614.1 MB jawiki-20150901-abstract2.xml 396.7 MB jawiki-20150901-abstract3.xml 379.4 MB jawiki-20150901-abstract4.xml 349.0 MB
2015-09-08 16:33:56	done	List of all page titles jawiki-20150901-all-titles.gz 16.8 MB
2015-09-08 16:33:46	done	List of page titles in main namespace jawiki-20150901-all-titles-in-ns0.gz 9.4 MB
2015-09-02 13:43:57	done	List of pages' geographical coordinates jawiki-20150901-geo_tags.sql.gz 1.3 MB

図 4.8 20150901 の stub-meta-history

第 5 章

結果

第 6 章

考察

第 7 章

結論

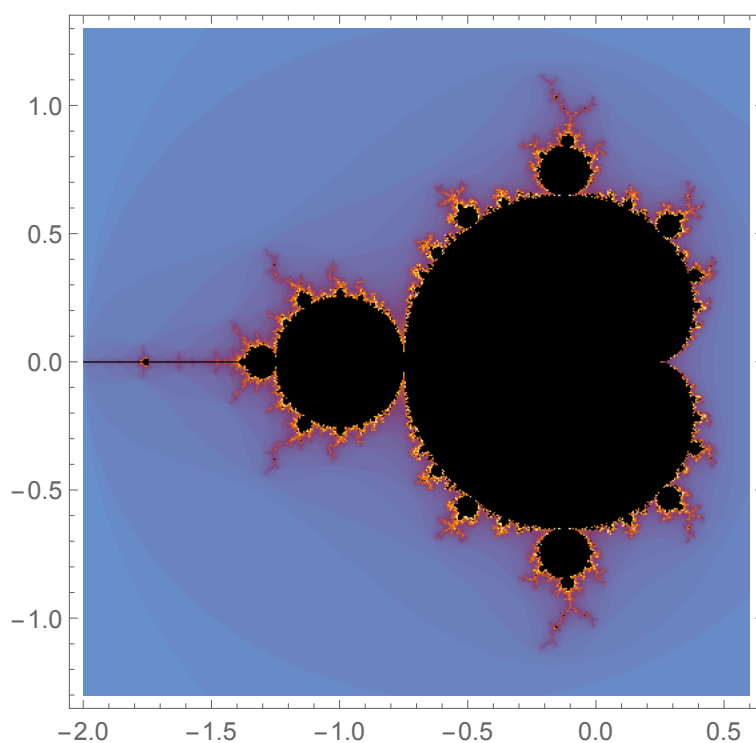


図 7.1 図の挿入例

参考文献は文献ファイル（この文書では biblio.bib）に記述し，\cite で参照する．例：データベースのための問い合わせ言語 SQL で数独を解く方法が提案されている [1]．このように参照すると，参考文献リストに自動的に登録される．文献の種類には，雑誌論文 [1] や会議録論文 [2]，卒業論文 [3]，書籍 [4]，ウェブサイト [5] などがある．文献の種類によって必要な項目が異なるため，biblio.bib を見て確認すること．

参考文献

- [1] 矢吹太郎, 佐久田博司. SQL による数独の解法とクエリオプティマイザの有効性. 日本データベース学会論文誌, Vol. 9, No. 2, pp. 13–18, 2010.
- [2] 矢吹太郎, 増永良文, 森田武史, 石田博之. 知識体系のエリア自動抽出のためのユニット分類手法. 第 5 回データ工学と情報マネジメントに関するフォーラム (DEIM2013). 電子情報通信学会データ工学研究専門委員会, 日本データベース学会, 情報処理学会データベースシステム研究会, 2013.
- [3] 久保孝樹. チケットを活用するオープンソースソフトウェア開発の実態調査. 卒業論文, 千葉工業大学, 2014.
- [4] 奥村晴彦, 黒木裕介. L^AT_EX2e 美文書作成入門. 技術評論社, 第 6 版, 2013.
- [5] 矢吹太郎. 自分のコードを出力するプログラム. <http://www.unfindable.net/article/self.html> (2012.12.01 閲覧).