SNS に書かれた疑問記事の自動的な収集・分類方法に関する研究

筑波大学

図書館情報メディア研究科

2018 年 3 月

荒井俊介

概要

SNS に書かれた疑問記事の自動的な収集・分類方法に関する研究

本論文では，SNS 上の疑問の書かれた記事 (以下，疑問記事) を自動的に収集・分類する手法を提案し，その有効性について検証する。近年，SNS を利用した個人による私的な書き込みが増加し続けている。その中でも Twitter のようなマイクロブログは，スマートフォンやタブレット PC の普及に伴い，ユーザを増やしている。マイクロブログのユーザは思いついたことやアイデアや疑問を気軽に投稿することができる。しかし，このようなマイクロブログに日常的な内容の疑問記事が書き込まれたとしても，疑問記事に対して回答できるユーザ (以下回答ユーザ) がその疑問記事を発見できなければ，疑問記事に回答が付くことはない。偶然回答ユーザが疑問記事を発見し回答が付く場合もあるが，回答ユーザが疑問記事を発見することは稀である。この問題を解決するためには，疑問記事が回答ユーザの目にとまり，疑問記事に回答が与えられる可能性を高めるための Web サイトを構築することが有効であると考えられる。そこで，その基盤となる手法として，疑問記事の自動収集と自動分類について検討を行った。

本論文ではまず，Web サイトの有用性を確認するために，以下の 3 点を確認する予備調査を行った (3.1 節)。即ち，(a) 疑問記事が Web 上に一定量存在すること，(b) 疑問記事に書かれた疑問はレファレンスの専門家でなくとも回答が可能であること，(c) ブログ著者は見知らぬ人からの回答でも好意的に受けとめること，の 3 点である。結果として，(a) 検索結果 800 件中 16 件の疑問記事が存在する事，(b) その中の回答のついていない疑問記事 13 件中 7 件で回答が得られたこと，(c) 疑問記事 7 件への回答のうち 5 件で回答に対して感謝されたこと，を確認した。

Web サイトを構築する際に，課題となる点が 2 つある。即ち，(1) 疑問記事の収集方法，(2) 収集した疑問記事の分類方法，の 2 つである。疑問記事は将来にわたって増え続けていくため，疑問記事を収集することも，疑問記事を分類することも，人手で行おうとするとその都度大きなコストがかかってしまう。これらの課題を解決するために，本論文ではそれぞれの課題について以下の手法を提案する。即ち，(1) 疑問記事の自動的な収集手法，(2) 収集した疑問記事の自動的な分類手法，である。

(1) に関しては，2 つのステップによって疑問記事を自動的に収集する手法を提案した。即ち，(1-1) 特徴的な表現による検索，(1-2) テキスト自動分類による抽出，である。(1-1) は疑問記事にのみ頻繁に現れる特徴的な表現を特定するステップである (3.2.1 項)。実験データとして，ブログと Twitter それぞれに関して疑問記事と非疑問記事の 2 種類を 100 件ずつ集めた計 400 件を用いた。実験の評価には検索結果上位 100 件中疑問記事が何件あるかを

表す疑問記事率を用いた。(1-2) は検索された特徴的な表現が現れる記事をテキスト自動分類によって疑問記事と非疑問記事に分類するステップである (3.2.2 項)。実験データとして，ブログに関して疑問記事と非疑問記事を 150 件ずつ集めた計 300 件を用い，Twitter に関して疑問記事と非疑問記事を 800 件ずつ集めた計 1600 件を用いた。実験の評価には疑問精度，非疑問精度，疑問再現率，非疑問再現率，疑問 F 値，非疑問 F 値を用いた。

(1-1) の実験の結果 (4.1.1 項), ブログにおいて疑問記事率が最も高かった特徴的な表現は「タイトル_が_思い出せ_ない」で疑問記事率は 19%であり，Twitter において疑問記事率が最も高かった特徴的な表現も「タイトル_が_思い出せ_ない」で，疑問記事率は 13%であることが明らかになった。(1-2) の実験の結果 (4.1.2 項)，ブログに関して最も精度，再現率，F 値が高かったのは，ブースティングの 1 形態素によってテキスト自動分類を行った場合であり，精度 94.8%，再現率 94.3%，F 値 0.943 であった。Twitter に関して最も精度，再現率，F 値が高かったのは，決定木の 1 形態素によってテキスト自動分類を行った場合であり，精度 94.3%，再現率 94.3%，F 値 0.941 であった。

(1-1)，(1-2) の結果から，ブログに対してブースティングの 1 形態素を用いた場合，その疑問再現率は 89.4% (＝疑問記事数 150 件のうち正しく判定された疑問記事数 134 件)，非疑問再現率は 99.3% (非疑問記事数 150 件のうち正しく判定された非疑問記事数 149 件) であった。そのため 17 件 (16.9 = 19×0.894) の疑問記事が出力される一方，非疑問記事は 0 件から 1 件 (0.567 = 81×(1－0.993)) しか出力されない。従って例えば検索だけでは 100 件に目を通さねばならなかったのが，テキスト自動分類によって 17 件の疑問記事と 0 件から 1 件の非疑問記事に目を通し，疑問記事を選び出すだけで済むようになる。同様の条件において Twitter に対して決定木の 1 形態素を用いた場合，その疑問再現率は 94.4%，非疑問再現率は 94.1%であった。そのため 12 件 (12.2 = 13×0.944) の疑問記事が出力される一方，非疑問記事は 5 件 (5.133 = 87×(1－0.941)) しか出力されない。ブログでも Twitter でも，疑問記事かどうかの判定にかかるコストの削減率は大きい。以上から，提案手法である特徴的な表現による検索とテキスト自動分類による抽出の 2 つのステップによって，疑問記事の収集にかかるコストを大幅に削減し，効率的に収集できることを明らかにした (4.1.2.4 項)。

ここで，こうして収集した疑問記事を利用して実際に疑問記事を提示する Web サイトを構築し，その有効性の確認を行った (4.1.3 項)。構築した Web サイトにはブログ記事 30 件，ツイート 31 件を掲載し，2010 年 12 月から 2011 年 2 月までの間回答を呼びかけた。その結果，構築した Web サイトには 1,251 人の利用者から 1,511 回のアクセスがあり，ブログに関しては 6 件に回答が付き，ツイートに関しては 5 件の回答が付いた。ブログ記事に関しては 2 件の回答に感謝のコメントが付き，ツイートには 5 件の回答に感謝のコメントが付いた。ここで，感謝のコメントが付いてないブログ記事はいずれも 1 ヶ月以上前に投稿された記事であった。この結果から，実際に Web サイトを通した疑問に対する回答はブログ著者や Twitter ユーザからも好意的に受け止められると考えられる。

次に (2) に関しては，レファレンス協同データベースのレファレンス事例 (以下事例) の参考資料と質問文を用いて疑問の書かれた文章に対して NDC (日本十進分類法) 記号を自動付与する手法を提案した。本研究では以下の 3 つの手法を用いて，事例に対する NDC 記号の自動付与を行った。即ち，(2-1) 参考資料を用いた手法，(2-2) 質問文を用いた手法，(2-3) 参考資料と質問文の両方を用いた手法，である。(2-1) は事例の参考資料の項目に記入されている図書の NDC 記号を利用して特徴ベクトルを作成し, NDC 記号の自動付与を行う手法である (3.3.1 項)。実験データとして，事例のうち NDC 記号と参考資料が両方付与されている 17,181 件を用い，実験の評価には精度，再現率を用いた。(2-2) は事例の質問の項目に記入されている文章を利用して特徴ベクトルを作成し，NDC 記号の自動付与を行う手法である (3.3.2 項)。実験データとして，事例のうち NDC 記号の付与されている 40,288 件を用い，実験の評価には精度，再現率を用いた。(2-3) は上記の参考資料の項目と質問の項目の両方から作成した特徴ベクトルを同時に利用して NDC 記号の自動付与を行う手法である (3.3.3 項)。実験データとして，事例のうち NDC 記号と参考資料が両方付与されている 17,181 件を用い，実験の評価には精度，再現率を用いた。

(2-1) の実験の結果 (4.2.1 項)，参考資料を用いた手法での NDC 記号の 1 桁目 (類) までの自動付与の精度は 53.4%であり, 再現率は 52.0%であった。(2-2) の実験の結果 (4.2.2 項)，質問文を用いた手法での NDC 記号の 1 桁目 (類) までの自動付与の精度は 54.2%であり，再現率は 54.5%であった。(2-3) の実験の結果 (4.2.3 項)，参考資料と質問文の両方を用いた手法での NDC 記号の 1 桁目 (類) までの自動付与の精度は 51.8%であり，再現率は 51.7%であった。

先行研究である原田ら (2007) の手法の精度と比較すると，参考資料を用いた手法では NDC 記号の自動付与精度を 8.5 ポイント向上させることができた。また，質問文を用いた手法では NDC 記号の自動付与精度を 10 ポイント向上させることができた。以上から以下の 2 つのことが明らかになった。即ち，(2-1) 事例の参考資料の項目に記入されている図書の NDC 記号を利用することで原田らの手法よりも高い精度で NDC 記号の自動付与を行えること (4.2.1.1 項)，(2-2) 特徴ベクトルの作成方法を工夫することで，事例の質問文の項目に記入されているような自然言語で書かれた文章を利用しても，参考資料を用いた場合と同程度の高い精度で NDC 記号の自動付与を行えること (4.2.2.1 項)，の 2 つである。

最後に上記の (1)，(2) の 2 つの提案手法を用いて，実際に疑問記事を自動的に収集し，収集した疑問記事に NDC 記号を自動付与する実験を行った (3.4 節)。実験データとして，「タイトル」が含まれるツイート 15,000 件を用いた。

この実験の結果 (4.3.1 項)，15,000 件の中から 118 件のツイートを疑問記事として自動抽出し，その中の 40 件が疑問記事であった。このツイートの疑問記事 40 件に対して NDC 記号を自動付与したところ，精度 71.2%，再現率 52.5%で疑問記事に対する NDC 記号の自動付与を行うことができた。以上から，(1)，(2) の実験データと提案手法を用いて，収集したツイートの疑問記事に対して事例の質問文を用いた場合よりも高い精度，同程度の再現率

で NDC 記号を自動付与できることが明らかになった。

本論文によって，以下のことが明らかとなった (5 章)。(1)「タイトルが思い出せない」というキーワードを用いて疑問記事の検索を行い，検索結果として得られた記事群に対して機械学習を用いることで，疑問記事を発見するコストを大幅に削減できる。(2) Web 上に投稿された疑問記事を事例の質問文と同質のものと考えるならば，疑問記事に対して NDC 記号を自動付与することが可能である。以上から，(3) 自動的な疑問記事の収集と疑問記事に対する NDC 記号の付与を高い精度で行える。

Abstract

Methods to Automatically Collect and Categorize Questions Posted on SNS

In this paper, we propose methods to automatically collect and categorize articles on SNS containing questions (hereinafter, "question articles") and examine their effectiveness. In recent years, the number of articles on SNS written by individuals has increased. Furthermore, the number of articles and users on Twitter is increasing with the widespread use of smartphones and tablet PCs. Twitter users can easily post questions using the platform. However, these questions are left unanswered if other users able to answer them (hereinafter, "answering users") cannot find the questions. Question articles are scattered all across the Web, and it is difficult for answering users to find them if they are not following the users who posted these articles. Within this context, constructing a website which collects, categorizes, and displays question articles will be an effective way for the question articles to be answered. Accordingly, we discuss methods to automatically collect and categorize question articles in this paper.

First, in order to confirm the usefulness of the previously mentioned website, we conducted preliminary research on the following three hypotheses (Section 3.1): (a) there is a certain number of question articles on the Web; (b) the questions in the question articles can be answered by non-experts; and (c) the users who posted the questions appreciate answers from answering users. As a result, we confirmed: (a) there were 16 question articles in 800 search results; (b) seven articles were answered out of the 13 question articles in the temporarily constructed website; and (c) among the seven question articles answered, five answers were appreciated by the users who posted the questions.

There are two problems to be solved when constructing the website: (1) how to collect question articles effectively, and (2) how to classify the collected articles correctly. Question articles will continue to increase and collecting and categorizing them manually is costly. In order to solve these problems, we propose the following two solutions: (1) automatic collection of question articles, and (2) automatic categorization of collected question articles.

Regarding solution (1), we proposed a method to automatically collect question articles by two steps; they are (1-1) finding candidate articles by using a search engine with specific keywords, and (1-2) extracting question articles from search results by text categorization. For (1-1), we identified specific words that appear frequently only in

question articles (Section 3.2.1). We used 100 question articles and 100 non-question articles found in blogs and tweets (400 articles in total). We focused on the rate of question articles in the top 100 pages, which the search engine returned (hereinafter, “question article rates”). For (1-2), we used 150 question and non-question articles (300 articles in total) from blogs and 800 question and non-question articles (1,600 articles in total) from tweets (Section 3.2.2). For the evaluation of the experiment, we used precision, recall, and F-measure.

In the experiment (1-1) (Section 4.1.1), the specific phrase with the highest question article rate from blogs was "タイトル_が_思い出せ_ない" (meaning "I cannot remember the title"). The question article rate for this phrase was 19%. Similarly, the specific phrase with the highest question article rate from Twitter was also "タイトル_が_思い出せ_ない," with a question article rate of 13%. In the experiment (1-2) (Section 4.1.2), the highest precision, recall, and F-measure from blogs were obtained when text categorization was performed by using one morpheme in boosting. The precision, recall, and F-measure were 94.8%, 94.3%, and 0.943, respectively. Similarly, the highest precision, recall, and F-measure from Twitter were obtained when text categorization was performed by using one morpheme in the decision tree. The precision, recall, and F-measure were 94.3%, 94.3%, and 0.941, respectively.

In the experiments (1-1) and (1-2), when we used one morpheme in boosting for blogs, we obtained 89.4% question recall (134 correctly extracted question articles/150 question articles) and 99.3% non-question recall (149 correctly extracted non-question articles/150 non-question articles). Therefore, if we apply our method to the top 100 articles from the search engine output, a total of 18 articles would be extracted and among them 17 (19 * 0.894 = 16.9) would be question articles and only 1 (81 * (1 - 0.993) = 0.6) would be a non-question article. In this case, we only have to check 18 articles instead of 100 and can find 17 question articles. Similarly, when we used one morpheme in the decision tree for tweets, we obtained 94.4% question recall and 94.1% non-question recall. Therefore, if we apply our method to the top 100 articles from the search engine output, a total of 17 articles would be extracted and among them 12 (13 * 0.944 = 12.2) would be question articles and 5 (87 * (1 - 0.941) = 5.1) would be non-question articles. In this case, we only have to check 17 articles instead of 100 and can find 12 question articles. Therefore, we can say that the cost of collecting question articles can be reduced by the two steps of our method, i.e., using a search engine with specific words, and extracting question article candidates by text categorization (Section 4.1.2.4).

We temporarily constructed a website that presents question articles and confirmed its

effectiveness (Section 4.1.3). We presented 30 blogs and 31 tweets on the website we constructed, and called for answers from December 2010 to February 2011. As a result, the website was accessed 1,511 times by 1,251 users. Users wrote 6 answers to the blogs and 5 answers to the tweets. Among them, 2 blog answers and 5 tweet answers were appreciated by the users who posted the questions. Four (6 - 2) blog users did not respond to the answer; however, all of them had already stopped posting for more than a month. Therefore, they might not have noticed the answer. From these results, it can be concluded that blog and Twitter users appreciate answers to their questions.

Regarding solution (2), we proposed a method of automatically assigning Nippon Decimal Classification codes (hereinafter, "NDCs") to question sentences in reference records of the Collaborative Reference Database, which is being maintained by the National Diet Library in Japan. Reference records are classified based on Nippon Decimal Classification (NDC), and accordingly, most of them have their own NDCs. We can use such pairs of records and NDCs as our training data for machine learning. In the present study, we examine the following three methods to automatically assign NDCs to reference records: (2-1) using reference materials (reference books, etc.) that were listed in the records, (2-2) using question sentences in the records, (2-3) and using both of them. Method (2-1) automatically assigns NDCs by creating feature vectors based on the NDCs of the reference materials in the records (Section 3.3.1). In the experiment, we used 17,181 records with NDCs and reference materials. Method (2-2) automatically assigns NDCs by creating feature vectors based on question sentences in the records (Section 3.3.2). We used 40,288 records with NDCs. Method (2-3) automatically assigns NDCs by creating a feature vector using reference materials and question sentences in the records (3.3.3). We used 17,181 records with NDCs and reference materials. For the evaluation of these three experiments, we used precision and recall.

In the experiment from (2-1) (Section 4.2.1), we found that the precision and recall were 53.4% and 52.0%, respectively, from assigning the first digit of the NDCs using the reference material. In the experiment from (2-2) (Section 4.2.2), the precision and recall were 54.2% and 54.5%, respectively. In the experiment from (2-3) (Section 4.2.3), the precision and recall were 51.8% and 51.7%, respectively.

We have implemented the method proposed by Harada et al. (2007) for assigning NDCs and found that its precision and recall for assigning NDCs to question sentences were 44.9% and 45.0%, respectively. By comparing these results with our previously mentioned results, it can be stated that (1) our method performs with higher performance than that of Harada et al. (2007) (Section 4.2.1.1) and (2) by devising

methods to create feature vectors, automatic assignment of NDCs can be performed with high accuracy (Section 4.2.2.1).

Finally, by using the two proposed methods (1) and (2) mentioned above, we automatically collected question articles and assigned NDCs to the collected question articles (Section 3.4). We used 15,000 tweets containing the keyword “タイトル.” We found that 118 out of 15,000 tweets were automatically extracted as potential question articles. In addition, 40 out of the 118 tweets were actually question articles (Section 4.3.1). The precision and recall to automatically assign NDCs to these articles were 71.2% and 52.5%, respectively. It was shown that by using our methods (1) and (2), NDCs can be assigned automatically with higher precision and similar recall to tweets than reference records.

This study clarified that (1) by searching with the specific phrase "タイトルが思い出せない" and text categorization using machine learning, the cost of finding question articles can be greatly reduced, (2) if we assume that the question articles and the question sentences of the reference record are of the same quality, it is possible to automatically assign NDCs to the question articles and (3) we can automatically collect question articles and assign NDCs to them with high precision. Therefore, we conclude that it is possible to construct a website we proposed (Section 5).

目次

1　はじめに

本研究では，疑問に対する回答を得るための支援を行う Web サイトを構築するための基盤的な手法として，SNS に書かれた疑問記事を自動的に収集する手法と，分類した上で人々に提示する手法を提案し，その実現性について検証を行った。本章ではこの研究の背景 (1.1 節) と目的 (1.2 節), そしてその目的を達成するために解決しなければならない課題 (1.3 節) について説明し，最後に本論文の構成 (1.4 節) について述べる。

1.1　研究の背景

近年，Social Networking Service (SNS) を利用した個人による私的な書き込みが増加し続けている。SNS とは，人と人とのコミュニケーションを促すように設計された Web サービスおよびそのサービスによって提供されている Web サイトを指す。その中でも Twitter のようなマイクロブログは，スマートフォンやタブレット PC の普及に伴い，ユーザを増やしている。Twitter Japan は 2016 年 2 月 18 日に行われた記者説明会 [1]において，日本における 2015 年 12 月時点での月間アクティブユーザ数は 3,500 万人であると発表している。Twitter Japan ができた 2011 年 3 月時点の 670 万人と比べると約 5.2 倍に増加している。同記者説明会において，アクティブユーザの増加率を国ごとに比較すると，2015 年 12 月時点でのアクティブユーザの増加率は世界の中で日本が最も高いことも示されている。Twitter.Inc は，Twitter のミッションを「言語や文化などの障壁をなくして，思いついたアイデアや見つけた情報を一瞬にして共有する力をすべての人に提供すること」としている [2]。

Twitter のユーザは，思いついたことやアイデアや情報を気軽にツイートすることができる。それらのツイートの中には，日常的な疑問が書かれるものもある。例えば「〜ってなんだっけ」,「〜のことが思い出せない」,「〜について知りたい」といった形の疑問である。このような疑問をツイートするユーザは，それへの回答がつくことを多少は期待していると考えられる。しかし，上で述べたように Twitter ユーザは増加しており，それに伴いツイート数も日々増加を続けている。仮にその疑問に答えることのできる人がいて答える気が

[1] http://www.huffingtonpost.jp/2016/02/18/twitter-japan_n_9260630.html (accessed 2016-6-16)
[2] https://about.twitter.com/ja/company (accessed 2016-6-16)

あったとしても，フォロー関係がなければそのツイートを見つけることはほぼ不可能である。このため疑問の書かれたツイートには回答がつかないまま，他のツイートの中に埋もれてしまうことが多い。このことはツイート以外にも，ブログ記事などの Web 上で行われる書き込み全般に言える。本研究ではコメントやリプライのような形で書き込みに返信できる Web サイトを SNS と呼ぶ。本研究では手始めにブログと Twitter への書き込みを扱い，これらの Web サイトに書き込まれた疑問を疑問記事と呼ぶ。また，疑問記事や後述するレファレンス事例の質問文などの，何かに関する疑問を質問の形で表現した文を総称して疑問文と呼ぶ。

1.2 研究の目的

本研究は疑問記事を自動的に収集・分類するための手法を提案し，その有効性を検証することを目的とする。この提案手法を利用して，発見が難しい疑問記事を提示する Web サイトを構築することで，疑問記事がその疑問に回答できるユーザ (以下回答ユーザ) の目にとまり，回答が与えられるようになることを目指す。

こうした Web サイトの最も重要な意義としては，SNS ユーザの疑問を解消できることが挙げられる。このような Web サイトを構築することによって，回答がつかなかった疑問にも回答がつく可能性が出てくる。加えて，人々の新たな結びつきや人的交流を促進するという意義もある。疑問記事を投稿したユーザと回答ユーザとはフォロー関係になくても同じ趣味・嗜好を持っている可能性があり，回答を通じて新たな交流が生まれるかもしれない。SNS では人と人をつなぐフォロー関係等を通じて，もともと人的交流を促す仕掛けが組み入れられているが，このような Web サイトはそれを更に強化するものと言える。

さらに，本研究では疑問記事を提示する Web サイトの回答ユーザには広く一般人を想定しているが，図書館員がこの Web サイトを通じ自分達の業務として回答するようになった場合，それはレファレンスサービスのアウトリーチと呼ぶこともできる。スマートフォンやタブレットパソコンから気軽にツイートされる疑問の中には，図書館員が調べれば簡単に分かる内容のものもある。図書館から離れた場所で，疑問を持ったユーザにとって調べることが難しい内容であっても，図書館にいて資料に熟知した図書館員であれば，データベースや参考図書を利用して簡単に調べることができるかもしれない。もし，図書館員が

この Web サイトを通じてブログ記事やツイートに回答を書くことによって図書館の存在を PR すれば，レファレンスサービスに関する新たな広報活動と位置付けることもできる。この Web サイトには，そのような活動を可能あるいは効率的にする意義もある。

以上のような意義があるにもかかわらず，こうした Web サイトを構築するための基盤的な手法は研究されてこなかった。

1.3 研究の課題

疑問記事を収集し分類した上で人々に提示する Web サイトを構築する際に，課題となる点が 2 つある。即ち，(1) 疑問記事の収集方法，(2) 収集した疑問記事の分類方法，の 2 つである。これらの課題を解決するために，本研究ではそれぞれの課題について以下の手法を提案する。即ち，(1) 疑問記事の自動的な収集手法，(2) 収集した疑問記事の自動的な分類手法，である。以下でそれぞれについて具体的に述べる。

まず (1) であるが，検索可能な記事は膨大な数にのぼり，疑問記事を人手で収集するのは非常にコストがかかる。そのため，検索結果に対して何らかのフィルタリングを行い，疑問記事を効率よく集めることが望まれる。そのような課題を解決する手法の一つにテキスト自動分類がある。テキスト自動分類はスパムメールの自動判定などに応用されている。テキスト自動分類では，機械学習手法を用いてあらかじめラベルが付与されている学習用コーパスから分類器を作成する。この分類器を用いることで，ラベルの付与されていない文書にどのラベルを付与すべきかを自動的に判別できる。本研究で提案する方法は，既存の機械学習手法を用いているという点では自動分類に関する先行研究と同じであるが，疑問記事を分類・抽出対象にしている点が新しい。分類対象ごとに重要な特徴量は変わるため，対象が新しくなれば，用いるのに適切な特徴量も既存の研究とは異なる。疑問記事とその他の記事 (以下非疑問記事) を判別する問題に対して有効な特徴量はまだ十分に研究されていない。その意味で本研究には十分な新規性・有用性があると考える。そこで提案手法 (1) では，機械学習を用いて疑問記事と非疑問記事を判別する分類器を作成し，疑問・非疑問のラベルが付与されていない文書に自動的にラベルを付与することで疑問記事の収集を行う。本研究では，疑問記事として図書のタイトルに関する疑問の書かれた記事を扱う。

次に (2) であるが，疑問記事は膨大な数にのぼるため，これらを Web サイトに無秩序に並べるだけでは，回答ユーザが興味のある疑問記事にたどり着くことは容易ではない。また回答ユーザがわざわざ自分が得意とするキーワードを入力して検索してくれるとも思えない。回答ユーザが興味のある疑問記事を見つけることができなければ，疑問に回答がつく可能性が低くなってしまう。このような問題を回避するために，回答ユーザが興味のある疑問記事を見つけやすいよう工夫して提示する必要がある。そのための方法の 1 つに，レファレンス協同データベースで用いられているようなディレクトリ型検索がある。レファレンス協同データベースは，2005 年 4 月から国立国会図書館によって始められたサービスで，個々の図書館で行われるレファレンスサービスに関する知見を広く共有するために，レファレンス事例 (以下，事例) を収集・提供するものである。2017 年 5 月時点で，このレファレンス協同データベース事業には公共図書館，大学図書館などを合わせて 745 の図書館が参加しており，98,000 件を超える事例が一般に公開されている。レファレンス協同データベースには一般的なキーワード検索の他に，「テーマから探す」という検索方法が用意されている。「テーマから探す」では，明確なキーワードが思い浮かばないときなどに，日本十進分類法 (Nippon Decimal Classification; NDC) に基づいた分類を選択していくことで，特定の主題に関係する事例を見つけることができる (このような検索を以下ではディレクトリ型検索と呼ぶ) 。「テーマから探す」においては，階層関係にある分類を選択していく中で，類似した分類に属する事例を多く目にすることになる。その中に，データベースの利用者にとって興味のある内容の事例が含まれているかもしれない。キーワード検索で類似した分類に属する事例を発見するためには，キーワードを変えて複数回の検索を行わなければならない。しかし探索上の明確なニーズを持たない回答ユーザがこのような複雑な検索を行うことは，先述のように考えにくい。以上から，ディレクトリ型検索はキーワード検索に比べて，(a) キーワードが思いつかなくても検索できる，(b) 階層関係を辿っている途中に興味のある内容を発見できる可能性がある, という 2 つの利点があると思われる。そこで提案手法 (2) では, Web サイトに NDC に基づくディレクトリ型検索を実装するために，疑問記事に対して NDC の分類記号 (以下，NDC 記号) の自動付与を行う。疑問記事の分類に NDC 記号を用いたのは，本研究で扱う疑問記事が主に図書のタイトルに関するものだからである。

1.4　本論文の構成

以下に本論文の構成を述べる。第 2 章では，先行研究を概観しそれらの研究と本研究の関連および位置付けを述べる。第 3 章では，SNS に書かれた疑問記事を自動的に収集し，見つけやすい形で回答ユーザに疑問記事を提示する Web サイトを構築するための提案手法について詳説する。まず，疑問記事を提示する Web サイトの有用性を確認するための予備調査行い，その結果について述べる。次に，Web サイトの構築にあたって解決すべき課題を解決するために行う，2 つの実験方法について説明する。即ち，(1) 疑問記事の自動的な収集実験 (3.2 節)，(2) 疑問文に対する NDC 記号の自動付与実験 (3.3 節)，である。最後にこれら 2 つの実験から得られた知見をもとに，疑問記事に対する NDC 記号の自動付与実験 (3.4 節) を行うための手順を説明する。第 4 章では，以上の 3 つの実験の結果と考察を述べる (4.1 節，4.2 節，4.3 節)。図 1 に本研究における課題と提案手法の関係を，本論文における構成と同時に示した。最後に第 5 章では本研究によって得られた結論と今後の課題について述べる。

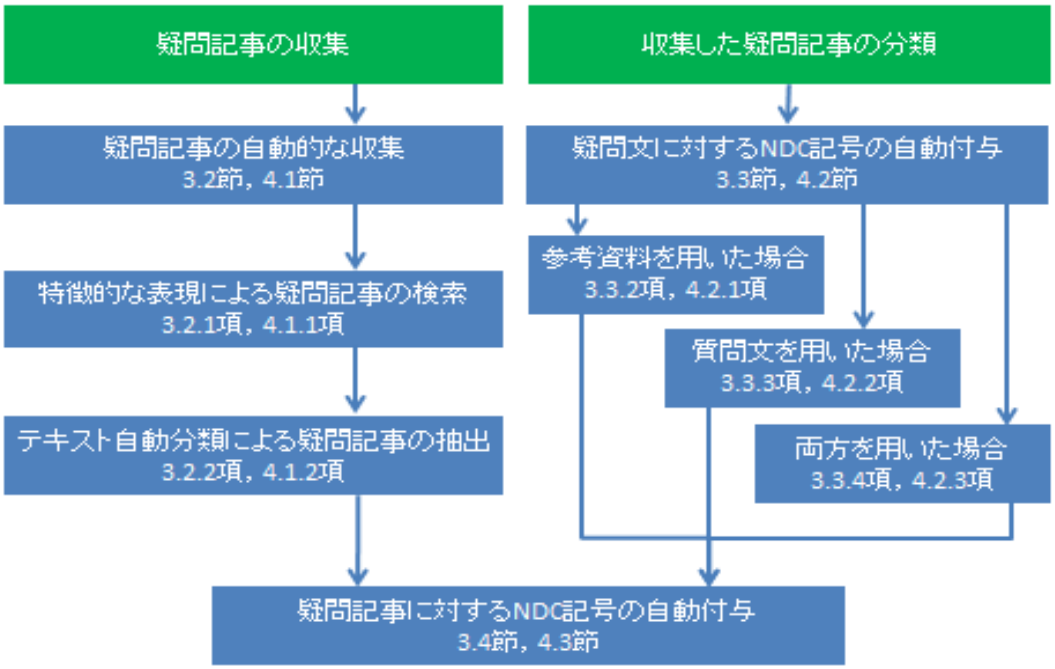

図 1　本研究における課題と課題を解決するための提案手法

2　先行研究

本研究では質問の形で書かれた疑問文として，SNS に投稿された疑問記事とレファレンス協同データベースに収録されている事例の質問文を用いる。このような言葉に対する分析手法としてテキストマイニングがある。疑問記事を提示する Web サイトは，単純な方法では得る事の出来ない知見を得るためのテキストマイニング手法と言うことができる。この Web サイトが持つ機能はレファレンスサービスのアウトリーチとも呼ぶことができ，これはデジタルレファレンスサービスの一種と考える事もできる。本研究では疑問記事の自動収集，自動分類において機械学習手法によるテキスト自動分類を利用する。本章では本研究に関連する先行研究について，以下の 3 つに大別し概観する。即ち，(1) テキストマイニングに関する研究 (2.1 節), (2) デジタルレファレンスサービスに関する研究 (2.2 節), (3) 機械学習に関する研究 (2.3 節)，の 3 つである。

2.1　テキストマイニングに関する研究

テキストマイニングとは人手では処理できないほどの大量の文書に対して，自然言語処理や機械学習などの手法を利用して語や句などの出現頻度や相関関係といった指標を算出し，そこから新たな知識を発見するための方法である。本節ではまず，テキストマイニングの概要や活用例に関する研究について述べる (2.1.1 項)。次に，本研究で扱う日本語で書かれたものを主としてブログ, Twitter, その他の Web テキストの 3 つのそれぞれについて，これまでに行われたテキストマイニングに関する研究を概観する (2.1.2 項)。最後に，テキストマイニング技術を図書館情報学分野で応用した研究について述べる (2.1.3 項)。

2.1.1　テキストマイニングの概要

那須川ら (1999) によれば, テキストマイニングの活用は 1999 年以前から行われていた。その活用例として，日本 IBM に寄せられた問い合わせの文書に対してテキストマイニングを行った例を紹介している。その結果として，よく寄せられる質問とそれに対する回答をホームページ上で公開することで問い合わせ件数を減らし，問い合わせがあってもすぐに答えられるようにすることが可能になった，としている。富田 (2004) は有効な意思決定を

迅速に行う為には，大量のテキスト情報を集計・集約してユーザに提示するテキスト集約技術が必要であるとしている。文書は統計的な数値データに比べて，コンピュータで処理することが難しいため，高度な利用があまり進んでいない。このような状況の中で重要なのは，コンピュータによる完全な言語の理解ではなく，意思決定を支援する形で大量の文書を集約する言語処理という観点である，としている。那須川 (2009) は大きなインパクトにつながっているテキストマイニングの活用成功例が少ないという問題意識から，テキストマイニングの本質と可能性に対する誤解と，普及の難しさについて述べている。その中でテキストマイニングを，「複数のテキストを分析する事で初めて得られる知見を得る事を目的とした技術」として定義している。テキストマイニングは得られる知識の候補を与えてくれるものであり，それが有益な知識であるかどうかはユーザが判断する必要がある。即ち，テキストマイニングはあくまでの知識発見を支援するものでしかなく，システムが全て自動で知識を発見してきてくれるわけではない。

近年，文書に含まれる単語の共起頻度を利用して隠れた構造を見つける手法としてトピックモデルが注目されている。トピックモデルを用いたテキストマイニングについて解説を行ったものには岩田 (2014) がある。実際にトピックモデルを用いてテキストマイニングを行った研究として，佐々木ら (2013)，山田ら (2013)，単ら (2014)，白井ら (2014)，清水ら (2016) などがある。

2.1.2 ブログ・Twitter・その他の Web テキストに関するテキストマイニング

ブログや Twitter などに書き込まれる投稿は，インターネットの普及とともに急速に増加してきた。これらの投稿から有用な知見を得るために，これまでにブログや Twitter を対象とする，様々なテキストマイニング研究が行われている。以下ではそれらの研究について概観する。

2.1.2.1 ブログに関するテキストマイニング

ブログに対するテキストマイニングの動向をまとめたものに奥村 (2008) がある。奥村はブログがテキストマイニングにおいて注目を集める理由として，一般の多くの人びとから発信された情報を蓄積しており，そこから情報を抽出することで，一般の人びとの速報性，リアルタイム性のある新鮮な生の声を得ることができる点を挙げている。

ブログを対象としたテキストマイニングの研究として，奥村ら (2004)，関口ら (2005)，平野ら (2005)，数原ら (2006)，古瀬ら (2006) がある。奥村らはブログを定期的に監視し，そこから自動で情報を抽出するシステム blogWatcher の開発を行っている。blogWatcher ではブログ記事の横断検索を行うことができ，評判情報検索やおすすめブログの推薦などの機能もある。関口らはブログ記事の省略を多く含む口語的な記述を考慮して，ブログ記事に対する話題語句抽出を行っている。ブログ著者の興味分野を抽出し，同様の興味を持つ人々の間で共通して用いられる語句を，話題語句として抽出している。平野らは複数のポータルサイトの分類体系を参考に，独自の分類体系を作成し，日本語圏のブログ記事の自動分類を行っている。数原らは述語とその述語に係る格助詞を伴う名詞の組み合わせに着目して，ブログ記事に現れる固有名詞と関連する固有名詞の間の動作関係を抽出する手法を提案している。このとき，名詞はしばしば指示語に置換される，あるいは省略されてしまう。この問題を解決するために，大量のブログ記事を解析し情報の補完を行っている。古瀬らはブログ記事を対象とする意見文検索システムを開発し，その評価を行っている。このシステムでは，あらかじめブログ記事を収集し意見文か否かの分類を行っておき，意見文として分類されたブログ記事の中からユーザの入力したクエリと類似しているブログ記事を提示する。

ブログ記事には口語的な特有の表現が含まれる場合が多い。そのために，特別な修正ルールが必要になる場合がある。池田ら (2009) はブログ的表現を文語的な表記に修正するためのルールを提案している。ブログ記事の収集方法について概説したものとしては石川ら (2012) がある。

2.1.2.2 Twitter に関するテキストマイニング

Twitter を対象としたテキストマイニングの研究として，安藤ら (2014)，久米ら (2015)，森國ら (2015)，阿部 (2016) などがある。安藤らは Twitter や動画のコメントなどから有益な意見を抽出し，これらの意見をテレビ番組や Web 放送に活かすために意見を分類する手法を提案している。しかしこれらの文書に対して，テキストマイニングをそのまま用いるのは，以下の 2 つの理由で難しいとしている。即ち，(1) 短い文章や誤った文章が多い，(2) 未知語が用いられることが多い，の 2 つである。安藤らはこのような問題を，ユーザに選択してもらった意見の対象，意見の種類，未知の名詞の出現頻度，の 3 つを特徴量として

用いることで解決している。久米らは興味領域を考慮した Twitter フォローユーザの推薦手法を提案している。Twitter には 140 字以内という投稿規則があるため，単語が省略される場合が多い。そのため同じ意味をもつ単語の表現方法が統一されていない。結果として，テキストマイニングを行う際に特徴を示す単語とノイズである単語が差別化されず，精度が低下してしまう。久米らはこの問題を，各単語を単語ごとではなく，分類ごとに管理することで解決している。森國らは位置情報の付与されていないツイートの投稿位置を，ツイートに含まれる地域ごとの単語の出現頻度を用いて推定する手法を提案している。精度を向上させるために，ノイズとなる単語を除去し，単語の出現頻度が高い地域の，周辺の地域における単語の出現頻度を高くするスムージングを行っている。阿部は Twitter のハッシュタグを利用して，特定の話題の中に現れるドイツ語の単語を収集する方法を提案している。

以上から，ブログや Twitter などの気軽に投稿できるマイクロブログの記事を対象としてテキストマイニングを行う場合，記事中に現れる口語的な表現が問題になることが多いことが分かる。本研究においては第 3 章においてブログ記事とツイートを対象としたテキストマイニングを行っている。本研究ではこの問題に対して，第 3 章ではコーパスにブログ記事とツイートそれ自身を用いることで対処している。

2.1.2.3 その他の Web テキストに関するテキストマイニング

その他の Web テキストを対象としたテキストマイニングの研究として，板倉ら (2004)，鍛冶ら(2009)，疋田ら (2012)，杉本ら (2015) ，が挙げられる。板倉らはアドホックな閾値を必要としないラフ集合理論を利用して，Web ページの Yahoo!JAPAN トップカテゴリへの自動分類を行っている。鍛冶らは片仮名用言を，Web テキストから自動獲得する手法を提案している。Web テキストに対する形態素解析の精度低下の一因が，「ググる」などの片仮名用言であることに着目し，これらの片仮名用言を自動的に取得できれば，くだけた表現の多い Web テキストの解析に対する頑健性が向上する，としている。鍛冶らは片仮名用言の自動獲得を，大量の Web テキストから片仮名列だけを抽出し，各片仮名列が動詞の語幹になるか否か，または形容詞の語幹になるか否か，を判定する分類器にかけることで行っている。ここで語幹であると判定された場合，その片仮名列を語幹とする動詞，または形容詞を獲得している。疋田らは組織研究のためにテキストマイニングを用いている。東証 1

部上場企業 500 社，三重県の中小企業 198 社，米国のビジョナリーカンパニー18 社のホームページから経営理念，社是，行動方針，行動規範を収集し，これらを用いて各企業の間の差異の検出，各企業の分類を行っている。杉本らは口コミ情報が書き込まれる価格.com, coneco.net，トリップアドバイザーへの書き込みから，感情を表す語を抽出する手法を提案している。感情語の抽出は辞書を用いたマッチングによって行っている。

特に質問回答に関連の深い Web テキストのテキストマイニング研究として，藤井ら (2007)，吉田ら (2008) ，Choi et al. (2013)，が挙げられる。藤井らは Web 検索において，検索質問によって必要とされる検索モデルが異なることから，検索に用いられた質問文から質問のタイプを自動的に分類したうえで，質問のタイプによって検索手法を動的に変化する手法を提案している。質問の自動分類のための尺度として，検索質問に含まれる文字列と同じアンカーテキストからリンクされているページの出現分布の歪度の高低を用いている。吉田らは Q&A サイトの口コミ情報から，関連する単語同士を結んだネットワークを抽出する手法を提案している。あらかじめ単語とその単語の属性を含む用語辞書を作成し，質問文とそれに対応する回答文に現れる，同じ属性を持つ単語同士を関連語として抽出している。こうして得られた関連語の中から，共起頻度，Jaccard 係数，Simpson 係数の高かったものを関連語として扱っている。Choi et al. はテキストマイニング技術を用いて，Q&A サイト「Yahoo! Answers」において回答が付きにくい質問が，どのような特徴を持っているかを調査している。実験の結果，回答が付きにくい質問は，以下の 3 つの特徴によって，大きく影響されることを明らかにしている。3 つの特徴とは即ち，(1) 質問がどんな言葉で始まるか，(2) 質問の中で用いられている語彙数，(3) 質問の複雑さ，の 3 つである。このようにして，質問に回答が付かない確率を予測できれば，情報を求める Q&A サイト利用者による質問の構成を手助けし，回答が付く可能性が高まるという点で有用である，としている。

2.1.3 テキストマイニングの図書館情報学分野での応用

テキストマイニングの技術を図書館情報学分野で応用した研究として，Moony et al. (2000)，Chang et al. (2006)，が挙げられる。Moony et al. は，利用者の興味に基づいて図書や他の情報源を推薦することは電子図書館にとって重要なサービスとなることを指摘し，テキスト自動分類によるコンテンツベースの方法で図書推薦を行うシステムを提案している。

その中で，テキスト自動分類を用いたコンテンツベースの方法による推薦システムは，推薦システムによく用いられる協調フィルタリングとは異なり，ユニークな嗜好を持つユーザへの質の高い推薦や，未評価の図書の効果的な推薦が可能であるとしている。Chang et al. はデータマイニング技術を利用し，図書館利用者の過去の貸出履歴から利用者を，いくつかのクラスタに分類する実験を行っている。その結果，図書館利用者は在学生や卒業生，研究者など異なる貸出履歴の特徴を持つ，5 つのクラスタにクラスタリングできることを明らかにしている。このデータマイニングの結果から，あらかじめ利用者のニーズを把握しておくことができるようになるとしている。

特にデジタルレファレンスサービスに関連の深い研究として，Nicholson et al. (2007), Kucukyilmaz et al. (2008)，Yu et al. (2012)，が挙げられる。Nicholson et al. はデジタルレファレンスサービスにおける処理の記録が，検索可能な形で残されていない点を問題として挙げ，検索可能なナレッジベースを作成するためのスキーマの必要性について論述している。この中で，もしデジタルレファレンスサービスに関する整備されたビッグデータが入手可能になれば，ビブリオマイニング (図書館に関するデータマイニング) はその研究の結果から，デジタルレファレンスサービスの新しい形の評価ツールの作成や，レファレンス処理の中の新規性があり実行可能なパターンの抽出を可能とするだろう，と述べている。しかし Nicholson et al. はそれと同時に，レファレンスサービスはその内容が自由記述であるため，個人情報が入り込んでしまう危険性がある，としている。そのために，利用する際にはその中に個人情報が含まれていないことに注意する必要があることも指摘している。Kucukyilmaz et al. はチャットメッセージから，そのチャットを行っているユーザとメッセージの属性を推定する実験を行っている。実験データとして，Heaven BBS で行われた 218,742 件のチャットメッセージを用いている。実験の結果，テキスト自動分類を利用しチャットユーザの言葉の選択と文体の好みを用いることで，99.7%の精度でユーザの属性を推定できる，としている。同時に語彙の選択や文体の好みには，チャットを行った時間やメッセージの受信者によって大きく影響を受けることも指摘している。Yu et al. は従来の情報探索とは異なる，コンピュータを通した情報探索の特性の検証のために，デジタルレファレンスサービスの質問回答の中にどのようなプロセスが含まれており，何がそのプロセスに寄与しているかを調査している。加えて，機械学習を用いたテキスト自動分類による質問回答の特性の抽出と，質問回答の自動的な最適化の可能性について検証している。実験

データとして，OCLC (Online Computer Library Center) の提供するデジタルレファレンスサービスの対話記録から 211 個を選択し用いている。実験の結果，Yu et al. は機械学習を用いることで，質問回答の対話の中から特性を抽出し，質問回答の概念モデルを提供できることを明らかにしている。これにより，デジタルレファレンスサービスの記録に基づく，完全に自動的な質問回答システムを構築するためのプロトタイプモデルを示すことができた，としている。

2.2 デジタルレファレンスサービスに関する研究

本節ではデジタルレファレンスサービスに関する研究について概観する (2.2.1 項)。その後に，本研究で用いる実験データであり，一般に公開されておりレファレンスサービスのアウトリーチとしての一面も持つレファレンス協同データベースに関する研究について概観する (2.2.2 項)。

2.2.1 デジタルレファレンスサービス

デジタルレファレンスサービスに関する研究には，Ferguson et al. (1997)，Kwon (2007)，Shachaf (2008)，が挙げられる。Ferguson et al. は，図書館はネットワーク環境の整備や技術の統合，他部署との連携から生まれる新たな図書館の価値を模索しながら，個々の利用者のニーズに沿ったサービスと，全員が公平にアクセスできるサービスを保持しなければならない，としている。そして，予算と人員に制限があり利用者のニーズの量と複雑さが増加する状況の中で，レファレンスデスクが質の高いサービスに繋がらずむしろ障害になる，ということに図書館職員が気付き始めたと指摘している。Ferguson et al. はこのような問題の解決策の一つとして，レファレンスデスクの外に出て仕事をし，支援を提供する必要があるときにだけデスクの中で仕事をする，専門の図書館職員を作ることを挙げている。加えて，レファレンスサービスを遠隔地へ提供することが比較的容易になったことを挙げ，電話や e-mail だけでなく，ネットワークを通したリアルタイムで行うレファレンスサービスの可能性を指摘している。Kwon は協同チャットレファレンスサービスにおける質問の型ごとの，質問回答の有効性について検討している。実験の結果，主題に基づく調査質問と単純な事実に関する質問の間の質問回答の有効性にはほとんど差がないこと，ただし地域

固有の質問に関しては地域固有でない質問と比べ，利用者の満足度が下がることを明らかにしている。この結果から，協同チャットレファレンスサービスはシームレスなサービスとして設計するとよいことが示唆された，としている。Shachaf は e-mail を用いたデジタルレファレンスサービスの IFLA (International Federation of Library Associations) と RUSA (Reference and User Services Association) によって公表された 2 組のガイドラインへの順守レベルを比較評価している。実験データとして 54 の図書館で行われたデジタルレファレンスサービスの記録 324 件が用いられている。その結果として，以下の 4 つのことを明らかにしている。4 つとは即ち，(1) 全体的にガイドラインへの準拠レベルが低い，(2) どちらのガイドラインでも質問の要求レベルと利用者の名前によって準拠レベルが変動する，(3) 評価のためのガイドラインを変えることで図書館のガイドラインへの準拠度合のラインキングが変動する，(4) いずれかのガイドラインへの準拠レベルと利用者の満足度の間には相関関係がない，の 4 つである。

2.2.2 レファレンス協同データベース

レファレンス協同データベースは，レファレンスサービスに関連した 4 種類のデータを提供している。即ち，「事例」，「調べ方マニュアル」，「特別コレクション」，「参加館プロファイル」の 4 つである。本研究で用いるのはこの中の「事例」である。事例は，定められたフォーマット (表 1) に沿って記入される。記入が必須とされている項目は，質問，公開レベル，管理番号，回答，の 4 つであり，それ以外の項目は任意となっている。

レファレンス協同データベースの概要や取組みについて説明したものとして，依田 (2006a)，依田 (2006b)，堤ら (2011)，依田 (2013) がある。依田 (2006a) はレファレンス協同データベースの利用法について紹介し，特に薬学図書館に対して登録してほしい事例のタイプについて考えを示している。そして薬学図書館に対しては，以下の 3 つのタイプの事例の提供を期待している。即ち，(1) 専門的なもの，(2) 情報リテラシー教材となるもの，(3) 基礎的な知識が記載されたもの，の 3 つである。依田 (2006b) は，レファレンス協同データベースの概要と事業への参加に伴うレファレンスサービスの変化について述べ，レファレンス協同データベース事業を，「全国の図書館で行われているレファレンスサービスの記録や，そこで蓄積された調べ方に関する情報などをデータベース化し，図書館におけるレファレンス業務や，一般の人々の情報検索に役立てる事を目的とする協同事業である」

表 1　レファレンス事例記入フォーマット

No	項目名	記入区分
1	質問	必須
2	公開レベル	必須
3	管理番号	必須
4	回答	必須
5	事例作成日	任意
6	解決／未解決	任意
7	キーワード	任意
8	NDC の版	任意
9	NDC	任意
10	調査種別	任意
11	内容種別	任意
12	参考資料	任意
13	回答プロセス	任意
14	照会先	任意
15	事例調査事項	任意
16	備考	任意
17	質問者区分	任意
18	寄与者	任意
19	関連画像	任意
20	登録番号	自動付与
21	登録日時	自動付与
22	最終更新日時	自動付与
23	提供館コード	自動付与

としている。事例の中核的な事項として，「質問」「回答」「事前調査事項」「回答プロセス」「参考資料」「照会先」「寄与者」の 7 つを挙げ，レファレンス協同データベースの利用者として図書館員，一般利用者，図書館学研究者，の 3 つのグループを想定している。依田 (2006b) はこの事業に参加する事で参加館に以下のような 4 つの変化があるとしている。即ち，(1) レファレンス情報源として利用できるようになる，(2) 図書館サービスの改善に利用できるようになる，(3) 事例を登録する分の業務が増える，(4) 参加館同士のネットワー

ク構築の支援機能を利用できるようになる，の 4 つである。堤らは，レファレンス協同データベースの登場によって，レファレンスデータベースの状況にどのような変化がおきたかを考察している。レファレンス協同データベースが現れる以前は，レファレンス業務のデータベース化に取り組んでいた図書館は少なく，事例は記録されない，あるいは紙媒体で記録されていたため，検索が困難な場合が多かったことを明らかにしている。堤らは考察の中でレファレンス協同データベースができたことによる成果として，以下の 5 つを挙げている。即ち，(1) 事例の標準的な記録のフォーマットの提供，(2) 事例の共有の有用性の実証，(3) 図書館サービスの改善のためのデータの提供，(4) レファレンスサービスの一般利用者への広報，(5) 新たな図書館評価指標としての利用，の 5 つである。依田 (2013) はレファレンス協同データベースの実践コミュニティとしての側面について考察している。レファレンス協同データベースの実践コミュニティとしての可能性について，以下の 2 つを挙げている。即ち，(1) コミュニティの活動領域の拡大，(2) コミュニティ形成の経験の図書館外部への応用，である。

レファレンスサービスの現状と課題について分析したものとして，川瀬ら (2012)，谷本ら (2012)，谷本ら (2013) がある。川瀬らは公共図書館を中心として，レファレンス協同データベースについての考察と，改善点の検討を行っている。レファレンス協同データベースの問題点として，全事例のうちの約 30%である 21,403 件が一般公開されていないことを指摘し，この理由として以下の 3 つを挙げている。即ち，(1) レファレンス協同データベースへの登録の困難さ，(2) その事例を一般利用者に閲覧されたくない，(3) その事例が未解決である，の 3 つである。加えて，川瀬らは参加館ごとの事例の登録件数に大きなばらつきがある本質的な理由として，事例を登録するか否かの基準が大きく異なる点を指摘している。このことから，一般利用者によるレファレンス協同データベースの利用を促すような取り組みを行っていくことを，一つの方向として示している。このとき重要なポイントとして以下の 2 点を挙げている。即ち，(1) 一般利用者を対象としたレファレンス協同データベースの広報，(2) レファレンス協同データベースの教育の場での活用，である。谷本ら (2012) はレファレンス協同データベース事業の取り組みに関する分析を行い，事業の進むべき方向について考察している。分析の結果，事業における取り組みは事例の生産という点が重視されており，事例の利用についての取り組みが少ない点を指摘している。谷本ら (2013) はレファレンス協同データベース事業における，事例の登録の問題点についてアン

ケート調査を行い，その結果に基づいて分析と考察を行っている。分析の結果，事例はあるが大多数の参加館はそれらを登録していない点を指摘し，事例が登録されない原因として以下の 3 つを挙げている。即ち，(1) 登録作業が業務上の負担となっている，(2) 登録に対する業務上の位置付けが曖昧になっている，(3) 参加館の担当者に登録すべき事例を選択しようとする意識がある，の 3 つである。

レファレンス協同データベースを図書館業務の中で活用した例として，宮川 (2007)，寺尾 (2008) がある。宮川は公共図書館職員の立場からレファレンス協同データベース事業について解説し，多くの館の事業への参加を促している。未解決の事例を公開することによって，他の参加館である専門図書館から所蔵資料に記載があると連絡を受けて解決することができ，これまでに照会をかけたことがなく協力の乏しかった図書館との協力のきっかけとなった，としている。寺尾は人にわかりやすく伝えるように事例の記録をまとめることが良い研修となり，レファレンス協同データベース事業への参加は図書館職員に良い影響をもたらすとしている。レファレンス協同データベースのコメント機能を使って他館との情報のやりとりを行うことで，日本の図書館界においても「レファレンスの相互協力」を行う段階に入ったことを実感した，としている。

レファレンス協同データベースを利用した研究として，原田ら (2007)，伊藤ら (2008)，間部ら (2011)，三津石ら (2012)，吉田 (2015)，などがある。原田らはレファレンス協同データベースのまだ NDC 記号が付与されていない事例に対して Support Vector Machine (以下 SVM) を用いて NDC 記号の類 (1 桁目) までの自動付与実験を行っている。原田らの研究については 3.2.2.1 節の中で詳述する。伊藤らは日本の公共図書館および大学図書館における，Web 上で公開されているパスファインダーの現状を各機関のホームページの調査，アンケート調査，パスファインダーの集積サイトの調査によって分析している。パスファインダーの集積サイトの一つとして，レファレンス協同データベースの調べ方マニュアルを挙げている。間部らはレファレンスサービスにおいて用いられる参考図書について，以下の 3 つを明らかにするために，レファレンス協同データベースの分析を行っている。(1) 調査によく使用される参考図書，(2) 回答を可能とした参考図書，(3) 主題別のよく使用される参考図書，である。三津石らはマイニング探検会によってレファレンス協同データベースを活用して開発された, レファレンススキル向上のためのツールである Ref.Master について概説し，運用の結果得られた知見について考察している。Ref.Master はゲーム感覚で遊び

ながらレファレンススキルを学習できるツールである。Game with a Purpose のアプローチをとっており，学習者がゲームを進めた副産物としてコンピュータでは困難な処理を行うことができるよう設計されている。三津石らはこのアプローチを用いることで，レファレンス協同データベースの品質向上に貢献することができる，としている。吉田は図書館員向けの研修において，事例を活用することの効果を検証し，具体的な実施方法について検証している。このときの事例としてはレファレンス協同データベースに収載された事例を用いている。

2.3 機械学習による文書の分類に関する研究

手書き数字を認識し，1,…,9 のそれぞれに機械的に分類する問題を考える。このような問題を，識別のためのルールやヒューリスティックを作成するアプローチを用いて解決しようとすると，手書き数字の多様性によってルールの例外も発散してしまい，実際にはうまくいかない場合が多い。これに対して，機械学習的なアプローチについてビショップ (2012a) は以下のように述べている。

> 一方，機械学習アプローチを採用すれば，はるかに良い結果が得られる。機械学習では，まず訓練集合 (training set) と呼ぶ N 個の手書き数字の大きな集合 $\{x_1, \cdots, x_N\}$ を使って，モデルのパラメータを適応的に調整する。ただし，訓練集合の手書き数字のカテゴリは，あらかじめ人間が手で 1 つ 1 つラベル付けすることなどにより既知とする。1 つ 1 つの数字に対応するカテゴリは目標ベクトル (target vector) **t** を用いて表現できる。(ビショップ (2012a)，p.1-2)

高村 (2010) の中で，自然言語処理の近年実用化され注目を集めている技術として，以下の 3 つが挙げられている。即ち，(1) 質問応答システム，(2) 情報抽出，(3) 機械翻訳，の 3 つである。また，永田ら (2001) は「テキスト情報を自動的に分類する技術がさまざまなところで非常に重要な役割を果たす」として，Web コンテンツフィルタとスパムメールフィルタを機械学習の社会での応用例として挙げている。詳細は後述するが，本研究では疑問の書かれた文書の自動収集 (即ち情報抽出) を行う際と，文書に対する NDC 記号の自動付

与 (即ち自動分類) を行う際に，機械学習を利用している。以下では文書を対象とした機械学習に関する詳細を述べる。

2.3.1 機械学習の概要

機械学習手法は以下の 2 つに大別できる。即ち，(1) 教師あり学習，(2) 教師なし学習，である。(1) の教師あり学習は，クラスラベルや値があらかじめ付与されている文書を学習用データとして，モデルを学習しそこから未知の文書の適切なラベルや値を予測する手法である。クラスラベルを予測する場合は分類，連続値を予測する場合は回帰と呼ばれる。教師あり学習手法としては，パーセプトロン，ロジスティック回帰，SVM，ニューラルネットワーク，決定木，ランダムフォレスト， Naive Bayes，などの手法が用いられる。(2) の教師なし学習は，正解となるクラスラベルや値が分かっていない文書同士の類似度や分布を求める手法である。文書同士の類似度を測定し似た文書を同じ集合としてクラスタリングするときや，文書に現れる語の中で各文書の特徴をよく表す語だけを選別し次元削減を行うときに用いられる。教師なし学習手法としては，k 平均法，EM アルゴリズム，PLSI (Probabilistic Latent Semantic Indexing)，LDA (Latent Dirichlet Allocation)，などの手法が用いられる。本研究では，後述するが，教師あり学習手法である SVM，決定木，Naive Bayes，を用いてテキスト自動分類を行う。

2.3.1.1 機械学習による文書の分類

本項では機械学習による分類について述べる。機械学習による分類とは即ち，「ある入力ベクトル $\mathbf{x}$ を K 個の離散クラス C_k の 1 つに割り当てる事」(ビショップ(2012a)，p.177) である。入力空間は各クラスを表す決定領域に分離され，この決定領域の境界を決定面と呼ぶ。また，入力空間に対して線形な決定面によってクラスを識別するモデルの事を，線形識別モデルと呼ぶ。分類問題では，

> 離散値をとるクラスラベルを予測, あるいはもっと一般的に領域 (0,1) の値をとる事後確率を予測したい。そこで，線形モデルを，パラメータ $\mathbf{w}$ の線形関数を非線形関数 $f(\cdot)$ によって変換するように一般化する。

$$y(\mathbf{x}) = f(\mathbf{w}^{\mathrm{T}}\mathbf{x} + w_0)$$

機械学習の分野では, この $f(\cdot)$は活性化関数 (activation function) として知られている。(ビショップ (2012a), p.178)

ここで, **w** は M 次の多項式の係数をまとめたベクトルであり, $\mathbf{w} = (w_1, w_2, \cdots, w_M)$ である。この活性化関数が非線形であっても，決定面は入力ベクトル **x** の線形関数になる。

文書の分類に対して機械学習を用いる場合，文書をそのままの形で入力ベクトル **x** として扱うことができない。そこで，文書を機械学習で利用できる入力ベクトル **x** の形に変換するために，文書を単語に分割し各単語の出現頻度を要素 (特徴量) の値とした，ベクトルの形で表現する方法を用いる。ただし，このベクトル表現では文書の中の文の構造や，単語の現れる順番などの情報が失われている簡略表現であることに注意する必要がある。単語の出現頻度を要素として，単語の現れる順番などの情報が失われている文書のベクトル表現を bag of words と呼ぶ。bag of words によるベクトル表現の形は，特徴量の選び方や値の取り方によって変わる。この例のように，文書内に単語が現れた回数の情報を利用して表現されたベクトルを，頻度ベクトルと呼ぶ。その他にも文書内に単語が現れたか否かの情報を利用する，二値ベクトルなどがある。教師あり学習において，こうして作成した各文書に関するベクトルのうち，モデルの学習に使われるものを学習用コーパス，モデルのテストに使われるものを評価用コーパスと呼ぶ。

2.3.1.2 機械学習手法

機械学習手法として様々なものが開発されている。本項ではその主なものとして，以下の 5 つの手法について解説する。即ち，(1) 初期の機械学習手法であるパーセプトロン，(2) ロジスティック回帰，(3) SVM，(4) 決定木とそのアンサンブル学習であるランダムフォレスト，(5) Naive Bayes，の 5 つである。

(1) パーセプトロン

パーセプトロンは Rosenblatt によって提案された機械学習手法である。この手法は McCulloch と Pitts によって提案されたニューロンモデルに基づいており，誤分類率を最小

化する最適なパラメータを計算した後に，そのパラメータを入力とかけ合わせることで，2つのクラスのどちらに属しているかを分類することができる。パーセプトロンの具体的な分類方法について，ビショップ (2012a) は以下のように述べている。

まずある決まった非線形関数を用いて入力ベクトル $\mathbf{x}$ を変換して特徴ベクトル $\boldsymbol{\phi}(\mathbf{x})$ を得て，以下の式で表される一般化線形モデルを構成する。

$$\mathrm{y}(\mathbf{x}) = f(\mathbf{w}^{\mathrm{T}}\boldsymbol{\varphi}(\mathbf{x}))$$

ここで非線形活性化関数はステップ関数

$$f(a) = \begin{cases} +1, & a > 0 \\ -1, & a < 0 \end{cases}$$

で与えられる。 (中略) パーセプトロン基準は

$$E_p(\mathbf{w}) = -\sum_{n \in M} \mathbf{w}^{\mathrm{T}}\boldsymbol{\varphi}_n t_n$$

で与えられる。ここで $\boldsymbol{\varphi}_n = \boldsymbol{\varphi}(\mathbf{x}_n)$ であり，M は誤分類されたすべてのパターンの集合を表す。(ビショップ (2012a)，p.190-192)

また，t_n はクラスに対する目的変数値であり+1 か - 1 をとる。このときの $E_p(\mathbf{w})$を誤差関数と呼び，誤差関数を最小化するようなパラメータ $\mathbf{w}$ を求める事で，誤分類の少ない最適な決定面を得ることができる。誤差関数の最小化のために，

確率的最急降下アルゴリズムを適用する。そのとき，重みベクトル $\mathbf{w}$ の変化は

$$\mathbf{w}^{(\tau+1)} = \mathbf{w}^{(\tau)} - \eta \nabla E_p(\mathbf{w}) = \mathbf{w}^{(\tau)} + \eta \boldsymbol{\varphi}_n t_n$$

ここで η は学習率パラメータであり， τ は整数で，アルゴリズムのステップ数である。(ビショップ (2012a)，p.192)

パーセプトロンの問題点としてよく挙げられるものが 2 つある。即ち，(a) 線形分離可能でない場合学習がいつまでも終わらない，(b) コーパスに含まれる文書が全て分類されてし

まった段階で学習が終わってしまう，の 2 つである。(a) の線形分離可能でない場合とは，文書 A の重みを更新すると文書 B が正しく分類されず，文書 B の重みを更新すると文書 A が正しく分類されない，というような場合である。この場合，パーセプトロンによる分類境界は，更新を繰り返し続けることになる。この問題は，学習用コーパスに対するトレーニングの最大回数や，誤分類の最大数を設定することで回避できる。あるいは，完全に線形分離するのではなく，文書のクラスラベルに分類される確率を計算する方法もある。このような機械学習手法として，後述するロジスティック回帰がある。(b) に関しては，学習用コーパスに含まれる文書が全て正しく分類されると全ての重みの更新値が 0 になるため，それ以上重みが更新されなくなる問題である。重みが更新されなくなっているにもかかわらず，このモデルを評価用コーパスに対して用いたときに，全ての文書を正しく分類できるとは限らない。即ち，このモデルは学習用コーパスに過学習したモデルである可能性がある。上記のような単純なパーセプトロンの学習規則では，この過学習を防ぐことができない。しかし，この問題はコスト関数に正則化項を加えることで回避することができる。パーセプトロンでは誤分類率を最小化することで文書を分類する決定境界を求めたが，これはマージンが 0 の場合に等しい。マージンを最大化する決定境界を求めることで，学習用コーパスだけでなく評価用コーパスにも最適な分類を行うことができる。このような機械学習手法が後述する SVM である。

(2) ロジスティック回帰

ロジスティック回帰は分類のためのモデルである。パーセプトロンと同様に，2 クラス分類のための線形モデルであるが，最尤法を用いて一般化線形モデルのパラメータを直接決定する点に違いがある。本項ではロジスティック回帰の一般化線形モデルについて述べる。

クラス C_1 の事後確率は入力ベクトル $\mathbf{x}$ に対して変換をほどこした特徴ベクトル $\boldsymbol{\phi}$ の線形関数のロジスティックシグモイド関数として以下のように書ける。つまりその事後確率は

$$p(C_1|\boldsymbol{\varphi}) = y(\boldsymbol{\varphi}) = \sigma(\mathbf{w}^{\mathrm{T}}\boldsymbol{\varphi})$$

となる。(ビショップ (2012a)，p.204)

ここで非線形活性化関数 $\sigma(\cdot)$ はロジスティックシグモイド関数

$$\sigma(a) = \frac{1}{1 + \exp(-a)}$$

で与えられる。最尤法を用いる事でこのロジスティック回帰モデルのパラメータ $\mathbf{w}$ を，以下のように決定することができる。

データ集合 $\{\boldsymbol{\phi}_n, t_n\}$，$t_n \in \{0, 1\}$ であり，$\boldsymbol{\phi}_n = \boldsymbol{\phi}(\mathbf{x}_n)$ で $n = 1, \cdots, N$ に対する尤度関数は,

$$p(\mathbf{t}|\mathbf{w}) = \prod_{n=1}^{N} y_n^{t_n}\{1 - y_n\}^{1-t_n}$$

と書ける。ここで $\mathbf{t} = (t_1, \cdots, t_N)^{\mathrm{T}}$ であり，$y_n = p(C_1|\boldsymbol{\phi}_n)$ である。この尤度の負の対数を取って誤差関数を定義する。この誤差関数は，以下の式で与えられる交差エントロピー誤差関数 (cross-entropy error function)

$$E(\mathbf{w}) = -\ln p(\mathbf{t}|\mathbf{w}) = -\sum_{n=1}^{N}\{t_n \ln y_n + (1 - t_n)\ln(1 - y_n)\}$$

を与える。ここで，$y_n = \sigma(a_n)$ であり，$a_n = \mathbf{w}^{\mathrm{T}}\boldsymbol{\phi}_n$ である。$\mathbf{w}$ に対する誤差関数の勾配を取って,

$$\nabla E(\mathbf{w}) = \sum_{n=1}^{N}(y_n - t_n)\boldsymbol{\varphi}_n$$

が得られる。(ビショップ (2012a), p.204-205)

この誤差関数の勾配は即ち，パラメータ $\mathbf{w}$ の更新式である。ただし，これにより求められる解は，最尤法を用いて見つけた解である点に注意が必要である。この解は最適化アルゴリズムとパラメータ $\mathbf{w}$ の初期値に依存して，様々な解に収束してしまうという問題がある。

(3) SVM

SVM はパーセプトロンの発展形と考えることができ，広く利用されている機械学習手法

である。SVM が広く利用されるようになった理由は, SVM の持つ以下の 2 つの特徴にある。即ち，(1) オーバーフィッティングしにくい分類を行うことができる，(2) 非線形な決定面を容易に作ることができる，の 2 つである。(1) はマージンを最大化する決定境界を求めることで実現できる。(2) はカーネル関数を利用することで実現できる。

2 値分類問題に対する SVM の一般化線形モデルは，入力ベクトル $\mathbf{x}$ の変換関数 $\phi(\mathbf{x})$を用いて

$$y(\mathbf{x}) = \mathbf{w}^{\mathbf{T}}\boldsymbol{\varphi}(\mathbf{x}) + b$$

と表すことができる。SVM における決定境界は，マージンを最大化することを条件として選ばれる。即ち，決定境界に沿った正の超平面と負の超平面の間の距離を最大化する。以下に $n = 1,\cdots,N$ の訓練データを用いたマージンを最大化するパラメータ $\mathbf{w}$ の具体的な求め方について，ビショップ (2012b) は以下のように述べている。

分類境界から点 x_n までの距離は次のように表される。

$$\frac{t_n y(\mathbf{x}_n)}{||\mathbf{w}||} = \frac{t_n(\mathbf{w}^{\mathbf{T}}\boldsymbol{\varphi}(\mathbf{x}_n) + b)}{||\mathbf{w}||}$$

マージンは訓練データと分類境界の最短距離であり，今求めたいのはそのマージンを最大化するパラメータ $\mathbf{w}$ と b である。従って，解は次の最適化問題を解くことで得られる。

$$\text{argmax}\{\frac{1}{||\mathbf{w}||}\min[t_n(\mathbf{w}^{\mathbf{T}}\boldsymbol{\varphi}(\mathbf{x}_n) + b)]\}$$

(中略) また，このスケールの下ではすべてのデータについて次の制約式が成立する。

$$t_n(\mathbf{w}^{\mathbf{T}}\boldsymbol{\varphi}(\mathbf{x}_n) + b) \geq 1$$

(中略) 結局，マージンを最大化する解は，

$$\text{argmin}\frac{1}{2}||\mathbf{w}||^2$$

を制約の下で解くことで得られる。(ビショップ (2012b)，p.38-39)

制約条件のついた最適化問題は，ラグランジュの未定乗数法を用いることで解くことができる。ラグランジュ乗数を $a_n \geqq 0$ とすると，ラグランジュ関数は

$$L(\mathbf{w}, b, \mathbf{a}) = \frac{1}{2}||\mathbf{w}||^2 - \sum_{n=1}^{N} a_n \{t_n(\mathbf{w}^{\mathbf{T}}\boldsymbol{\varphi}(\mathbf{x}_n) + b) - 1\}$$

これをそれぞれの重みで偏微分し 0 とおくと，以下の数式が得られる。

$$\mathbf{w} = \sum_{n=1}^{N} a_n\, t_n \boldsymbol{\varphi}(\mathbf{x}_n)$$

$$0 = \sum_{n=1}^{N} a_n\, t_n$$

これをラグランジュ関数に代入すると

$$L(\mathbf{a}) = \sum_{n=1}^{N} a_n - \frac{1}{2}\sum_{n=1}^{N}\sum_{m=1}^{N} a_n\, a_m t_n t_m k(\mathbf{x}_n, \mathbf{x}_m)$$

が得られる。このラグランジュ関数を以下の制約条件のもとで最大化することで **a** の最適解が求まり，最終的にパラメータ **w** が求まる。

$$\sum_{n=1}^{N} a_n\, t_n = 0$$

$$a_n \geq 0, \quad \mathrm{n} = 1, \ldots, \mathrm{N}$$

このように，もともとの最適化問題 (**w** に対する最適化) である主問題に対し，置き換えられた最適化問題 (**a** に対する最適化) を双対問題と呼ぶ。

SVM は疎な解を持つサポートベクトルだけを計算に利用するため，計算にかかるコストが少なく，カーネルトリックの導入が容易であるという特徴を持つ。カーネルトリックとは，線形分離できない入力ベクトルを射影関数で高次元の空間に射影し，高次元空間における決定面を求めることで，疑似的に非線形分類を行う手法である。本来は，入力ベクトルを単純に高次元空間に射影しようとすると，計算コストが増えてしまうという問題がある。しかし，SVM においてはサポートベクトルに関する計算だけができればよく，サポートベクトルでない入力ベクトルは計算から除外されるため，計算コストの増加による影響が小さい。カーネルトリックにおいてよく用いられるカーネル関数として，木構造カーネル，多項式カーネル，動径基底関数カーネルなどがある。

(4) 決定木とランダムフォレスト

決定木は入力データの特徴量に基づいて情報利得を計算し，情報利得が最大となる特徴量で文書を分類する手法である。この分類は全ての入力データが正しいクラスラベルに分類されるまで行うことができる。しかし，全ての入力データを正しく分類してしまうと，パーセプトロンと同様にオーバーフィッティングの問題が起こる。極端に言えば，全ての文書に対して，1 つずつ分割条件を持たせてしまうことも可能である。この問題を解決するために，通常は決定木の分割の深さに制限を設ける。決定木はどの特徴量が分類に大きく寄与したかがわかりやすいため，他の機械学習手法に比べて意味を解釈しやすいという利点がある。以下に決定木の具体的な分類方法について，ビショップ (2012b) は以下のように述べている。葉ノードを τ=1,⋯,T としたときに，

葉ノード τ は入力空間の領域 R_τ を表し，N_τ 個のデータ点を持つとする。そして葉ノードの総数を|T|とする。領域 R_τ に対する最適な予測は以下で与えられる。

$$y_\tau = \frac{1}{N_\tau} \sum_{x_n \in R_\tau} t_n$$

従って，それに対応する残留の二乗和残差の寄与は

$$Q_\tau(T) = \sum_{x_n \in R_\tau} \{t_n - y_\tau\}^2$$

である。そして枝刈りの基準は以下で与えられる。

$$C(T) = \sum_{\tau=1}^{|T|} Q_\tau(T) + \lambda |T|$$

正規化パラメータλは，全体に残留する二乗和誤差と，葉ノードの数|*T*|で測られるモデルの複雑さとの間のトレードオフを決定し，その値は交差確認法により選ばれる。(ビショップ (2012b)，p.383)

このときの $Q_\tau(T)$ は置き換えが可能であり，他に交差エントロピー誤差関数やジニ係数，分類誤差などがよく用いられる。

交差エントロピー誤差関数

$$Q_\tau(T) = -\sum_{k=1}^{K} p_{\tau k} \ln p_{\tau k}$$

ジニ係数

$$Q_\tau(T) = \sum_{k=1}^{K} p_{\tau k}(1 - p_{\tau k})$$

分類誤差

$$Q_\tau(T) = 1 - \max\{p_{\tau k}\}$$

ここで k はクラスラベル，$p_{\tau k}$ は特定の葉ノードにおいてクラスラベル k に属する入力データの割合を表す。交差エントロピー誤差関数とジニ係数は，その葉ノードの中に全てのクラスラベルが同じ割合で混ざっているときに最大になる。

決定木を発展させたものとして，ランダムフォレストがある。SVM と同様に，分類性能が高いとされており，広く利用されている。ランダムフォレストは 1 つの文書集合から複数の決定木を作成し，それらの決定木の分類結果の多数決によって分類を行う，アンサンブル学習と見なすことができる。ランダムフォレストのアルゴリズムは，以下のように表すことができる。

1. 学習用コーパスから n 個の文書をランダムに抽出する。
2. n 個の文書に対して決定木を作成する。
   2-1. d 個の特徴量をランダムに非復元抽出する。
   2-2. 目的関数を計算し，最適な分割を行う特徴量によってノードを分割する。
3. ステップ 1-2 を繰り返す。
4. 複数個の決定木のクラスラベルの予測の多数決をとり，クラスラベルを予測する。

(5) Naive Bayes

Naive Bayes は確率に基づいた分類器であり，入力ベクトル **x** に対して P(k|**x**) が最大になるようなクラスラベル k を出力する。この分類器はベイズの定理を利用し，特徴量同士が独立であるという，単純な仮定を前提とするため Naive Bayes と呼ばれる。高村 (2010) は，多変数ベルヌーイモデルを使った Naive Bayes について解説している。多変数ベルヌーイモデルは，ある特徴量が入力ベクトル **x** の中に現れたか否か (1 か 0 か) をその特徴量の値として考えるモデルである。

入力ベクトルに現れる全特徴量を V としたときに, V に含まれる特徴量 f とクラスラベル k について，ベルヌーイ分布に従う確率変数 X_{fk} を考えると，各確率変数は特徴量 f が入力

ベクトル **x** 内に現れるときに 1，現れなかったときに 0 をとる。ここで X_{fk} が 1 になる確率を p_{fk} で表す。また，クラスラベルが k である確率を $p_k = P(k)$ とする。このとき，入力ベクトル **x** の生起確率は

$$P(\mathbf{x}|k) = \prod_{f \in V} p_{fk} \prod_{f \in /V} (1 - p_{fk})$$

で表される。ここでベイズの定理を導入する。

$$P(k|\mathbf{x}) = \frac{P(k)P(\mathbf{x}|k)}{P(\mathbf{x})}$$

$P(k|\mathbf{x})$ を事後確率，$P(k)$ を事前確率、$P(\mathbf{x}|k)$ を尤度と呼ぶ。左辺は入力ベクトル **x** のクラスラベルが k である確率を示している。即ち，$P(k|\mathbf{x})$ を最大になるようなクラスラベル k を求めることができれば, そのクラスラベル k が入力ベクトル **x** にとって最も適切なクラスラベルである。ここで右辺の分母は k に依存しないため計算から除外できる。入力ベクトル **x** に最も適切なクラスラベル k は

$$k = \text{argmax } P(k) \prod_{f \in V} p_{fk} \prod_{f \in /V} (1 - p_{fk})$$

を求めることに等しい。この最大化問題を解くための方法として，最尤推定法，最大事後確率推定，ベイズ推定などの方法が知られているが，ここでは最尤推定法で解いた場合について述べる。計算を楽にするために対数をとると，

$$\log\left(P(k) \prod_{f \in V} p_{fk} \prod_{f \in /V} (1 - p_{fk})\right)$$
$$= \sum_k N_k \log p_k + \sum_k \sum_{f \in V} N_{fk} \log p_{fk} + \sum_k \sum_{f \in /V} (N_k - N_{fk}) \log(1 - p_{fk})$$

となる。ここで N_k はクラスラベルが k である入力ベクトル **x** の数，N_{wk} はクラスラベルが k

で特徴量 f を含む入力ベクトル $\mathbf{x}$ の数である。最尤推定において求めるべきパラメータは，この式を最大化する p_{fk} と p_k である。また，$P(k)$ は確率であることから，以下の制約条件が付く。

$$\sum_k p_k = 1$$

上記の対数尤度の最大化問題を制約条件のもとで解くためには，ラグランジュの未定乗数法を用いればよい。このときのラグランジュ関数は以下の式で表せる。

$$L(\theta, \lambda) = \log P(k)P(\mathbf{x}|k) + \lambda(\sum_k p_k - 1)$$

ここで θ は求めたいパラメータの集合 (つまり p_{fk} と p_k) である。各パラメータについて偏微分し 0 とおくと，結果として

$$p_{fk} = \frac{N_{fk}}{N_k}, \qquad p_k = \frac{N_k}{\sum_k N_k}$$

が求まる。ここで求められる p_{fk} は $f \times k$ 個，p_k は k 個存在する。これらは全てパラメータであり，クラスラベルが未知の入力ベクトル $\mathbf{x}$ に対する分類器として働く。

最尤推定法を用いた Naive Bayes において，学習用コーパスの中に 1 度も現れていない特徴量 f が評価用コーパスに現れた場合，注意が必要である。確率の積を求めているために，その他の全ての特徴量 f を無視してクラスラベルが k である確率は 0 になってしまう。これはパーセプトロンにおける，オーバーフィッティングと同様の問題である。このような問題を解決するための手法として，ラプラススムージングがある。ラプラススムージングは全ての特徴量 f の出現頻度とクラス数に 1 を足してパラメータを計算する。これによって、1 度も現れていない特徴量 f が新たな入力ベクトル $\mathbf{x}$ の中に現れたときに，その入力ベクトル $\mathbf{x}$ の確率が 0 になってしまうことを避けられる。

2.3.2 機械学習に関する先行研究

テキスト自動分類における機械学習手法を概説した研究としては，永田ら (2001)，Sebastiani (2002)，石田 (2003)，が挙げられる。永田らは，テキスト自動分類の問題設定および機械学習の代表的な手法について解説している。なかでも分類精度の高さが注目されている SVM に関して，RWCP テキストコーパスを用いたテキスト自動分類の例を紹介している。また，情報抽出・質疑応答，ポータルサイトの自動構築へのテキスト分類の応用について言及している。Sebastiani は機械学習を用いたテキスト自動分類について解説している。特に以下の 3 点に関して詳細に議論している。即ち，(1) 文書のベクトル表現における次元削減手法，(2) 識別モデルと生成モデルによる分類手法，(3) 精度，再現率を用いた分類性能の評価方法，である。Sebastiani はテキスト自動分類がメジャーな研究領になった理由として以下の 4 点を挙げている。即ち，(1) 応用の範囲が広い，(2) 人手では処理できないサンプル数でも処理が可能である，(3) 自動分類が不可能であったとしてもその結果を参考にすることで人手での分類を援助できる，(4) 専門家と同程度の精度で分類ができるようになった，の 4 点である。石田は，テキスト自動分類に関する研究を再検討し，その将来の方向性について述べている。ここでの方向性は Web ページを対象とした自動分類を想定したものである。単位当たりの分量が少ない記事を含む Web ページを対象としたテキスト自動分類において，テキスト構造が従来よりも重要な要素になり，特徴量の決定は従来の問題に加えて新たな問題を抱えると予想している。その一方でカテゴリ表現や類似度計算などはほとんど影響を受けないとしている。

機械学習を行う際の特徴量に工夫を加えた文書の分類の研究としては，相澤 (2003)，太田ら (2004)，藤野ら (2006)，平田ら (2007)，後藤 (2010)，が挙げられる。相澤は，特徴量として低頻度語を利用した場合のテキスト自動分類の性能の改善方法について検証している。改善方法として具体的には以下の 3 つが挙げられている。即ち，(1) 確率重み付き情報量による語の重み付け，(2) 統計的ディスカウンティングによる確率推定，(3) 抽出した複合語の利用，である。実験データとして，NTCIR に含まれる文書 309,999 件を学習用コーパスとして, 10,000 件を評価用コーパスとして用いている。結果として, SVM に関しては (3) を用いる事で性能の改善が見られたとしている。太田らは，テキスト自動分類の特徴量選択のために用語抽出の技術を用いて専門用語を抽出し，これを特徴量と用いることを提案している。RWCP テキストコーパスの中から 10 分類の記事を 100 件ずつ，計 1,000 件を実

験に用いている。結果として，名詞のみを特徴量として用いた場合と比べて，専門用語を特徴量として用いることで少ない特徴量で同程度の精度を得ることができたとしている。その一方で，軍事，国際関係の分類を除いて再現率は低下している。藤野らは，生成，識別アプローチのハイブリッドに基づく分類器を用いて，文書に付随する付加情報を利用することで分類の性能を向上させる方法を提案している。付加情報とは，テキスト構造としての本文，タイトル，アンカーテキストなどであり，提案手法ではテキスト構造ごとに異なる重み付けを行う事で分類精度を向上させる。実験データとして，学習用コーパスに 20 Newsgroups に含まれる文書 8,000 件，WebKB に含まれる文書 2,000 件，NIPS に含まれる文書 500 件，残りを評価用コーパスとして用いている。結果として，生成，識別アプローチのハイブリットに基づく分類器を用いることで従来の分類器に比べ分類精度を向上させることができたとしている。付加情報を利用した場合も，提案手法による分類精度が最も高かったとしている。平田らは，反復度を統計量として用いて文書の特徴量となる文字列を抽出し，これを特徴量として SVM によるテキスト自動分類を行っている。反復度とは，文書中で何度も繰り返される文字列は文書の特徴を表す重要な文字列であるという仮定に基づく統計量である。実験データとして，Reuter-21578 テキストコレクションに含まれる文書 9,603 件を学習用コーパスとして，3,299 件を評価用コーパスとして用いている。結果として，条件付確率を用いて特徴量を抽出した場合と比較して，反復度による特徴量を用いた場合に全体としての分類性能が向上したとしている。特にサンプル数が少ない分類において F 値が高かったことから，反復度はサンプル数の少ない場合に有効な特徴量であるとしている。後藤らは，テキスト自動分類におけるベクトル空間の特徴として，(1) 高次元，(2) 不必要な特徴量の混入，(3) スパースネスの問題を持つ，の 3 つを挙げ，これらの問題の漸近的な挙動を理論的に解析する枠組みを構築した上で，(2)，(3) について分析している。結果として，高次元かつスパースネスの問題を有するテキスト自動分類のための距離として余弦尺度を用いることの正当性を理論的に裏付けることができたとしている。ただし，後藤らは余弦尺度による距離が分類誤り率に与える影響を評価できておらず，実際にテキスト自動分類を行う場合，次元削減を行うことで分類性能が向上することも指摘されている。

ブースティングを用いたテキスト自動分類の研究としては塚本ら (2001)，平ら (2002) が挙げられる。塚本らは，能動学習がテキスト分類にどの程度有効かを明らかにするために，AdaBoost の能動学習法である Query-By-Boosting を新たにテキスト分類に適用した結果

を報告している。実験データとして，Reuters21578 に含まれる文書から学習用コーパスとして 9,603 件，評価用コーパスとして 3,299 件を用いている。結果として，AdaBoost の能動学習を用いると 1/10 程度のサンプル数で全てのサンプルを使った場合と同程度の精度を得る事が出来るとしている。平らは，トランスダクティブ法を用いたブースティング法によるテキスト自動分類手法を提案し，従来のブースティング法，SVM との比較実験を行っている。トランスダクティブ法とは学習用コーパスだけでなく分類クラスの付与されていないコーパスの分布も考慮に入れる方法である。実験データとして，RWCP テキストコーパスに含まれる文書から学習用コーパスとして 1,000 件，評価用コーパスとして 1,000 件を用いている。の結果として，特に訓練データが少ない場合に従来手法に比べて提案手法での精度が向上したとしている。

SVM を用いたテキスト自動分類の研究としては，Joachims (1998)，平ら (1998)，平ら (2000)，Tong et al. (2001)，が挙げられる。Joachims は SVM が以下の 3 つの点で，理論的にもまた実証的にもテキスト自動分類に適した機械学習手法であるとしている。3 つとは即ち，(1) 高次元の特徴空間を持つ，(2) 多くの特徴量が似通っている，(3) 疎なベクトルを持つ，の 3 つである。平ら (1998) は，従来手法に比べて過学習せずに最適解が得られるとされている SVM を用いたテキスト自動分類を提案している。分類精度に関する実験を行う事によって，SVM に関して以下の 3 点を確認している。(1) 特徴ベクトルの要素数を増加させても過学習を起こらない，(2) 多項式 Kernel 関数の次元を変化させてもこの実験の範囲では精度が大きく向上することはないが，過学習が起こらないことから次元を上げていく事で高い精度が得られるかもしれない，(3) 目的関数の正例と負例に関する部分を分離する事で，分類精度が向上する。平ら (2000) は，日本語新聞記事コーパスを使用して SVM によるテキスト分類における最適な属性選択についての調査を行っている。SVM と C4.5 に関して，属性選択として以下の 2 つの方法を比較しながら，SVM による手法における最適の精度となる属性選択を調査している。2 つの方法とは即ち，(1) 相互情報量を基準とした選択方法，(2) 品詞を基準とした選択方法，である。結果として，品詞を基準とした選択手法の性能が高く，SVM に関しては用いる特徴量を選択するのではなく単純に実験における全ての特徴量を用いることが有効であるとしている。Tong et al. は SVM を基にラベル無し文書を利用したテキスト自動分類手法を提案している。Reiter-21578 を用いた実験から，大規模なラベル付けされた文書の必要性を軽減し，一部の文書だけを用いて全文書を用いる場合に匹敵

する分類性能を持たせることができたとしている。

Naive Bayes を用いたテキスト自動分類の研究としては，岩永ら (2005)，花井ら (2005)，田端 (2006)，が挙げられる。岩永らは， Naive Bayes を用いたスパムメールフィルタリング手法において，日本語と英語のメールが混在する環境を想定し，特徴量単位で使用するコーパスを切り替える手法を提案している。特徴量単位で切り替える提案手法によって，メール単位で切り替える従来手法よりも高い精度でスパムメールフィルタリングを行えたとしている。花井らは，各特徴量の間の独立を仮定している Naive Bayes によるテキスト分類において，2 つの特徴量同士の依存性を考慮する事で精度を上昇させる手法を提案している。提案手法では確率計算を行う際に依存性があると考えられる特徴量の組について，個々の特徴量の生起確率の積のかわりに同時確率を用いて計算を行っている。結果として，提案手法によって正解率を向上させることができたとしている。田端は，メールのフィルタリング技術の中でも，特に性能が良いとされている Naive Bayes を用いたベイジアンフィルタについて解説している。ベイジアンフィルタの精度を向上させるためには，バランスのよい閾値の設定，学習コーパス量の増加，定期的な学習コーパスの更新が必要であるとしている。また，ベイジアンフィルタを意図的に回避する方法として以下の 3 つを紹介している。即ち，(1) スパムメールに含まれにくい単語をメール内に付加する，(2) スパムメールに含まれやすい単語に対してわざと誤字や脱字を行う，(3) 本文を短くして添付されている画像の中に文章を書く，である。

複数の機械学習手法の比較を行ったテキスト自動分類の研究としては，相澤 (2002)，金 (2003)，高須ら (2003)，Khan et al. (2010)，が挙げられる。相澤は，大規模なテキスト分類問題を想定し，計算コストを削減するための単純な手法に対する性能改善の検討・評価を行っている。大規模テキスト分類において問題となる低頻度語を利用するという立場から，提案手法である確率重み付き情報量を分類尺度とする手法と SVM の分類性能について比較している。結果として，同条件での学習の実行時間は提案手法では 135 秒，SVM では約 1 日であり，提案手法により計算コストが削減されたとしている。分類性能に関しては基本的に SVM が高いが，サンプル数が少ない領域では提案手法の方が高い性能を持つとしている。またいずれの手法においても複合語を特徴量に含めることで性能が改善されたとしている。金は，SIR によるテキスト分類を試み，k-近傍法，SVM，ニューラルネットワークとの比較実験を行っている。青空文庫に保存されている 4 人の著者による著書 80 冊を学習用

コーパスとし，書き手の特徴を表す特徴量として助詞の unigram と bigram を用いて実験を行っている。実験の結果として SIR に基づく分類方法は計算量が少ないにも関わらず，他の分類手法より高い分類の精度を得る事ができ，サンプル数の減少による影響も少ないとしている。高須らは，テキスト自動分類に用いる訓練データの大きさを変えた場合の分類器の性能の比較を行っている。この結果，Rocchio の分類法のような単純な手法においては少ない訓練データで性能が収束するのに対して，決定木や SVM のような複雑な分類規則では訓練データ数を増やす事で高い分類性能を発揮することができるようになることを示している。また，数千記事程度の訓練データを用いれば，分類法の性能比較ができるとしている。 Khan et al. は既存の研究で用いられた，文書の自動分類とテキストマイニングの理論と方法についてレビューしている。その結果，文書の自動分類手法としては k-近傍法，SVM，Naive Bayes，がよく用いられていることを明らかにしている。また特徴量に関しては，文書のベクトル表現方法として多くの研究で，単純な Bag of Words が用いられており，統計的なテクニックとして十分でないことを指摘している。

2.4 2 章のまとめ

本章では，先行研究を以下の 3 つに大別した。3 つとは即ち，(1) テキストマイニングに関する研究 (2.1 節)，(2) デジタルレファレンスサービスに関する研究 (2.2 節)，(3) 機械学習による文書の分類に関する研究 (2.3 節)，の 3 つである。

2.1 節ではまず，大量の文書を自然言語処理や機械学習などの技術によって処理し，新たな知見を得る方法であるテキストマイニングの概要について過去に行われた例を挙げてを説明した。次に，テキストマイニングをブログや Twitter，その他の Web テキストに応用した研究をみた。これらの研究から，特にブログや Twitter において，それぞれの記述に特有の口語表現が用いられる問題がある点について確認した。最後に，テキストマイニングを図書館情報学領域で図書推薦や図書館利用者のクラスタリング，デジタルレファレンスサービスの分析などに応用した研究について概観した。

2.2 節では第一に，疑問記事を提示する Web サイトに関係のある，デジタルレファレンスサービスの評価や利用者満足度に関わる研究について述べた。第二に，本研究で実験データとして用いるレファレンス協同データベースの概要について説明し，レファレンス協同

データベースに関する研究を概観した。

2.3 節では，本研究で利用する文書の分類に関する研究について機械学習の概要を述べ，具体的な機械学習手法として基礎的なパーセプトロンとロジスティック回帰，本研究で用いる SVM，決定木，Naive Bayes について詳説し，機械学習に関する研究を概観した。

3　SNS に書かれた疑問記事を収集・提供する Web サイトの構築手法

本章では SNS に書かれた疑問を自動的に収集し，それらを回答できる可能性のある人々に提示する Web サイトを構築するための提案手法について述べる。

3.1　疑問記事を提示する Web サイトに関する予備調査

疑問記事を提示する Web サイトに関して，以下の 3 点を確認する予備調査を行った。即ち，(1) 疑問記事が Web 上に一定量存在すること，(2) 疑問記事に書かれた疑問はレファレンスの専門家でなくとも回答が可能であること，(3) ブログ著者は見知らぬ人からの回答でも好意的に受けとめること，の 3 点である。本予備調査は，疑問記事としてブログに書かれた図書のタイトルに関する疑問に限定して行った。ブログに限定した理由は，人が何か日常的な疑問を持ったときにブログに書き込むか Twitter に書き込むかの判断は各自が利用しているツールを用いるだけで，大きな偏りはないと考えたからである。また図書のタイトルに関する疑問に限定した理由は，レファレンスサービスでよく扱われる疑問であり，正解か否かの判断が容易だからである。以下ではそれぞれの調査方法とその結果を述べる。

まず (1) に関しては，「本　タイトル　思い出せない」，「本　題名　思い出せない」という 2 組のキーワードを用いてサーチエンジンで検索を行い，それぞれ検索結果の上位 200 件に疑問記事が何件含まれているかを調べた。調査は 2009 年 5 月に行い，サーチエンジンには Google ブログ検索と Yahoo!ブログ検索を用いた。その結果，「本　タイトル　思い出せない」をキーワードとして用いた場合，疑問記事を 7 件発見することができた。また「本　題名　思い出せない」をキーワードとして用いた場合，疑問記事を 9 件発見することができた。ここで，検索エンジンごとの検索結果として得られた疑問記事に重複はなかった。即ち全 800 (=200×2×2) 件を調査した結果，互いに異なる疑問記事が重複なく 16 件存在した。さらに，図書のタイトルが思い出せないという疑問記事は今後も現れ続ける可能性が高い。即ち，予備調査で対象とした疑問記事は Web 上に一定量存在することが確認できた。

(2) に関しては，記事中の疑問はレファレンスの専門家でなくとも回答可能なものであるかを調査する為，(1) の 16 件中でまだ回答がコメント欄に書きこまれていない 13 件の疑問

記事を, 記事の著者に代わって図書館のデジタルレファレンスサービス[3]と Q&A サイト (教えて!goo と Yahoo!知恵袋) に質問した。質問は 2009 年 7 月に行った。その結果，デジタルレファレンスサービスからは 8 件に対する回答を，Q&A サイトからは 7 件に対する回答を得ることができた。Q&A サイトによって得られた 7 件の回答は，デジタルレファレンスサービスによって得られた 8 件の回答のうちの 7 件の回答と同じものであった。以上のことから疑問記事の多くはレファレンスの専門家でなくとも回答可能であることがわかった。

(3) に関しては，(2) で得られた 8 件の回答を疑問記事のコメント欄に書き込み，ブログ著者からどのような反応が得られるかを調べた。調査は 2009 年 8 月に行った。結果，5 人のブログ著者から感謝のコメントを得ることができた。逆に迷惑である旨を伝えるブログ著者は存在しなかった。コメントが返ってこなかった著者は 3 名であるが，それらのブログは全てコメントを書きこむ 5 カ月以上前に更新が終わっており，著者らは回答を見ていない可能性も高い。以上のことから，見知らぬ者が回答を与えたとしても，ブログ著者からは感謝される方が多いことが分かった。

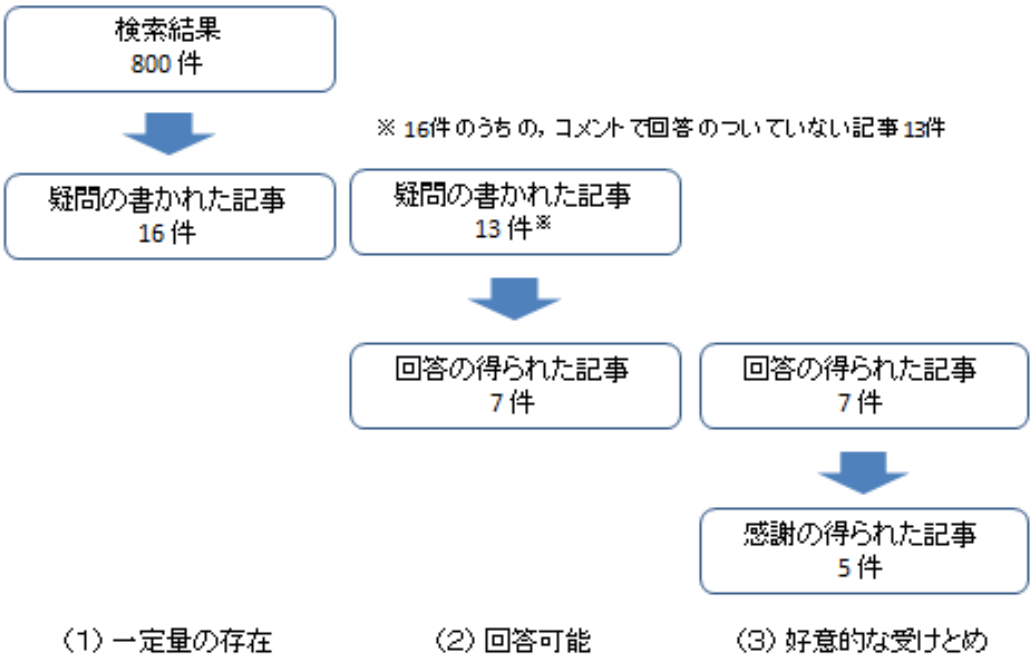

図 2 疑問記事を提示する Web サイトに関する予備調査

[3] メールで質問を受け付けている都道府県立図書館 7 館，市区町村立図書館 6 館に 1 問ずつ質問を行った。

以上の予備調査の結果から，以下の 3 点を確認した。即ち，(1) 疑問記事が Web 上に一定量存在する，(2) 疑問記事に書かれた疑問はレファレンスの専門家でなくとも回答可能である，(3) ブログ著者は疑問記事に対する回答を好意的に受けとめる，の 3 点である (図 2)。本研究はこれら 3 点を前提として進める。

3.2 疑問記事の効率的な収集実験

3.1 節における予備調査によって，サーチエンジンで検索することで一定量の疑問記事を発見できることがわかった。また，後述の 4.3 節における実験では Twitter において，2017 年 2 月 14 日 15:00 から 2 月 15 日 03:00 までの約 12 時間の間に書き込まれた「タイトル」を含むツイート 15,000 件のうち, 40 件がタイトルに関する疑問記事であることがわかった。これらの結果から，ブログと Twitter の両方で疑問記事は一定量存在するが，単純な検索方法で疑問記事を発見しようとするとその割合は 0.2%から 2%程度であり, その中から疑問記事を発見するにはそれなりのコストがかかることがわかる。疑問記事は将来にわたって書き込まれ増え続けるため，疑問記事の検索ごとにコストがかかる状態は望ましくない。そこで本研究では，自動的に疑問記事を収集するための手法を提案する。この提案手法は以下の 2 つのステップから成る。即ち，(1) 特徴的な表現をサーチエンジンに入力することによる疑問記事の検索，(2) 機械学習を用いた疑問記事の抽出，の 2 つである。これらの 2 つのステップは，どちらか一方を行うだけでも単純な検索方法に比べて疑問記事を効率よく収集することができる。しかし，(1) だけでは疑問記事とそれ以外の記事を完全に分類することは難しく，(2) だけでは検索結果全体に対する疑問記事の割合が少ないため，分類を行う際に非常に高い精度が要求される。そこで本研究においては，(1) で様々な記事が混在する Web 上から本研究で収集したい疑問記事を絞り込むためのキーワードを発見し，(2) で (1) の検索結果からテキスト自動分類を用いて疑問記事だけを抽出する手法を提案する。以下に提案手法の詳細を述べる。

3.2.1 特徴的な表現による検索

本項では検索に用いる特徴的な表現の定義 (3.2.1.1 項) を述べ，特徴的な表現を用いた検索実験に用いるデータ (3.2.1.2 項) とその実験の評価方法 (3.2.1.3 項) について説明する。

3.2.1.1 特徴的な表現

本研究では，疑問記事中に頻繁に現れ，それ以外の記事にはあまり現れない形態素列を「特徴的な表現」と呼ぶ。特徴的な表現をキーワードとしてサーチエンジンで検索を行えば，検索結果の中に疑問記事を多く含めることができると思われる。

本研究手法では特徴的な表現を導出するために，学習用コーパスとして疑問記事とそれ以外の記事をあらかじめ手作業で収集する。そして収集した疑問記事を形態素解析ソフト MeCab 0.993[4]で形態素に分解する。次に，疑問記事とそれ以外の記事において，3~5 個の連続する形態素の出現頻度を求める。例えば「子供の頃読んだ絵本」というテキストから 3 形態素列を抽出する場合，「子供_の_頃」「の_頃_読ん」「頃_読ん_だ」「読ん_だ_絵本」の 4 つの形態素列が得られる (ここで“_”は形態素の切れ目を表す)。その上で記事の中に現れる形態素列 w に関して以下の特徴度

$$F(w) = \frac{n_q(w)}{N_q} - \frac{n_o(w)}{N_o} \qquad \cdots \quad (1)$$

を算出する。ここで N_q は疑問記事に現れる全形態素列の出現頻度の合計，$n_q(w)$ は疑問記事における形態素列 w の出現頻度，N_o はそれ以外の記事に現れる全形態素列の出現頻度の合計，$n_o(w)$ はそれ以外の記事における形態素列 w の出現頻度である。特徴度 $F(w)$の値が大きくなるのは，$n_q(w)／N_q$ の値が $n_o(w)／N_o$ の値に比べて大きいときである。従って w は疑問記事に頻繁に現れ，かつそれ以外の記事にはあまり現れない形態素列である。本研究手法ではこの特徴度 $F(w)$ の値が大きな形態素列を特徴的な表現とし，検索に用いることで効率的に疑問記事を収集する。

3.2.1.2 特徴的な表現による検索で用いる実験データ

本研究では特徴的な表現を抽出するための学習用コーパスとして，ブログと Twitter それぞれに関して疑問記事とそれ以外の記事の 2 種類を 100 記事ずつ，計 400 記事を手作業で収集した。記事の収集には，ブログに関しては Google ブログ検索，Yahoo!ブログ検索を利

[4] http://taku910.github.io/mecab/

用し，Twitter に関しては Twitter 検索を利用した。記事の収集は 2010 年 2 月から 3 月にかけて行った。疑問記事は，これらのサーチエンジンにキーワードとして「タイトル　思い出せない」,「題名　思い出せない」,「タイトル　忘れた」,「題名　忘れた」の 4 つを入力し，得られた検索結果の中から「作品の内容について説明があり，かつその作品のタイトルが思い出せない」という内容のブログ記事，Twitter 記事を選択し疑問記事として収集した。また，それ以外の記事は，どのような記事にも含まれると考えられる助詞の「は」をキーワードとして同様に検索を行い，得られた検索結果の中から含まれる形態素数が 100 個を超える記事を選択し「それ以外の記事」として収集した。形態素解析には MeCab 0.993 を用い，設定はデフォルトのまま使用した。

3.2.1.3　実験の評価方法

特徴的な表現を用いることでどの程度効率的に疑問記事を収集できるかを評価するために，実際に特徴的な表現をキーワードとしてサーチエンジンで検索し，得られた検索結果の上位 100 記事の中に含まれている疑問記事の割合を疑問記事率として評価に用いる。即ち

$$\text{疑問記事率} = \frac{\text{疑問記事数}}{\text{検索結果の上位 100 記事}} \quad \cdots \ (2)$$

である。ここでサーチエンジンには，ブログに関しては Google ブログ検索と Yahoo!ブログ検索を用い, Twitter に関しては Twitter 検索を用いる。Google ブログ検索における精度を「疑問記事率(G)」，Yahoo!ブログ検索における精度を「疑問記事率(Y)」，Twitter 検索における精度を「疑問記事率(T)」と表す。

サーチエンジンを用いたキーワード検索では複数のキーワードを組み合わせることができる。一般に，複数のキーワードを論理演算子の AND で繋いで検索を行うことで検索の精度は高くなる。そこで，本研究で得られた特徴的な表現を二つ組み合わせた検索についても，同様の実験を行った。

3.2.2 テキスト自動分類による抽出

特徴的な表現による検索だけでは，疑問記事とそれ以外の記事を完全に分類する事は難しい。本項では特徴的な表現による検索結果 (3.2.1 項) から，テキスト自動分類によって疑問記事だけを抽出する手法 (3.2.2.1 項) について述べる。その後に，実験で用いるデータ (3.2.2.2 項) と実験の評価方法 (3.2.2.3 項) について述べる。

3.2.2.1 テキスト自動分類

疑問記事とそれ以外の記事を機械的に分類することができれば，2 つの記事が混在する記事集合から疑問記事だけを抽出することができる。このような問題を解く手法として機械学習を用いたテキスト自動分類がある。本研究では。MeCab 0.993 を用いて記事を形態素解析し，得られた形態素から，1 形態素，2 形態素列，3 形態素列の 3 種類の形態素列の出現頻度を特徴量 (2.3.1.1 項参照) として分類実験を行った。ここで，1 形態素 だけでなく複数の形態素列を特徴量として用いた理由は，疑問記事の分類は文書を分類するタスクとは違い，1 形態素だけを特徴量として用いてもうまくいかないことが予想されるからである。即ち，疑問記事の分類を行う場合，1 形態素より形態素列の方が分類に寄与する度合いが高まると考え，連続する形態素列を特徴量として用いた。ただし，形態素列のみを用いた場合，スパースネスの問題から分類結果が不安定になる可能性があるため，単語の場合も同時に特徴量として用いた。加えて，本研究では全記事における出現頻度が 5 回未満の特徴的な表現は分類に対する寄与が少ないと考え，用いる特徴量から排除した。

テキスト自動分類にはこれまで様々な手法が提案されているが，今回はその中でよく利用されている決定木，ブースティング，Naive Bayes，SVM の 4 種類の手法を用いてテキスト自動分類性能の比較実験を行う。ツールとしてはデータマイニングソフト Weka 3.6.4[5]を用いた。決定木には J48，ブースティングには AdaBoostM1，Naive Bayes には NaiveBayes，SVM には SMO を用いている。また，オプション設定はすべてデフォルト値を用いている。

3.2.2.2 テキスト自動分類による抽出で用いる実験データ

本研究では以下の記事群 A と記事群 B の 2 つの記事群についてテキスト自動分類を行った。記事群 A は Google ブログ検索に「タイトルが思い出せない」をキーワードとして入力

[5] http://www.cs.waikato.ac.nz/ml/weka/

した際の検索結果として得られる，ブログ記事 300 件を手作業で収集したものである。これらの記事の中の疑問記事は 52 件であり，非疑問記事は 248 件であった。次に記事群 B は，分類性能の向上のために疑問記事と非疑問記事のサンプル数を同じにした記事群として，ブログに関して疑問記事とそれ以外の記事を 150 記事ずつ，Twitter に関して疑問記事とそれ以外の記事を 800 記事ずつ手作業で収集したものである。ここで，記事群 B の疑問記事の収集においては，3.2.1 項で特徴的な表現として「が思い出せない」が頻出したことから，「タイトルが思い出せない」と「題名が思い出せない」の 2 つを検索のキーワードとして用いた。これらの 2 つの記事群に対して実験を行った [6]。これらの記事群の収集は 2010 年 8 月に行った。

3.2.2.3 実験の評価方法

本研究では精度，再現率，F 値によってテキスト自動分類の性能を評価した。ここで実際の疑問記事数を *Q*，疑問記事のうち正しく分類できた記事数を *Q_a*，疑問記事として分類された記事数を *Q_b*，実際の非疑問記事数を *O*，非疑問記事のうち正しく分類できた記事数を *O_a*，非疑問記事記事として分類された記事数を *O_b* としたとき，疑問記事，非疑問記事に関する精度，再現率，F 値はそれぞれ以下のように定義される。

$$\text{疑問精度} = \frac{Q\_a}{Q\_b} \quad \cdots \ (3)$$

$$\text{非疑問精度} = \frac{O\_a}{O\_b} \quad \cdots \ (4)$$

$$\text{疑問再現率} = \frac{Q\_a}{Q} \quad \cdots \ (5)$$

$$\text{非疑問再現率} = \frac{O\_a}{O} \quad \cdots \ (6)$$

[6] Twitter に関して，SVM に限り筆者の環境ではメモリ容量が足りず結果が得られなかったため， 500 記事ずつに減らしてテキスト自動分類実験を行った。

$$疑問\ F\ 値 = \frac{2}{\frac{1}{疑問精度} + \frac{1}{疑問再現率}} \quad \cdots (7)$$

$$非疑問\ F\ 値 = \frac{2}{\frac{1}{非疑問精度} + \frac{1}{非疑問再現率}} \quad \cdots (8)$$

本実験では 10 回交差検定を行い，その平均値から精度，再現率，F 値を求める。10 回交差検定とは収集したサンプルの中から 9／10 を学習用，1／10 を評価用として分割し，評価用と学習用を入れ替えて分類実験を 10 回繰り返すことで，サンプル数を増やさずにより正確な精度や再現率を求める手法である。

3.3 レファレンス事例に対する NDC 記号の自動付与実験

本節では，3.2 節の方法で収集した疑問記事に対してディレクトリ型検索を実装するために，疑問文に対して NDC 記号を自動付与する手法を提案する。本研究ではレファレンス協同データベースに収録されているレファレンス事例を例として NDC 記号の自動付与を行う。事例を用いる理由としては，本研究では機械学習手法を利用して NDC 記号の自動付与を行うが，そのためにはあらかじめ作成された正しい分類が付与されている学習用コーパスが必要になる点が挙げられる。レファレンス協同データベースには，質問の項目とその質問の分類を表す NDC の項目の組が多数登録されている。これらはあらかじめ作成された疑問文と NDC 記号の組として考えることができる。本研究では以下の 3 つの手法を用いて NDC 記号の自動付与を行った。即ち，(1) 参考資料を用いた手法 (3.3.1 項)，(2) 質問文の形態素を用いた手法 (3.3.2 項)，(3) その両者を同時に用いた手法 (3.3.3 項)，である。

本研究ではレファレンス協同データベースに登録され，2013 年 3 月 19 日に一般に公開されていた事例 62,328 件に記入されている NDC の項目 (NDC 記号)，参考資料の項目 (以下参考資料)，質問の項目 (以下質問文)，キーワードの項目 (以下キーワード) を実験に用いた (2.2.2 項の表 1 参照)。これらの事例の収集にはレファレンス協同データベース API を利用した。

3.3.1　参考資料を用いた NDC 記号の自動付与

本項では参考資料を用いた NDC 記号の自動付与手法について述べる。事例の参考資料に挙げられているのは，回答を作成するにあたって参考にした資料である。従って参考資料の NDC 記号は，事例に付与されるべき NDC 記号と類似していると考えられる。そこで参考資料の NDC 記号を調べ，それを特徴量として特徴ベクトルを作成し，この特徴ベクトルを利用して機械学習を行った。具体的には以下の流れで NDC 記号の自動付与を行う。

1. レファレンス協同データベースに登録されている事例を入手する。
2. 事例の参考資料から図書のタイトル部分を自動抽出し，利用可能な状態に自動整形する (3.3.1.1 項)。
3. (2)で自動抽出した図書のタイトルで国立国会図書館サーチ [7]を用いた検索を行い，その図書の NDC 記号を判定する (3.3.1.2 項)。
4. 機械学習ソフトウェア Weka 3.6.6 を利用して参考資料の NDC を特徴量とする特徴ベクトルを作成する (3.3.1.3 項)。

以下では 2〜4 のステップについて詳述する。

3.3.1.1　図書のタイトルの抽出

図書のタイトルの抽出においては，収集した事例に記入されている参考資料 32,657 件を実験に用いた。これらの事例において，参考資料の書誌事項の書き方が統一されていないことに注意が必要である。例として以下のような場合がある。

・1.『千葉の建築探訪』(中村哲夫　崙書房出版 2004)|0200797764;
・【資料 1】「クラシック音楽作品名辞典」(三省堂，2009)
・日本神話事典 / 青木周平[ほか]編　大和書房 , 1997　ISBN:4479840435　p.31

即ち，(a) 挙げられた参考資料の数え方 (1.と【資料 1】など) が異なる，(b) 図書のタイト

[7] http://iss.ndl.go.jp/

ルを囲う括弧が異なる，(c) 書かれている項目（著者名の有無など）が異なる，などの違いがある。

本研究ではこのように異なる参考資料に関する記入から，図書のタイトル部分を適切に抽出する為のデータクリーニングを行う。具体的には以下の 2 つのステップによって図書のタイトル部分を抽出する。即ち，(i) 『』で囲まれた部分を図書のタイトルとみなす，(ii) 『』がない場合，【資料 1】などの資料番号に関するノイズを除き，半角スペース，カンマで区切ったときの最初の単語列を図書のタイトルとみなす，の 2 ステップである。このときノイズは以下の正規表現を用いて削除した。

s/bibl_desk = (¥(|【文献|【資料|資料|¥s|)(¥d|[０-９][①-⑨])(¥)|】|.|.　)/bibl_desk = /

3.3.1.2　図書のタイトルからの NDC 記号の判定

次に NDC 記号の判定だが，本研究では図書のタイトルからその図書の NDC 記号を判定するために，検索用 API として提供されている OpenSearch[8]を利用して国立国会図書館サーチの検索を行う。

国立国会図書館サーチでは，図書のタイトルを入力することでその図書の NDC 記号も表示される。しかし，タイトルの検索だけではその図書を一意に同定することができない。表記の揺れにより，他の図書の NDC 記号が表示されてしまう可能性もある。そこで 3.3.1.1 項で述べた方法で抽出した図書のタイトルを形態素に分解し，その形態素の中の名詞のみを検索のキーワードとして用いる。これにより，表記のゆれの問題を解決し，3.3.1.1 項で削除できなかったノイズも同時に落とすことができる。例えば参考資料に書かれている図書のタイトルが『千葉の建築探訪』であれば，図書のタイトルに含まれる名詞をスペースで区切った「千葉 建築 探訪」で検索を行う。これにより得られる検索結果は「千葉」「建築」「探訪」をタイトルに含む 1 つあるいは複数の図書の情報である。

複数の検索結果が得られた場合，検索結果に現れた図書のタイトルを同様に形態素に分解する。そして参考資料から得られた図書のタイトルに含まれる名詞の集合 N_R と検索結果に現れた図書のタイトルに含まれる名詞の集合 N_i の間で以下の Dice 係数 S_i が最も高い図書 i を目的の図書として扱う。

[8] http://www.opensearch.org/Home

$$S_i = \frac{2s(N_R \cap N_i)}{s(N_R) + s(N_i)} \quad \cdots \quad (9)$$

ここで，$s(N)$ は集合 N の要素数を表す。例えば国立国会図書館サーチでタイトルに対してキーワード「千葉 建築 探訪」(N_R) を用いて検索すると，以下の 3 つの図書が検索結果として得られる。即ち，(1)『千葉の建築探訪』，(2)『千葉文華』，(3)『沖縄スタイル』の 3 つである。(2)，(3) に関してはメインタイトルの中にはキーワードが含まれていないが，サブタイトルの中に「千葉」，「建築」，「探訪」が含まれていた。これらの図書のタイトルはそれぞれ，(1)「千葉」「建築」「探訪」(N_1)，(2)「千葉」「文華」(N_2)，(3)「沖縄」「スタイル」(N_3) という名詞の集合に分解できる。それぞれに対して Dice 係数を求めると，(1) S_1 = 2 ×3 / (3+3) = 6/6 = 1.0，(2) S_2 = 2×1 / (3+2) = 2 / 5 = 0.4，(3) S_3 = 2×0 / (3+2) = 0/5 = 0.0，が得られ，この中で最も Dice 係数が高い図書である (1)『千葉の建築探訪』を目的の図書として扱う。その NDC 記号は 520 (建築学) となる。

3.3.1.3 NDC 記号の出現回数を用いた特徴ベクトルの作成

参考資料に挙げられた図書の NDC 記号を参照し, NDC 記号の出現回数を用いて特徴ベクトルを作成する。例として，ある事例に 4 つの参考資料が挙げられており，その参考資料の NDC 記号 [9]がそれぞれ 010 (図書館，図書館学)，013 (図書館管理)，016 (各種の図書館)，210 (日本史) であった場合を考える。1 桁目 (類) までの NDC 記号の自動付与を行いたい場合，その事例を表現するベクトルは 10 個の要素を持つベクトルとなる。このベクトルの要素の値は，0 (総記) に対応する値が 3，2 (歴史) に対応する値が 1，その他の要素はすべて 0 となる。

本研究では以下の 2 種類の場合に関して実験を行う。即ち, (1) NDC 記号の 1 桁目 (類) までを特徴量として自動付与を行った場合 (10 次元ベクトル) と, (2) NDC 記号の 2 桁目 (綱) までを特徴量として自動付与を行った場合 (100 次元ベクトル)，の 2 種類である。

[9]データを採集した時点でレファレンス協同データベースに使われているのは第 9 版までなので，ここでは第 9 版における NDC 表記を採用した。

3.3.1.4 NDC 記号の自動付与実験

こうして作成した特徴ベクトルから NDC 記号の自動付与を行い，その性能に関する評価を行う。NDC 記号の自動付与においては，収集した事例のうち NDC 記号と参考資料の両方が挙げられている事例 17,181 件を実験に用いた。評価は事例を各 NDC 記号へ分類した際の精度と再現率によって行った。

ここで，1 桁目までの自動付与を行う際に，実際の NDC 記号が i の事例を E_i，NDC 記号が i の事例のうち正しく付与できた事例数を E_i_a, NDC 記号が i の事例として付与された事例数を E_i_b としたとき，NDC 記号が i である事例に関する精度 P_i，再現率 R_i は，それぞれ以下のように定義される。

$$P_i = \frac{E_i\_a}{E_i\_b} \quad \cdots \quad (10)$$

$$R_i = \frac{E_i\_a}{E_i} \quad \cdots \quad (11)$$

精度および再現率の全体の平均を出す際には加重平均を用いた。精度と再現率の加重平均は以下の式で定義される。

$$\text{精度の加重平均} = \frac{\sum_{i=1}^{n} r_i P_i}{\sum_{i=1}^{n} r_i} \quad \cdots \quad (12)$$

$$\text{再現率の加重平均} = \frac{\sum_{i=1}^{n} r_i R_i}{\sum_{i=1}^{n} r_i} \quad \cdots \quad (13)$$

ここで r_i は各 NDC 記号に属する事例の全件数である。n は NDC 記号の数であり，1 桁目までの自動付与であれば n=10，2 桁目までの自動付与であれば n=100 となる。

参考資料を用いて NDC 記号を自動付与する際には，機械学習ソフトウェア Weka 3.6.6 を用いた。機械学習手法としては Random Forest, SVM ，Complement Naive Bayes (以下 CNB), を用いた。また，評価のベースラインとしては，原田ら (2007) の手法による精度，再現率を用いた。即ち，質問文に関して形態素の出現頻度をベクトルの要素として用いて NDC 記

号を自動付与した場合の精度，再現率である。本実験では 3 回交差検定を行った。

3.3.2 質問文を用いた NDC 記号の自動付与

事例における参考資料の記入は任意であるため，参考資料が空白の場合もある。参考資料を用いた手法は，参考資料が挙げられていない事例に対して NDC 記号を自動付与することができない。そこで，全ての事例に NDC 記号を付与したい場合，記入が必須である質問文を用いた NDC 記号の自動付与も行う必要がある。本項では 質問文を用いた NDC 記号の自動付与手法について述べる。

3.3.2.1 質問文を用いた特徴ベクトルの作成

参考資料の NDC 記号を用いる場合とは異なり，質問文に現れる形態素 1 つ 1 つを素性として特徴ベクトルを作成する場合，以下の 2 つの問題がある。即ち，(1) ベクトルの要素数は非常に大きくなり計算に時間がかかる，(2) 特徴量が NDC を基準としていないため精度が下がる，の 2 つである。(1) に関しては，参考資料を用いた場合の特徴ベクトルの要素数は，その分類で用いる NDC 記号の総数にクラスラベルを足した数に一致する。即ち 1 桁目までの分類であれば 10+クラスラベル個である。しかし質問文を用いて特徴ベクトルを作成しようとした場合，質問文内に現れた形態素の数だけベクトルの要素が必要になる。そのため，質問文に現れる形態素の出現頻度を特徴量として特徴ベクトルを作成した場合，要素数が非常に大きくなってしまい，計算に非常に多くの時間が必要になる。(2) に関しては，参考資料の場合の特徴ベクトルは参考資料の NDC 記号の出現頻度が特徴量となるために，その特徴ベクトルをそのまま用いて NDC 記号の付与をスムーズに行うことができる。しかし質問文における特徴量は質問文の中に現れる単語の出現頻度になるため，今回付与したいと考えている NDC 記号の構造を特徴ベクトルにそのまま用いることができない。このために，参考資料を用いた場合の NDC 記号の付与に比べ精度が下がってしまうことが懸念される。

そこで本研究では以下のような特徴ベクトルの作成手法を提案する。まず質問文と NDC の組である事例を学習用コーパスとして，すべての質問文に含まれる形態素の NDC ごとの出現頻度を計算する。そのときの形態素 j についての形態素ベクトルの i 番目の要素 v_{ij} を以下の式で計算する。

$$v_{ij} = \frac{w_{ij}}{\sum_{i=1}^{n} w_{ij}} \quad \cdots \ (14)$$

ここで w_{ij} は形態素 j の NDC 記号ごとの出現回数である。このベクトルの要素数は NDC 記号の数 n である。これにより質問文に現れたすべての形態素 j を形態素ベクトル $v_{ij} = (v_{0j}, v_{1j}, \cdots, v_{nj})$ で表すことができる。この具体的な例については後述する。

次に質問文 t に関して，その質問文 t に現れる形態素 j について形態素ベクトルの i 番目の要素 v_{ij} を加算することで，質問文 t の特徴ベクトルの i 番目の要素 $Vi(t)$ を以下の式で求める。

$$V_i(t) = \sum_{j=1}^{m} v_{ij} \quad \cdots \ (15)$$

ここで m は質問文 t に含まれる全形態素数である。こうして作成した質問文 t についての特徴ベクトル $\mathbf{V}(t)$ を用いて事例に対する NDC 記号の自動付与を行い，その性能の評価を行う。

以下に特徴ベクトルの作り方に関して，(1) 先行研究でとりあげた原田ら (2007) の手法 (2.2.2 項参照) を再現した作り方，(2) 提案手法を用いた作り方，の 2 つに関して例を挙げる。以下のような NDC 記号を付与された 3 つの質問文があり，これらを用いて NDC 記号の 1 桁目 (類) までを自動付与したい場合を考える。

・NDC 記号: 351，明治以降の日本の裁判統計を調べたい。
・NDC 記号: 350，『世界の統計』のデータを調べたい。
・NDC 記号: 216，明治時代にあった京都国技館の場所を知りたい。

どちらの方法においても，まず MeCab 0.993 を用いて形態素解析を行い，これらの質問文からすべての名詞を抽出する。上記の質問文に含まれる名詞は，明治，日本，裁判，統計，世界，統計，データ，明治，時代，京都，国技，館，場所の 13 個である。

原田ら (2007) の手法は，これら名詞の出現頻度をベクトルの要素として用いている。即

ち，{明治，日本，裁判，統計，世界，データ，時代，京都，国技，館，場所}が出現頻度を数え上げる名詞となる。名詞の出現頻度を数え上げることで，3 つの質問文の特徴ベクトルを以下のように表す。

・NDC 記号: 3，(1, 1, 1, 1, 0, 0, 0, 0, 0, 0, 0)
・NDC 記号: 3，(0, 0, 0, 1, 1, 1, 0, 0, 0, 0, 0)
・NDC 記号: 2，(1, 0, 0, 0, 0, 0, 1, 1, 1, 1, 1)

この方法では質問文を増やした際にその中に新しい名詞が含まれていると，そのたびにベクトルの要素数が増加していき，機械学習の実行に時間がかかってしまう問題がある。

これに対して，本研究が提案する手法ではすべての名詞に対してどの NDC 記号に属しているかを計算し, 形態素ベクトル v_{ij} を作る。このベクトルの要素は (NDC=0, NDC=1, NDC=2, …, NDC=9) である。即ち

明治　(0, 0, 1, 1, 0, 0, 0, 0, 0, 0)
裁判　(0, 0, 0, 1, 0, 0, 0, 0, 0, 0)
統計　(0, 0, 0, 2, 0, 0, 0, 0, 0, 0)
京都　(0, 0, 1, 0, 0, 0, 0, 0, 0, 0)

というように，すべての名詞に対して形態素ベクトル v_{ij} を作る。質問文に含まれる名詞の形態素ベクトル v_{ij} を足し合わせることで, 3 つの質問文の特徴ベクトル $Vi(t)$を以下のように表す。

・NDC 記号: 3，(0, 0, 1, 4, 0, 0, 0, 0, 0, 0)
・NDC 記号: 3，(0, 0, 0, 4, 0, 0, 0, 0, 0, 0)
・NDC 記号: 2，(0, 0, 6, 1, 0, 0, 0, 0, 0, 0)

このように, 3 つの質問文の NDC 記号と特徴ベクトル $Vi(t)$の各要素の中の最も大きな値 (それぞれ 4 回，4 回，6 回) が一致する。

この方法であれば，学習用コーパスをどんなに増やしても，ベクトルの要素数は学習用コーパスで用いられている名詞数ではなく, NDC 記号の数 (NDC 記号の 1 桁目であれば 10) +クラスラベル個に一致する。本研究では，提案手法を用いることで，コーパス数が増えると特徴ベクトルの要素数も増大してしまう問題の解決を図る。本研究では形態素ベクトルとして質問文以外を用いて特徴ベクトルを作成する場合に関しても実験を行う。即ち，(a) 事例の質問文を用いた場合，(b)事例のキーワードを用いた場合，(c) 図書のタイトルを用いた場合，の 3 種類の形態素ベクトルを作成し，そこから作成した 3 種類の異なる特徴ベクトルに対して分類性能の評価を行う。

3.3.2.2 NDC 記号の自動付与実験

このように事例の質問文から本研究で提案する特徴ベクトルを作成し，事例への NDC 記号の自動付与を行い，その性能に関する評価を行う。質問文を用いた NDC 記号の自動付与においては, 収集した事例のうち NDC 記号が付与されているすべての事例 40,288 件を用いた。また，図書のタイトルから形態素ベクトルを作る際には，日本全国書誌からランダムに抽出した 214,673 件の図書のタイトルと NDC 記号を用いた。評価は事例を各 NDC 記号へ分類した際の精度と再現率によって行った。本実験では 3 回交差検定を行った。

参考資料を用いて NDC 記号を自動付与する際には，機械学習ソフトウェア Weka 3.6.6 と LIBSVM[10]を用いた。機械学習手法としては Random Forest，SVM，CNB，を用いた。ここで SVM を用いる際に，カーネルには動径基底関数カーネルを使用し，C パラメータ (誤りに対するペナルティ) とγパラメータ (カーネル関数の係数) は LIBSVM のパッケージに含まれている easy.py を用いてグリッドサーチによる最適化を行っている。また，評価のベースラインとしては，原田ら (2007) の手法と同時に，参考資料を用いた NDC 記号の自動付与実験の結果を用いた。本実験では 3 回交差検定を行った。

3.3.3 参考資料と質問文の両方を用いた NDC 記号の自動付与

これまでに参考資料を用いる手法と，質問文を用いる手法という 2 種類の NDC 記号の自動付与手法を提案した。だが，参考資料が記入されている事例に対しては，参考資料と質問文の手法を同時に用いることで，自動付与の精度をより向上させられる可能性がある。

[10] https://www.csie.ntu.edu.tw/~cjlin/libsvm/

そこで本研究では，参考資料と質問文の両方が挙げられている事例に対して，上記の 2 種類の手法を同時に用いる NDC 記号の自動付与手法を提案する。

3.3.3.1 参考資料と質問文を用いた特徴ベクトルの作成

参考資料と質問文の特徴ベクトルを同時に用いる際には，以下の 2 つの方法でベクトルを作成し，それぞれの有効性を検証した。即ち，(a) 得られたベクトルを規格化し 2 つのベクトルの各要素を足し合わせる，(b) 得られたベクトルを規格化し 2 つのベクトルの各要素を独立に並べる，の 2 つである。例えば参考資料から得られたベクトルが (0, 0, 1, 0)で，質問文から得られたベクトルが (1, 0, 1, 0) だった場合, 規格化により後者は (0.5, 0, 0.5, 0) となり，(a) の方法では (0.5, 0, 1.5, 0) が得られ，(b) の方法では (0, 0, 1, 0, 0.5, 0, 0.5, 0) が得られる。

3.3.3.2 NDC 記号の自動付与実験

事例の参考資料と質問文の両方を用いて特徴ベクトルを作成し，事例への NDC 記号の自動付与を行い，その性能に関する評価を行う。NDC 記号の自動付与においては，収集した事例のうち NDC 記号と参考資料の両方が挙げられている事例 17,181 件を実験に用いた。評価は事例を各 NDC 記号へ分類した際の精度と再現率によって行った。参考資料と質問文の両方を用いた NDC 記号の自動付与では，機械学習ソフトウェア Weka 3.6.6 を用いた。ここで 3.3.2.2 項において使用した LIBSVM 3.12 を用いなかったのは，グリッドサーチを実行する際に非常に時間がかかり, かつ LIBSVM 3.12 で最適化された場合の SVM の精度と, Weka 3.6.6 での CNB の精度に大きな違いが見られなかったためである。また，評価のベースラインとしては，参考資料を用いた NDC 記号の自動付与実験の結果を用いた。本実験では 3 回交差検定を行った。

3.4 疑問記事の自動収集および NDC 記号の自動付与実験

本節では，これまでに述べてきた 2 つの提案手法を用いて，疑問記事を収集しその疑問記事に NDC 記号を自動付与する実験の手順を述べる。即ち，3.2 節の手法で収集した疑問記事に対して，3.3 節の手法を用いて NDC 記号を自動付与する。本研究においては，3.2 節

の手法で用いた学習用コーパスが図書のタイトルに関する疑問記事を収集するものであるため，収集できる疑問も図書のタイトルに関するものが多くなる。そこで本研究では，日本の公共図書館で用いられている主要な図書分類法である NDC を用いる。本研究では，対象をツイートに限定し以下の 2 つのステップによって疑問記事に対する NDC 記号の自動付与を行う。即ち，(1) 疑問記事の自動分類による抽出，(2) 疑問記事に対する NDC 記号の自動付与，である。以下の項ではそれぞれの具体的な内容について述べる。

3.4.1 実験データ

実験データとして，2017 年 2 月 14 日の 15:00 から 15 日の 03:00 までの約 12 時間に投稿され，文字列「タイトル」が含まれているツイート 15,000 件を用いた。このとき，ツイートの検索には TwitterAPI[11]を用いた。ここでリツイートに関しては本研究で扱いたい疑問の書かれたツイートではない可能性が高いため，先頭に RT の文字列を含むツイートは除外した。検索のキーワードをタイトルに限定したのは，3.2 節において疑問記事の収集を行う際にタイトルに関する疑問に限定したためである。しかしツイート上にはタイトル以外にも様々な疑問が書かれると思われる。こういった疑問を扱うことは今後の課題としたい。

3.4.2 疑問記事のテキスト自動分類による抽出

3.2.2 項ではテキスト自動分類を利用した効率的な疑問記事の収集手法を提案した。本研究ではこの提案手法を用いて，ツイートの集合から疑問記事だけを自動分類により抽出する。テキスト自動分類の詳しい結果については 4.1 節で述べるが，テキスト自動分類を行う際にツイートに対して最も F 値の高かった決定木 (J48) を機械学習アルゴリズムとして用い，特徴量としてツイート中の形態素の出現頻度を用いる。特徴量に関して 3.2.2 項と同様に，全記事における出現頻度が 5 回未満の特徴量は分類に対する寄与が少ないと考え，用いる特徴量から除外する。機械学習を行う際の学習用コーパスとしては 3.2.2.2 項で作成した記事群 B を用いる。

3.4.3 疑問記事に対する NDC 記号の自動付与

3.4.2 項ではツイートの中から疑問記事だけを抽出する方法を述べた。本研究ではこれら

[11] https://dev.twitter.com/docs

の疑問記事に対して, 3.3.2 項で述べた質問文を用いた NDC 記号の自動付与手法を利用して NDC 記号の 1 桁目を付与する。NDC 記号の自動付与の詳しい結果については 4.2 節で述べるが，NDC 記号の自動付与を行う際に最も F 値の高かった SVM を機械学習アルゴリズムとして用いる。また 3.3.2.1 項で述べた提案手法を用いて特徴ベクトルを作成する。機械学習を行う際の学習用コーパスとしては，レファレンス協同データベースのうちの NDC 記号が付与されている事例 40,288 件の質問文とその NDC 記号を用いる。

3.4.4 実験の評価方法

3.4.3 項の方法でツイートの疑問記事に対して人手で NDC 記号を付与し, それらのツイートに NDC 記号の自動付与を行うことで精度と再現率を算出し，自動付与性能を評価する。

3.5 3 章のまとめ

本章では，自動的に疑問記事を収集し分類して提示する Web サイトの構築するための提案手法について述べた。

まず，疑問記事を提示する Web サイトの有用性について予備調査 (3.1 節) を行い，その結果以下の 3 点を確認した。即ち，(i) 疑問記事が Web 上に一定量存在する，(ii) 疑問記事に書かれた疑問はレファレンスの専門家でなくとも回答可能である，(iii) ブログ著者は疑問記事に対する回答を好意的に受けとめる，の 3 点である。

次に，この Web サイトの構築における課題を解決するための 2 つの提案手法について述べた。2 つの課題とは即ち，(1) 疑問記事の自動的な収集手法 (3.2 節)，(2) 収集した疑問記事の自動的な分類手法 (3.3 節)，である。(1) では，2 つのステップによって疑問記事を自動的に収集する手法を提案した。即ち，(1-1) 特徴的な表現による検索，(1-2) テキスト自動分類による抽出，である。(1-1) は疑問記事にのみ頻繁に現れる特徴的な表現を特定するステップ，(1-2) は検索された特徴的な表現が現れる記事をテキスト自動分類によって疑問記事と非疑問記事に分類するステップである。(2) では，レファレンス協同データベースの事例の参考資料と質問文を用いて事例に対して NDC 記号を自動付与する手法を提案した。この提案手法を用いることで，質問文のような自然言語で書かれた疑問文に対しても，効率的に機械学習を行えるような特徴ベクトルを作成することができる。

最後に，上記の (1), (2) の 2 つの提案手法を用いて，実際に疑問記事を自動的に収集し，収集した疑問記事に NDC 記号を自動付与する実験の手順 (3.4 節) について述べた。

4　結果と考察

本研究では，以下の 3 つの実験を行った。即ち，(1) 疑問の書かれた記事の効率的な収集 (4.1 節)，(2)レファレンス事例に対する NDC 記号の自動付与 (4.2 節)，(3) 疑問記の自動収集および NDC 記号の自動付与 (4.3 節)，である。本章ではそれぞれの結果とその考察を述べる。

4.1　疑問記事の効率的な収集

本研究では，疑問記事を効率的に収集するために，特徴的な表現による検索 (4.1.1 項) によって得られた記事に対して, テキスト自動分類による疑問記事の抽出 (4.1.2 項) を行った。さらに収集した疑問記事を提示する Web サイトの有効性調査 (4.1.3 項) を行った。以下の項ではそれぞれの結果を示す。

4.1.1　特徴的な表現による検索

本項では特徴的な表現による検索の結果 (4.1.1.1 項) と考察 (4.1.1.2 項) について述べる。

4.1.1.1　特徴的な表現による検索結果

表 2 ~ 表 5 に特徴度 F(w) (3.2.1.1 項の (1) 式) の高かった特徴的な表現とその疑問記事率 (3.2.1.3 項の (2) 式) を示す。表 2 から，最も特徴度 F(w) の高い特徴的な表現 w「が_思い出せ_ない」の Twitter 検索における疑問記事率は 2%であることがわかる。ここで表 3, 表 4, 表 5 に現れる「EOS」は改行を表す。本研究では改行も一つの形態素として扱った。この理由は改行自体が疑問記事に特徴的な表現に含まれている可能性を考えたためである。

Twitter において最も疑問記事率が高かったのは「タイトル_が_思い出せ_ない」で疑問記事率(T) は 13%であった (表 3)。ブログにおいて最も疑問記事率が高かったのは「タイトル_が_思い出せ_ない」で, 疑問記事率(G) は 16%, 疑問記事率(Y) は 19%であった (表 5)。

表 2 Twitter における特徴的な表現 (3 形態素列)

	特徴的な表現 w	特徴度 F(w)	疑問記事率(T)(%)
1	が_思い出せ_ない	0.01392	2
2	タイトル_が_思い出せ	0.01347	11
3	思い出せ_ない_。	0.01055	1
4	の_タイトル_が	0.00606	2
5	ん_だけ_ど	0.00516	0
6	な_い_。	0.00336	0
7	思い出せ_ない_…	0.00314	1
8	が_あっ_た	0.00247	0
9	だけ_ど_、	0.00247	0
10	な_ん_だ	0.00247	0

表 3 Twitter における特徴的な表現 (4 形態素列)

	特徴的な表現 w	特徴度 F(w)	疑問記事率(T)(%)
1	タイトル_が_思い出せない	0.01325	13
2	が_思い出せない_。	0.00696	2
3	の_タイトル_が_思い出せ	0.00561	7
4	思い出せ_ない_。_EOS	0.00292	1
5	ん_だけ_ど_、	0.00201	1
6	、_タイトル_が_思い出せ	0.00201	11
7	ん_だけ_ど_タイトル	0.00201	4
8	が_思い出せ_ない_…	0.00201	2
9	映画_の_タイトル_が	0.00179	12
10	だけ_ど_、_タイトル	0.00157	0

表 4 ブログにおける特徴的な表現 (3 形態素列)

	特徴的な表現 w	特徴度 F(w)	疑問記事率(G)	疑問記事率(Y)
1	。_EOS_EOS	0.00248	0	0
2	ん_だけ_ど	0.00142	0	0
3	・_・_・	0.00139	0	0
4	が_思い出せ_ない	0.00107	0	0
5	た_。EOS	0.00103	0	0
6	て_い_た	0.00080	0	0
7	た_よう_な	0.00074	0	0
8	だけ_ど_、	0.00073	0	0
9	タイトル_が_思い出せ	0.00070	7	4
10	ん_です_が	0.00068	0	0

表 5 ブログにおける特徴的な表現 (4 形態素列)

	特徴的な表現 w	特徴度 F(w)	疑問記事率(G)	疑問記事率(Y)
1	た_。_EOS_EOS	0.00144	0	0
2	タイトル_が_思い出せ_ない	0.00067	16	19
3	ん_です_が_、	0.00051	0	0
4	な_ん_だけ_ど	0.00049	0	0
5	ん_だけ_ど_、	0.00047	0	0
6	だっ_た_か_な	0.00040	0	0
7	思い出せ_ない_。_EOS	0.00039	0	0
8	た_ん_だけ_ど	0.00038	0	0
9	・_・_・_。	0.00038	0	0
10	だっ_た_よう_な	0.00038	0	0

4.1.1.2 特徴的な表現による検索に関する考察

予備実験においてキーワード「タイトル　思い出せない」を用いて検索したときには，Google ブログ検索，Yahoo!ブログ検索のいずれにおいても疑問記事率が 2%であった。「タイトル_が_思い出せ_ない」という表現を用いる事で，疑問記事率は Twitter では 13%，ブログでは 16%～19%に向上した。このことから，本研究の特徴的な表現を用いることで，疑問記事の収集効率を向上させられたと言える。

Twitter における特徴度 F(w)が高い 3 形態素列，4 形態素列として「が_思い出せ_ない」(表

2),「が_思い出せ_ない。」(表 3) がある。これらの疑問記事率は共に 2%で，先ほどの「タイトル_が_思い出せ_ない」に比べるとはるかに疑問記事率が低かった。これらのキーワードは「タイトル」という語を含まないため，本研究で収集したい「タイトルに言及する記事」をヒットさせる能力が低いものと思われる。表 2 の「の_タイトル_が」や，表 3 の「ん_だけ_ど_タイトル」,「だけ_ど_、_タイトル」はタイトルが含まれているにも拘らず，疑問記事率が低い。この原因として，これらは「思い出せない」という語を含まないため，疑問記事をヒットさせる能力が低いことが考えられる。

キーワード検索は複数のキーワードを組み合わせて行うことができる。4.1.1.1 項で得られた特徴的な表現を組み合わせたものをキーワードとして用いることで，精度を向上させることができる可能性がある。そこで，表 5 から「タイトル」,「思い出せない」が含まれている「タイトル_が_思い出せ_ない」,「思い出せ_ない_。_改行」とどの文章にも含まれている可能性が高い「た_。_改行_改行」,「・_・_・_。」の 4 つを除いた 6 つの特徴的な表現と「タイトルが思い出せない」をキーワードとして組み合わせ, Google ブログ検索で AND 検索を行った (表 6)。

表 6 表 5 の特徴的な表現と「タイトルが思い出せない」を組み合わせた際の疑問記事率

	特徴的な表現 w	特徴度 F(w)	疑問記事率(G)
3	ん_です_が_、	0.00051	6
4	な_ん_だけ_ど	0.00049	6
5	ん_だけ_ど_、	0.00047	14
6	だっ_た_か_な	0.00040	4
8	た_ん_だけ_ど	0.00038	9
10	だっ_た_よう_な	0.00038	13

その結果最も精度が高かったのは「タイトルが思い出せない」と「んだけど,」を組み合わせた場合の精度 14%であった。即ち，特徴的な表現を組み合わせても「タイトルが思い出せない」を単独で用いた場合の精度 16%を上回ることはなかった。なお,「タイトルが思い出せない」と他の表現を組み合わせた疑問記事率は以上のような結果となった。

4.1.2　テキスト自動分類による抽出

本項では，記事群 A におけるテキスト自動分類による抽出の結果 (4.1.2.1 項) と考察 (4.1.2.2 項) ，記事群 B におけるテキスト自動分類による抽出の結果 (4.1.2.3 項) と考察 (4.1.2.4 項) について述べる。

4.1.2.1　記事群 A におけるテキスト自動分類による抽出結果

記事群 A に対してテキスト自動分類を行った場合の精度，再現率，F 値を表 7 に示す。例えば表 7 から，決定木の 2 形態素列によって分類を行った場合の精度は 71.4%，再現率は 70.3%，F 値は 0.709 であることが分かる。

表 7　テキスト自動分類の結果 (記事群 A)

機械学習手法	媒体	形態素数	精度(%)	再現率(%)	F 値
決定木	ブログ	1	72.6	75.7	0.740
		2	71.4	70.3	0.709
		3	73.2	72.0	0.726
ブースティング	ブログ	1	68.3	82.7	0.748
		2	68.3	82.3	0.747
		3	68.2	82.0	0.745
Naive Bayes	ブログ	1	71.5	69.3	0.704
		2	73.4	69.0	0.709
		3	71.2	63.3	0.666
SVM	ブログ	1	74.5	77.7	0.759
		2	73.8	78.0	0.749
		3	72.4	75.3	0.738

最も精度，F 値が高かったのは SVM の 1 形態素によってテキスト自動分類を行った場合であり，精度 74.5%，再現率 77.7%，F 値 0.759 であった。また最も再現率が高かったのは SVM の 1 形態素ではなくブースティングの 1 形態素によってテキスト自動分類を行った場合であり，再現率 82.7%であった。

4.1.2.2　記事群 A におけるテキスト自動分類による抽出に関する考察

記事群 A におけるテキスト自動分類において，最も再現率が高かったのはブースティングの 1 形態素によってテキスト自動分類を行った場合であった。この結果を確かめるためにブースティングによる分類の結果を細かく見たところ，ブースティングによる分類では 300 記事のすべての記事が非疑問記事と判定されていた。疑問再現率 (3.2.2.3 項の (5) 式，52 件，0%) と非疑問再現率 ((6) 式，248 件，100%) の加重平均を取るために再現率が高い ((0×52+100×248) / 300 = 82.7%) ように見えるが，実際の疑問再現率は上記のように 0%である。このことは，他の手法に比べてブースティングでの精度が低い原因にもなっている。同様に SVM による分類でも，再現率は比較的高い (77.7%) が，疑問再現率は 19.2% (＝疑問記事のうち正しく判定された疑問記事数 10 件／疑問記事数 52 件, 3.2.2.3 項) となっており，これでは本研究で収集したい疑問記事を効率よく抽出することは難しいと考えられる。

記事群 A に対する分類精度が悪い原因としては以下のような点が考えられる。即ち，(1) 疑問記事 (52 件) と非疑問記事 (248 件) のサンプル数が大きく異なる，(2) サンプル数 (全 300 件) が少ない，などである。記事群 B では全体のサンプル数を増やし，疑問記事と非疑問記事のサンプル数を揃えた。

4.1.2.3　記事群 B におけるテキスト自動分類による抽出結果

記事群 B に対してテキスト自動分類を行った場合の精度，再現率，F 値を表 8 に示す。例えば表 8 から，Twitter 記事に対して決定木の 2 形態素列によって分類を行った場合の精度は 92.2%，再現率は 92.1%，F 値は 0.921 であることが分かる。

ブログに関して最も精度，再現率，F 値が高かったのは，ブースティングの 1 形態素によってテキスト自動分類を行った場合であり，精度 94.8%，再現率 94.3%，F 値 0.943 であった。Twitter に関して最も精度，再現率，F 値が高かったのは，決定木の 1 形態素によってテキスト自動分類を行った場合であり，精度 94.3%，再現率 94.3%，F 値 0.941 であった。このように，提案手法によって高い精度・再現率・F 値で疑問記事を抽出することができた。

表 8 テキスト自動分類の結果 (記事群 B)

機械学習手法	媒体	形態素数	精度(%)	再現率(%)	F 値
決定木	ブログ	1	93.1	93.0	0.930
		2	88.1	88.0	0.880
		3	82.5	82.3	0.823
	Twitter	1	94.3	94.3	0.941
		2	92.2	92.1	0.921
		3	91.9	91.6	0.916
ブースティング	ブログ	1	94.8	94.3	0.943
		2	91.3	90.7	0.906
		3	85.3	81.3	0.808
	Twitter	1	93.9	93.6	0.936
		2	93.0	92.6	0.926
		3	91.2	91.0	0.910
Naive Bayes	ブログ	1	77.7	76.7	0.764
		2	77.5	75.7	0.752
		3	76.7	76.3	0.762
	Twitter	1	88.6	88.5	0.884
		2	91.6	90.8	0.908
		3	91.2	90.9	0.909
SVM	ブログ	1	91.0	91.0	0.910
		2	91.9	91.7	0.917
		3	90.1	90.0	0.900
	Twitter	1	93.4	93.4	0.934
		2	93.2	93.1	0.931
		3	93.1	92.9	0.929

機械学習手法ごとの違いを比較すると，決定木，ブースティングに関しては，特徴量として用いる形態素数が少ないほど分類性能が良い傾向が見られたが，Naive Bayes，SVM に

関しては，特徴量として用いる形態素数によって分類性能が大きく変わることはなかった。

媒体ごとの違いを比較すると，ブログ記事に関しては決定木，ブースティングにおいて，形態素数が小さいほど分類性能が良い傾向が見られた。しかし Twitter 記事に関しては形態素数による分類性能の変化はなかった

4.1.2.4 記事群 B におけるテキスト自動分類による抽出に関する考察

本項では，3 つの視点から本研究におけるテキスト自動分類の性能について考察する。3 つの視点とは即ち，(a) 疑問記事の抽出性能，(b) 機械学習手法による分類性能の違い，(c) 媒体による分類性能の違い，である。最後にいくつかの文書を無作為に抽出し，テキスト自動分類の結果誤分類された記事に関して誤り分析を行った結果を示す。

(a) 疑問記事の抽出性能

4.1.1.1 項で述べたようにブログに関して特徴的な表現を使った検索だけでは，19%の疑問記事率で，Twitter に関しては 13%の疑問記事率で疑問記事を得られることが分かった。しかし，例えば 100 件のブログ記事から Web サイトにアップする疑問記事を選ぶには検索結果の中に含まれてしまう残りの 81 件の非疑問記事にも目を通さねばならない。しかし上記テキスト自動分類を記事選択の第 2 段階として用いることで，その数は大幅に減らすことができる。

例えばブログに対してブースティングの 1 形態素を用いた場合，その疑問再現率 (表 9) は 89.4% (＝疑問記事のうち正しく判定された疑問記事数 134 件／疑問記事数 150 件)，非疑問再現率 (表 10) は 99.3% (=非疑問記事のうち正しく判定された非疑問記事数 149 件／非疑問記事数 150 件) であった (3.2.2.3 項の (5) 式, (6) 式)。そのため 17 件 (16.9 = 19×0.894) の疑問記事が出力される一方，非疑問記事は 0 件から 1 件 (0.567 = 81×(1－0.993)) しか出力されない。従って例えば検索だけでは 100 件に目を通さねばならなかったのが，テキスト自動分類によって 17 件の疑問記事と 0 件から 1 件の非疑問記事に目を通し，疑問記事を選び出すだけで済むようになる。コストの削減率はかなり大きいと言えよう 。

同様の条件において Twitter に対して決定木の 1 形態素を用いた場合，その疑問再現率は 94.4%，非疑問再現率は 94.1%であった。そのため 12 件 (12.2 = 13×0.944) の疑問記事が出力される一方，非疑問記事は 5 件 (5.133 = 87×(1－0.941)) しか出力されない。ブログ同様，

表 9 記事群 B における疑問精度，疑問再現率，疑問 F 値

機械学習手法	媒体	形態素数	精度(%)	再現率(%)	F 値
決定木	ブログ	1	95.1	90.7	0.929
		2	89.7	86.1	0.878
		3	85.0	78.8	0.818
	Twitter	1	94.2	94.4	0.943
		2	90.1	94.8	0.924
		3	88.5	95.8	0.92
ブースティング	ブログ	1	99.3	89.4	0.941
		2	96.2	84.8	0.901
		3	97.0	64.9	0.778
	Twitter	1	90.7	97.3	0.939
		2	88.7	97.6	0.929
		3	88.4	94.5	0.913
Naive Bayes	ブログ	1	72.4	86.8	0.789
		2	70.5	88.7	82.8
		3	73.5	82.8	0.779
	Twitter	1	86.2	91.7	0.888
		2	86.0	97.6	0.914
		3	87.8	95.0	0.913
SVM	ブログ	1	91.3	90.7	0.910
		2	95.0	88.1	87.4
		3	92.3	87.4	0.898
	Twitter	1	91.8	95.3	0.953
		2	91.5	95.1	0.933
		3	90.4	96.0	0.931

表 10　記事群 B における非疑問精度，非疑問再現率，非疑問 F 値

機械学習手法	媒体	形態素数	精度(%)	再現率(%)	F 値
決定木	ブログ	1	91.0	95.3	0.931
		2	86.5	89.9	0.882
		3	80.0	85.9	0.823
	Twitter	1	94.4	94.1	0.942
		2	94.5	89.5	0.919
		3	95.4	87.5	0.913
ブースティング	ブログ	1	90.2	99.3	0.946
		2	86.2	96.6	0.911
		3	73.4	98.0	0.839
	Twitter	1	97.0	90.0	0.934
		2	97.4	87.5	0.922
		3	94.1	87.5	0.907
Naive Bayes	ブログ	1	83.2	66.4	0.739
		2	84.5	62.4	0.718
		3	80.0	69.8	0.746
	Twitter	1	91.0	85.2	0.880
		2	97.2	84.0	0.901
		3	94.5	86.7	0.905
SVM	ブログ	1	90.7	91.3	0.910
		2	88.8	95.3	0.919
		3	87.9	92.6	0.902
	Twitter	1	95.1	91.5	0.932
		2	94.9	91.1	0.93
		3	95.7	89.7	0.926

コストの削減率はかなり大きいと言える。本研究が行ったようにサーチエンジンに「タイトルが思い出せない」といった疑問記事に特徴的な表現を入力することで得られる記事に

対しては，本手法は十分有効であると言える。

決定木手法はどの要素が分類に大きく寄与したかが，分析者にわかりやすいという利点を持つ。そこで決定木の 3 形態素列によって分類を行った結果を詳しく見たところ，最も疑問記事の分類に有効な特徴量はブログ, Twitter のいずれにおいても「が_思い出せ_ない」であった。この「が_思い出せ_ない」は 4.1.1.1 項でも特徴的な表現として抽出されていたが，そこで述べたように疑問記事をヒットさせる能力は低かった。このことから，分類に有効な特徴量とサーチエンジンで疑問記事を効率的にヒットさせる特徴的な表現は，重なりはあるかもしれないものの完全に一致するわけではないことがわかった。

(b) 機械学習手法による分類性能の違い

前述したが，Naive Bayes，SVM に関しては，特徴量として用いる形態素数によって分類性能が大きく変わることはなかった (表 8)。表 11 は記事群 B における，延べ形態素数と異なり形態素数を連続する形態素列数ごとに算出したものである。

表 11 コーパスの延べ形態素数と異なり形態素数

	延べ形態素数		異なり形態素数	
形態素数	ブログ	Twitter	ブログ	Twitter
1	131,070	58,482	14,148	9,137
2	137,979	58,483	62,551	31,166
3	137,679	56,882	102,452	44,234

特徴量として用いる形態素列を長くすると，異なり形態素数が増える。その結果として各語の出現頻度は減ることになる。このことから，特徴量として用いる形態素列の長さによって分類性能の変わった決定木，ブースティングは，異なり形態素数が増えることで性能が悪化する分類方法であることが考えられる。また分類性能の変わらなかった Naive Bayes，SVM は，異なり形態素数が増えることによる影響をあまり受けない分類方法であることが考えられる。異なり形態素数が増えるということは入力する次元数，即ちベクトルの要素数が増えることを意味する。故に今回実験に用いたコーパスに関して，Naive Bayes, SVM は過学習の影響をあまり受けていないと考えられる。マージンを最大化するという条

件で分類を行う SVM に関して，過学習の影響が少ない点 (2.3.1.2 項) は，先行研究の結果と一致する。

(c) 媒体による分類性能の違い

ブログ記事に関しては決定木，ブースティングにおいて，形態素数が小さいほど分類性能が良い傾向が見られた。しかし Twitter 記事に関しては形態素数による分類性能の変化はなかった (表 8)。

Naive Bayes は他の手法と異なり，Twitter 記事に比べてブログ記事において分類精度を顕著に下げている。この理由としては，ブログ記事は Twitter 記事より 1 記事あたりの文章量が多いことが影響していることが考えられる。表 11 に示したように，1 形態素の全延べ語数はブログでは 131,070 語，Twitter では 58,482 語である。1 記事あたりに換算するとブログは 873.8 語, Twitter は 73.1 語となり, 前者は後者に比べて 10 倍以上の語を含むことになる。その分 Naive Bayes にとって分類上のノイズとなる語が多く含まれていた可能性が高い。一方，例えば SVM は Naive Bayes と異なり，ノイズによる過学習を起こしにくい。そのためブログ記事と Twitter 記事とで分類性能が変わらなかったと考えられる。

検証のため，Naive Bayes を用いて以下の分類実験を行った。即ち，ブログに書かれた疑問記事 150 記事と非疑問記事 150 記事のそれぞれに対して 1 記事あたりに含まれる形態素数の中央値を特定し，中央値以上であった記事を「ブログ Long」，中央値未満の記事を「ブログ Short」，同様に Twitter に書かれた疑問記事 800 記事と非疑問記事 800 記事も中央値を境として「Twitter Long」と「Twitter Short」に分け，それぞれで分類実験を行った。結果は以下のようになった (表 12)。

表 12 Naive Bayes での文章の長さごとの分類性能

媒体	文章の長さ	精度(%)	再現率(%)	F 値
ブログ	Long	75.3	75.2	0.751
	Short	83.0	82.7	0.826
Twitter	Long	87.6	87.5	0.875
	Short	90.4	90.3	0.903

表 12 からブログ記事, Twitter 記事ともに延べ語数が小さい記事で分類性能が良いことが分かる。このことから長い記事中には Naive Bayes にとってノイズとして機能する語が多く含まれ，それが分類精度を下げていることが考えられる。

最後に，テキスト自動分類の結果において観察された誤分類について述べる。ブログに関しては，疑問記事 30 個のうち 2 個で判定を誤り，それ以外の記事 30 個のうち判定を誤ったものはなかった。ここで判定を誤った 2 つの記事を x_1，x_2 とする。記事 x_1 は，本のあらすじを箇条書きで書いた記事である。即ち，4.1.1.1 項で求めた特徴的な表現である「んですが,」,「なんだけど」,「だったかな」のような表現が一つも含まれていない記事である。この記事には「タイトルが思い出せない」という表現は含まれていたが，それ以外の特徴的な表現が含まれていなかったため，疑問記事として判定できなかったと考えられる。記事 x_2 は記事内に「タイトル」という単語が現れない記事である。本研究で用いたコーパスでは疑問記事に「タイトル」という単語が多く含まれている。そのために疑問記事として判定できなかったと考えられる。

Twitter に関しては，疑問記事 100 個のうち判定を誤ったものはなかったが，それ以外の記事 100 個のうち 4 個で判定を誤った。これらはいずれも記事中に「タイトルが思い出せない」という表現が入っているがそれ以外の記事に属している記事であった (例えば「だめだ、乗ってねえ、誰かレビュー書いてよ・・　タイトルが思い出せない」「聞いてる曲のタイトルが思い出せない時って良くあるよね」など)。Twitter は文字数が 140 字以下に限定されるため，疑問記事に多く現れる表現が 1 つでも現れるとその影響は大きく，判定を誤る可能性が考えられる。

4.1.3　構築した Web サイトの有効性調査

4.1 節の結果として収集した疑問記事を提示する Web サイト (図 3) を実際に構築し，その有効性を確認した。この Web サイトには，ブログと Twitter に関して，収集した各疑問記事の本文とリンクと日付を記述した。また，ハイライトを使い，疑問の書かれた部分を見分けやすくした。今回構築したような Web サイトでは，情報の新鮮さや閲覧者に注目してほしい部分を強調するのは特に重要と考えられる。有効性の評価は，Web サイトに疑問記事としてブログ 30 記事，Twitter31 記事を掲載し，広く回答を呼び掛け [12]，その結果各記事

[12] 呼び掛けは Twitter 上で多数のフォロワーを持つ知人に依頼する形で行った。

にどの程度回答が付いたか，付いた回答に対して感謝のコメントが疑問記事著者から付いたかを調べることで行う。ブログについては疑問記事のコメント欄を随時チェックすることで調べ，Twitter に関しては答えを書いた者へのリプライがないかをチェックすることで回答が付いたか否かを調べた。

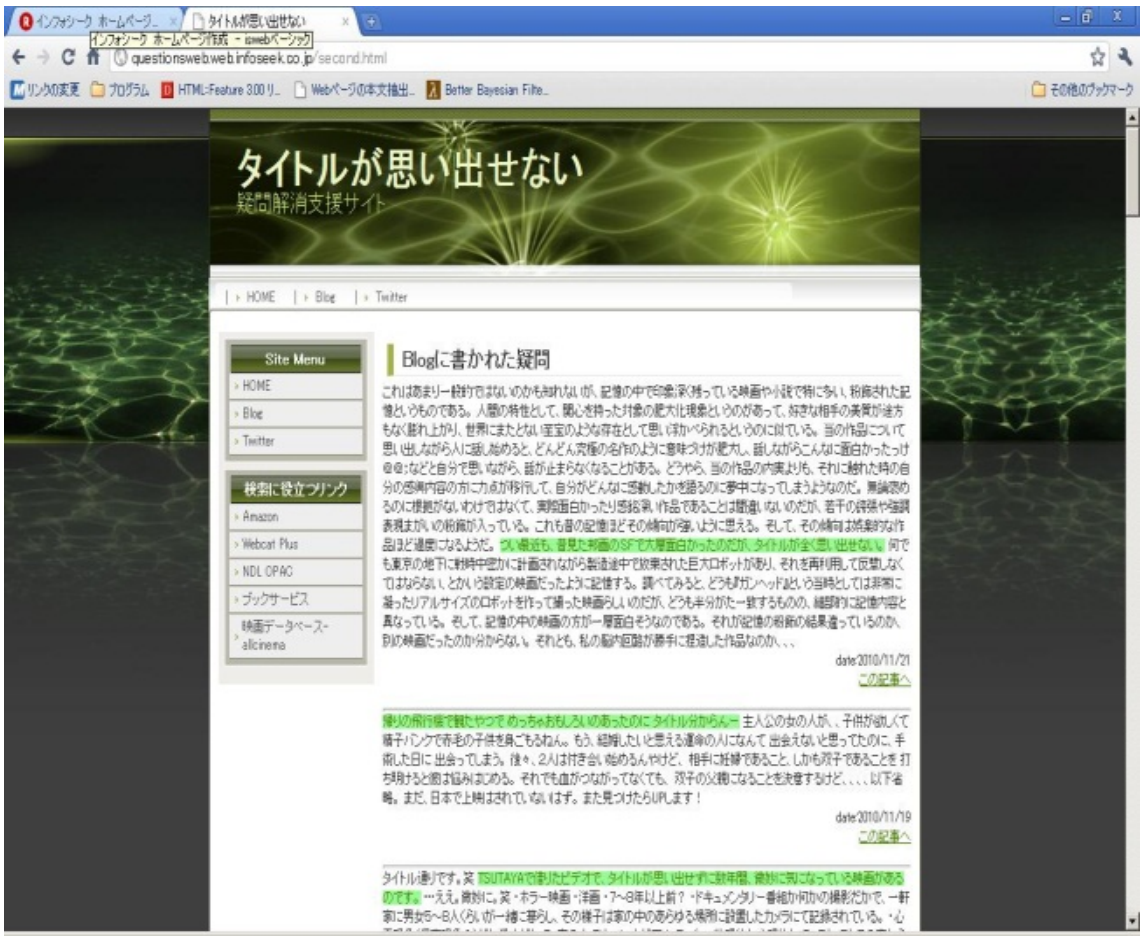

図 3　構築した Web サイト

その結果，2010 年 12 月 9 日から 2011 年 2 月 9 日の間に 1,251 人の利用者から 1,511 回のアクセスがあり，ブログに関しては 30 記事中 6 記事，Twitter に関しては 31 記事中 5 記事に回答が付いた。

回答が付いた記事の中で著者からの感謝のコメントがあったのは，ブログに関しては 6 記事中 2 記事，Twitter に関しては 5 記事中 5 記事であった。ブログに関して，感謝のコメントが付いていない記事は，書かれてから 1 カ月以上経った古い記事であった。そのため

著者は既にその記事に関して関心を失っているとも考えられる。Twitter に関しては全ての提示した記事が書かれてから 1 週間以内の記事であったためか，回答が付いた記事には必ず感謝のコメントが返されていた。ここで「感謝」の定義であるが，感謝か否かは筆者の主観で判断している。具体的には，「教えてくれてありがとう」「そうそう，それだ！」といった書き込みは感謝とみなした。ちなみにブログや Twitter の著者の書き込みの中で，感謝ではないと判断されたものはなかった。

Q&A サイトやデジタルレファレンスサービスに回答を依頼した場合に比べると回答率が低いが (3.1 節参照)，これには以下の 3 つの理由が考えられる。即ち，(1) 呼び掛けが十分でなく，まだ本 Web サイトの知名度が低いため，図書館員など回答を知る人を誘導できていない，(2) 回答を知ってはいるが既に答えが出ているかもしれないと考えた人が回答を控えている，(3) 答えを知っていても回答するメリットが少ないと考える者がいる，の 3 つである。(1) に関しては，本 Web サイトの知名度を上げ，役に立つコンテンツを設置することで解決できると考えられる。(2) に関しては，回答が付いた疑問記事は 1 週間程度で本 Web サイトから削除し，かつそのような削除方針を採っていることを Web サイト中に明記することで，解決できると考えられる。(3) に関しては，本 Web サイトは Q&A サイトの利用者のように無償で回答してくれる人に頼ったものである為，経済的金銭的メリットを打ち出すのは難しいが，例えば Yahoo!知恵袋の知恵コインのようなインセンティブをこちらで設定するなどして回答のモチベーションを高めることなどが考えられる。

4.2 レファレンス事例に対する NDC 記号の自動付与

本節では疑問記事を提示する Web サイトでディレクトリ型検索を実現する為に行った，事例に NDC 記号を自動付与する実験結果を示す。本研究では以下の 3 つの手法を用いて事例に対する NDC 記号の自動付与を行った。即ち，(1) 参考資料を用いた手法 (4.2.1 項)，(2) 質問文を用いた手法 (4.2.2 項)，(3) 参考資料と質問文の両方を用いた手法 (4.2.3.1 項)，である。

4.2.1 参考資料を用いた NDC 記号の自動付与

本項では参考資料を用いた NDC 記号の自動付与の結果(4.2.1.1 項) と考察 (4.2.1.2 項) に

ついて述べる。

4.2.1.1 参考資料を用いた NDC 記号の自動付与結果

表 13 にベースライン手法である原田ら (2007) の手法を用いた NDC 記号の自動付与の結果を示した。以下ではこのベースライン手法の結果と提案手法の結果を比較することで, 提案手法の評価を行う。1 桁目 (類) までの NDC 記号の自動付与の結果を表 14 に示す。この表から事例の類が 3 の場合に関する精度は 40.6%, 再現率は 69.1%であることなどが分かる。

表 13 原田ら (2007) の手法による自動付与結果

	精度(%)	**再現率(%)**
類 (1 桁目)	44.9	45.0
綱 (2 桁目)	32.4	32.4

表 14 NDC 記号の自動付与結果 (類)

NDC	**精度(%)**	**再現率(%)**	**件数(個)**
0	35.5	16.1	2,189
1	51.8	50.2	1,308
2	58.8	54.5	5,300
3	40.6	69.1	4,589
4	59.6	53.4	1,617
5	61.5	44.1	1,729
6	54.6	44.8	1,393
7	64.2	52.2	2,207
8	57.4	55.8	754
9	62.8	55.7	2,246
平均	53.4	52.0	

この実験における全 NDC 記号の精度の加重平均 (3.3.1.4 項の (12) 式) は 53.4%であり，再現率の加重平均 ((13) 式) は 52.0%であった。参考資料の NDC 記号を用いることで約半数の事例において学習用サンプルと同じ NDC 記号を付与できた。原田らの (2007) の手法 (表 13) の精度と比較すると，本研究手法によって精度を 8.5 ポイント向上させることができた。

次に 2 桁目 (綱) までの NDC 記号の自動付与の結果を表 15，表 16 に示す。表 15 には実験によって得られた精度が高い NDC 記号上位 10 件を示し，表 16 には事例件数の多い NDC 記号上位 10 件を示す。

表 15 NDC 記号の自動付与結果 (綱，精度上位)

NDC	精度(%)	再現率(%)	全件数(個)
65	88.9	21.6	74
92	81.4	41.2	308
82	80.0	11.1	72
60	77.8	10.3	68
79	75.8	29.4	85
71	75.0	3.4	87
14	73.3	16.4	67
34	70.0	7.4	94
41	66.7	9.1	44
69	66.7	3.2	62

この実験における精度の加重平均は 45.6%であり，再現率の加重平均は 36.1%であった。半数程度の場合に関して，2 桁目 (綱) まで含む場合においても学習用サンプルと同じ NDC 記号を付与できた。表 13 の綱 (2 桁目) から，原田ら (2007) の手法による精度，再現率はどちらも 32.4%であった。従って本研究手法は，2 桁目 (綱) までの NDC 記号の自動付与においても，原田ら (2007) の手法より精度を 13.2 ポイント向上させることができた。

表 16　NDC 記号の自動付与結果 (綱，件数上位)

NDC	精度(%)	再現率(%)	全件数(個)
21	13.6	63.0	2,108
28	40.8	39.4	1,601
91	59.8	55.2	1,552
38	51.2	40.7	953
29	37.1	37.8	943
02	28.9	8.4	784
37	56.1	52.2	760
09	3.4	0.1	682
32	60.0	41.8	625
31	46.8	33.0	621

4.2.1.2　参考資料を用いた NDC 記号の自動付与に関する考察

本項では参考資料を用いた NDC 記号の自動付与を行うために本研究で提案する 2〜4 のステップ (3.3.1 項参照) のそれぞれについて考察を行う。2〜4 のステップとは即ち，(a) 図書のタイトルの抽出，(b) 図書のタイトルからの NDC 記号の判定，(c) NDC 記号の自動付与，である。

(a) 図書のタイトルの抽出

本研究では以下の 2 つのステップによって事例の参考資料の項目からの図書のタイトル部分の抽出を行った。

(i)『』で囲まれた部分を図書のタイトルとみなす。

(ii)『』がない場合，【資料 1】などのノイズを除き，半角スペース，カンマで区切ったときの最初の区切りまでを図書のタイトルとみなす。

(i) の『』で囲まれた部分を図書のタイトルとみなした場合については，全 32,657 件中 15,905 件の参考資料の項目に『』が含まれていた。この 15,905 件の中から無作為に 500 件

を抽出し誤り分析を行ったところ，以下の 2 件 (0.4%) に関して図書のタイトルを抽出できていなかった。

・"「コワレフスキー少将実在せず」(『産経新聞　2010/10/19　3 面』)"
・"中国医学古典と日本 : 書誌と伝承 / 小曽戸洋著　塙書房　1996　ISBN:4827311420 p.346～　第三節　『小品方』"

1 つ目に関しては参考資料として挙げられていたものが新聞記事であるため，図書のタイトルが文の中に含まれていない。そのため図書のタイトルを抽出できなかった。2 つ目に関しては図書のタイトルではなく小品方という節のタイトルが『』で囲われていたため抽出できなかった。

(i) で扱わなかった残りの 16,752 件のうち「http」の文字列が含まれる参考資料の項目を除外した 15,465 件に対して，(ii) の条件によって『』がない場合，【資料 1】などのノイズを除き，半角スペース，カンマで区切ったときの最初の区切りまでを図書のタイトルとみなした。こうして得られた本のタイトルの中から無作為に 500 件抽出し誤り分析を行った結果，以下の例を含む 17 件 (3.4%) に関して図書のタイトルを抽出できていないことがわかった。

・"図説　成田の歴史　普及版成田市史編集委員会編 成田市 1994.12 L210"
・"日本教育方法学会編．現代教育方法事典．図書文化社．2004．p.293 に記述がある．"
・"館内文献調査"

1 つ目に関しては，図書のタイトルの中の図説の部分のみが抽出されてしまい，2 つ目に関しては，日本教育方法学会編という編者が抽出されてしまった。また，図書のタイトルを抽出できていなかった 17 件中 11 件において，3 つ目の"館内文献調査"のようにテキスト中に図書のタイトルが含まれていなかった。以上の結果から，図書のタイトルを抽出できる事例の参考資料は，(i) のステップで約 15,841 件 (15,905×(1-0.004))，(ii) のステップで約 14,939 件(15,465×(1-0.034)) 程度存在すると考えられる。参考資料に http が含まれていない，即ち Web ページを参考資料に挙げていない参考資料全 31,370 件のうち，30,780 件 (約

97%) という高い割合で図書のタイトルを抽出できていると考えられる。

(b) 図書のタイトルからの NDC 記号の判定

この検索は 2013 年 5 月 10 日から 5 月 12 日の間に行った。図書のタイトルをキーワードとして国立国会図書館サーチで検索した場合，その検索結果には通常複数の図書が含まれる。本研究ではその中から先述のように最も Dice 係数の高い図書を選択し, その図書の NDC 記号を参考資料の NDC 記号として用いた (3.3.1.2 項)。このとき，図書の NDC 記号と事例の NDC 記号が必ずしも一致するとは限らず，その結果は以下の 4 つに分けられた。即ち，(A) 図書を正しく検索でき，その NDC 記号が事例の NDC 記号と一致する場合，(B) 図書を正しく検索できたが，その NDC 記号が事例の NDC 記号と一致しない場合，(C) 図書を正しく検索できておらず, 抽出された図書の NDC 記号が事例の NDC 記号と一致しない場合，(D) 図書を正しく検索できていないが, 抽出された図書の NDC 記号がたまたま事例の NDC 記号と一致してしまう場合，の 4 つである。ここで，事例の NDC 記号とその参考資料は，その事例を登録した担当職員によってあらかじめ決められているため，図書を正しく検索できたが NDC 記号が一致しない (B) の数を減らすことはできない。しかし，本研究ではここで得た NDC 記号をそのまま事例に付与するのではなく，事例の NDC 記号の特徴を表す特徴量として用いる。そのため(A)，(B) の数が十分に多ければ NDC 記号の自動付与という本研究の目的を達成できると考えられる。

事例の参考資料から無作為に 100 件を抜きだし NDC 記号の抽出を行い，(A)，(B)，(C)，(D) のそれぞれの該当数を集計した結果を表 17 に示す。

表 17　図書のタイトルからの NDC 記号の抽出結果

	A (%)	**B (%)**	**C (%)**	**D (%)**
1 回目	48	42	10	0
2 回目	49	46	5	0
3 回目	58	38	4	0
4 回目	45	46	9	0
5 回目	47	49	4	0
平均	49.4	44.2	6.4	0

偏りを軽減するためにこの実験は 5 回繰り返し行った。本研究ではテキスト自動分類による 2 桁目 (綱) までの NDC 記号の付与を行う。そのため，綱まで一致していれば，一致しているものとみなした。結果として，9 割を超える参考資料が (A)，(B) のタイプに含まれており図書を正しく検索できていることが分かった。(C), (D) に関してはそれぞれ 6.4%, 0.0%と非常に少なく，タイトルから図書を正しく検索できていると考えられる。

(c) NDC 記号の自動付与

1 桁目までの分類に関しては，各類でどの類を誤って付与するケースが多かったかを分析するために，自動付与の結果の詳細を表 18 に示す。表 18 から，例えば 0 類の事例は全部で 2,189 件存在し，うち 352 件で 0 類と正しく判定し，732 件で 3 類と誤って判定していることなどが分かる。全体的に 3 類と判定してしまう場合が多く，誤りのうちの約 40%がこの誤り方であった。これが，3 類が他の類に比べて再現率が高い原因であると考えられる。

表 18　NDC 記号の自動付与の結果 (類) の詳細

	実験により付与された NDC 記号											計
正解の NDC 記号	NDC	0	1	2	3	4	5	6	7	8	9	
	0	352	85	470	732	69	60	69	110	16	226	2,189
	1	51	657	189	252	24	7	6	44	36	42	1,308
	2	159	282	2,890	1,154	107	112	140	188	42	226	5,300
	3	139	81	536	3,170	155	132	101	117	82	76	4,589
	4	38	18	92	410	864	62	77	15	25	16	1,617
	5	50	13	178	473	81	763	93	45	20	13	1,729
	6	36	7	165	396	72	56	624	18	10	9	1,393
	7	47	51	189	553	27	36	20	1,152	30	102	2,207
	8	19	25	43	165	22	9	3	17	421	30	754
	9	101	50	165	500	29	3	9	87	52	1,250	2,246
計		992	1,269	4,917	7,805	1,450	1,240	1,142	1,793	734	1,990	23,332

2 桁目までの分類に関しては，表 16 の NDC 記号 21 における精度と全件数から，2 桁目

(綱) までを誤って 21 と判定してしまう場合が非常に多いことが分かる。これは 1 桁目 (類) までの実験と同様に，21 の再現率が高い原因であると考えられる。このことは全体の精度を下げる大きな原因となっている。NDC 記号 09 は「貴重書，郷土資料，その他の特別コレクション」であり，そこに含まれる参考資料同士の共通点が少ないことが予想される。これが NDC 記号 09 の件数が多いにもかかわらず，精度・再現率が共に非常に低い原因であると考えられる。NDC 記号 02 に関しても総記であるために，おそらく同様の理由で精度，再現率が共に低くなっている。NDC 記号 96 や NDC 記号 85 など，全件数が数個しかないものを中心に，1 つもその NDC 記号に判定されない場合が 32 個あった。これは学習用サンプル数の不足によると思われる。もしサンプル数を増やすことができればこの問題は解決すると考えられる。

4.2.2 質問文を用いた NDC 記号の自動付与

本項では質問文を用いた NDC 記号の自動付与の結果(4.2.2.1 項) と考察 (4.2.2.2 項) について述べる。

4.2.2.1 質問文を用いた NDC 記号の自動付与結果

質問文を用いた NDC 記号の自動付与 (3.3.2.2 項参照) を行った結果を表 19，表 20 に示す。本研究で評価のベースラインとして用いている原田ら (2007) の手法，即ち質問文に関して形態素の出現回数をベクトルの要素とした場合の精度と再現率も表 19 に同時に示してある。表 19 の「原田ら (2007) の手法」から，原田ら (2007) の手法を再現した場合の 1 桁目 (類) までの精度は 44.9%，再現率は 45.0%であることがわかる。また，4.2.1.1 項で述べた，参考資料を用いて NDC 記号の自動付与を行った場合の結果も「参考資料」に改めて示した。用いた機械学習手法である Random Forest，SVM，CNB のうち最も精度，再現率の高かった CNB の結果と，LIBSVM 3.12 で実装されている SVM によって計算した精度を示す。

3.3.2.1 項の最後で述べたように，本研究では形態素ベクトルを作成する際に以下の 3 つを用いた。即ち，(a) 事例の質問文の項目を用いた場合，(b) 事例のキーワードの項目を用いた場合，(c) 図書のタイトルを用いた場合，の 3 つである。以下にそれぞれの結果について考察を述べる。

表 19 NDC 記号の自動付与の結果 (1 桁目 (類))

	精度(%)	再現率(%)
原田らの手法	44.9	45.0
参考資料	53.4	52.4
質問文 CNB	59.7	58.9
質問文 SVM	58.8	61.0
キーワード CNB	54.2	53.8
キーワード SVM	54.2	54.5
図書のタイトル CNB	42.4	42.3
図書のタイトル SVM	42.9	43.5

表 20 NDC 記号の自動付与の結果 (2 桁目 (綱))

	精度(%)	再現率(%)
原田らの手法	32.4	32.4
参考資料	45.6	36.1
質問文 CNB	51.5	49.7
質問文 SVM	51.6	57.7
キーワード CNB	41.1	40.6
キーワード SVM	36.9	41.7
図書のタイトル CNB	29.1	29.3
図書のタイトル SVM	22.1	24.4

まず, (a) の質問文の項目を用いた場合だが, 表 19 の「質問文」の行より, 1 桁目 (類) までの自動付与で CNB を用いた場合，精度は 59.7%，再現率は 58.9%であり，SVM を用いた場合，精度は 58.8%，再現率は 61.0%であった。表 19 に示したように参考資料を用いた場合の精度は 53.4%，再現率は 52.4%である。即ち (a) の場合は参考資料を用いた NDC 記号の自動付与より高い精度で分類を行うことができた。また，表 20 の「質問文」の行より，

2 桁目 (綱) までの自動付与で CNB を用いた場合，精度は 51.5%，再現率は 49.7%であり，SVM を用いた場合，精度は 51.6%，再現率は 57.7%であった。表 20 に示したように参考資料を用いた場合の精度は 45.6%，再現率は 36.1%である。即ち，この場合も参考資料を用いた場合より高い精度で分類を行うことができた。ベースラインの原田ら (2007) の手法より精度，再現率共に約 14 ポイント向上させることができた。

次に，(b) のキーワードの項目を用いた場合だが，表 19 の「キーワード」の行より，1 桁目 (類) までの自動付与では CNB を用いた場合，精度は 54.2%，再現率は 53.8%であり，SVM を用いた場合, 精度は 54.2%, 再現率は 54.5%であった。また表 20 より 2 桁目 (綱) までの自動付与では CNB で精度は 41.1%，再現率は 40.6%であり，SVM で精度は 36.9%，再現率は 41.7%であった。この結果から，質問文を用いた場合よりは精度が下がったが，原田ら (2007) の手法と比較すると約 10 ポイント精度，再現率が向上した。

最後に，(c) の図書のタイトルを用いた場合の結果を示す。表 19 の「タイトル」の行より，1 桁目 (類) までの自動付与では CNB で精度は 42.4%，再現率は 42.3%であり，SVM で精度は 42.9%，再現率は 43.5%であった。即ち，この場合，ベースラインや参考資料を用いた場合よりも精度を向上させることはできなかった。

4.2.2.2 質問文を用いた NDC 記号の自動付与に関する考察

機械学習手法の違いを比較すると，1 桁目までの分類では，CNB と SVM の間に大きな差は見られなかった。しかし，2 桁目までの分類では質問文を用いて形態素ベクトルを作成した場合は，CNB より SVM の再現率が 8 ポイント高かった。

形態素ベクトルを作成するためのコーパスの違いを比較すると，質問文による精度，再現率が最も高く，図書のタイトルによる精度，再現率が最も低かった。本研究では事例とは異なる学習用コーパス，即ち図書のタイトルも学習用コーパスとして形態素ベクトルを計算することができた。しかし，事例ではなく図書のタイトルを学習用コーパスとして用いているため，平均としてみると分類性能が低くなってしまったと考えられる。

4.2.1.2 項の最後でも述べたが，NDC 記号の 2 桁目までの分類では，レファレンス協同データベースの中にその NDC 記号に対応する事例がほとんど存在しない場合がある。そのような NDC 記号は，事例件数の多い NDC 記号に引っ張られてしまうため，自動付与されづらい傾向がある。このような，もともと事例の中でも特にサンプル数の非常に少ない類や

綱 (例えば 96 や 85 など) だけを選択して，上記のような図書のタイトルで足りないサンプル数を補うことで，分類性能を向上させることができるかもしれない。この点は今後の課題としたい。

4.2.3 参考資料と質問文の両方を用いた NDC 記号の自動付与

本項では参考資料と質問文を用いた NDC 記号の自動付与の結果(4.2.3.1 項) と考察(4.2.3.2 項) について述べる。

4.2.3.1 参考資料と質問文の両方を用いた NDC 記号の自動付与結果

4.2.1 項，4.2.2 項の結果から，参考資料と質問文のどちらを用いても，原田ら (2007) の手法より高い精度で自動付与を行うことができたと言える。3.3.3.1 項で述べた通り，本研究では参考資料と質問文を同時に用いる方法において，以下の 2 つの方法で特徴ベクトルを作成した。即ち，(a) 参考資料と質問文から得られた特徴ベクトルを規格化し各要素について足し合わせる，(b) 参考資料と質問文から得られた特徴ベクトルを規格化し各要素を独立に並べる，という 2 つの方法である。それらの結果を表 21, 表 22 に示す。表中の「(a) の方法」「(b) の方法」にはそれぞれ上記の (a), (b) の方法を用いて, 機械学習手法である CNB, SVM のうち，最も精度，再現率の高い CNB の結果を示した。また，表 19，表 20 の中から同条件で行った実験結果について「キーワード」に改めて示した。

表 21 NDC 記号の自動付与の結果 (1 桁目 (類))

	精度(%)	**再現率(%)**
（a）の方法	51.8	51.7
（b）の方法	51.8	50.5
キーワード	54.2	53.8

表 22 NDC 記号の自動付与の結果 (2 桁目 (綱))

	精度(%)	**再現率(%)**
（a）の方法	41.0	39.8
（b）の方法	43.2	43.2
キーワード	41.1	40.6

4.2.3.2 参考資料と質問文の両方を用いた NDC 記号の自動付与に関する考察

表 21 の (a) の方法，(b) の方法の精度とキーワードを用いたときの精度を比較すると，NDC 記号の 1 桁目 (類) までの自動付与に関しては，精度，再現率は向上させることができなかった。しかし表 22 の (a) の方法，(b) の方法とキーワードを用いたときの精度を比較すると，NDC 記号の 2 桁目 (綱) までの自動付与に関しては，精度，再現率共に向上させることができている。2 桁目 (綱) までの自動付与において精度，再現率が向上した理由としては以下の点が考えられる。即ち，1 桁目 (類) までの自動付与において，各 NDC 記号の中でサンプル数の最も少ない NDC 記号は 8 (言語) で，754 個であった (表 14)。これに対して，2 桁目 (綱) までの自動付与においてはサンプル数が 10 個に満たない NDC 記号を持つ事例も多く，このような事例に対して正しい NDC 記号の自動付与を行うことは非常に難しい。参考資料と質問文を同時に用いることで，サンプル数を増やすことと同様の効果が得られたため，2 桁目 (綱) までの自動付与の精度，再現率が向上したと考えられる。

4.3 疑問記事の自動収集および NDC 記号の自動付与

本研究では 3.4 節で述べたように，3.2 節の手法を用いて「タイトル」をキーワードとして検索したツイートから疑問記事を抽出し，3.3 節の手法を用いて抽出した疑問記事に対して NDC 記号を自動付与する実験を行った。本節では，この実験の結果 (4.3.1 項) と考察 (4.3.2 項) について述べる。

4.3.1 疑問記事の自動収集および NDC 記号の自動付与結果

ツイートの疑問記事をテキスト自動分類によって抽出した結果，15,000 件のツイートから 118 件の疑問記事を抽出することができた (付録)。ただし，これらの 118 件のツイートは機械的に抽出されたものであり，これらの中には回答が不可能なツイートも存在する。例えば，非常に短いため何の疑問なのか分からないツイートや，自動分類では疑問と判定されたが実際には疑問が書かれていないツイートなどである。これらのツイートは本研究で扱いたい疑問記事とは異なる。そこでこれらツイートの中から，回答可能と思われる疑問記事を人手で抽出した。ここで回答可能と思われる疑問記事とは「作品の内容について説明があり，かつその作品のタイトルが思い出せない」という内容が書かれたツイートを

指す。具体的には「あとタイトル忘れたけど機銃掃射から一人だけ生き残った子が大人になってから、当時の同級生たちに出会う話」や,「チョコミントの出てくる少女マンガ(うろ覚え)読んだ気がするけどタイトル忘れた」といった内容のツイートである。この結果として 118 件の自動抽出した記事の中から，40 件の回答可能な疑問記事を得ることができた。回答可能な疑問記事か否かは筆者が判断した。こうして得られた疑問記事 40 件に対して NDC 記号を自動付与したところ，表 23 のように NDC 記号の付与を行うことができた。この表から例えば，疑問記事 16 件に NDC 記号 7 を自動付与していることが分かる。

表 23 NDC 記号ごとの回答可能な疑問記事件数

NDC	**回答可能な疑問(件)**
0	0
1	0
2	3
3	5
4	2
5	4
6	1
7	16
8	0
9	9
計	40

ここで抽出した 40 件の疑問記事に人手で正しい NDC 記号が付与し，テキスト自動分類による結果と比較し，精度，再現率を算出した。その結果を表 24 に，詳細を表 25 に示す。表 24 から, NDC 記号が 7 の疑問記事が 26 件存在し, それらの疑問記事に関して精度 93.8%, 再現率 57.7%で NDC 記号を自動付与できたことがわかる。

表 24 に示すように，全体の加重平均としては精度 71.2%，再現率 52.5%で疑問記事に対する NDC 記号の自動付与を行うことができた。レファレンス事例に対する NDC 記号の自動付与が精度 53.4%，再現率 52.0%であることを考えれば (表 14)，疑問記事に対する NDC 記号の自動付与はレファレンス事例に対する自動付与よりも高い精度かつ同程度の再現率で行えると考えられる。

表 24　ツイートの疑問記事に対する NDC 記号の自動付与

NDC	精度(%)	再現率(%)	件数(個)
0	0.0	0.0	0
1	0.0	0.0	1
2	33.3	100.0	1
3	20.0	100.0	1
4	0.0	0.0	0
5	0.0	0.0	0
6	0.0	0.0	3
7	93.8	57.7	26
8	0.0	0.0	0
9	44.4	50.0	8
平均	71.2	52.5	40

表 25　ツイートの疑問記事に対する NDC 記号の自動付与詳細

	実験により付与された NDC 記号											計
正解の NDC 記号	NDC	0	1	2	3	4	5	6	7	8	9	＼
	0	0	0	0	0	0	0	0	0	0	0	0
	1	0	0	0	0	0	1	0	0	0	0	1
	2	0	0	1	0	0	0	0	0	0	0	1
	3	0	0	0	1	0	0	0	0	0	0	1
	4	0	0	0	0	0	0	0	0	0	0	0
	5	0	0	0	0	0	0	0	0	0	0	0
	6	0	0	0	2	0	0	0	0	0	1	3
	7	0	0	1	2	1	2	1	15	0	4	26
	8	0	0	0	0	0	0	0	0	0	0	0
	9	0	0	1	0	1	1	0	1	0	4	8
計	＼	0	0	3	5	2	4	1	16	0	9	40

4.3.2 疑問記事の自動収集および NDC 記号の自動付与に関する考察

本研究で扱った疑問記事の NDC 記号は 7 類や 9 類であることが多い。これは，ツイートされるタイトルを求める疑問には，映画，漫画，ゲーム，音楽 (7 類) や，小説 (9 類) に関する疑問が多いからである。表 24 から，7 類の精度が 93.8%と非常に高くなっているが，これには全体の 2／3 が 7 類の疑問であることも影響していると考えられる。3 類の精度が 20.0%と低くなっている。この原因の 1 つとして，特徴ベクトルを作る際に用いたレファレンス事例に，3 類の事例が多いことが挙げられる。このために誤って 3 類を付与してしまうことが多くなり，3 類において再現率は高いが精度は低くなったと考えられる。2 類についても 3 類と同様の理由で精度が低くなったと考えられる。このような傾向は 4.2.2 項で行ったレファレンス事例に対する NDC 記号の自動付与実験においても見られた。ただし，3 類や 2 類に属するツイート数自体は少ない。そのため, NDC 記号ごとの精度が低く見えても，それが精度の加重平均に与える影響は小さい。

4.4 4 章のまとめ

以下の 3 つの実験を行った。即ち，(1) 疑問記事の自動的な収集 (4.1 節)，(2) レファレンス事例に対する NDC 記号の自動付与 (4.2 節)，(3) 疑問記事の自動収集および NDC 記号の自動付与 (4.3 節)，である。

(1) の実験では，提案手法である特徴的な表現による検索とテキスト自動分類による抽出の 2 つのステップによって，疑問記事の収集にかかるコストを大幅に削減し，効率的に収集できることを明らかにした。

(2) の実験では，以下の 2 つのことを明らかにした。即ち，(2-1) 事例の参考資料の項目に記入されている図書の NDC 記号を利用することで先行研究である原田ら (2007) の手法よりも高い精度で NDC 記号の自動付与を行えること，(2-2) 特徴ベクトルの作成方法を工夫することで事例の質問文の項目に記入されている文章を利用しても，参考資料を用いた場合と同程度の高い精度で NDC 記号の自動付与を行えること，の 2 つである。

(3) の実験では，(1)，(2) の実験データと提案手法を用いて，収集した疑問記事に対して事例よりも高い精度，同程度の再現率で NDC 記号を自動付与できることを明らかにした。

5　おわりに

本研究では，疑問に対する回答を得るための支援を行う Web サイトを構築するための基盤的手法を提案し，その実現性を検討した。より具体的には以下の 2 点について検証した。即ち，(1) 疑問記事の自動的な収集手法，(2) 収集した疑問記事の自動的な分類手法，の 2 つである。

(1) に関しては，以下の 2 つのステップによって，Web サイトに提示する疑問記事の効率的な収集方法を提案した。即ち，(1-1) 特徴的な表現による疑問記事の検索，(1-2) テキスト自動分類による疑問記事の抽出，である。(1-1) では，特徴的な表現を用いて疑問記事を検索する際には，「タイトルが思い出せない」という検索ワードを用いることで，効率よく疑問記事を検索できることが明らかになった。次に (1-2) では，テキスト自動分類を用いて疑問記事を抽出する際に最も分類性能が良い方法は，ブログ記事の場合はブースティングによってテキスト自動分類を行い，1 形態素を用いた場合であり，Twitter 記事の場合は決定木によってテキスト自動分類を行い, 1 形態素を用いた場合であった。これらの結果から，「タイトルが思い出せない」というキーワードを用いて疑問記事の検索を行い，検索結果として得られた記事群に対して上記の機械学習を用いることで，疑問記事を発見するコストを大幅に削減できると考えられる。こうして収集した疑問記事に関して，実際に Web サイトを構築し回答を呼び掛けたところ，実際に疑問に対する回答が付き，その疑問が最近書かれたものであれば，著者から好意的に受け止められることがわかった。

(2) に関しては，疑問記事に対してその記事が持つ主題を特定するための実験として，レファレンス協同データベースに蓄積された事例に対して機械学習を用いて，NDC 記号を自動付与する実験を行った。実験の結果，以下の 2 点が示された。即ち，(2-1) NDC 記号を自動付与したい事例の参考資料に記入されている図書の NDC 記号を用いることで，原田ら (2007) の手法より高い精度で 2 桁目 (綱) まで NDC 記号を自動付与することができる, (2-2) NDC 記号を自動付与したい事例の質問文に記入されている文章だけを用いた場合でも特徴ベクトルを工夫することで, 原田ら (2007) の手法より高い精度で 2 桁目 (綱) まで NDC 記号を自動付与できる，の 2 点である。Web 上に投稿された疑問記事を事例の質問文と同質のものと考えるならば，特に (2-2) の結果に基づいて，疑問記事に対して NDC 記号を自動付与することが可能である。これにより疑問記事を効果的に分類し提示できると思われる。

以上の 2 つの実験から得た知見を利用して，最後に収集したツイートの疑問記事に対して NDC 記号の自動付与を行う実験を行った。この実験により，自動的な疑問記事の収集と疑問記事に対する NDC 記号の付与を高い精度で行えることが明らかとなった。この結果から，本研究で提案する自動収集・自動分類手法を用いて疑問記事を提示する Web サイトを構築することは十分に可能であると考えられる。こうした Web サイトによって疑問記事がその疑問に回答できるユーザの目にとまり，回答が与えられるようになることに貢献できるのではないだろうか。

以下に，本研究における課題と今後の展望について述べる。

4.1 節の疑問の書かれた記事の効率的な収集においては，主に図書のタイトルに関する疑問記事を自動的に収集する手法を提案した。しかし，日常的な疑問としては辞書や事典を調べることで解決するような，タイトル以外に関する疑問も存在する。このような疑問も扱うことができれば，Web サイトの有用性は増すであろう。ここで，本研究において図書のタイトルに関する疑問が主な収集対象になった原因は，コーパス作成時に図書のタイトルに関する疑問に対して疑問記事のラベルを付与したからである。即ち，コーパスとして図書のタイトル以外に関する疑問に対して疑問記事のラベルを付与すれば，疑問記事に現れる特徴的な表現とテキスト自動分類による疑問記事の自動分類の結果が変わり，疑問記事を自動的に収集できるようになると考えられる。ただし，本研究においては図書のタイトルを主に扱ったことから，疑問記事を分類して提示する際に NDC 記号を用いた。もし，図書のタイトル以外の疑問も同時に扱うのであれば，分類法として他の枠組みを用いることも検討する必要がある。このことに関しては後述する。

4.2 節の事例に対する NDC 記号の自動付与においては，事例の質問文を用いて NDC 記号の自動付与を行う際に，以下の 3 つから学習用コーパスを作成した。即ち，(1) 事例の質問文, (2) 事例のキーワード, (3) 図書のタイトル, の 3 つである。このときの正解ラベルは (1), (2) では事例の NDC 記号であり，(3) では図書の NDC 記号である。本研究で提案する NDC 記号の自動付与手法では，形態素と正解ラベルである NDC 記号の組さえあれば，それらを学習用コーパスとして用いることができる。もし多様な情報源から学習用コーパスを作成できれば，分類に寄与しない部分の偏りが減るため，NDC 記号の自動付与の精度は向上すると考えられる。そこで，精度の向上のために，事例の内容や図書のタイトル以外も同時に用いて学習用コーパスを作成することが考えられる。例えば，図書の目次や索引，本文

とその図書の NDC 記号や，国立国会図書館件名標目表に含まれる件名とその件名の NDC 記号を共に用いることなどが挙げられる。

4.3 節では，本研究で提案した 2 つの手法を用いて，ツイートの疑問記事を収集しその疑問記事に対する NDC 記号の自動付与を行った。しかし，人々が日常的な疑問を Web 上に書き込むときに利用できる SNS は Twitter だけではない。Instagram や Facebook，はてなブックマーク，SNS ではないが 2 ちゃんねるなども広く用いられているサービスであり，これらのサービスにも日常的な疑問が書き込まれることがある。また Twitter に類似したサービスとして Mastodon も挙げられる。これらのサービスの間には匿名と記名の違い，投稿文字数の違い，利用者の層の違いなど様々な異なる点がある。そのために，書き込みに用いられる表現や単語もまた異なることが考えられる。本研究は，書き込みに用いられる表現や単語の違いが結果に大きく影響するため，これらのサービスを Web 上のサービスとして一括して扱うことで精度が低下してしまう可能性がある。Twitter 以外のサービスに書き込まれる疑問も同時に扱う場合には，サービスごとのサンプル数を揃えるなどの操作を行い，用いられる表現や単語の違いによる偏りを減らす工夫が必要になるだろう。

加えて前述したように，図書のタイトル以外の疑問を扱う場合，NDC 以外の分類を用いることも検討する必要がある。疑問記事を提示する Web サイトにとって有用な分類としては，Wikipedia のカテゴリや既存の Q&A サイトのカテゴリが挙げられる。NDC 記号だけでなくこれらの分類を疑問記事に付与することができれば，それぞれのカテゴリの分類記号に含まれる記事を参照することができるため，疑問記事に書かれた疑問を解決する際に有用な情報源となりうる。これらの分類記号は，本研究で提案した手法を用いることで，NDC 記号と同様の操作で自動付与を行うことができる。Twitter に限れば，ハッシュタグを分類として扱うことも考えられるが，ハッシュタグ同士の間に階層関係が無いため，ディレクトリ型検索を行う為の使用には適さない。

今後の展望としては以下の 2 つが挙げられる。即ち， (a) 精度向上を目的とした深層学習の利用，(b) 回答支援を目的とした疑問記事に対する参考図書の自動推薦システムの構築，の 2 つである。 (a) に関しては現在様々な分野で実績を上げている機械学習手法の一つである深層学習を用いて NDC 記号の自動付与を行いたい。深層学習は大規模な学習用コーパスが存在することを前提とした手法なので，手作業でコーパスを作成する必要がある疑問記事の分類には使えないが，既存のコーパスが存在する NDC 記号の自動付与に対しては有

望な方法である可能性が高い。(b) に関しては，自動付与された NDC 記号も援用しながら疑問記事への回答の助けとなる参考図書を自動推薦するシステムを構築したい。疑問記事に対して適切な参考図書が表示されれば，疑問記事を提示する Web サイトの回答ユーザ，特に序章で述べたレファレンスサービスのアウトリーチを意識した図書館員にとって有用であろう。

謝辞

本研究を進めるにあたり，丁寧で熱心なご助言を賜りました指導教員である緑川信之教授に，心より感謝申し上げます。副指導教員である辻慶太准教授には，長きにわたり懇切なご指導と多くの貴重なアドバイスを頂きました。ここに感謝の意を表します。要所において参考になるアドバイスを頂きました副指導教員である佐藤哲司教授に，感謝申し上げます。また，審査委員をお引き受け下さった芳鐘冬樹教授と高久雅生准教授には予備審査において的確なご意見を頂きました。最後に，日常的な議論を通して様々な示唆を頂いた研究室の皆様に御礼申し上げます。

参照文献

英語論文に関してはアルファベット順，日本語論文に関しては五十音順に並べた。

Blum, Avrim L.; and Langley, Pat (1997); “Selection of Relevant Features and Examples in Machine Learning.”, *Artificial Intelligence*, vol.97, no.1, pp.245-271.

Chen, Guo; and Xiao, Lu (2016); “Selecting Publication Keywords for Domain Analysis in Bibliometrics: A Comparison of Three Methods.”, *Journal of Informetrics*, vol.10, no.1, pp.212-223.

Chang, Chan Chine; and Chen, Ruey Shun (2006); “Using Data Mining Technology to Solve Classification Problems: A Case Study of Campus Digital Library.”, *The Electronic Library*, vol.24, no.3, pp.307-321.

Choi, Erik; Kitzie, Vanessa; and Shah, Chirag (2013); “A Machine Learning-Based Approach to Predicting Success of Questions on Social Question-Answering.”, *Proceedings of the iConference 2013*, pp.409-421.

Ferguson, Chris D.; and Bunge, Charles A. (1997); “The Shape of Services to Come: Values-Based Reference Service for the Largely Digital Library.”, *College & Research Libraries*, vol.58, no.3, pp.252-265.

Joachims, Thorsten (1998); “Text Categorization with Support Vector Machines: Learning with Many Relevant Features.”, *Proceedings of the 10th European Conference on Machine Learning*, pp.137-142.

Khan, Aurangzeb; Baharudin, Baharum; Lee, Lam Hong; and Khan, Khairullah (2010); “A Review of Machine Learning Algorithms for Text-Documents Classification.”, *Journal of Advances in Information Technology*, vol.1, no.1, pp.4-20.

Kucukyilmaz, Tayfun; Cambazoglu, Berkant Barla; and Aykanat, Cevdet (2008); “Chat Mining: Predicting User and Message Attributes in Computer-Mediated Communication.”, *Information Processing & Management*, vol.44, no.4, pp.1448-1466.

Kwon, Nahyun (2007); “Public Library Patrons' Use of Collaborative Chat Reference Service: The Effectiveness of Question Answering by Question Type.”, *Library & Information Science Research*, vol.29, no.1, pp.70-91.

Mladenić, Dunja (1998); “Feature Subset Selection in Text-Learning.”, *Proceedings of the 10th European Conference on Machine Learning*, pp.95-100.

Mooney, Raymond J.; and Roy, Loriene (2000); “Content-Based Book Recommending using Learning for Text Categorization.” *Proceedings of the Fifth ACM Conference on Digital Libraries*, pp.195-204.

Nicholson, Scott; and Lankes, R. David (2007); “The Digital Reference Electronic Warehouse

Project: Creating the Infrastructure for Digital Reference Research through a Multidisciplinary Knowledge Base.”, *Reference & User Services Quarterly*, vol.46, no. 3, pp.45-59.

Sebastiani, Fabrizio (2002); “Machine Learning in Automated Text Categorization.”, *ACM Computing Surveys*, vol.34, no.1, pp.1-47.

Shachaf, Pnina; and Horowitz, Sarah M. (2008); “Virtual Reference Service Evaluation: Adherence to RUSA Behavioral Guidelines and IFLA Digital Reference Guidelines.”, *Library & Information Science Research*, vol.30, no.2, pp.122-137.

Tong, Simon; and Koller, Daphne (2001); “Support Vector Machine Active Learning with Applications to Text Classification.”, *Journal of Machine Learning Research*, vol.2, Nov, pp.45-66.

Yu, Bei; and Inoue, Keisuke (2012); “An Investigation of Digital Reference Interviews: Dialogue Act Annotation with the Hidden Markov Support Vector Machine.”, *2011 OCLC/ALISE Library and Information Science Research Grant Project*, pp.1-34.

相澤彰子 (2007);「共起に基づく類似性」, オペレーションズ・リサーチ 経営の科学, vol.52, no.11, pp.706-712.

相澤彰子, 大山敬三, 高須淳宏, 安達淳 (2005);「レコード同定問題に関する研究の課題と現状」, 電子情報通信学会論文誌, vol.J88-D-I, No.3, pp.576-589.

相澤彰子 (2003);「低頻度語の利用によるテキスト分類性能の改善と評価」, 情報処理学会論文誌, vol.44, no.7, pp.1720-1730.

相澤彰子 (2002);「Naive 手法による大規模テキスト分類問題へのアプローチ」, 情報処理学会研究報告 自然言語処理, vol.2002, no.4, pp.41-46.

安形輝, 石田栄美, 池内淳, 久野高志, 野末道子, 上田修一 (2006);「オープンアクセスを想定した日本語学術論文ファイルの自動判定」, 情報処理学会研究報告 デジタルドキュメント, vol.2006, no.33, pp.55-62.

秋葉友良 (2002);「自然言語処理におけるベイジアンネット」, 人工知能学会誌, vol.17, no.5, pp.553-558.

芦川将之, 池田朋男 (2014);「クラウドソーシングを用いたアノテーション」, 人工知能: 人工知能学会誌, vol.29, no.1, pp.54-59.

阿部一哉 (2016);「Twitter を利用した特定の話題に特徴的な語彙の収集」, 跡見学園女子大学人文学フォーラム, no.14, pp.A32-A40.

安藤駿, 猪瀬裕介, 増田英孝, 佐々木良一 (2014);「マイクロブログ中のリスクコミュニケーションに関する有益な意見を自動的に抽出する手法の提案と評価」, 情報処理学会論文誌, vol.55, no.9, pp.2149-2158.

池田和史, 柳原正, 松本一則, 滝嶋康弘 (2009);「ブログ的表記を正規化するためのルール自動生成方式の提案と評価」, 日本データベース学会論文誌, vol.8, no.1, pp.23-28.

石川雅弘, 池田潔, 加藤淳一 (2012);「ブログ記事の収集と予備分析: 大規模分析に向けて」,

つくば国際大学 研究紀要, vol.18, pp.41-55.
石田栄美 (2003); 「テキストの自動分類に関わる諸要素」, 日本図書館情報学会誌, vol.49, no.2, pp.65-78.
石田栄美 (1998); 「図書を NDC カテゴリに分類する試み」, Library and Information Science, vol.39, pp.31-45.
石田和成 (2015); 「地域特有の単語共起にもとづく位置推定と地域トピックの考察」, 情報処理学会研究報告 情報基礎とアクセス技術, vol.2015, no.2, pp.1-6.
板倉弘幸, 田村雅樹, 若木利子 (2004); 「ラフ集合理論援用による Web ページのテキスト分類」, 情報処理学会研究報告 知能と複雑系, vol.2004, no.85, pp.147-154.
伊藤民雄 (2004); 「インターネットで文献検索+デジタル・レファレンスの現状」, 館灯, vol.42, pp.1-12.
伊藤白, 小澤弘太 (2008); 「国内における Web 上パスファインダーの現況調査」, 情報の科学と技術, vol.58, no.7, pp.361-366.
岩井美樹, 二宮崇 (2015); 「word2vec に基づく述語項構造の分布表現獲得」, 言語処理学会第 21 回年次大会 発表論文集, pp.75-78.
岩田具治 (2014); 「潜在トピックモデルを用いたデータマイニング」, 電子情報通信学会誌, vol.97, no.5, pp.405-409.
巖寺俊哲, 菊井玄一郎 (1997); 「トレンド・トラッキング型テキスト自動分類の試み」, 情報処理学会研究報告 自然言語処理, 1997, vol.1997, no.53, pp.19-24.
岩永学, 田端利宏, 櫻井幸一 (2005); 「ベイジアンフィルタリングを用いた迷惑メール対策における多言語環境でのコーパス分離手法の提案と評価」, 情報処理学会論文誌, vol.46, no.8, pp.1959-1966.
上野良輔, 江口浩二 (2013); 「回帰分析のためのマージン最大化トピックモデルのギブスサンプリング推定」, 情報処理学会研究報告 情報基礎とアクセス技術, vol.2013, no.33, pp.1-6.
内田誠, 柴田尚樹 (2006);「ブログ記事ネットワークからの emerging topic の抽出と可視化」, 人工知能学会全国大会論文集, vol.6, pp.1-4.
遠藤雅樹, 廣田雅春, 大野成義, 石川博 (2015); 「マイクロブログを用いた生物季節観測によるピーク期推定手法の検討」, 情報科学技術フォーラム講演論文集, vol.14, no.2, pp.97-102.
大倉務, 清水伸幸, 中川裕志 (2007);「スケーラブルで汎用的なブログ著者属性推定法」, 情報処理学会研究報告 自然言語処理, vol.2007, no.94, pp.1-6.
太田晋, 美馬秀樹 (2004); 「用語抽出技術を利用したテキスト分類」, 情報処理学会研究報告 自然言語処理, vol.2004, no.93, pp.61-66.
小笠原崇, 高橋友和, 井手一郎, 村瀬洋 (2007); 「ニュース映像アーカイブにおける登場人物の顔照合を利用した名寄せ」, 電子情報通信学会技術研究報告 パターン認識・メデ

ィア理解, vol.106, no.606, pp.55-60.
奥村学 (2008); 「ブログマイニング技術の最新動向」, 電子情報通信学会誌, vol.91, no.12, p.1054-1059.
奥村学, 南野朋之, 藤木稔明, 鈴木泰裕 (2004); 「日本語 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」, 情報科学技術レターズ, vol.3, pp.71-74.
小田光宏 (2006); 「総論: デジタルレファレンスサービスの現在」, 情報の科学と技術, vol.56, no.3, pp.84-89.
小野亘 (2008); 「フリーソフトによるテキスト処理入門」, 情報の科学と技術, vol.58, no.5, pp.232-236.
鍛治伸裕, 福島健一, 喜連川優 (2009); 「大規模ウェブテキストからの片仮名用言の自動獲得」, 電子情報通信学会論文誌 情報・システム, vol.J92-D, no.3, pp.293-300.
梶田将司, 太田芳博, 田島嘉則, 田島尚徳, 平野靖, 内藤久資, 間瀬健二 (2008); 「生涯利用可能な名古屋大学 ID の導入に伴う名寄せ問題とその解決法」, 情報処理学会研究報告 分散システム／インターネット運用技術, vol.2008, no.15, pp.73-78.
鹿島久嗣, 馬場雪乃 (2014); 「ヒューマンコンピュテーション概説」, 人工知能: 人工知能学会誌, vol.29, no.1, pp.4-11.
神谷直樹, 向後礼子 (2004); 「自然言語処理を利用して質的データから客観的評価指標を抽出する方法の検討: 職業リハビリテーションにおける実証的検討」, 日本教育工学会論文誌, vol.28, no.1, pp.49-55.
川浦康至, 山下清美, 川上善郎 (1999); 「人はなぜウェブ日記を書き続けるのか: コンピュータ・ネットワークにおける自己表現」, 社会心理学研究会, vol.14, no.3, p.133-143.
川口克則, 横尾昭男, 奥田英範 (2007); 「ブログの視覚化によるナビゲーションシステム」, 情報処理学会研究報告 デジタルドキュメント, vol.2007, no.77, pp.7-12.
川瀬綾子, 北克一 (2012); 「レファレンス協同データベースの課題: 都道府県立, 政令指定都市立, 公共図書館を中心に」, 情報学, vol.9, no.1, pp.109-131.
岸田和明 (2007); 「インターネット時代における統制語彙の意義と役割」, 情報の科学と技術, vol.57, no.2, pp.62-67.
岸田和明 (2003); 「文書クラスタリングの技法: 文献レビュー」, Library and Information Science, vol.49, pp.33-75.
岸田和明 (2001); 「図書館情報学における自動分類と自動索引作成のための統計的手法:文献レビュー」, 日本図書館情報学会誌, vol.47, no.1, pp.17-28.
喜連川優 (1997);「データマイニングにおける相関ルール抽出技法」, 人工知能学会誌, vol.2, no.4, pp.513-520.
金紅亜, 張軼 (2003); 「ネット上での合同レファレンスシステムと文献資源共有との関係-上海図書館と知識ナビゲーション合同ネットワークサイト」, カレントアウェアネス, vol.278, pp.2-4.

金明哲 (2003); 「SIR によるテキストの分類: 助詞分布を用いた書き手別の分類を中心に」, 日本行動計量学会大会発表論文抄録集, vol.31, pp.40-43.
久米雄介, 打矢隆弘, 内匠逸 (2015); 「興味領域を考慮した Twitter ユーザ推薦手法の提案と評価」, 情報処理学会研究報告 知能システム, vol.2015, no.1, pp.1-8.
後藤正幸, 石田崇, 鈴木誠, 平澤茂一 (2010); 「高次元ベクトル空間モデルによるテキスト分類問題について: 分類性能と距離構造の漸近解析」, 日本経営工学会論文誌, vol.61, no.3, pp.97-106.
齋藤泰則 (2007); 「デジタル環境の進展による図書館と利用者との関係の変容: レファレンスサービスの仲介的機能の展開を中心に」, 情報の科学と技術, vol.57, no.9, pp.429-433.
佐々木謙太朗, 吉川大弘, 古橋武 (2014); 「複数のトピックの時間的依存関係を考慮した時系列トピックモデル」, 情報処理学会研究報告 数理モデル化と問題解決, vol.2014, no.3, pp.1-6.
佐々木謙太朗, 吉川大弘, 古橋武 (2013); 「Twitter におけるユーザの興味と話題の時間発展を考慮したオンライン学習可能なトピックモデルの提案」, 情報処理学会研究報告 数理モデル化と問題解決, vol.2013, no.3, pp.1-6.
佐々木稔, 新納浩幸 (2004); 「文書分類を用いたスパムメール判定手法」, 情報処理学会研究報告 自然言語処理, vol.2004, no.93, pp.75-82.
佐々木悠人, 古宮嘉那子, 森田一, 小谷善行 (2014); 「周辺語義モデルによる日本語の教師無し語義曖昧性解消」, 情報処理学会研究報告 自然言語処理, vol.2014, no.3, pp.1-14.
清水琢也, 岡留剛 (2016); 「Dyamic Stacked Topic Model: 階層構造を持つ文書に対する動的トピックモデル」, 人工知能学会論文誌, vol.31, no.2, M-F92_pp.1-8.
単壮, 加藤昇平 (2014); 「カウンセリングデータにおけるトピックモデルを用いた文書分類」, 情報科学技術フォーラム講演論文集, vol.13, no.2, pp.297-300.
白井匡人, 三浦孝夫 (2014); 「トピックモデルに基づくニュースストリームのオンライン分類」, 情報処理学会研究報告 データベース・システム, vol.2014, no.3, pp.1-6.
数原良彦, 戸田浩之, 櫻井彰人 (2006); 「話題語を手がかりとしたブログからのイベントマイニングの検討」, 情報処理学会研究報告 自然言語処理, vol.2006, no.124, pp.67-73.
杉本祐介, 佐藤太一, 土井千章, 中川智尋, 太田賢, 稲村浩, 内藤克浩, 水野忠則, 菱田隆彰 (2015); 「口コミを利用したレコメンドに適した感情語の分類方法の検討」, 情報処理学会研究報告 モバイルコンピューティングとユビキタス通信, vol.2015, no.50, pp.1-6.
鈴木誠 (2008); 「カテゴリ間の単語頻度の差分を用いたテキストの自動分類」, 日本経営工学会論文誌, vol.59, no.4, pp.355-363.
関口裕一郎, 佐藤吉秀, 川島晴美, 奥田英範, 奥雅博 (2005);「blog ページ集合に対する話題語句抽出手法」, 情報処理学会研究報告 自然言語処理, vol.2005, no.117, pp.27-32.
平博順, 春野雅彦 (2002); 「トランスダクティブ・ブースティング法によるテキスト分類」, 情報処理学会論文誌, vol.43, no.6, pp.1843-1851.

平博順, 春野雅彦 (2000); 「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.

平博順, 向内隆文, 春野雅彦 (1998); 「Support Vector Machine によるテキスト分類」, 情報処理学会研究報告 自然言語処理, vol.1998, no.99, pp.173-180.

高須淳宏, 相原健郎 (2003); 「テキスト分類における訓練データと性能の実験的考察」, NII Journal, vol.6, pp.1-8.

高村大也 (2010); 『言語処理のための機械学習入門』, コロナ社.

竹崎あかね, 大浦裕二, 河野恵伸, 木浦卓治, 林武司 (2016);「自然言語処理を利用した農産物関連テキストからの概念抽出: 野菜商品レビューを対象事例として」, 農業情報研究, vol.25, no.1, pp.47-58.

但馬康宏, 北出大蔵, 中野未知子, 中林智, 藤本浩司, 小谷善行 (2008); 「HMM とテキスト分類器による対話の段落分割」, 情報処理学会研究報告 数理モデル化と問題解決, vol.2008, no.85, pp.91-94.

谷本達哉, 兼松芳之 (2013); 「レファレンス事例データベースの協同構築事業におけるデータ登録の現状と問題点; 国立国会図書館レファレンス協同データベースを対象として」, 図書館情報メディア研究, vol.11, no.1, pp.11-21.

谷本達哉, 兼松芳之 (2012); 「図書館の情報サービスが持つ可能性: 国立国会図書館レファレンス協同データベース事業,その軌跡と展開」, 図書館界, vol.64, no.2, pp.142-153.

田端利宏 (2006); 「SPAM メールフィルタリング: ベイジアンフィルタの解説」, 情報の科学と技術, vol.56, no.10, pp.464-468.

田村俊作 (2001);「デジタルレファレンスサービスの動向」, カレントアウェアネス, no.267, pp.9-12.

塚本浩司, 颯々野学 (2001); 「AdaBoost と能動学習を用いたテキスト分類」, 情報処理学会研究報告 自然言語処理, vol.2001, no.112, pp.81-89.

堤恵, 佐藤久美子, 牧野めぐみ (2011); 「レファ協で拓くレファレンスサービスの新たな地平」, 情報の科学と技術, vol.61, no.5, pp.187-193.

常川真央, 松村敦, 宇陀則彦 (2013); 「日本十進分類法を用いた類似読者発見手法」, 情報メディア研究, vol.12, no.1, pp.42-51.

寺尾隆 (2008);「レファレンス協同データベース事業」, 病院図書館, vol.28, no.4, pp.193-196.

富田準二 (2004); 「ビジネスインテリジェンスをめぐる展望: 意思決定を支援するテキスト集約技術」, 電子情報通信学会技術研究報告 オフィスインフォメーションシステム, vol.103, no.707, pp.51-58.

中嶋琢美, 酒井浩之, 増山繁 (2003); 「SVM (Support Vector Machine) を用いた経済記事の著者の見解に基づく分類」, 情報処理学会研究報告 情報学基礎, vol.2003, no.51, pp.175-180.

中谷知博, 星野准一 (2008); 「経験的価値の分類に基づくゲーム推薦システム」, 情報処理

学会研究報告 エンタテインメントコンピューティング, vol.2008, no.129, pp.49-56.
永田昌明, 平博順 (2001); 「情報論的学習理論とその応用：テキスト分類ー学習理論の「見本市」ー」, 情報処理, vol.42, no.1, pp.32-37.
那須川哲哉 (2009); 「テキストマイニングの普及に向けて: 研究を実用化につなぐ課題への取り組み」, 人工知能学会誌, vol.24, no.2, pp.275-282.
那須川哲哉, 諸橋正幸, 長野徹 (1999); 「テキストマイニング: 膨大な文書データの自動分析による知識発見」, 情報処理, vol.40, no.4, pp.358-364.
野口幸生 (2003); 「デジタル・レファレンス・サービス: 動向と問題点」, 情報管理, vol.45, no.10, p.696-706.
伯田晃, 高橋友一, 小林幸雄 (1990); 「自然言語で指示された対象物の同定方法」, 情報処理学会論文誌, vol.31, no.12, pp.1-8.
八田俊之, 三輪祥太郎 (2015); 「トピックモデルに基づく人行動分析技術」, 情報科学技術フォーラム講演論文集, vol.14, no.4, pp.367-370.
花井拓也, 山村毅 (2005); 「単語間の依存性を考慮したナイーブベイズ法によるテキスト分類」, 情報処理学会研究報告 自然言語処理, vol.2005, no.22, pp.101-106.
原田隆史 (2008); 「Web アーカイブの仕組みと技術的な特徴」, 情報の科学と技術, vol.58, no.8, pp.383-388.
原田隆史, 江藤正己, 大西美奈子 (2007);「レファレンスデータに対する NDC の自動付与」, 情報知識学会誌, vol.17, no.2, pp.61-64.
林賢紀, 松山龍彦, 新元公寛 (2006); 「QuestionPoint: 導入事例と今後の予定」, 情報の科学と技術, vol.56, no.3, pp.96-102.
林幸記, 江口浩二, 高須淳宏 (2011); 「カテゴリ階層構造を考慮した確率的トピックモデルとその応用」, 情報処理学会研究報告 情報基礎とアクセス技術, vol.2011, no.7, pp.1-8.
疋田眞也, 萩原克幸, 鶴岡信治 (2012); 「組織研究におけるテキストマイニングを用いた系統的分析法」, 日本情報経営学会誌, vol32, no.3, pp.97-109.
ビショップ, C.M. (2012a); 『パターン認識と機械学習上』, (元田浩, 栗田大喜夫, 樋口知之, 松本裕治, 村田昇監訳), 丸善出版
ビショップ, C.M. (2012b); 『パターン認識と機械学習下』, (元田浩, 栗田大喜夫, 樋口知之, 松本裕治, 村田昇監訳), 丸善出版
平田勝大, 岡部正幸, 梅村恭司 (2007);「文字列を特徴量とし反復度を用いたテキスト分類」, 情報処理学会研究報告 自然言語処理, vol.2007, no.76, pp.121-126.
平野耕一, 古林紀哉, 高橋淳一 (2005); 「日本語圏ブログの自動分類」, 情報処理学会研究報告 自然言語処理, vol.2005, no.117, pp.21-26.
廣瀬怜那, 松村敦, 宇陀則彦 (2011); 「分類体系と位置情報を組み合わせたディスカバリインターフェースの開発: 検索結果の構造的理解を目指して」, 情報知識学会誌, vol.21, no.2, pp.131-136.

福重貴雄, 菅野祐司 (2003); 「対応分析とベイジアンネットワークを用いた文書分類」, 情報処理学会研究報告 情報学基礎, vol.2003, no.51, pp.167-174.
福田淳一 (2011); 「テキストマイニングのシステム開発上流工程適用の試み」, プロジェクトマネジメント学会誌, vol.13, no.2, pp.18-23.
藤井敦, 秋葉友良 (2007); 「Web 検索質問の自動分類と質問応答への応用」, 情報処理学会研究報告 情報学基礎, vol.2007, no.54, pp.21-28.
藤野昭典, 上田修功, 斉藤和巳 (2006); 「最大エントロピー原理に基づく付加情報の効果的な利用によるテキスト分類」, 情報処理学会論文誌, vol.47, no.10, pp.2929-2937.
古瀬蔵, 廣嶋伸賞, 山田節夫, 片岡良治 (2006); 「ブログ記事からの意見文検索」, 情報処理学会研究報告 自然言語処理, vol.2006, no.124, p.121-128.
ボッレーガラダヌシカ (2014); 「自然言語処理のための深層学習」, 人工知能: 人工知能学会誌, vol.29, no.2, pp.195-201.
本間維, 永森光晴, 杉本重雄 (2010);「日本十進分類法と Wikipedia のカテゴリを用いた蔵書検索クエリの拡張: 蔵書検索結果を用いた語彙構造化」, 情報科学技術フォーラム講演論文集, vol.9, no.2, pp.355-358.
前田新一 (2014); 「コントラスティブダイバージェンス法とその周辺」, 人工知能学会誌, vol.29, no.4, pp.366-380.
増田正 (2012); 「地方議会の会議録に関するテキストマイニング分析: 高崎市議会を事例として」, 地域政策研究, vol.15, no.1, pp.17-31.
松崎友洋, 冨澤眞樹 (2008); 「カテゴリー分類にもとづく FAQ の自動生成」, 情報処理学会全国大会講演論文集 データベースとメディア, 2008, vol.70, pp.687-688.
松本慎平, 川口大貴, 鳥海不二夫 (2015); 「東日本大震災前後の Twitter 利用者の投稿活動に基づく定量化と自動判別への応用」, 人工知能学会論文誌, vol.30, no.1, pp.393-402.
間部豊, 小田光宏 (2011); 「レファレンス質問への回答を可能にしたレファレンスブックの特性に関する研究」, 日本図書館情報学会誌, vol.57, no.3, pp.88-102.
丸山宏 (1994); 「N グラムモデルによる日本語単語の並べ換え実験」, 情報処理学会 全国大会講演論文集, vol.49, pp.181-182.
三津石智巳, 外崎みゆき, 河村俊太郎, 中塚寛幸, 愛宕翔太, 岡本真, 清田陽司 (2012); 「RefMaster: レファレンススキル向上を目的とした E ラーニング・ゲーム」, 情報の科学と技術, vol.62, no.12, pp.508-513.
港真人, 相澤彰子 (2010);「名前同定のための SVM 特徴素の抽出と適用」, 情報処理学会 全国大会講演論文集, vol.72, pp.659-660.
宮川陽子 (2007); 「レファレンス協同データベースへの招待状」, 図書館界, vol.58, no.5, pp.284-288.
村上義継, 坂本比呂志, 有村博紀, 有川節夫 (2001); 「HTML からのテキストの自動切り出しアルゴリズムと実装」, 情報処理学会論文誌 数理モデル化と応用, vol.42, no.14,

pp.39-49.
森國泰平, 吉田光男, 岡部正幸, 梅村恭司 (2015); 「ツイート投稿位置推定のための単語フィルタリング手法」, 情報処理学会論文誌 データベース, vol.8, no.4, pp.16-26.
安井照昌 (1993); 「蟻の行動パタンを用いたテキスト分類の試み」, 情報処理学会　全国大会講演論文集, vol.46, pp.247-248.
山田太造, 野村朋弘, 井上聡 (2013); 「日本南北朝期史料を対象とした潜在的トピックによる史料分類と関連史料提示の手法」, 人文科学とコンピュータシンポジウムじんもんこん 2013 論文集, vol.2013, no.4, pp.145-152.
山田智之, 西信能, 佐藤友思, 棚橋佳子, 渡辺麻子, 松邑勝治, 黒沢努, 矢口学 (2010); 「高精度研究者人名名寄せによる効率的な研究成果情報の集積方法」, 情報プロフェッショナルシンポジウム予稿集, vol.2010, pp.117-122.
山本仁志, 諏訪博彦, 岡田勇, 山本浩一 (2008); 「ブログ空間上のコミュニケーション発生メカニズムの分析」, 日本社会情報学会学会誌, vol.20, no.1, p.29-39.
吉田昭子 (2015); 「レファレンス事例の活用と評価」, 文化学園大学紀要 人文・社会科学研究, pp.21-33.
吉田昭子 (2014); 「レファレンスツールの評価」, 文化学園大学紀要 人文・社会科学研究, vol.22, pp.1-14.
吉田知訓, 間瀬心博, 北村泰彦 (2008); 「質問応答 Web サイトからの関連語ネットワークの自動抽出」, 電子情報通信学会技術研究報告 人工知能と知識処理, vol.108, no.119, pp.75-80.
吉田光男, 荒瀬由紀 (2016); 「トレンドキーワードに関するウェブリソースの横断的分析」, 情報処理学会論文誌 データベース, vol.9, no.1, pp.20-30.
吉田将人, 福原知宏, 増田英孝 (2009); 「ブログ記事と Web ページを用いたイベント情報抽出手法の提案」, 情報処理学会研究報告 デジタルドキュメント, vol.2009, no.35, pp.37-44.
吉田稔, 中川裕志 (2010); 「テキストマイニングの活用」, 情報の科学と技術, vol.60, no.6, pp.230-235.
吉原輝, 関和広, 上原邦昭 (2015); 「ニュース記事の時間的特性を考慮した株価動向予測」, 情報処理学会研究報告 数理モデル化と問題解決, vol.2015, no.4, pp.1-6.
依田紀久 (2013); 「レファレンス協同データベース事業: レファレンスサービスに関わる人たちの実践コミュニティとしての側面とその可能性」, 情報学, vol.10, no.2, pp.1-7.
依田紀久 (2006a); 「レファレンス協同データベース事業について―活用方法の紹介と薬学図書館への期待」, 薬学図書館, vol.51, no.3, pp.220-226.
依田紀久 (2006b); 「レファレンス協同データベース事業に見るデジタルレファレンスサービス」, 情報の科学と技術, vol.56, no.3, pp.90-95.

全研究業績のリスト

① 査読制度のある学術雑誌

- Tsuji, Keita, Arai, Shunsuke, Suga, Reina, Ikeuchi, Atsushi and Yoshikane, Fuyuki, (2013) "Analysis of the Question Asked through Digital and Face-to-face Reference Services," US-China Education Review, vol.8, no.4, pp.51-58.
- 荒井俊介，辻慶太，(2013)「Blog・Twitter に書かれた疑問を収集・提供する Web サイトの構築」，情報知識学会誌, vol.23, no.1, pp.1-19.
- 荒井俊介，辻慶太，(2015)「機械学習を用いたレファレンスデータへの NDC の自動付与」，情報知識学会誌, vol.25, no.1, pp.23-40.

② 査読制度のある国際会議録

- Arai, Shunsuke, Tsuji, Keita, (2012) "Development of a Website to Collect and Provide Questions Posted in Blogs and on Twitter," Proceedings of the 3rd IIAI International Conference on e-Services and Knowledge Management (IIAI ESKM 2012).
- Arai, Shunsuke, Tsuji, Keita, (2013) "Automatic Classification of Reference Service," Proceedings of the 3rd International Conference on Integrated Information (IC-ININFO 2013) .
- Nomura, Nozomi, Arai, Shunsuke and Tsuji, Keita, (2014) "Investigations on Reference Books Held in Japanese Public Libraries," Proceedings of the 5th International Conference on E-Service and Knowledge Management (ESKM 2014).
- Nomura, Nozomi, Arai, Shunsuke and Tsuji, Keita, (2015) "Analysis of Reference Books in Japanese Public Libraries Regarding their Number of Holdings, Frequency of Use, and Price," Proceedings of the Asia-Pacific Conference on Library & Information Education & Practice 2015 (A-LIEP 2015).

付録

テキスト自動分類によって 15,000 件のツイートから自動抽出した，疑問記事 118 件を以下に示す。

2 類 (歴史)

タイトル忘れたけどオアシスのドキュメンタリー映画観たい。

縄文式ドジ弱点満点あとなんやっけかタイトル忘れたけどとにかくアンコ the KANCREW 最高

ハヤカワ SF 文庫で、昔読んだタイムスリップとパラドックスを掛け合わせためっちゃ後味悪い小説のタイトルが思い出せない！！！誰かわかる人いたら、ほんと、助けて欲しい。じつは、パソ通時代のニフティでも空振りだったんだよねー…

「とてもなじゆうのくわわりかた」(僕が今夢で見たベルリ君風の青年の解放戦から数年後、世間はそんな戦いのことなどてんで忘れている。処刑を噂された僕であったが、姫様と洞窟内の研究所からお昼を食べに行くカットが入る。そしてカメラは寄り、銀の常時監視腕輪を映してタイトルバック)、100 点

上司に薦められてた本をようやく買った　タイトルいっつも忘れてしまってたんだよな

【拡散】mizutani_yutaka タイトル忘れたけど見た幾つかの動画がどんなだったか覚えてる。チェルノから２００キロ。白血病になった父親。入院して家族と離ればなれ。毎日４０度の発熱。事故当時は少年で何も気にせず外でスポーツをしていた。私はその動画の後福島東京距離で検索した。

平子:えっと…今日発売のセガのゲームで…ちょっとタイトル思い出せへんけど…ありますか？店員:赤ちゃんはどこからくるのですか？平子:なんやいきなり

「作画の技術もセンスも非常に素晴らしいものがある WIT スタジオの TV アニメで、あらすじとしては人類滅亡後の世界で人間軍の主人公と吸血鬼貴族軍の親友が対立と協調を進めていく」という作品のタイトルが思い出せないこと何度かあり、自分の記憶力に絶望する。(『終わりのセラフ』です)

昨日、情報リテラシー講義のパワポ資料が RT で流れてきて面白かったんだけど、もう一度読もうと思ったらタイトルとか URL とか当該ツイートとか全然思い出せない。確か「ファクトよりデマの方が優先される」といった記述があった気がする。

私立應南学院高等部が DJ を務めるポドキャス！メッセージは、タイトルに「音遊塾メッセージ」と記入して下記まで送信！ラジオネーム忘れずに！★宛先…osaka@rey-s-in.co.jp ⊙毎月第 4 木曜更新「音遊塾」https://t.co/yp1dbculC7

3 類 (社会科学)

あとかなり好きだった海外ドラマで、不良が神様(？)に犬にされて良い行いを 100 個しないと人間に戻れないやつがあったんだけど、最後のほう観れなかったからめっちゃ観たいんだけど内容覚えてるのにタイトルが思い出せなくて詰んでる

タイトル：あのね 本文：携帯忘れてるよ！

その他タイトル打点 1.#1 鹿島 5 得点 1.#0RINS 41.#1 鹿島 4 盗塁 1.#2 グチ 21.#00 鍬本 2 出塁率(規 1.#1 鹿島 .765 二塁打 1.#1 鹿島 11.#5J-宇 11.#59 ユー1 四死球 1.#0RINS 51.#12 飯塚 5

【Kindle】KADOKAWA フェア、対象タイトルが希望小売価格から 50%OFF〜。2 月 16 日(木) 23 時 59 分迄 https://t.co/JQbF9p1gRy https://t.co/Pgg2XxPDtp

あ、タイトルかえんの忘れた。♪♪

タイトル：迷宮奇譚録-Master of Mystery-システム：インセインテーマ：ミステリー表紙：フルカラー（絵：接続設定）サイズ/ページ：A5/184phttps://t.co/ejoKp59TRz 掲載シナリオ数：13 本値段：￥2,000

シーモンド：前に母さんに簡単なパスタの作り方を聞いたんだ。忘れないようにメモしてたんだが、今そのメモを見たらタイトルが「スパゲテカルボンボンボナーラ」で…色々足りないやら多いやら…

「おそ松さん」を題材にしたアプリ 6 タイトル合同のイベント「第二回ニート ... - https://t.co/Mn5vz2UHca https://t.co/v59Ec8BDsr

昔読んでめっちゃ怖かったのあったんだけどタイトル忘れた内容は覚えてるのにリゾートバイトもなかなか怖かった記憶

あ！アンケートタイトルに「進撃の巨人」入れるの忘れてた…今度やなぁ(˘ω˘)

怖い話でタイトル思い出せないんだけど、遊びのつもりでやった儀式で出しちゃった霊に苦しめられる、解決しない系の話、面白かったからもっかい見たいけどタイトル…

小学生のとき同級生から聞いた稲垣淳二の怖い話が本当に怖かったんだけどタイトルが思い出せない

時に余計な情報まで次々盛られてゴシップ意識を刺激してくる報道スタイル、香ばしいアロマが旨すぎです〜😋この暗殺劇、幼い頃に誘拐されて殺人英才教育を施された女スパイ殺し屋が暗躍する、10 年くらい前の香港映画(タイトル忘れた)思い出しました。しかも毒針って。。

タイトル……考えついたのに忘れた……

あまりにも昔の話なのでタイトルが思い出せない

以上 ED にあたしンちのお母さんが写るアニメのレポートでしたタイトルは最終話カスすぎて忘れました

あとタイトル忘れたけど機銃掃射から一人だけ生き残った子が大人になってから、当時の同級生たちに出会う話

【ブログ更新しました！】タイトルは「他力本願の考えは早く捨てて、択一試験の点数を引き上げましょう。」です。内容は、前回のブログで書き忘れたことで、知っていて損のないことですので、受験生の方、是非ご一読ください。-「https://t.co/t7vytr5aRU 」

「おそ松さん」を題材にしたアプリ 6 タイトル合同のイベント「第二回ニートフェス〜バレンタイン編〜」がスタート。限定衣装の 6 つ子達をまとめてお届け - https://t.co/Mn5vz2UHca https://t.co/v59Ec8BDsr

4 類 (自然科学)

何かのドラマで見たけどタイトルが思い出せない…老化現象 ww

今日の予定。掃除軽くして、飯食って、火星の映画（タイトル忘れた）を借りてノンカロリーなコーラと一緒にみて、ff して、ローカロリーなおつまみをいっぱい作ってお酒買って、Skype 準備してこたつでぬくぬく待機ジャな

夢の中に一瞬だけでてきた、Fate/Zero のキャスターの旦那が流す滂沱の涙を、片手を高々とあげてそっと拭いてあげている龍之介、という構図のイラストに、『これが俺の《自由》の女神』というタイトルがついていたのにくっそ萌えたんで、忘れないうちにつぶやいておきますw

電車の座席に忘れられている本のタイトルが『機密漏洩』でちょっと笑った。

悩んでも悩まなくても心配しても心配しなくても何もかもみんななるようになるとタイトル忘れちゃった◆けどなんかの本に書いてありましたよ

サメ…あー、なんかありましたねタイトルど忘れしましたけど

第一話タイトルも伏線からませて「夜、満月と邂逅す」で行こう忘れてなければ

5 類 (技術，工学)

ああ、やつがしらを写真で見せていただいた。少し前に読んだシバの女王の本（タイトル思い出せない）にシンボル的な存在として出て来た。ほんとにこんな風にいるんだ。

小学生の時に読んだ小説がどんなタイトルでどんな話か思い出せぬ…、なんか鬼だとか少年とかバトルとかそんなんで表紙が月をバックに和服の少年が舞っていた感じだったような気がするんだけどググり方が悪いのかそれらしきものが見当たらぬ

文労の BL 本のタイトルは「凍てついた花のように」で、帯のフレーズは【 一度触れてしまったら、二度と離せなくなる 】です#BL タイトルと帯 https://t.co/dS2LLyKtpn 文労だったか労文だったか忘れました

えむえむでぃーでしょっきりさんがウィザードに変身する動画があったんだけどタイトル忘れた

そうですよねー！全然おかしくないですよね☺□笑本人も爪汚いって言われたことないしむしろ褒められる的なこと言ってました！それ言ってた放送のタイトル名ちょっと忘れちゃったので後で探しますね☺ガチマッチはイカよりも手を見る動画だと思ってますwww

朝から頭の中で、タイトルを思い出せないファミコンソフトの BGM がリフレイン。大好きな曲だ。この音とフレーズは間違いなく任天堂だな。一時間粘っても思い出せなかったので、ネットで任天堂ソフト一覧を眺めて判明した。「アイスホッケー」だ。すぐに開けられる引き出しが減っていくのを実感する。

(横浜の某セブンに「大人はきたない」というタイトルのコンテ？をお忘れの方…プリンターに置きっぱなしでございます……お心当たりがあったら……)

カーテンの刑するやつ観にいきたい(タイトル忘れた)

((MC ランボーでの CD リリース、おめでとうございます。すみません、1st Single のタイトルを忘れてしまいました(汗)。教えていただけますか？FROM 山下健二郎)) 『最後の戦場～アフガンの砂と共に～』
※このツイートは実験データの中に全く同じものが 2 つ存在した。

イキまくる動画 ミセスハント No.2 ～二子玉川 vs.銀座の奥さんナンパ～ タイトル ミセスハント No.2 ～二子玉川 vs.銀座の奥さんナンパ～ https://t.co/59T44y6leC

応募者全員プレゼントって名前が出てこなかったからみはるちゃんに言われて、それ！！！ってなった！懐かし～な～(๑´ㅂ`๑)普段忘れてるけど、タイトルとか聞くとどんどん思い出すしまた読みたくなるよね w

6 類 (産業)

SONG BY：種ともこ ♪はっきり言ってあげよか 鈍い人ね始めから あなたとは出会うつもりじゃなかった♫ のタイトルが思い出せない。

貴方の秋本の海城本は「曖昧な言葉でいつもごまかされる。」から始まり、タイトルは『墜落予定』、煽りは【忘れられるわけないだろう】です。#BL 本のタイトルと煽りと出だし https://t.co/cXz1BxeSDt 別れ話から再びくっつくまでの海城で読みたい

あまりにぶっ飛んだタイトルで気になっていた本。丸善で購入し夜帰ってきてから時間を忘れて一気に読み終えた。しばらく涙がとまりそうにない。男女年齢関係なく全ての人に読んでほしい　#夫のちんぽが入らない https://t.co/EBPJevZDbF

奥さん作タイトル！「旦那と入りたかったミルキーウェイ！今日も入れなかったけど」https://t.co/6vr8WStH59

かなでりかの新刊タイトルは『ごめん、ネギ買い忘れた』#あなたの新刊タイトル https://t.co/jxaRMvAgtc

#高知動画 #高知 タイトル【高知単車盗難情報】拡散ディスクリプション twitter 画像より　音 193 音　tw... https://t.co/gwc59KyBBh https://t.co/wY2UuPr6uv

【Android アプリ紹介】「SoundHound」・・・音楽を読み込んで情報を掲示するアプリ。店舗などで耳にした音楽の名前が思い出せないときなど、これを起動して聞かせれば、タイトルや歌手をすぐに知ることができます♪https://t.co/QKimqddrQT

タイトル忘れの結果😳 https://t.co/k0f6APJ9OV

7 類 (芸術，美術)

「豪炎寺の決意！」という回があるんだけど、当時その回を観ていた時タイトルコールの瞬間にたまたま母が通りかかって、「"豪炎寺"ってお寺が燃えてるみたいで何か縁起悪いなあ」って言っていたのが未だに忘れられない

タイトル忘れたけど 3 話でクラインがキリトを止めようとするシーン好きよ(⌒‿⌒)

フォグホーン「そりゃお前えっとな・・・今日の発売のセガのゲームで・・・タイトルが思い出せない」バーンヤード「あ！わかった。あれだ、赤ちゃんはどこからくるの？」フォグホーン「そりゃお前いきなり何だよ」 バーンヤード「えっ」 フォグホーン「えっ」

出勤中の車のなかでふと思った。忍ミュの鐘ヶ江仙ちゃんと北村仙様を例えるとしたら、松浦亜弥のタイトル忘れたけど「迷うな♪♪セクシー(北村仙様)なのキュート(鐘ヶ江仙ちゃん)なのどっちが好きなの？(゜∀゜　)」だな。※全国の忍ミュ立花仙蔵ファンの皆様すみません💦💦

オーヤサン：甘党から簡単なパスタの作り方を聞いたトリハ。忘れないようにとメモしてたんだけど、そのメモのタイトルが「スパゲテカルボンボンボナーラ」なんか、いろいろ足りないやら多いやら…姐さん：爆発しそうなパスタね

ホットラインマイアミもちょっと気になるしタイトル忘れたけど荒川アンダーザブリッジの星みたいなのが主人公のゲームもやりたい

人の話ってちゃんと聞かなきゃだめだよね、電話で何を彼が何を言ったのか思い出せない映画を見たって言ってたな、タイトルは？内容は？話してくれたはずなのに覚えてない君は何かあるたび電話でいっぱいいっぱい楽しそうに話してくれるのにそれなのに僕の耳にはもう届かない

最近読んでないけど紡木たくよりいくえみ綾が好きだった〜好きな漫画のタイトル忘れちゃった(^^;)単行本捨てずにお納戸にあるはずだから帰ったら見てみよう(^-^)いくえみ綾のペンネームの元になってるくらもちふさこも好きだった〜いつもポケットにショパン、東京のカサノバ〜＞RT

ARMS 描いた人の別の漫画の主人公タイトル忘れた

北村尚登さんの絵。ポールペンを使って下書き。画材も色々油絵の具やらコンテや コーヒーをかけたり自由！可愛い❤タイトルが面白い W 黄色い方「王子さまが迎えに来るのを待ってる　キリン」赤い方「しばらく忘れてい… https://t.co/0mVY6y8tkO

大尉「うーん…セガのゲームでさ…ちょっとタイトル思い出せないんだが…あるか？」中尉「赤ちゃんはどこからくるの？」大尉「なんだいきなり」

タイトル忘れちゃったが昔相撲漫画というかギャグマンガも描いてたのう。砂漠の野球部はキャッチャーがオカマっぽくて、相撲のは主人公がゲイっぽいアレだったが。

異次元からの色彩だ、タイトルが思い出せなくて宇宙からの色って書いたけど、クトゥルフ原作ベスト 5 には入る

見たい映画が 5 タイトルくらいあるんだけど、どういう風に見ていけば全部正規の値段(1800 円)払わずに見られるかめっちゃ考えてる。今日行こうと思ってて忘れてた。あしたは黒執事とドクターストレンジ見ようと思ってるけど 2 本立て続けいけるかな、体力的

に。

あー！タイトル忘れましたがこの作品ありましたねぇ！いいですねぇ！

ヘンゼルとグレーテルを見て思い出したんだけど、前に見たホラー映画……タイトルが思い出せないけど……姉弟が休暇に母親の実家か何かに行ってワーーーってなるやつ……カマドの掃除をするところドキドキしながら見てた。閉じ込めて丸焼きにされるのではないかと。

エックス「えっと…今日発売のセガのゲームで…ﾀｲﾄﾙ思い出せない、なんだったかな…」 ゼロ「赤ちゃんはどこからくるの？」 エックス「え、どうしたの急に」 ゼロ「えっ」 エックス「えっ」

チョコミントの出てくる少女マンガ(うろ覚え)読んだ気がするけどタイトル忘れた

徹くんおはようございます☀魅力とばれんさっきコメントしてきたんですけど魅力のほうのタイトルどうしても思い浮かばなくて先に本文を打っていたら忘れて送信してました〜いつも必ず書いてたのに悔しい☹

大倉忠義の疾風ロンド。横山裕の破門。錦戸亮の映画（タイトル忘れた）重岡大毅の溺れるナイフ。濵田崇裕の破門。小瀧望のプリンシパル。映画多い〜あああああ。

もう老人だからこの間やろうと思ってたゲームのタイトルも思い出せない、なんだっけ？

そう言えばバレンタインで思い出したんだけど、昔蔵王さんが挿し絵描いてて確かパティシエの話でチンコチョコが出てくる小説読んだな…。タイトルも著者も思い出せないけど、蔵王さんだったのは確実に覚えている…。

今日のキャンプ中継で流れてた懐メロのタイトルをやっと思い出せた！　GLAY の『SHUTTER SPEEDS のテーマ』だったわ。シングルカットされてないと思ったけどよく流れたなあ

唐澤と key の話してたら最近アニメやった登場人物が DeNA 選手の苗字ばかりの作品の存在を思い出したがタイトルが思い出せない　バカつまらなかったことだけは覚えてる

初めて触れた R-18 コミックはなんだったか覚えてるけど（タイトルは忘れた）、レーティングがなくてえっちな作品というとなんだったかなあ……

タイトルに三毛猫入れるの忘れてた←

(■∋■)＜『ニューヨークに行きたいかー！』が何の曲の仮タイトルか忘れちゃった。

また AV 落ちのなんちゃらかんちゃらアイドルもの漫画のタイトル忘れた

小学生か幼稚園のころにやったペンギンを滑らせてギミッククリアするゲームのタイトルが思い出せない 2D で何匹か引き連れてる　なんだっけか

今日ゆーちゅーぶ任せで音楽流していたとき、伊福部昭さんのゴジラ 1 作目の音楽が流れた。すごく綺麗で悲しげで胸を打たれる印象的な曲なのに、どの場面で使われていたかちっとも思い出せなくて調べてみたら、タイトルが「海底下のゴジラ」だったので納得。そこ、めちゃくちゃ泣いてたわ。

何の映画か忘れてしまったけど、主人公が馬と一緒に 希望を持たない者は沈む沼 みたいなところを渡ろうとして馬は次第に沈んでしまうっていうシーンがあって、観たくてもタイトルが出てこない…

マンガ読みたい！！！大正時代らへんが舞台のやつ！うわぁ、好きなマンガのタイトル思い出せない ww 大正デモクラシーしか出てこない ww

この曲すきなんだよな～🖤あとはなんだっけ、タイトル忘れたけどもういっこめっちゃ好きなやつあるんよな～☺#bnann

ララランドという映画がテレビで紹介されてますが、この女優さんが出てた映画をこの前飛行機の中で観たはず。ギャング映画でしたがタイトルを忘れた・・・調べてみよう。

卒業の合唱曲のタイトルが思い出せない。終わらない明日へ

【まよチキ！】正しいタイトルは迷える何とかとチキンな…忘れた。ヒロインがどれも可愛いアニメだ。メインヒロインは男装執事だ‼ホモ作品ではないぞ。アニメが気に入ったら、是非原作を読んで欲しい。

グランプリワンマンが蘇った(´д⊂)今日はいい曲いっぱい聴けた□どの曲も自分の曲のように歌いこなすな〜おはやっぱりすごいよ‼あっ久々に見れたタンクトップ🫦腕の筋肉にｷｭﾝｷｭﾝしたよ😆あっ聴くの忘れた♠新曲のタイトルはあれで行くのかななな👀笑

承太郎「うーん…セガのゲームでさ…ちょっとタイトル思い出せないんだが…あるか？」花京院「赤ちゃんはどこからくるの？」承太郎「なんだいきなり」

8 類 (言語)

歌えるけどタイトル忘れた…あおぞらに稲妻〜らんらんらんらんランナウェイ…稲妻パラダイスか！

9 類 (文学)

どっかで主人公の親族の葬式中にヤりはじめるヤンデレが出てくるエロゲ見かけたんだけどタイトル忘れた

エクストラバージンはホンマでっか TV までおあづけ。アレグラは今朝(昨日）8 時台テレビ大阪のアジア系ドラマ(タイトル忘れ）で 2 種。明日もありますように(⋈◍＞◡＜◍)。✧♡

ずっと思い出したくて思い出せない小説が気になる。25 年前に読んだ #いとうせいこう氏（だったはず）の短編小説で恋人同士がすれ違ったり婚約したり空港に行ったりする話（漠然すぎ)。読み返したいのに色々うろ覚えすぎて誰にも質問できず。誰かご存知ならタイトル教えてください〜

どの作品も面白いデスよね p(^^)q あと、カナンはどうでしょう？(タイトルのスペルは忘れました（泣）)

中学生のころに学級文庫で読んだ短編 SF のタイトルが思い出せない内容は強烈に覚えてるのに

というか、タイトル思い出せなくて「団地、団地…… 団地まさおだ！」と、そのまま書きそうになったぐらいのアレ。

拝読しました！風邪で弱っている朔ちゃんの問いかけに、うんうん不安だよね…と思った矢先に、最後の犀くんの台詞に打ちのめされました…！文アルでほのぼのしてると忘

れてしまいますが、そうなんですよね……読み終えてからタイトルを見たときも衝撃を受けました…！

クロビの 88 のやつ、タイトル忘れたけど。アレやってみたけど思考が停止した。譜面覚えるのに五千円くらい使いそう。

そんな話のヤンデレ系百合 ntr 漫画を読んだことがあるのですが、誰か知りませんか？タイトル、作者が思い出せない

イル「題名忘れたけど、ジュリアロバーツの「飲む打つ買う」みたいなタイトルの映画、楽しみに借りたのに途中で寝てしまった…」みる「『食べて祈って恋をして』？w」まこ「ただのギャンブラーになっててワロタ」

史実を漁りたい気もするがまず作品を読みたい気がする。作品はおーがい先生の高瀬舟、舞姫と敦君の李陵、山月記、タイトル忘れたけど弓の人の話と弟子、夏目先生のこころと坊ちゃんは読んだ。

#カルテット今回も全編やられまくった。コミカルな部分とぶっ刺して来る部分の落差が…ほんと一つの理想。タイトル出たのが３０分過ぎと言うのもアレだけど（忘れてた位惹き付けられてた)、やっぱ真紀さんの夫のクドカンが。もしかしたら顔は出ないのかとも思ってただけにほんとやられた…！

日付変わったからもう遅いかもしれんが論文タイトルとページ作成〆切は火曜までだからお忘れなく

何年か前に Youtube か Vimeo に UP されたイヴォンヌ・レイナーの映像はすぐに消去されていたな。タイトル忘れたけど、もう一度アレを見たい。

フェリちゃんは少し本を読んでいました 。タイトルは .. 「不思議の国のアリス」という誰でも知っているアレです 。これを読んでいると心が落ち着けるので , つい時間を忘れてこんな時間にまで ..?

ゆきちゃん、おはよう ^^タイトルは忘れちゃったけど、何冊か東野圭吾さんの本をツイートしたことあったよね。意外とチャンちゃんと話が合うかも、と思った事を覚えてます(遠い目)

タイトル思い出せないんだけど小学校の図書室に置いてあった少女向けっぽい今で言うとこのラノベだななんだっけあれ、なんか主人公の少女のうなじに目みたいなデキモノがあって〜てやつ>RT

菊池「題名忘れたんだが、「飲む打つ買う」みたいなタイトルの映画を借りたのに途中で寝ちまった…」横光「『食べて祈って恋をして』では？」吉川「ただのギャンブラーじゃないか」芥川「いつもの寛だね」

あ！そう『グール』！(゜ｗ゜)(言っておいてタイトルは忘れてた)

タイトル忘れたけど去年の夏コミの新曲はイントロ涙出るくらいいいゾ

タイトル付けるの忘れてた。『べっぴんさん』、嘘がつけない新人君の「僕は何も知りません」に「それは知ってるということやな」って言い放つ紀夫さんが最高に面白かった。

よかった…これからもぱっと見では意味の分からないタイトルを心掛けよう……同人誌の仕舞い忘れは怖い😭😭

彼女の事を検索しようとして名前を思い出せなくて Chinese wife って検索したらどうやって中国人妻と出会うか、とか中国人妻はいいぞ、とかが Google のトップに出てきたのに Japanese wife って入れたら AV タイトルしか出て来なかった件 :-(