Techniques for Analysis and Classification of Online Opinions

Submitted to
Graduate School of Information Science and Technology
Osaka University

January 2013

Kazushi IKEDA

List of Publications

Journal Papers

1. Kazushi Ikeda, Tadashi Yanagihara, Kazunori Matsumoto and Yasuhiro Takishima, "An Automatic Rule Generation Method for Modifying Informal Expression in Blog Documents", *Journal of the Database Society of Japan (DBSJ)*, vol. 8, no. 1, pp. 23–28, 2009.

2. Kazushi Ikeda, Tadashi Yanagihara, Kazunori Matsumoto and Yasuhiro Takishima, "Automatic Rule Generation Approach for Morphological Analysis of Peculiar Expressions on Blog Documents", *Journal of Information Processing Society of Japan (IPSJ) Transactions on Databases*, vol. 3, no. 3, pp. 68–77, 2010.

3. Kazushi Ikeda, Tadashi Yanagihara, Gen Hattori, Kazunori Matsumoto, Chihiro Ono and Yasuhiro Takishima, "Detection of Malicious Web Pages Based on HTML Elements", *Journal of Information Processing Society of Japan (IPSJ)*, vol. 52, no. 8, pp. 2474–2483, 2011.

4. Kazushi Ikeda, Gen Hattori, Kazunori Matsumoto, Chihiro Ono and Teruo Higashino, "Demographic Estimation of Twitter User for Marketing Analysis", *Journal of Information Processing Society of Japan (IPSJ) Transactions on Consumer Devices & Systems*, vol. 2, no. 1, pp. 82–93, 2012.

5. Kazushi Ikeda, Gen Hattori, Chihiro Ono, Hideki Asoh and Teruo Higashino, "Twitter User Profiling Based on Text and Community Mining for Market Analysis", *Journal of Knowledge Based Systems*, (under submission)

Conference Papers

1. Kazushi Ikeda, Tadashi Yanagihara, Kazunori Matsumoto and Yasuhiro Takishima, "Unsupervised Text Normalization Approach for Morphological Analysis of Blog Documents", in *Proceedings of the 22nd Australasian Joint Conference on Artificial Intelligence (AI)*, 2009, pp. 401–411.

2. Kazushi Ikeda, Tadashi Yanagihara, Kazunori Matsumoto and Yasuhiro Takishima, "Detection of Hazardous Information Based on HTML Elements", in *Proceedings of the 7th IEEE International Conference on Research, Innovation and Vision for the Future in Computing & Communication Technologies (RIVF)*, 2010, pp. 111–114.

3. Kazushi Ikeda, Tadashi Yanagihara, Kazunori Matsumoto and Yasuhiro Takishima, "Hazardous Document Detection Based on Dependency Relations and Thesaurus", in *Proceedings of the 23rd Australasian Joint Conference on Artificial Intelligence (AI)*, 2010, pp. 455–465.

4. Kazushi Ikeda, Gen Hattori, Kazunori Matsumoto, Chihiro Ono and Yasuhiro Takishima, "Social Media Visualization for TV", in *Proceedings of the 22nd International Broadcasting Convention (IBC) Conference D–243*, 2011.

Other Related Journal Papers

1. Kazushi Ikeda, Thilmee M. Baduge, Akihito Hiromori, Hirozumi Yamaguchi and Teruo Higashino, "An Application Layer Multicast Protocol for Stable Streaming and Its Evaluation on PlanetLab", *Journal of Information Processing Society of Japan (IPSJ)*, vol. 50, no. 6, pp. 1549–1560, 2009.

2. Kazushi Ikeda, Thilmee M. Baduge, Takaaki Umedu, Hirozumi Yamaguchi and Teruo Higashino, "ALMware: A Middleware for Application Layer Multicast Protocols", *Journal of Computer Communications*, vol. 34, no. 14, pp. 1673–1684, 2011.

Other Related Conference Papers

1. Kazushi Ikeda, Thilmee M. Baduge, Takaaki Umedu, Hirozumi Yamaguchi and Teruo Higashino, "A Middleware for Implementation and Evaluation of Application Layer Multicast Protocols in Real Environments", in *Proceedings of the 17th ACM International Workshop on Network and Operating Systems Support for Digital Audio & Video (NOSSDAV)*, 2007, pp. 125–130.

2. Kazushi Ikeda, Shunsuke Mori, Yuya Ota, Takaaki Umedu, Akihito Hiromori, Hirozumi Yamaguchi and Teruo Higashino, "D-sense: An Integrated Environment for Algorithm Design and Protocol Implementation in Wireless Sensor Networks", in *Proceedings of the 11th IFIP/IEEE International Conference on Management of Multimedia and Mobile Networks and Services (MMNS)*, 2008, pp. 20–32.

3. Chihiro Ono, Kazushi Ikeda, Gen Hattori, Kazunori Matsumoto and Yasuhiro Takishima, "TV Viewing Support System Considering Both Individual and Family Preferences", in *Proceedings of the 18th User Modeling, Adaptation and Personalization (UMAP)*, 2010, pp. 31-33.

4. Hideki Asoh, Kazushi Ikeda and Chihiro Ono, "A Fast and Simple Method of Twitter User Profiling", *Proceedings of the International Workshop on Mining Ubiquitous and Social Environments (MUSE)*, 2012, pp. 19-26.

5. Kazushi Ikeda, Gen Hattori, Chihiro Ono, Hideki Asoh and Teruo Higashino, "Early Detection Method of Service Quality Reduction Based on Linguistic and Time Series Analysis of Twitter", in *Proceedings of the 9th International Symposium on Frontiers of Information Systems and Network Applications (FINA), 2013*, (to appear)

Abstract

With the advent of the Internet, an increasing number of online opinions on Web pages, blogs and Bulletin Board System (BBS) are posted by consumers. These opinions can be regarded as a large amount of text information posted in real-time. Analysis techniques for utilizing these real-time and huge amounts of opinions on both businesses and consumers are strongly desired.

There are three steps for the utilization of online opinions, namely: collection, analysis and distribution. For each step, various studies have been addressed, respectively. This thesis studies about three major problems in the step of analysis of online opinions. As a problem for linguistic analysis, online opinions often contain peculiar expressions, which greatly reduce the performance of linguistic analysis since morphological analyzers regard them as unknown sequences. Techniques for increasing the accuracy of linguistic analysis of these peculiar expressions are required. Online opinions contain malicious information such as spams and illegal information. These malicious information should be removed because it reduces the credibility of services for businesses and consumers. Existing term-based methods have difficulties in detecting malicious information in some cases. For example, a term has several meanings in different use cases. In addition, Web pages and blogs sometimes contain more pictures and less text information. However, advanced image processing requires large processing time. User profiles such as age and gender are important to deeply understand online opinions. However, only a few users show their profiles. Obtaining user profiles is essential for credible services such as marketing analysis.

Considering the above facts, this thesis studies the following three research topics: (1) linguistic analysis techniques for morphological analysis of peculiar expressions, (2) highly accurate detection techniques for malicious information,

and (3) user profiling techniques for online opinions.

The first method aims at reducing the number of unknown words on online opinions by replacing peculiar expressions with formal expressions. Manual registration of peculiar expressions to the morphological dictionaries is a conventional solution, which is costly and requires specialized knowledge. In our algorithm, substitution candidates of peculiar expressions are automatically retrieved from formally written documents such as newspapers and stored as substitution rules. For the correct replacement, a substitution rule is selected based on three criteria; its appearance frequency in the retrieval process, the edit distance between substituted sequences and the original text, and the estimated accuracy improvements of word segmentation after the substitution.

The second method aims at detecting online opinions that contain malicious information. Existing term-based methods have difficulties in detecting malicious information because (a) existing methods ignore the usage of terms which often effects on maliciousness of information, and (b) it is difficult to detect Web pages and blogs which mainly contain pictures but few text information. We propose a method which detects malicious information based on the dependency relations of terms for the former problem. We also propose a method which detects malicious information by analyzing HTML of Web pages for the latter problem.

In the former method of using dependency relations of terms, we propose an algorithm to increase the accuracy of malicious information detection by correcting the classification of a conventional text-based method based on the dependency relations of the malicious terms and their neighboring segments. In order to apply this method for many examples, we also propose a practical algorithm to increase performance by expanding the malicious segment pairs using a thesaurus.

In the latter method of analyzing HTML of Web pages, strings that appear especially in HTML elements of malicious Web pages are automatically chosen. For example, features such as the background colors of Web pages, the server names related to malicious Web pages, or the names of javascript functions that makes browsers perform unusual actions are detected. In order to detect malicious Web pages correctly, we train classifiers to learn the complicated combination of these features. Since our algorithm does not rely on the text parts of

Web pages, our method can detect Web pages that existing text-based methods have difficulty in detecting.

The third method aims at extracting user profiles such as age and gender. Several text-based approaches are proposed to extract user profiles. Considering practical use, however, we found that a large number of users have few posts or few features on their posts. Text-based demographic estimation methods lack the accuracy for those users. To solve these problems, we propose a hybrid of a text-based method and a community-based method for demographic estimation of users. The text-based method estimates the users who have plentiful text features on their posts. For the rest of users, the community-based method analyzes their related persons such as followers and followees who have plentiful text features.

Analysis techniques discussed in this thesis offer differentiated values for services utilizing online opinions. Through experiments, we have confirmed that the performances of the proposed techniques are high enough to allow practical uses. Part of these techniques has already been deployed as functions of a marketing analysis tool for online opinions.

Contents

List of Figures

List of Tables

Chapter 1

Introduction

With the advent of the Internet, an increasing number of online opinions on Web pages, blogs and Bulletin Board System (BBS) is posted by consumers. These opinions can be regarded as a large amount of text information posted in real-time. Analysis techniques for utilizing these real-time and huge amounts of opinions on both businesses and consumers are strongly desired. There are three steps for the utilization of online opinions, namely: collection, analysis and distribution (Figure 1.1). For each step, various studies have been addressed, respectively. This thesis studies about three major problems in the step of analysis of online opinions.

In the research area of information collection, many studies have been addressed such as, information retrieval to give exact answers for search requests from users [1, 2, 3, 4, 5], query expansion to give a larger amount of information for search requests [6, 7, 8, 9, 10, 11, 12], and search algorithm to find information quickly from a large database [13, 14, 15, 16, 17, 18]. Conventional studies mainly focus on information retrieval for a huge set of documents or Web pages. In recent years, methods for collecting online opinions have been proposed [19, 20, 21]. Authors have also proposed an efficient method for collecting online opinions related to television programs [22].

In the research area of analysis of online opinions, various studies have been addressed. Linguistic analysis is important in order to correctly interpret information, which includes morphological analysis [23, 24], dependency analysis [25, 26], and case analysis [27, 28]. Classification methods of opinions based on their sentiment [29, 30, 31] and the categories of topics [32, 33, 34] have been

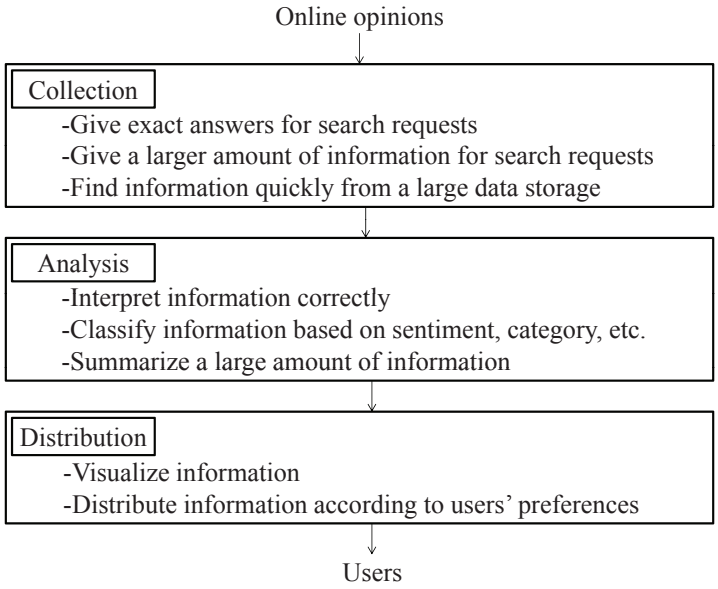


Figure 1.1: Research Areas Related to Collection, Analysis and Distribution of Online Opinions

proposed. Techniques for summarizing a large amount of information are also major research topics [35, 36, 37, 38].

In order to distribute information to users, techniques for information visualization and recommendation are essential. Many methods for visualization of text information have been proposed [39, 40, 41, 42]. Studies related to recommendation systems which allow to present information according to the user's preferences draw many attentions. These methods are classified into collaborative filtering based on the relations among users' history information[43, 44, 45], and content-based recommendation based on features extracted from the contents [46, 47, 48]. Authors have also proposed an recommendation method for multiple users [49].

This thesis particularly focuses on the techniques for the step of analysis of online opinions described above. As a problem for linguistic analysis of online opinions, they often contain peculiar expressions, which greatly reduce the performance of linguistic analysis since morphological analyzers regard them as unknown sequences. Techniques for increasing the accuracy of linguistic analy-

sis of these peculiar expressions. Online opinions contain malicious information such as spams and illegal information. These malicious information should be removed because it reduces the credibility of services for businesses and consumers. Existing term-based methods have difficulties in detecting malicious information in some cases. For example, a term has several meanings in different use cases. In addition, Web pages sometimes contain more pictures and less text information. However, advanced image processing requires large processing time. User profiles such as age and gender are important to deeply understand online opinions. However, only a few users show their profiles. Obtaining user profiles is essential for credible services such as marketing analysis.

Considering the above facts, this thesis studies the following three research topics: (1) linguistic analysis techniques for morphological analysis of peculiar expressions, (2) highly accurate detection techniques for malicious information, and (3) user profiling techniques for online opinions.

The first method aims at reducing the number of unknown words on online opinions by replacing peculiar expressions with formal expressions. Manual registration of peculiar expressions to the morphological dictionaries is a conventional solution, which is costly and requires specialized knowledge. For example, information related to the word such as its part of speech and inflected form is required. Compatibility of the dictionary should be maintained. From our experience, only 30,000 unknown words per month are manually registered by an experienced worker. On the other hand, in our pre-examination by using popular Japanese morphological analyzer MeCab [23], 6,000,000 blog sentences contain about 650,000 kinds of unknown words, which show the difficulty in the manual registration of unknown words.

Most peculiar expressions seen in Japanese blogs are derived from formal expressions and categorized in some typical patterns based on their way of derivation. For example, visually-similar characters tend to be substituted, such as, “i @m” instead of “I am” in English, where “a” is substituted by its visually-similar symbol “@”. In the same manner, “cute” can be substituted by “$c\mu\tau e$”. In another derivation pattern, words are written to reflect their pronunciation in conversation. For example, “amazing” is emphatically pronounced in conversation. In blogs, people exactly describe these pronunciation and create various kinds of expressions, such as, “amazzzzing”. Japanese is written with several

different character types such as *kanji*, *hiragana*, *katakana*, *romaji* (Roman alphabet), and so forth. People intentionally spell words in different character types.

As existing related work, the "Dictionary Expansion" algorithm [50] focuses on accuracy improvement of morphological analysis for Japanese colloquial expressions in Web chat applications. The authors find out the rules that generate colloquial expressions from formal expressions registered on morphological dictionaries. For example, they define rules like *"adjective words tend to be inserted with a prolonged sound symbol"* from the case seen in Web chat applications (like in English , the formal adjective "cool" tend to be written as "coooool". Since this method relies heavily on the subjective view and skills of the operator, it is difficult to generate generic rules that can be usefully applied to many expressions. In [51], our experiments with 2,000,000 blogs show 37.2% of all the sentences affected by the algorithm of [50] increase errors in word segmentation. This suggests the algorithm proposed in [50] is not versatile enough to apply to any type of text data, including blogs.

In our algorithm, substitution candidates of peculiar expressions are automatically retrieved from formally written documents such as newspapers and stored as substitution rules. In our previous work [51], we have proposed the "Initial Rule Expansion" algorithm, which automatically creates highly accurate rules from manually given low-level rules based on the statistics. For example, given rules such as, (a) remove "@" and (b) replace "@" with "a" will result in specific rules (a) "I @m $\rightarrow$ I m" and (b) "I @m $\rightarrow$ I am" ("X $\rightarrow$ Y" means substitution of Y for X). Then, (b) "I @m $\rightarrow$ I am" is stored due to its statistically high correctness. Although this method is effective in reducing the number of unknown words of peculiar expression, only a limited number of sentences are modified by the rules. Due to many kinds of peculiar expressions, it is almost impossible to manually cover all the initial rules.

Making advanced from our previous work, We propose a new method that automatically retrieves substitution candidates for peculiar expressions from formally written documents such as newspapers. For the correct replacement, a substitution rule is selected based on three criteria; its appearance frequency in the retrieval process, the edit distance between substituted sequences and the original text, and the estimated accuracy improvements of word segmentation

after the substitution. We compare the performance of our method to conventional two methods. In our experiments, we modify 100,000 blog sentences and evaluate the number of unknown words and the accuracy of word segmentation.

The second method aims at detecting online opinions that contain malicious information. One of the most popular approaches to detecting malicious Web pages is the blacklist/whitelist approach. However, this method has several problems such as the high cost of database management, performance deterioration in blogs, etc. where black and white pages exist under the same domain and where there is no utility for newly launched Web pages whose information does not yet exist in the database. To solve these problems, text-based approaches that detect malicious Web pages by analyzing described texts have been proposed. However, existing term-based methods have difficulties in detecting malicious information because (a) existing methods ignore the usage of terms which often effects on maliciousness of information, and (b) it is difficult to detect Web pages and blogs which mainly contain pictures but few text information. We propose a method which detects malicious information based on the dependency relations of terms for the former problem. We also propose a method which detects malicious information by analyzing HTML of Web pages for the latter problem.

Related to the former method of using dependency relations of terms, several text-based algorithms have been proposed to detect malicious Web pages [52, 53]. The algorithm proposed in [52] automatically generates a set of malicious keywords that appear unusually often in malicious Web pages in the training datasets. The algorithm proposed in [53] calculates the malicious score of Web pages based on the similarity of the feature vectors extracted from Web pages of the training datasets and the evaluation datasets. In these algorithms, however, documents are split into morphemes thus the contexts where they appear are ignored. As a result, they have difficulty in accurately discriminating between documents that contain morphemes that are classified both as malicious and harmless depending on the context, such as “kill” and “drugs”. For example, the word “kill” is used in both malicious documents such as “kill a man” and harmless documents such as “kill a process”. An accurate document retrieval algorithm based on dependency relations have been proposed [54]. The algorithm proposed in [54] splits documents into morphemes, analyze their de-

pendency relations, and makes a binary tree called a "Structured Index" in advance. Users' queries are written in pseudo-natural language, and are also analyzed into the Structured Index and matched with the documents. In these dependency relation based approach, the recall is often reduced due to the lack of applicable examples.

To solve these problems, we propose a method to increase the accuracy of malicious Web page detection by correcting the classification of a conventional text-based algorithm based on the dependency relations of the malicious keywords and their neighboring segments. In addition, we propose a practical algorithm to increase performance by expanding the malicious segment pairs using a thesaurus. In our experiments, we conduct a large-scale performance evaluation using 220,000 manually labeled Japanese Web pages as training data for our method, and another 20,000 Web pages for the evaluation itself. The experimental results show our method increases both the recall and precision of malicious document detection.

Related to the latter method of analyzing HTML of Web pages, a proposal involves the use of categorization algorithms of Web pages based on their HTML features such as the number of image files inserted and the number of the links of Web pages [55]. In the algorithms of [55], the authors observe Web pages and find some useful features whereby malicious Web pages and harmless Web pages can be classified and combine them using Bayesian networks. However, enhancing the accuracy of classification based on manually observed features is quite difficult because of the lack of generality in Web pages observed. For example, the authors of [55] identified the number of image files inserted and the number of links as useful features; however, in our preliminary experiments employing these features and using 10 thousand malicious Web pages and 10 thousand harmless Web pages, we found that the features had low ability to correctly identify Web pages. In our experiments, when detecting Web pages containing more than 10 links as malicious, 75.7% of all malicious Web pages were identified as being malicious (recall rate of 75.7%), while in fact only 56.8% of the detected Web pages were actually malicious (precision rate of 56.8%).

We propose a high-speed, accurate method for detecting malicious Web pages by analyzing HTML of Web pages. In order to extract effective features for detection of malicious documents, our method automatically chooses strings based

on statistics that appear especially in the HTML elements of malicious Web pages. We use these strings in combination as features of SVMs (support vector machines) in order to detect malicious Web pages. Since our method does not rely on the text parts of Web pages, it can detect those Web pages that existing text-based methods have difficulty in detecting. We conduct a large-scale performance evaluation using manually labeled Web pages. The experimental results show that the hybrid method of the HTML-based method and existing text-based methods increased the performance of malicious document detection compared to either of the methods.

The third method aims at extracting user profiles such as age and gender. Extracting author information from the Web has been attempted for long time. An extracting method from Web information sources is proposed for the purpose of judging whether the information is trustworthy or not [56]. Koppel et al. classify three author attribution problems [57]: the profiling problem, where the challenge is to provide as much demographic or psychological information as possible about the authors [58, 59, 60]; the needle-in-a-haystack problem, where there are many thousands of candidates for each of whom we might have a very limited writing sample [61]; and the verification problem, where the challenge is to determine whether the target is the author or not [62]. The problem that we tackle in this thesis is considered a form of the profiling problem. Common approaches to author profile estimation from documents are inputting the volume of each term that appears in the document into classifiers. Argamon et al. estimate the authors' age, gender, native language, and personality from blogs and essays that university students write [58]. Estival et al. estimate age, gender, nationality, education level, and native language from English e-mails [59]. Pham et al. estimate age, gender, and area from Vietnamese blogs [60]. However, in these previous studies, the evaluations are only on the original platform, such as blog, essay, or e-mail.

Considering practical use, however, we found that a large number of users have few posts or few features on their posts. Text-based demographic estimation methods lack the accuracy for those users. To solve these problems, we propose a hybrid of a text-based method and a community-based method for demographic estimation of users. The text-based method estimates the users who have plentiful text features on their posts. For the rest of users, the

community-based method analyzes their related users such as followers and followees who have plentiful text features. The hybrid method covers almost all users by making the most of the information of users and their related users which includes both text information and community information. In the text-based method, characteristic terms used by each demographic segment are automatically detected based on linguistic and statistical analysis by tracking the content of users' past posts. In the community-based method, demographic information is estimated from the relation of the target user and other users. Existing approaches do not always work well when a user has multiple demographic categories at the same time such as age, gender, and area of residence. In the proposed method, characteristic biases in the demographic segments of users are detected from the community groups constructed by clustering their related users. For example, a user has several community groups, such as local friends, work colleagues and hobby groups, where the members of each group have something in common such as age, gender and regional area. We conduct a large-scale performance evaluation using Twitter platform. The experimental results show that the estimation accuracy of user profiles is high enough for practical use.

The rest of this thesis is organized as follows. The next chapter reviews related work. Chapter 3 details automatic rule generation approach for morphological analysis of peculiar expressions. Chapter 4 reffers to malicious document detection based on dependency relations and thesaurus. Chapter 5 shows the detection algorithm of malicious Web pages based on HTML elements. In chapter 6, we propose a hybrid method for user profiling based on text and community mining. Finally, chapter 7 summarizes this thesis.

Chapter 2

Related Work

2.1 Linguistic Analysis Techniques

To date, we have found no direct researches which involve automated text normalization for improving morphological analysis of blog documents. However, we have identified a similar work for improving morphological analysis towards casual expressions. For example, the "Dictionary Expansion" algorithm [50] focuses on accuracy improvement of morphological analysis for Japanese colloquial expressions in Web chat applications. The authors find out the rules that generate colloquial expressions from formal expressions registered on morphological dictionaries. For example, they define rules like *"adjective words tend to be inserted with a prolonged sound symbol"* from the case seen in Web chat applications, where the formal adjective "amazing" tend to be written as "amazzzzing". Since this method relies heavily on the subjective view and skills of the operator, it is difficult to generate generic rules that can be usefully applied to many expressions. In [51], our experiments with 2,000,000 blogs show 37.2% of all the sentences affected by the algorithm of [50] increase errors in word segmentation. This suggests the algorithm proposed in [50] is not versatile enough to apply to any type of text data, including blogs.

Linguistic analysis of colloquial expressions is reported in [63, 64, 65]. The approaches shown in these papers also use manually generated rules, which require specialized knowledge to generate. Considering research to reduce the number of unknown words, the method for fluctuations of words written in *katakana* format is offered in [66], while the algorithm in [67] automatically

obtains new words from Web pages. The estimation algorithm for parts of speech of unknown words is offered in [68].The estimation algorithm for word segmentation in Japanese sentences is offered in [69]. These works are not focused on recognizing peculiar expressions on blogs.

In our previous work [51], we proposed the "Initial Rule Expansion" algorithm, which automatically creates highly accurate rules from manually given low-level rules based on the statistics. For example, given rules such as, (a) remove "@" and (b) replace "@" with "a" will result in specific rules (a) "I @m → I m" and (b) "I @m → I am" ("X→Y" means substitution of Y for X). Then, (b) "I @m → I am" is stored due to its statistically high correctness. Although this method is effective in reducing the number of unknown words of peculiar expression, only a limited number of sentences are modified by the rules. Due to many kinds of peculiar expressions, it is almost impossible to manually cover all the initial rules.

2.2 Techniques for Malicious Document Detection

Several text-based algorithms to detect malicious Web pages have been proposed [52, 53]. The algorithms proposed in [52] automatically choose those words that appear especially in malicious Web pages of the training datasets. These algorithms classify evaluation datasets by simple keyword matching algorithms without morphological analysis, which reduce the processing time, but also reduce their precision. Although combination algorithms of several keywords and more advanced-type language analysis such as dependency parsing increase the precision, such high-intensity processing increases the total processing time. The algorithms proposed in [53] calculate the malicious score of datasets for evaluation based on the similarity of the feature vectors extracted from training datasets and evaluation datasets. These algorithms require morphological analysis of evaluation datasets, which increases the processing time.

Another proposal involves the use of categorization algorithms of Web pages based on their HTML features such as the number of image files inserted and the number of the links of Web pages [55]. In the algorithms of [55], the authors observe Web pages and find some useful features whereby malicious Web pages

and harmless Web pages can be classified and combine them using Bayesian networks. However, enhancing the accuracy of classification based on manually observed features is quite difficult because of the lack of generality in Web pages observed. For example, the authors of [55] identified the number of image files inserted and the number of links as useful features; however, in our preliminary experiments employing these features and using 10 thousand malicious Web pages and 10 thousand harmless Web pages, we found that the features had low ability to correctly identify Web pages. In our experiments, when detecting Web pages containing more than 10 links as malicious, 75.7% of all malicious Web pages were identified as being malicious (recall rate of 75.7%), while in fact only 56.8% of the detected Web pages were actually malicious (precision rate of 56.8%).

In text-based algorithms such as [52] and [53], documents are split into morphemes thus the contexts where they appear are ignored. As a result, they have difficulty in accurately discriminating between documents that contain morphemes that are classified both as malicious and harmless depending on the context, such as “kill” and “drugs”.

Accurate document retrieval algorithms based on dependency relations have been proposed [54, 70]. The algorithms proposed in [54] deal with Japanese documents. The algorithms split documents into morphemes, analyze their dependency relations, and make a binary tree called a “Structured Index” in advance. Users’ queries are written in pseudo-natural language, and are also analyzed into the Structured Index and matched with the documents. The algorithms described in [70] expand users’ query words by extracting contextual terms and relations from external documents. These algorithms aim at improving the accuracy of document retrieval by using the dependency relations of morphemes. Although the aims of our algorithms are different, focusing on dependency relations to detect malicious documents with high accuracy is a promising approach.

In addition, term expansion using a thesaurus is also a promising approach to improving performance. Query expansion algorithms are well researched and many types of algorithms have been proposed [71, 72, 73, 74]. Liu’s group [71] offers methods to improve query expansion for ambiguous words. In [72], they present an approach to combining WordNet and ConceptNet by assigning appro-

priate weights for expanded terms. Yoshioka's group [73] proposes algorithms to modify a given Boolean query by using information from a relevant document set by combining probabilistic and Boolean IR models. In the recent research in [74], query expansion algorithms based on users' browsing histories are proposed. In their algorithms, Web pages are clustered into a Web community and each query is represented by the Web communities to which its accessed Web pages belong.

A part of our contribution has been reported in [75]. Going beyond this achievement, we offer practical query expansion algorithms with a thesaurus that improve the detection accuracy for malicious documents and reduce the computation load and memory consumption by removing noisy segment pairs.

2.3 User Profiling Techniques

Extracting author information from the Web has been attempted for a long time. An extracting method from Web information sources is proposed for the purpose of judging whether the information is trustworthy or not [56]. Koppel et al. classify three author attribution problems [57]: (1) the profiling problem, where the challenge is to provide as much demographic or psychological information as possible about the author [58, 59, 60]; (2) the needle-in-a-haystack problem, where there are many thousands of candidates for each of whom we might have a very limited writing sample [61]; and (3) the verification problem, where the challenge is to determine whether the target is the author or not [62]. The problem that we tackle in this thesis is considered a form of (1).

Common approaches to author profile estimation from documents are inputting the volume of each term that appears in the document into classifiers. Argamon et al. estimate the authors' age, gender, native language, and personality from blogs and essays that university students write [58]. Estival et al. estimate age, gender, nationality, education level, and native language from English e-mails [59]. Pham et al. estimate age, gender, and area from Vietnamese blogs [60]. However, in these previous studies, the evaluations are only on the original platform, such as blog, essay, or e-mail. From the viewpoint of applying the method to applications, the method of application and the performance of the application should also be considered. In this thesis, we propose a method and its application applicable to Twitter, which is one of the largest,

most common, and most world-wide platform among social media.

There are challenges associated the author attribution problem of (2) and (3) on the Twitter platform [61], [62]. Layton et al. show that the important threshold was 120 tweets per user, at which point adding more tweets per user gave a small but non-significant increase in accuracy for the author attribution problem [76]. Silva et al. show that markers include highly personal and idiosyncratic editing options, such as emoticons, interjections, and punctuation, which are often seen in casual SNS such as Twitter [77]. In these studies, only text information was used, which was considered to limit accuracy. We propose a hybrid method comprising a text-based method and a community-based method, which enhances the accuracy.

Follower and followee relationships are regarded as directed links between two users. From this viewpoint, there has been some research reporting link-based document classification methods, such as for scientific papers based on co-citations [78] and for Web pages based on their hyperlinks [79]. Hybrid methods composed of text-based methods and link-based methods are reported to improve classification accuracy for Web pages [80, 81, 82, 83]. Calado et al. show that a classification method based on co-citation is effective [80]. Qi and Davison show that the contents and topic information of neighboring documents increased the accuracy of classification [81]. Zhang et al. propose an optimization method for text-based and graph structures to categorize Web pages [82]. These contributions are helpful for improving the performance of text-based methods; however, in most existing link-based approaches, a target object is classified into a single category. However, a twitter user belongs to multiple demographic categories, such as age, gender, and area of residence, so existing approaches do not always work well. In the proposed method, characteristic biases in the demographic segments of users are detected from the community groups constructed by clustering their followers and followees. For example, a user belongs to several community groups, such as local friends, work colleagues, and hobby groups, where the members of each group have something in common.

We have previously proposed a text-based demographic estimation method for Twitter users and its application to broadcast television programs [84]. In this previous work, the demographic estimation method was limited in terms

of the minimum functions required for consumer usage. Only the basic demographic categories such as age, gender and area of residence were estimated. Estimation of the users with few tweets was outside of its scope. Going beyond the previous work, in this work, we have conducted inquiry surveys of companies to realize a practical market analysis application. We found that (1) more varied demographic categories should be estimated, such as occupation, hobby, and marital status, and (2) the estimation of users with few tweets such as followers of corporate accounts is also important. We proposed a hybrid method to estimate multiple user demographics and also the users with few tweets by combining text-based and community-based estimation methods, and executed additional evaluation for various demographic categories.

Chapter 3

Automatic Rule Generation Approach for Morphological Analysis of Peculiar Expressions

3.1 Introduction

In this chapter, we propose a new method that automatically retrieves substitution candidates for peculiar expressions from formally written documents such as newspapers. For the correct replacement, a substitution rule is selected based on three criteria; its appearance frequency in the retrieval process, the edit distance between substituted sequences and the original text, and the estimated accuracy improvements of word segmentation after the substitution. We have compared the performance of our method to conventional two methods. In our experiments, we modify 100,000 blog sentences and evaluate the number of unknown words and the accuracy of word segmentation. In this chapter, we evaluate the performance of the proposed method for blogs; however, the essential algorithm of automatic rule generation for the modification of peculiar expressions are applicable to other media such as BBS, Web pages and SNSs. In the rest of this chapter, we describe the detail algorithm of the proposed method in Section 3.2, the results of performance evaluations in Section 3.3, and conclude this chapter in Section 3.4.

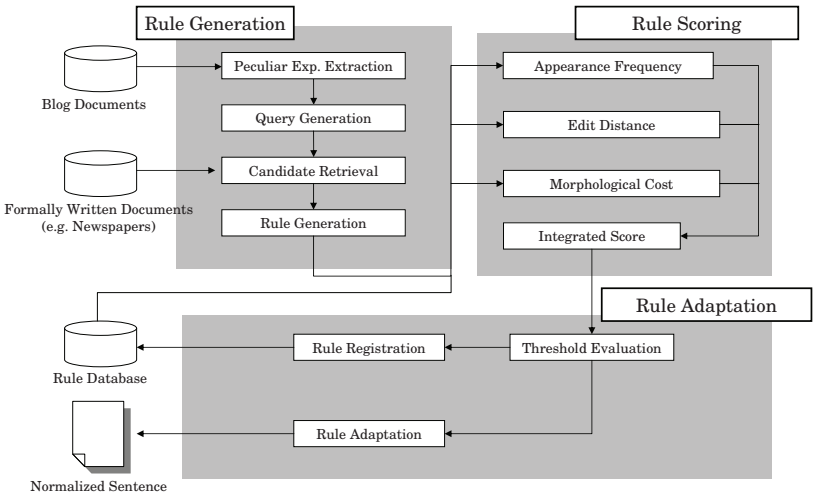


Figure 3.1: Overview of the Substitution Algorithm

3.2 Algorithm Design

Figure 3.1 shows an overview of our algorithm. Inputs of the algorithm are blog documents and formally written documents such as newspapers. Substitution candidates for a peculiar expression are listed up from formally written documents. Substitution rules are generated from the peculiar expression and its substitution candidates. Substitution rules are scored based on the following three criteria; (1) the appearance frequency in the retrieval process, (2) the edit distance between substituted sequences and original text, and (3) the estimated accuracy improvements of word segmentation after the substitution. A substitution rule with the highest score is selected as the most suitable expression for the context.

3.2.1 Generation of Substitution Rules

The rule generation algorithm has four steps, (1) extraction of a peculiar expression, (2) generation of a query for substitution candidates of the peculiar expression, (3) retrieval of the substitution candidates, and (4) generation of rules. Table 3.1 shows an example of rule generation. The sentence

Table 3.1: Generation of Substitution Rules

```
(1) Extraction of a peculiar expression
    Blog sentence: "dekirukadouxkaxwawakarimasen"
    Word segmentation: "dekiru/ka/dou/xkaxwa/wakari/mase/n"
    Unknown word: xkaxwa
(2) Generation of a query for substitution candidates
    "dou * wakari"
(3) Retrieval of the substitution candidates
    "... dou   kaha   wakari..."
    "... dou   ka   wakari..."
    "... dou   shitaraiinoka   wakari..."
    "... dou   kaha   wakari..."
    "... dou   ka   wakari..."
    "... dou   kaha   wakari..."
    "... dou   kaha   wakari..."
    "... dou   natteiruka   wakari..."
(4) Generation of substitution rules
    "xkaxwa -> kaha"
    "xkaxwa -> ka"
    "xkaxwa -> shitaraiinoka"
    "xkaxwa -> natteiruka"
"X->Y" means substitution of Y for X
```

“dekirukadouxkaxwawakarimasen” (“I wonder whether it is possible.”, in English) contains the peculiar expression “xkaxwa” and this expression should be substituted by the formal expression “kaha”. Most peculiar expressions are detected as unknown words in the morphological analysis because they are not listed in morphological dictionaries ((1) of Table 3.1 shows the peculiar expression “xkaxwa” is detected as an unknown word). Substitution candidates are retrieved from formally written documents. A query for substitution candidates is created by extracting the unknown word with its adjoining morphemes, and replacing the unknown word with the wild-card symbol. ((2) of Table 3.1 shows the peculiar expression “xkaxwa” and its adjoining morphemes “dou” and “wakari” are taken out, and “xkaxwa” is replaced with the wild-card symbol, “*”). As a result of the retrieval, substitution candidates are obtained ((3) of Table 3.1). Substitution rules are generated from the peculiar expression and the parts matched with the wild-card of the query ((4) of Table 3.1, where

Table 3.2: Scoring Based on Appearance Frequency

Substitution Rules	Appearance Frequency	Normalized Frequency
xkaxwa → kaha	4	0.5
xkaxwa → ka	2	0.25
xkaxwa → shitaraiinoka	1	0.125
xkaxwa → natteiruka	1	0.125

Table 3.3: Scoring Based on Edit Distance

Substitution Rules	Edit	Edit Distance
xkaxwa → kaha	2 Substitution, 1 Insertion	3
xkaxwa → ka	4 Deletion	4
xkaxwa → shitaraiinoka	5 Substitution, 7 Insertion	12
xkaxwa → natteiruka	5 Substitution, 4 Insertion	9

"X→Y" means substitution of Y for X). The substitution rules obtained by the above algorithm are scored based on the criteria described in Section 3.2.2.

3.2.2 Scoring Substitution Rules

Substitution rules are scored based on the following criteria.

Scoring Rules Based on the Appearance Frequency

The expressions appearing in the similar contexts to the peculiar expressions are expected to be suitable substitution candidates. The amount of retrievals for each substituted expression represents its appearance frequency. Table 3.2 is a summary of the appearance frequency of each substitution candidate retrieved from (3) of Table 3.1. The candidate "kaha" gains a high score because it often appears in a similar context to the peculiar expression "xkaxwa". The appearance frequency is divided by the total number of retrievals for normalization so as not to depend on the number of retrievals.

Scoring Rules Based on the Edit Distance

Since most peculiar expressions are derived from formal expressions, rules which greatly change a peculiar expression are considered not to be the best substi-

tution candidate. The edit distance such as the Levenshtein distance [85] is a criterion for measuring the amount of difference between two character strings. The edit distance between two strings is given by the minimum number of operations needed to transform one string into the other, where an operation is an insertion, deletion, or substitution of a single character. For example, the edit distance of the word "forum" and "farm" is 2 since the substitution of "a" for "o" and the deletion of "u" is the minimum way to change the former into the latter. Table 3.3 is a summary of the edit distance in each substitution rule generated in the example of Table 3.1. Substitution rules with a large edit distance gain a low score such as, "xkaxwa → shitaraiinoka" and "xkaxwa → natteiruka".

The weighted edit distance based on the way of derivation is expected to work effectively. For example, as previously stated, people intentionally write a word in *katakana* format on blogs, which is normally written in *hiragana* format on formal documents. Giving a small edit distance between *katakana* and *hiragana* characters is effective. The visual similarity of two characters and the similarity in the pronunciation of two words should also be reflected on the edit distance, which is our future work.

Scoring Rules Based on the Estimated Accuracy Improvements of Word Segmentation

Mistakes in rule selection lead to the generation of ungrammatical sentences. In our algorithm, morphological analysis cost is used for evaluating the relative unnaturalness of sentences. Morphological analysis cost is calculated from the appearance probability of a word and the joint probability of each word [23]. The validity of rule adoption is evaluated by comparing the morphological analysis costs of a sentence with peculiar expressions and substituted sentences.

Table 3.4 shows morphological analysis costs of each substituted sentence obtained in Table 3.1. Each sentence is segmented by a morphological analyzer. Each segment is given a morphological analysis cost (accumulated cost from the beginning of each sentence). The morphological analysis cost at the end of each sentence is considered to show the grammatical correctness of the whole sentence. Morphological analysis cost around peculiar expressions such as "xkaxwa" become higher since appearance probabilities and joint probabilities of peculiar expressions are low compared to formal expressions. The score of

Table 3.4: Scoring Based on Morphological Analysis Cost

```
Blog Sentence: dekirukadouxkaxwawakarimasen
Segmentation:  dekiru|ka|dou|xkaxwa|wakari|mase|n
Morph. cost:   5742|8263|11751|34685|39098|40388|39914
Total cost:    39914

Candidate 1 :  dekirukadoukahawakarimasen
Segmentation:  dekiru|ka|dou|ka|ha|wakari|mase|n
Morph. cost:   5742|8263|11751|14430|15438|19341|20631|20157
Total cost:    20157
Difference from before substitution: -19757

Candidate 2:   dekirukadoukawakarimasen
Segmentation:  dekiru|ka|douka|wakari|mase|n
Morph. cost:   5742|8263|16737|20120|21410|20936
Total cost:    20936
Difference from before substitution: -18978

Candidate 3:   dekirukadoushitaraiinokawakarimasen
Segmentation:  dekiru|ka|dou|shj|tara|ii|no|ka|wakari|mase|n
Morph. cost:   5742|8263|11751| ... |26035|27325|26851
Total cost :   26851
Difference from before substitution: -13063

Candidate 4:   dekirukadounatteirukawakarimasen
Segmentation:  dekiru|ka|dou|na|tte|iru|ka|wakari|mase|n
Morph. cost:   5742|8263|11751| ... |22975|24265|23791
Total cost:    23791
Difference from before substitution: -16123
```

a rule is defined as the difference of the morphological analysis cost between a substituted sentence and its original sentence. Although shorter sentences tend to get lower morphological analysis costs, rules deleting many characters get large edit distances. In the following section, we explain the calculation of the integrated score of a rule based on the above criteria.

Calculation of the Integrated Score

The integrated score of a substitution rule *score* is generally defined as Expression 3.1, where $freq$ is the appearance frequency of the substituted expression,

Table 3.5: Integrated Scores Based on Criteria ($\alpha = 1, \beta = -16, \gamma = -0.005$)

Substitution Rules	Freq. (%)	Edit Dist.	Morph. Cost	Total Score
tetaxyo→teta	20	3	-15757	83.3
tetaxyo→tetayo	0	1	-14037	54.2
tetaxyo→tetai	2	3	-10946	40.7
tetaxyo→tetatoieyou	2	4	-9108	-32.5
kyouxwahayameni →kyouhahayameni	33	2	-12721	80.6
kyouxwahayameni →kyoude, hayameni	0	3	-16449	50.2
kyouxwahayameni →kyouhasukoshihayameni	11	7	-13205	29.0
xokanenai→kanenai	0	2	-13131	49.7
xokanenai→zeikinnnai	8	6	-9887	41.4
xokanenai→okanenai	0	1	-10974	38.9
xokanenai→ukanenai	4	1	-6654	21.3

dist is the edit distance between the substituted expression and original peculiar expression, *cost* is the difference of the morphological analysis cost between the substituted sentence and the original one, and f, g, h are the functions for weighting criteria. In this chapter, functions f, g, h are simply defined as constants $\alpha, \beta, and \gamma$ as shown in Expression 3.2.

$$score = f(freq) + g(dist) + h(cost) \tag{3.1}$$

$$score = \alpha \cdot freq + \beta \cdot dist + \gamma \cdot cost \tag{3.2}$$

As an example, Table 3.5 shows examples of substitution rules that appeared in the experiments in Section 3.3. The values of constants in Expression (3.2) are set as $\alpha = 1, \beta = -16, \gamma = -0.005$. According to the table, the peculiar expression "tetaxyo" should be substituted by "teta" based on its appearance frequency and morphological analysis cost. The expression "tetatoieyou" retrieved as substitution candidates gets a lower score because of its large edit distance from its original expression. In the same manner, "kyouxwahayameni" should be substituted by "kyouhahayameni" based on its appearance frequency and edit distance. "xokanenai" ("no money") should be substituted by "kanenai" based on its morphological analysis cost. In this case, however, "zeikinnnai" ("no tax") also gets a relatively high score due to its high appearance frequency

and low edit distance. In Section 3.3, we evaluated the miss ratio of substitution, where the meanings of sentences have changed.

3.2.3 Adoption and Registration of Substitution Rules

Adoption of a substitution rule is decided depending on whether its score is higher than a given threshold. The number of substitutions and its accuracy are in a trade-off relation. In Section 3.3, we evaluated the trade-off by monitoring the reduced number of unknown words and word segmentation accuracy on several thresholds.

In our algorithm, rules with higher scores than the given threshold are registered on the database. When few substitution candidates are obtained from formally written documents, additional substitution rules are available from the database. When automatically created queries happen to be poor due to the neighbor characters of peculiar expressions, many unrelated expressions are retrieved. For example, a query generated from the sentence "kids are $c\mu\tau e$" may be "kids are *" which retrieves any kind of adjectives as substitution candidates. Calculation and comparison of many substitution candidates require much time, and substitution for them tends to result in errors. In this case, only the rules on the database are used because peculiar expressions are expected to be correctly substituted in other cases and correct rules are stored on the database.

3.3 Performance Evaluation

We implement our algorithm and compare its performance with two conventional algorithms, (a) "Dictionary Expansion Algorithm (DEA)" [50] and (b) "Initial Rule Expansion Algorithm (IREA)" [51]. The problem with using DEA is the high error ratio on the word segmentation due to the over adoption of rules. The problem with using IREA is the lack of scalability, where only limited sentences are normalized by the manually given initial rule sets. Considering these problems, we evaluate the reduction of unknown words and the accuracy of word segmentation. We also evaluate the changes in the meanings of sentences because our substitution algorithm may greatly change the meanings of sentences due to their nature as an unsupervised algorithm. The trade-off relation of our algorithm is shown on several thresholds.

Table 3.6: Categorization of Substitution Rules Based on their Effects on the Meanings

Befor Substitution	After Substitution
(a) Small Changes in the Meanings	
kyohanekochankitetaxyo-	kyohanekochankiteta
(“A cat stayed here today”)	
toxottemokimochiii	tottemokimochiii
(“so good feelings”)	
oishisoxodattandakedone	oishisoudattandakedone
(“It seemed delicious.”)	
mextsucchaasekaita	mecchaasekaita
(“I was sweating a lot.”)	
(b) Large Changes in the Meanings	
jaa,② jiniekimaede	jaa, 7jiniekimaede
(“See you on the station at 2.”)	(“See you on the station at 7.”)
bakappuru-	bakkassuru-
(“couples”)	(no meaning)
kawaisugixi	kawaisuginai
(“so cute”)	(“not so cute”)
ohhayoxo-	ohhayoi
(“Hello”)	(no meaning)
(c) Difficult to Categorize in (a) or (b)	
koxohennyoxtsu	kohennyo
(“I will not come”)	
watashimoosokunarutokixaruU	watashimoosokunarutokimoaru
(“I sometimes late”)	
zehixotameshiare	zehitameshiare
(“Lets try it”)	
hahahayokattane	yokattane
(“Thats good”)	

3.3.1 Experimental Settings

We execute morphological analysis of blog sentences by using DEA, IREA, and our algorithm and compare their performance based on the following four criteria; (1) The ratio of sentences improving the word segmentation accuracy to the total modified sentences. (2) The ratio of sentences deteriorating the word segmentation accuracy to the total modified sentences. (3) The ratio of sentences changing their meanings to the total modified sentences. (4) The ratio of unknown words to the total morphs.

The accuracy of word segmentation is defined in the same way as the conventional methods [50, 51, 86], where word segmentation is correct when it is divided in the same manner as that performed manually. The segmentation improvement of a sentence is defined as the case where the original sentence has word segmentation errors and the substituted sentence has no errors. The

segmentation deterioration of a sentence is defined as the opposite of the improvement case. Changes in the meanings of sentences are defined in three types, (a) almost no change, (b) obviously changed or the meanings of the sentence cannot be understood, and (c) difficult to categorize in categories (a) or (b). Table 3.6 shows examples of substitutions categorized in categories (a), (b), and (c). Substitutions in category (a) correctly replace peculiar expressions with formal expressions. They may cause a slight change in the impression of sentences, but no change in the meanings and the facts. Substitutions in category (b) obviously change the facts of sentences or make them incomprehensible. Substitutions in category (c) do not change the facts of sentences, but their impressions are slightly different. We define that substitutions in categories (b) and (c) change the meanings of sentences. In the following experiment, the ratio of improvement and deterioration in word segmentation and the ratio of the changes in the meanings are manually evaluated by sampling 600 sentences modified in each algorithm.

As a Japanese morphological analyzer, we use MeCab [23]. We additionally register 180,000 nouns or proper nouns, new words, and so forth since neither our substitution algorithm nor conventional algorithms focus on dealing with those words. The detail of experimental environments is as follows.

- Morphological Analyzer: MeCab version 0.97
- Morphological Dictionary: MeCab standard dictionary (IPADIC version 2.7.0) plus 180,000 nouns.
- Terminal Configuration: 8 CPUs of 2.33 GHz, 64GB RAM, Linux OS version 2.6.24, gcc version 4.1.2.
- Blog Documents: 1,000,000 sentences for the machine learning in IREA and the other 100,000 sentences for the targets of modification.
- Formally Written Documents: 1,000,000 sentences from Japanese newspapers.

3.3.2 Experimental Results

In our algorithm, the adoption of a substitution rule is decided according to the threshold. First, we evaluate the trade-off relation between the substitution

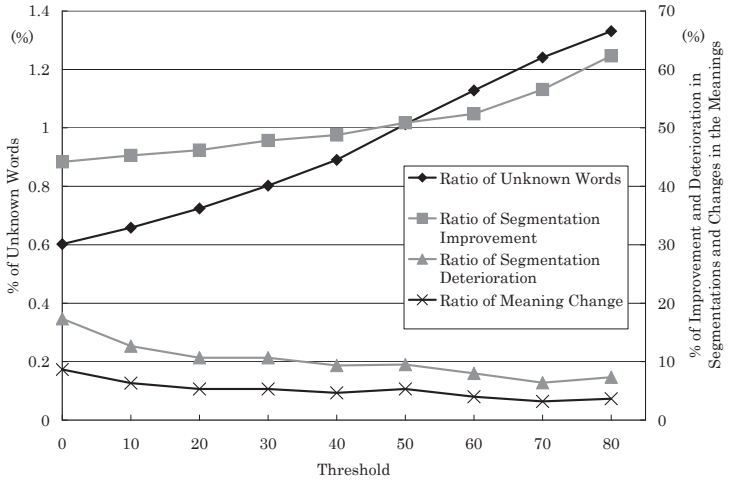


Figure 3.2: Ratio of Improvement and Deterioration in Word Segmentation Accuracy, Ratio of Changes in the Meanings of Sentences, and Ratio of Unknown Words on the Threshold 0 to 80 in our Algorithm.

accuracy and the ratio of unknown words. Figure 3.2 shows the ratio of improvement and deterioration in word segmentation, the ratio of changes in the meanings of the sentences, and the ratio of unknown words when the threshold changes from 0 to 80. When the threshold is low, many unknown words are replaced and the ratio of unknown words is low. However, manual evaluation of the accuracy in word segmentation and the ratio of changes in the meanings shows that most substituted words are recognized in other meanings on morphological analyzer. As the threshold becomes higher, the ratio of unknown words increases, but the accuracy of word segmentation and the ratio of changes in the meanings improve.

Table 3.7 shows the ratio of improvement and deterioration in word segmentation accuracy, the ratio of change in the meanings of sentences and the ratio of unknown words in each algorithm. We tune the threshold of our algorithm to 60 in order to maintain the same level in word segmentation accuracy and the ratio of changes in the meanings as IREA. Our algorithm has higher word segmentation accuracy and smaller ratio of unknown words compared to those in DEA. The ratio of unknown words in our algorithm is 1.128%, where the reduction ratio is 30.3% from 1.619% of MeCab. The reduction ratio of unknown words

Table 3.7: Performance Evaluation of Each Algorithm

Algorithm	Improvement(%)	Deterioration(%)	Meaning Change(%)	Unknown Words(%)
MeCab	-	-	-	1.619
DEA	48.1	32.1	-	1.458
IREA	52.1	9.7	4.0	1.377
Ours	52.4	8.0	4.0	1.128

in our algorithm is twice of that in IREA. This is because, in our algorithm, more substitution rules are obtained by the unsupervised algorithm, and the three criteria for accurate rule selection enable maintaining the word segmentation accuracy and the changes in the meanings of sentences in the same level as IREA. Considering scalability, the execution time of our algorithm is only 1 second to modify 1,000 blog sentences in the case of using 1,000,000 newspaper sentences.

3.4 Conclusion

In this chapter, we propose a method for reducing the number of unknown words by replacing peculiar expressions seen in blog documents with formal expressions. In our algorithm, candidates for the substitution of peculiar expressions are automatically retrieved from formally written documents and stored as substitution rules. In order to replace a peculiar expression with the most suitable expression for the context, a substitution rule is selected based on three criteria; its appearance frequency in the retrieval process, the edit distance between substituted sequences and the original text, and the estimated accuracy improvements of word segmentation after the substitution. The experimental results show our algorithm reduces 30.3% of unknown words in original blog documents at the same segmentation accuracy as conventional ones. This reduction rate is higher than twice of the rate of the conventional algorithm.

Chapter 4

Malicious Document Detection Based on Dependency Relations and Thesaurus

4.1 Introduction

In this chapter, we propose a method to increase the accuracy of malicious information detection by correcting the classification of a conventional text-based method based on the dependency relations of the malicious terms and their neighboring segments. In order to apply this method for many examples, we also propose a practical algorithm to increase performance by expanding the malicious segment pairs using a thesaurus. In our experiments, we conduct a large-scale performance evaluation using 220,000 manually labeled Japanese Web pages as training data for our method, and another 20,000 Web pages for the evaluation itself. The experimental results show our method increase both the recall and precision of malicious document detection. In this chapter, we evaluate the performance of the proposed method by focusing on malicious Web pages. The proposed method is applicable to other kinds of information such as words of mouth on blogs although it may be difficult to extract dependency relations from extremely short messages on Twitter or BBS. In the rest of this chapter, we describe the detail algorithm of the proposed method in Section 4.2, the results of performance evaluations in Section 4.3, and conclude this chapter

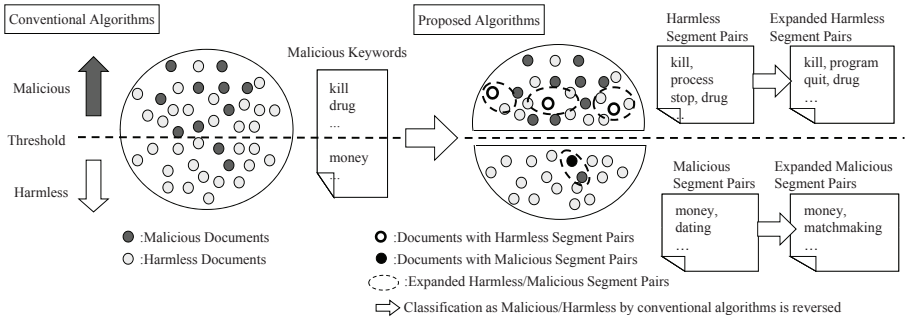


Figure 4.1: Overview of the Conventional Algorithms and the Proposed Algorithms

in Section 4.4.

4.2 Proposed Algorithms

In this chapter, we treat the algorithms of [52] as conventional text-based algorithms. Figure 4.1 shows an overview of the conventional and the proposed algorithms. Documents are classified into either malicious or harmless. The conventional algorithms have an automatically generated set of keywords with a malicious score.

Documents that contain any keywords with a higher score than a threshold are detected as malicious and the rest are considered harmless. In Figure 4.1, for example, the words “kill” and “drug” are regarded as malicious because their malicious scores are higher than the threshold. In contrast, words with low malicious scores such as “money” are regarded as harmless.

Although documents that contain keywords with high malicious scores are likely to be malicious, not all of them are malicious. For example, documents which contain sentences such as “kill a process” are not malicious even though they contain the malicious keyword “kill”. In the same way, documents that contain sentences such as “make money with dating” are malicious even though the keyword “money” is harmless. Our algorithms correct errors and improve accuracy by detecting malicious/harmless segment pairs from documents classified as harmless/malicious, respectively. In addition, we propose algorithms

Table 4.1: Number of Malicious/Harmless Documents where Morpheme m Appears

	Documents do contain m	Documents do not contain m	Sum
Malicious Documents	$N_{11}(m)$	$N_{12}(m)$	N_p
Harmless Documents	$N_{21}(m)$	$N_{22}(m)$	N_n
Sum	$N(m)$	$N(\neg m)$	N

Table 4.2: Example of the Obtained Malicious Keywords

Ranks	Keywords	$N_{11}(m)$	$N_{12}(m)$	$N_{21}(m)$	$N_{22}(m)$	$E(m)$
10	Actress	5802	102724	194	10833	6746
17	Blog	1091	97615	3354	10517	4495
46	Mobile phone	9253	99273	3259	10526	3167
106	Sponsor	2561	105965	708	10781	1129
110	Access	6573	101953	3361	10516	1105

to expand segment pairs with a thesaurus. For example, in our algorithms, the keyword "kill" is harmless when it appears as a segment pair with "process". In addition, expanded segment pairs such as "kill program" and "kill computation" are also regarded as harmless.

In the following section, we describe the conventional generation algorithms of the malicious keyword set in Section 4.2.1. We describe the proposed generation algorithms for malicious/harmless segment pairs in Section 4.2.2. Expansion algorithms for generated segment pairs are shown in Section 4.2.3.

4.2.1 Generation of Keyword Set

First we describe the conventional generation algorithms for a malicious keyword set. Algorithms shown in [52] split the documents manually labeled as malicious/harmless into morphemes by morphological analysis and extract malicious morphemes that appear particularly often in malicious documents. $E(m)$ which is the degree of bias of a morpheme m in malicious documents is calculated based on Akaike's Information Criterion (AIC) [87]. In AIC algorithms, the four criteria shown in Table 4.1 are used, where N_{11}/N_{21} is the number of malicious/harmless documents where morpheme m appears, N_{12}/N_{22} is the

number of malicious/harmless documents where morpheme m does not appear. In [52], $E(m)$ is defined as follows by using AIC dependent/ independent models of AIC_DM and AIC_IM based on the findings described in [88].

Table 4.2 shows examples of keywords (morphemes) with high $E(m)$ scores. We used 10,000 manually labeled Web pages[1] as training datasets (5,000 malicious and harmless Web pages each). Here, Web pages that contain information on dating, criminal declarations, libelous statements and porn are labeled as malicious. Table 4.2 shows how some keywords with high scores seem to be harmless. In contrast, our proposed algorithms aim to detect malicious documents with high accuracy by using the keywords' neighboring segments.

$$\begin{aligned} When, N_{11}(m)/N(m) \geq N_{12}(m)/N(\neg m) \\ E(m) = AIC\_IM(m) - AIC\_DM(m) \\ When, N_{11}(m)/N(m) < N_{12}(m)/N(\neg m) \\ E(m) = AIC\_DM(m) - AIC\_IM(m) \end{aligned} \tag{4.1}$$

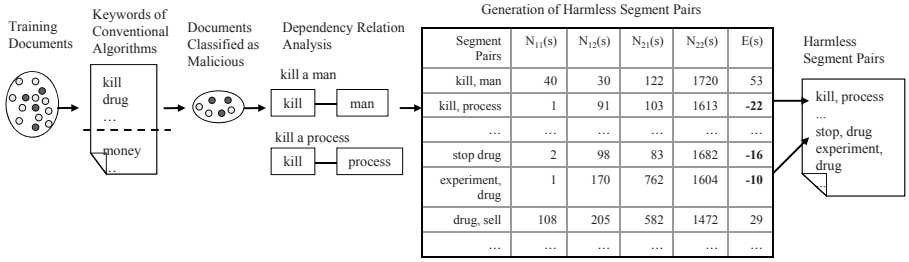
Here, $AIC\_IM(m)$ and $AIC\_DM(m)$ are defined as follows [87].

$$\begin{aligned} AIC\_IM(m) &= -2 \times MLL\_IM + 2 \times 2 \\ MLL\_IM &= N_p(m)logN_p(m) + N(m)logN(m) + N_n(m)logN_n(m) \\ &+ N(\neg m)logN(\neg m) - 2NlogN \\ AIC\_DM(m) &= -2 \times MLL\_DM + 2 \times 3 \\ MLL\_DM &= N_{11}(m)logN_{11}(m) + N_{12}(m)logN_{12}(m) \\ &+ N_{21}(m)logN_{21}(m) + N_{22}(m)logN_{22}(m) - NlogN \end{aligned} \tag{4.2}$$

4.2.2 Generation of Segment Pairs

Here we describe the algorithms for generating malicious/harmless segment pairs from the conventional harmless/malicious keyword sets. Figure 4.2 shows an overview of the generation algorithms. First, training datasets are classified as malicious or harmless by the conventional algorithms. Dependency relations

[1] Web pages are received from NetSTAR Inc. (http://www.netstar-inc.com/eng/) who engage in collection and manual classification of URLs



Segment Pairs	$N_{11}(s)$	$N_{12}(s)$	$N_{21}(s)$	$N_{22}(s)$	$E(s)$
kill, man	40	30	122	1720	53
kill, process	1	91	103	1613	**-22**
…	…	…	…	…	…
stop drug	2	98	83	1682	**-16**
experiment, drug	1	170	762	1604	**-10**
drug, sell	108	205	582	1472	29
…	…	…	…	…	…

Figure 4.2: Generation Algorithms of Harmless Segment Pairs

with malicious keywords in sentences from malicious documents are analyzed and segment pairs that contain malicious keywords are extracted. The number of malicious/harmless documents where each extracted segment pair "s" does/doesn't appear is evaluated in the same manner as in Table 4.1. In this case, the total number of documents N in Table 4.1 is defined as the number of malicious documents. Harmless segment pairs are obtained by calculating an $E(s)$ value based on expressions 4.2 and 4.3 in Section 4.2.1 In the same way, malicious segment pairs are obtained from the harmless keywords with scores below the threshold, and training datasets are classified as harmless by the conventional algorithms. For example, in Figure 4.2, "kill" and "drug" are malicious keywords, however, "kill a process" and "stop drug" are obtained as harmless segment pairs. When "money" has a below-threshold malicious score, the segment pair of "money" and "date" is malicious. Table 4.3 shows examples of segment pairs obtained from 10,000 training data sets. Segment pairs with negative scores are harmless.

Here, we describe the appropriateness of using dependency relations to reflect contexts when detecting malicious documents. The co-occurrence of morphemes is a possible approach to reflecting contexts. For example, "part-time girlfriend for a man" might be a malicious sentence found in dating Web pages and the co-occurrence of "part-time", "girlfriend" and "man" are learned as malicious. However, "a man's girlfriend quit her part-time" is a harmless sentence, yet it is regarded as harmful since it contains "part-time", "girlfriend" and "man". The dependency relationship between "part-time" and "girlfriend" does not appear

Table 4.3: Example of the Obtained Segment Pairs

Segment Pairs	$N_{11}(s)$	$N_{12}(s)$	$N_{21}(s)$	$N_{22}(s)$	$E(s)$
Actress, shot	106	144651	2	72293	74.7
Produce, Actress	0	144757	2	72293	-2.31
Sponsor, matchmaking	14	144743	1	72294	20.3
Sponsor, advertisement	2561	105965	708	10781	1129
Access, disguise	7	144750	3	72292	16.1
Access, guide	0	144757	27	72268	-9.20

in the latter sentence. Another approach to reflect contexts is using a simple n-gram; n adjacent morphemes. As a preliminary experiment, we evaluate the number of cases where 20 malicious segment pairs appear adjacently in 10,000 Web pages (for example, in the sentence "part-time girlfriend for a man", "part-time" and "girlfriend" are adjacent). The segment pairs appear 311 times, and 202 times (64.9%) they are adjacent and 109 times (35.1%) they are apart. Our algorithms are particularly effective in cases of separated segments because a bi-gram approach cannot detect them.

4.2.3 Expansion with a Thesaurus

In order to adapt segment pairs extracted in Section 4.2.2 to more expressions, we expand the segment pairs with a thesaurus. In our algorithms, morphemes that are not listed in malicious keyword sets are expanded. Figure 4.3 shows an overview of the base-line expansion algorithms in which morphemes are expanded to their one-level-higher concepts and their whole family of lower level concepts. For example, assume that "kill" is regarded as a harmless keyword in the conventional algorithms and a segment pair "kill" and "man" is extracted as malicious by the proposed algorithms, then "man" is expanded to the higher concept "human" and its entire family of lower level concepts such as "lady", "Alice", and "Bob". Here we believe that the malicious score for "kill" is almost the same as for "man" for all lower level concepts of "human".

In our implementation of the proposed algorithms, we used the Japanese EDR thesaurus [89], which consists of 410,000 concepts and 270,000 words in a tree topology. Each entry contains a concept ID, concept title, concept explanation, ID list of the upper/lower concepts, and more. Each entry can contain sev-

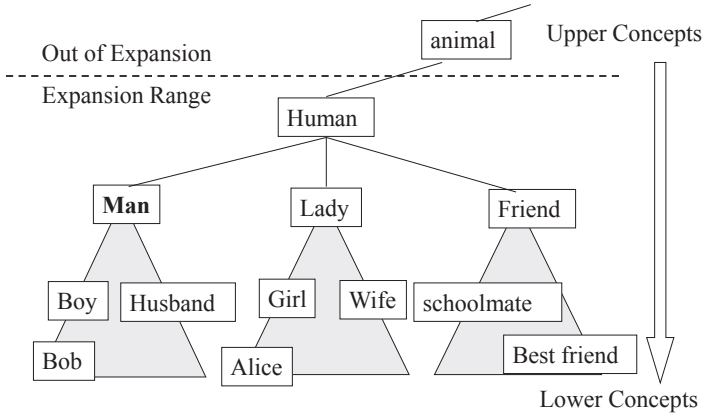


Figure 4.3: Overview of the Base-line Expansion Algorithms of Segment Pairs

eral words. For example, the concept "school" has one higher concept, which is "building for education" and 14 lower concepts, including "elementary school", "university", and so forth. The concept "school" contains the word "school" only. Due to the large number of concept entries, making use of all entries is impractical due to the huge requirement for processing time and memory for the number of expanded segment pairs. In addition to the removal of ineffective expansion paths described previously, concept entries are removed whose words do not appear in the training datasets. In our preliminary experiment with 220,000 training datasets and 10,000 evaluation datasets, we evaluate the number of words both by type and appearance that appear solely in training datasets or evaluation datasets or both. Figure 4.4 shows that only a few words appear solely in evaluation datasets, which means there is almost no need for a concept entry whose words do not appear in training datasets. The number of words in the thesaurus is reduced from 270,000 to 25,000. We compare their performance by the required processing time and memory consumption in Section 4.3.

In addition, we focus on the expansion paths and their correctness. For example, a much lower concept may differ from the original concept. We classified expansion paths into 10 types depending on (a) whether they trace a one-level-higher concept, and (b) the depth of the paths they trace (up to 5). We defined

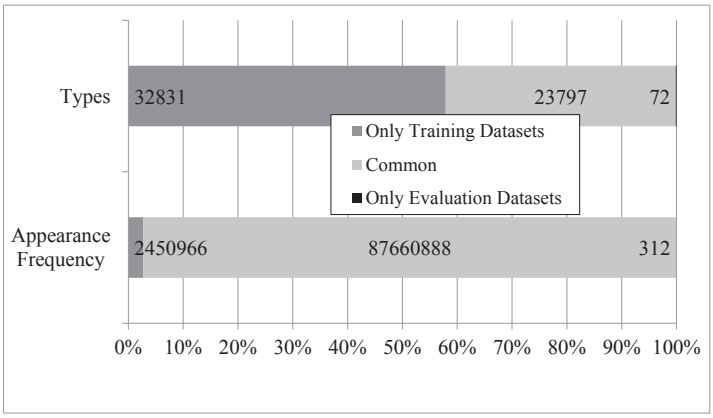


Figure 4.4: Types and Appearance of Morphemes that Appear in Datasets

expansion paths are correct when the expanded malicious/harmless segment pairs appear in malicious/harmless documents of the training datasets. Table 4.4 shows the number of expanded correct/incorrect segment pairs appearing in 200,000 Web pages. "Harmless to Malicious" and "Malicious to Harmless" mean that the classifications of documents are reversed from harmless/malicious to malicious/harmless by the proposed algorithms. In the "Harmless to Malicious" case, paths (1), (2), (6), and (7) have high correctness levels which show expansion paths to the same or nearby levels have high correctness and paths to low levels have low correctness. "Malicious to Harmless" has a similar tendency. In the experiment in Section 4.3, we confirm the improvement in the performance of the partial expansion algorithms that optimize the expansion paths by removing noisy paths, compared to the baseline algorithms of expanding to their one-level-higher concepts and all lower level concepts.

4.3 Performance Evaluation

4.3.1 Experimental Environments

Here we compare the performance of the conventional algorithms, the proposed base-line algorithms (BLA), the proposed base-line expansion algorithms (BLEA), and the proposed partial expansion algorithms (PEA). Experimental

Table 4.4: Example of the Expanded Segment Pairs in Each Path

Expansion Path	Expansion Types	Path Contain Higher Layer	# of Down Layers	Correct	Incorrect	Correct Ratio
(1)	Harmless to	No	1	433	82	84.1%
(2)	Malicious	No	2	146	36	80.2%
(3)		No	3	58	25	69.9%
(4)		No	4	41	6	87.2%
(5)		No	5	0	0	-
(6)		Yes	1	27983	1254	95.7%
(7)		Yes	2	1852	349	84.1%
(8)		Yes	3	802	216	78.8%
(9)		Yes	4	551	143	79.4%
(10)		Yes	5	280	68	80.5%
(11)	Malicious	No	1	107	23	82.3%
(12)	to Harmless	...	...	...	...	...

adjuncts and scenario are as follows.

Experimental Adjuncts: Japanese morphological analyzer: MeCab[23] Version 0.98, dictionary of morphological analyzer: IPADIC Version 2.7.0, (MeCab default), Japanese dependency analyzer: CaboCha[25] Version 0.53, dependency analysis models: CaboCha default, thesaurus: EDR thesaurus [89].

Datasets: 240,000 manually labeled Web pages (220,000 training data; 110,000 malicious and harmless Web pages each, 20,000 evaluation data; 10,000 malicious and harmless Web pages each).

Criteria for Evaluation: We evaluate the recall rate, the precision rate, and the F value of each proposed algorithm and the conventional algorithms. In this chapter, we define the recall, the precision, and F of the detection of malicious Web pages as follows based on the total number of malicious Web pages *All* (10,000 in this experiment), the number of Web pages detected as malicious *Judge*, and the number of detected Web pages that are actually malicious *Correct*.

$$Recall = Correct/All \tag{4.3}$$

$$Precision = Correct/Judge \tag{4.4}$$

$$F = 2/(1/Recall + 1/Precision) \tag{4.5}$$

Experimental Scenario:

Table 4.5: Thresholds of the Conventional Algorithms and their Performance (%)

Threshold	# of Keywords	Recall	Precision	F
A	2	45.3	91.0	60.5
B	7	54.7	86.3	67.0
C	12	66.1	80.6	72.7
D	21	71.3	78.2	74.6
E	36	77.3	71.6	74.4
F	84	82.1	67.8	74.3
G	161	90.5	60.6	72.6
H	359	95.5	57.6	71.8

1. Evaluate the trade-off of the recall rate and the precision rate of the conventional algorithms using several malicious score thresholds.

2. In each threshold of 1., correct the classification of the conventional algorithms by the proposed algorithms and evaluate the recall rate and the precision rate.

3. Expand the segment pairs with the thesaurus and evaluate the recall rate and the precision. We also evaluate the number of expanded segment pairs, the processing time and the memory consumption.

4.3.2 Experimental Results

In the conventional algorithms, malicious keywords are sorted by their malicious scores. With a high threshold, only a few malicious keywords are used and the recall rate is low and the precision rate is high. Conversely, with a low threshold, the number of keywords used increases and the recall rate increases, but the precision decreases. We test the proposed algorithms with 8 thresholds, producing the recall rates, the precision rates and the F values shown in Table 4.5.

Figures 4.5 and 4.6 show the relations between the recall and precision, and the F value of the conventional algorithms, the BLA, and the BLEA. The improvement in the recall results from correcting the “harmless” classifications by the conventional algorithms to “malicious”. The improvement in precision results from correcting both “malicious” and “harmless” classifications to “harmless” and “malicious” respectively. The improvement of the BLA is up to 7.6%

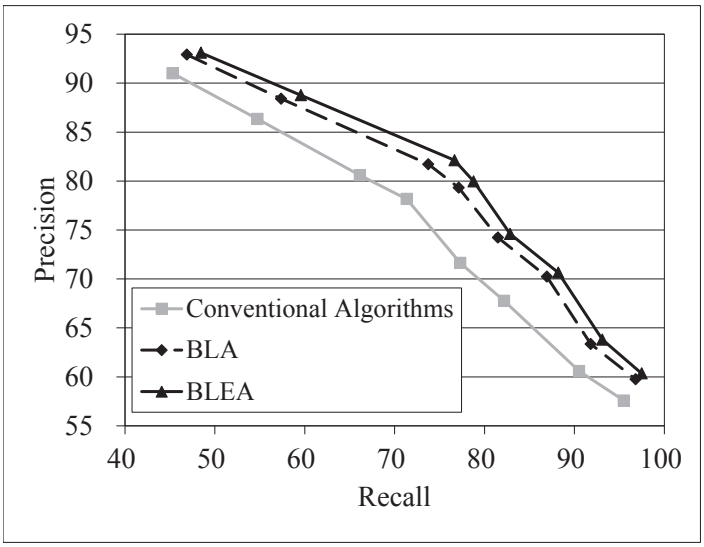


Figure 4.5: Performance Comparison of the Conventional Algorithms, BLA, and BLEA. (F Value, %)

in the recall rate, up to 2.0% in the precision rate, and up to 4.8% in the F value. The improvement of the BLEA is up to 10.6% in the recall rate, up to 3.2% in the precision rate, and up to 6.6% in the F value. This result means that segment pairs expanded by our algorithms detect more malicious/harmless expressions.

In our experiments, we use the dependency analyzer CaboCha with default settings. The accuracy of dependency analysis of CaboCha for Web documents is reported as about 85% [90]. In our algorithms, dependency analysis errors reduce the number of extracted segment pairs in the training phase and reduce the number of corrections in the evaluation phase. In our experiments, however, the effectiveness of our algorithms is confirmed even with the default setting of the dependency analyzer. Improvement in the accuracy of dependency analysis for Web documents is expected to improve the performance of our algorithms.

Finally we evaluate the effect of removing unnecessary concepts and expand noisy segment pairs based on their expanded path. Table 4.6 shows the number of segment pairs before/after expansion, processing time, and memory consump-

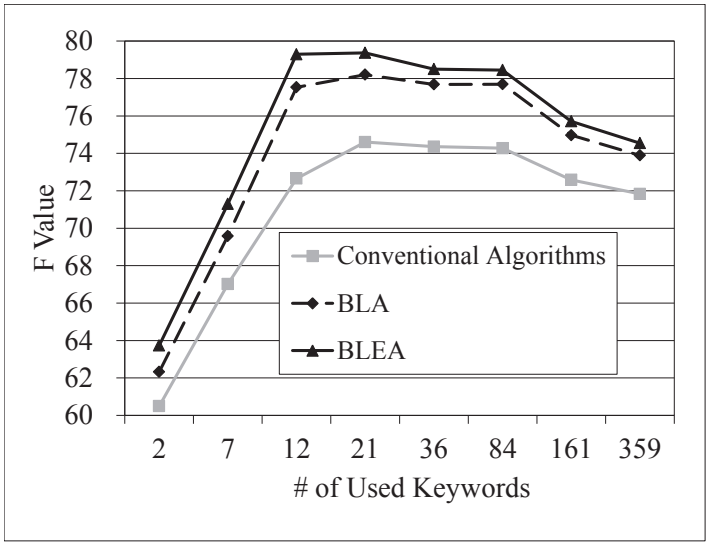


Figure 4.6: Performance Comparison of the Conventional Algorithms, BLA, and BLEA. (Recall vs. Precision, %)

tion. By removing unnecessary concepts, the average number of segment pairs is reduced to less than a quarter. The average processing time and memory consumption is also reduced about a quarter. Table 4.7 compares the performance of the BLEA and PEA. By removing several noisy expansion paths from Malicious to Harmless segment pairs, the recall rate is improved by up to 1.70 and the F value improved by 0.75% compared to the peak performance of the BLEA which is 7.3% higher than the performance of the same number of keywords in the conventional algorithms.

4.4 Conclusion

In this chapter, we propose a method to increase the accuracy of malicious Web page detection by correcting the classification of a conventional text-based method based on the dependency relations of the malicious keywords and their neighboring segments. In addition, we propose a practical algorithm to increase performance by expanding the malicious segment pairs using a thesaurus.

Table 4.6: Number of Segment Pairs vs. Processing Time, and Memory Consumption

	BLA	BLEA			PEA		
Key-words	Seg. Pair (thousand)	Seg. Pair (million)	Time (sec)	Memory (GByte)	Seg. Pairs (million)	Time (sec)	Memory (GByte)
2	304	19.3	208	26.3	5.31	87	7.22
7	224	14	158	19	3.98	68	5.41
12	138	74.2	943	101	17	168	23.1
21	82.1	46	592	62.6	10.9	113	14.8
36	52	30.5	341	41.5	7.44	91	10.1
84	36.4	20.8	201	28.3	5.09	75	6.92
161	18.7	10.2	113	13.9	2.42	57	3.29
359	8.48	3.86	71	5.25	0.953	49	1.3
Avg.	108	27.4	328	37.2	6.63	88.5	9.02

Table 4.7: Performance Comparison of the BLEA vs PEA (%)

	BLEA			PEA					
Key-words	Recall	Precision	F	Recall	Precision	F	Diff Recall	Diff Precision	Diff F
2	48.4	93.1	63.7	50.1	93.4	65.2	1.70	0.26	1.51
7	59.6	88.8	71.3	60.9	88.9	72.3	1.36	0.17	1.02
12	76.7	82.1	79.3	77.9	82.3	80.1	1.22	0.22	0.75
21	78.7	80.0	79.3	79.6	80.1	79.9	0.82	0.17	0.50
36	82.9	74.6	78.5	83.4	74.6	78.8	0.53	0.03	0.26
84	88.2	70.6	78.5	88.5	70.7	78.6	0.33	0.02	0.14
161	93.1	63.8	75.7	93.3	63.8	75.8	0.14	0.03	0.0
359	97.5	60.3	74.5	97.5	60.3	74.5	0	0	0

In our experiments with a large scale Web pages, the proposed base-line method improves the performance of the conventional method by up to 6.6% in F value. Removing noisy segment pairs based on their expanded path is also effective which increases the peak performance of the base-line method by 0.75% and the improvement from the conventional method iss 7.3% in F value.

Chapter 5

Detection of Malicious Web Pages Based on HTML Elements

5.1 Introduction

In this chapter, we propose a high-speed, accurate method for detecting malicious Web pages that existing text-based methods have difficulties with in terms of accurate detection. Our method automatically chooses strings that appear especially in the HTML elements of malicious Web pages. We use these strings in combination as features of SVMs (support vector machines) in order to detect malicious Web pages. Since our algorithm does not rely on the text parts of Web pages, our method can detect those Web pages that existing text-based methods have difficulty in detecting. Combining our methods with existing text-based methods is also an effective approach. In our experiment, we have conducted a large-scale performance evaluation using manually labeled Web pages. The experimental results show that the hybrid method of HTML-based method and existing text-based methods increased the performance of malicious document detection compared to either of the methods. In addition, the proposed method is also applicable to blogs and BBS, where malicious advertisements are included for the purpose of making a profit with affiliates. In the rest of this chapter, we describe the definition of malicious Web pages in Section 5.2, the detail algorithm of the proposed method in Section 5.3, the results of performance

evaluations in Section 5.4, and conclude this chapter in Section 5.5.

5.2 Definition of Malicious Web pages

In this chapter, we describe the detail categorization of Web pages and the definition of malicious/harmless. We received Web pages from a company who engages in collection and manual classification of URLs. Example of the categorization is shown in Table 5.1. Malicious/Harmless is determined by the policies defined by each user. For example, the categories of adult and injustice are malicious for most users. In addition, the categories of sports and games are sometimes malicious for users at school or office because they are not related their work. In this chapter, we define the categories such as advocacy, injustice and adult as malicious.

5.3 Proposed Algorithms

5.3.1 Overview of Proposed Algorithms

Figure 5.1 shows an overview of our algorithms. Our algorithms consist of a training phase in which training datasets manually labeled as malicious or harmless are used, and an evaluation phase in which non-labeled datasets are classified as malicious or harmless. In the training phase, strings that appear especially in HTML elements of malicious Web pages are automatically selected based on statistics. Then, the SVMs are trained with the features that represent the appearance frequency of each sentence in the HTML part of the training Web pages. In the evaluation phase, the evaluation datasets are classified by SVMs based on the features that represent the appearance frequency of each sentence in the HTML part of the Web pages.

Since our algorithms are able to classify Web pages based only on the HTML elements, their performance can be increased by combining them with existing text-based algorithms. In Section 5.3.5, we execute some preliminary experiments to show the differences between our proposed algorithms and existing text-based algorithms [52] in terms of their relative tendency to detect malicious Web pages. In the experiments described in Section 5.4, we evaluated the performance of the hybrid algorithms where our proposed algorithms clearly detect malicious Web pages with a high degree of accuracy, and apply text-based

Table 5.1: Detail categorization of Web pages and the definition of malicious/harmless

Main category	Sub category	Malicious/ Harmless
Security/Proxy Avoidance	Hacking, Cracking, Remote Proxies, Search Engine Caches, Translaters	Harmless
Dating	Dating, Matrimonial Agencies	
Finance	Market Rates, Online Trading, Insurance, Financial Products	
Gambling	Gambling in general, Lottery	
Games	Online games, Games in general	
Shopping	Auctions, Online Shopping, Real Estate, IT Online Shopping	
Communication	Web based Chat, Instant Messengers, Web based Mail, Mail Magazines/ML, Bulletin Boards, IT Bulletin Boards, SNS/Blog	
Downloads	Downloads, Program Downloads, Storage Services, Streaming Media	
Job Search	Employment, Career Advancement, Side Business	
Popular Topics	Special Events, Popular Topics	
Adult Indulgences	Adult Magazine/News, Smoking, Drinking, Alcoholic Products, Fetish, Sexual Expression(text), Costume Play/Enjoyment	
Occult	Occult	
Life Style	Gay/Lesbian Life Style	
Sports	Professional Sports, Sports in general, Leisure	
Travel	Tourism Info./Products, Public Agency Tourism, Public Transit, Accommodations	
Hobbies	Music, Fortune-telling, Entertainer/Celebrity, Dining/Gourmet, Entertainment in general	
Religions	Traditional Religions, Religions in general	
Political Acts/Parties	Political Acts/Parties	
Advertisements	Advertisements/Banners, Sweepstakes/Prizes	
News	News in general	
Injustice	Illegal Acts, Illegal Drugs, Inappropriate Drug Use	Malicious
Advocacy	Terrorism/Extremists, Weapons, Hate/Slander, Suicide/Runaway, Advocacy in general	
Adult	Sex,Nudity, Sexual Services, Adult Search/Links	
Grotesque	Grotesque	
Unsolicited Ads	Unsolicited Mail Links	

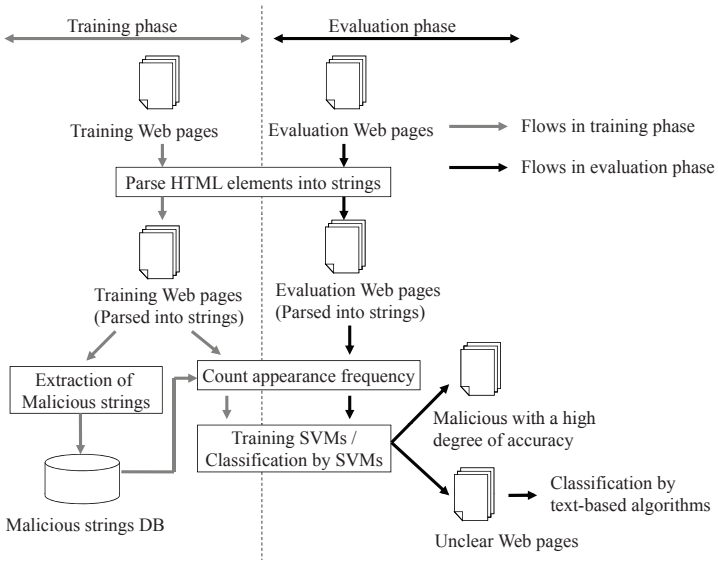


Figure 5.1: Overview of the Processing Flow in the Proposed Algorithms

algorithms to the rest.

5.3.2 Extraction of HTML Elements and Parsing

First, our algorithms extract HTML elements from a Web page and parse them into strings. Here, we define HTML elements as the range of the HTML source after the removal of main text contents, such as the range framed by <>. Algorithms to extract the main text contents are proposed in [91] and [92]. We use the algorithms proposed in [91] to remove the main text contents from HTML source because of their small processing load and because there is no need for preliminary training.

Second, our algorithms parse the extracted HTML elements into strings with the separating characters \t , . / ! " = % & { } [ ] _ and so forth. Table 5.2 is an example of HTML sources, main text contents extracted by the algorithms in [91], HTML elements after the removal of main text contents, and strings parsed with separating characters. For example, from <a href> tags, we can extract strings such as a, href, http, www, springer (server name), com, journal, 12652 (file/directory name), pdf, etc.

Table 5.2: Example of Extraction and Parsing of HTML Elements

HTML source

<h2 class="main left">Journal publication</h2> The Springer Jounal of Ambient Intelligence & Humanized Computing has accepted to take into consideration for publication the very best IEEE-RIVF papers related to its core subjects. See Journal CFP.

Main text part

Journal publication The Springer Jounal of Ambient Intelligence & Humanized Computing has accepted to take into consideration for publication the very best IEEE-RIVF papers related to its core subjects. See Journal CFP .

HTML elements after removed main text part

<h2 class="main left"></h2>   .

Parsed Strings from HTML elements (figures in parentheses are the appearance frequency)

a(4), class(2), h2(2), href(2), left(2), main(2), p(2), 12652, com, Cooperation, engineering, http, indent, jaihc, journal, new, pdf, springer, SpringerJAIHC, target, web, www

5.3.3 Extraction of Malicious Strings of HTML Elements

To extract malicious strings from Web pages, we use the algorithms proposed in [52], which detect the strings that appear especially in HTML elements of malicious Web pages. In the algorithms proposed in [52], $E(s)$ that is the degree of bias in the malicious Web pages of each string s is derived from AIC (Akaike's information criterion) [87]. As shown in Table 5.3, N_{11} and N_{21} are the number of malicious and harmless Web pages where string s appears. N_{21} and N_{22}

Table 5.3: Appearance Frequency of String s to Evaluate E(s)

	Web pages contain s	**Web pages do not contain s**	**Sum**
Malicious Web pages	$N_{11}(s)$	$N_{12}(s)$	N_p
Harmless Web pages	$N_{21}(s)$	$N_{22}(s)$	N_n
Sum	$N(s)$	$N(\neg s)$	N

are the number of malicious and harmless Web pages where string s does not appear. N_{11}, N_{12}, N_{21}, and N_{22} represent the appearance frequency of each string s in all Web pages. The authors of [52] derive the value of $E(s)$ from both independent model AIC_IM and dependent model AIC_DM as follows according to the findings in [88].

$$\begin{aligned} When, N_{11}(s)/N(s) \geq N_{12}(s)/N(\neg s) \\ E(s) = AIC\_IM(s) - AIC\_DM(s) \\ When, N_{11}(s)/N(s) < N_{12}(s)/N(\neg s) \\ E(s) = AIC\_DM(s) - AIC\_IM(s) \end{aligned} \tag{5.1}$$

Here, $AIC\_IM(s)$ and $AIC\_DM(s)$ are defined as follows according to the definition in [87] respectively.

$$\begin{aligned} AIC\_IM(s) &= -2 \times MLL\_IM + 2 \times 2 \\ MLL\_IM &= N_p(s)logN_p(s) + N(s)logN(s) \\ &+ N_n(s)logN_n(s) \\ &+ N(\neg s)logN(\neg s) - 2NlogN \end{aligned} \tag{5.2}$$

$$\begin{aligned} AIC\_DM(s) &= -2 \times MLL\_DM + 2 \times 3 \\ MLL\_DM &= N_{11}(s)logN_{11}(s) + N_{12}(s)logN_{12}(s) \\ &+ N_{21}(s)logN_{21}(s) + N_{22}(s)logN_{22}(s) - NlogN \end{aligned}$$

As an example, Table 5.4 shows S_1 that appears especially in HTML elements of malicious Web pages, S_2 that appears especially in harmless Web pages,

Table 5.4: Example of E(s) Values and Appearance Frequency

String s	$\mathbf{N_{11}(s)}$	$\mathbf{N_{12}(s)}$	$\mathbf{N_{21}(s)}$	$\mathbf{N_{22}(s)}$	**E(s)**
$\mathbf{S_1}$	100	1000	50	9850	122.9
$\mathbf{S_2}$	10	1090	900	9000	-55.6
$\mathbf{S_3}$	100	1000	900	9000	-2.0

and S_3 that uniformly appears in both malicious and harmless Web pages. Because of the feature of S_1, S_2, and S_3, the value of $E(s)$, which is the degree of bias in malicious Web pages, is large in S_1, small in S_2, and almost 0 in S_3 (in this example, the difference of -2.0 is the result of the difference between AIC-dependent models and -independent models).

These algorithms detect the features of malicious Web pages such as the name of the server related to malicious Web pages, or the name of javascript functions that makes browsers perform unusual actions in response to malicious Web pages, such as displaying a pop-up window.

To evaluate the detection performance of each extracted string, we have conducted a preliminary experiment. In this experiment, we evaluate the recall and precision of each extracted string when Web pages that contain the string more than N times are judged as malicious. Figure 5.2 shows the performance of S_1 and S_2, which are strings extracted by the proposed algorithms, the number of links, and the number of image files of Web pages that are said to be effective for detecting malicious Web pages in [55]. We used 10 thousand malicious and harmless Web pages respectively.

In this chapter, we define the recall and the precision of the detection of malicious Web pages as follows based on the total number of malicious Web pages *All* (10 thousand in this experiment), the number of Web pages detected as malicious *Judge*, and the number of detected Web pages that are actually malicious *Correct*.

$$Recall = Correct/All \tag{5.3}$$

$$Precision = Correct/Judge \tag{5.4}$$

Figure 5.2 shows that strings S_1 and S_2 have high precision compared to the same recall of the number of images and links of the Web pages. Although the

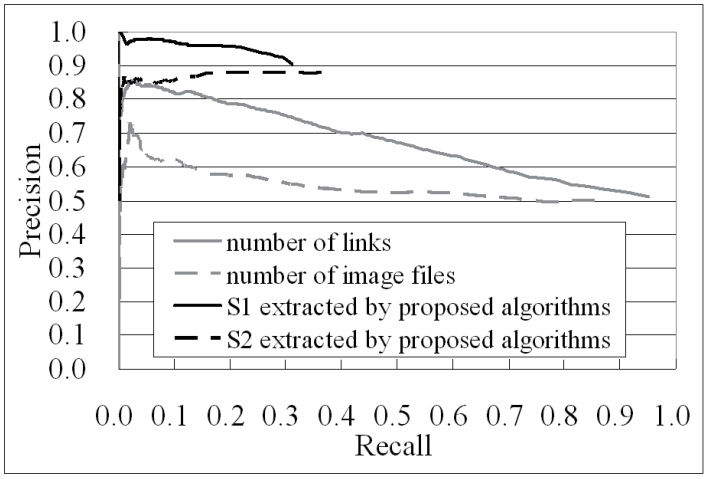


Figure 5.2: Comparison of the Peformance of S1 and S2, which are Extracted by the Proposed Algorithms, the Number of Links, and the Number of Image Files whereby Web Pages that Contain Each String More than N times are Judged as Malicious

maximum recall of S_1 and S_2 (when Web pages containing more than 1 string are judged as malicious) is low, compared to the number of images and links, it can be increased by combining several strings. In our algorithms, a filtering system having fairly high accuracy is realized by combining these high-precision strings. On the other hand, since the manually observed features introduced in [55] such as the number of images and links have lower precision, combining these features lessens the total accuracy of the filtering systems and complicates the optimization of the classifier parameters.

5.3.4 Training SVMs and Classification by SVMs

We use the strings extracted in Section 5.3.3 in combination as the features of SVMs (support vector machines) [93], and detect malicious Web pages. Table 5.5 shows examples of the input of SVMs. S_1, S_2, S_3, ..., S_m are the names of strings extracted. N_1, N_2, N_3, ..., N_m are the numbers of times each string appears in each Web page. In the training phase, adding the label of malicious or harmless to each Web page enables SVMs to train the datasets.

Table 5.5: Example of SVM Features

	$\mathbf{S_1}$	$\mathbf{S_2}$	$\mathbf{S_3}$	**...**	$\mathbf{S_m}$	**Label**
Page 1	N_{11}	N_{12}	N_{13}	...	N_{1m}	1
Page 2	N_{21}	N_{22}	N_{23}	...	N_{2m}	0
	...	...	...	...	...	...
Page X	N_{x1}	N_{x2}	N_{x3}	...	N_{xm}	0

Here, we describe the appropriateness of using SVMs to detect malicious Web pages. Considering the use cases of filtering systems, classifiers that show high generalization functions to evaluate datasets (unknown datasets) rather than training datasets (known datasets) are suitable. SVMs are known to have high generalization functions, so we used SVMs with our algorithms. As a preliminary experiment, we compare the performance of our algorithms by using SVMs and C4.5 [94], which is a well-known classifier based on decision trees, as the classifier of our algorithms. First, we train the SVMs and C4.5 with 20 thousand manually labeled Web pages (10 thousand malicious Web pages and 10 thousand harmless Web pages). Second, the F value of each classifier is evaluated through evaluation with another 20 thousand Web pages (10 thousand malicious Web pages and 10 thousand harmless Web pages). The F value of SVMs was 69.1%, and the F value of C4.5 was 59.1%, which shows that SVMs are more promising for our algorithms. Although C4.5 is a well-known classifier, neural networks [95] and Bayesian filtering [96] are also candidate classifiers, and evaluation of the algorithms with these classifiers is part of our future work.

The probability of classification accuracy calculated in SVMs can be regarded as the threshold for detecting malicious Web pages. By setting a high threshold, the recall rate is low and the precision rate is high. In contrast, by setting a low threshold, the recall rate is high and the precision rate is low. In our experiments in Section 5.4, we evaluated the trade-off between recall and precision by changing the threshold.

5.3.5 Characteristics of the Proposed and Text-based Algorithms

Since our algorithms classify Web pages depending only on HTML elements, they can detect other Web pages by being combined with existing text-based

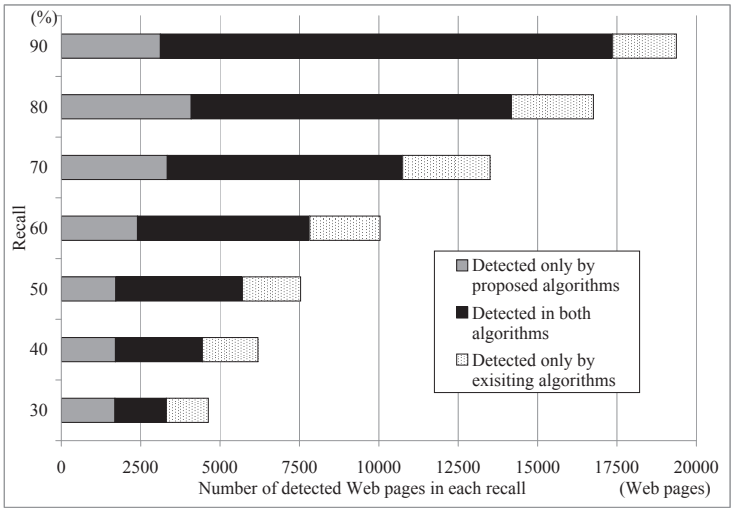


Figure 5.3: Number of Web Pages Detected as Malicious by the Proposed and Exsiting Algorithms

algorithms to increase detection accuracy. We conduct preliminary experiments to ascertain differences between our proposed algorithms and existing text-based algorithms [52] in terms of their tendency to detect malicious Web pages.

In the same way as the experiment in Section 5.3.4, we used 20 thousand manually labeled Web pages (10 thousand malicious Web pages and 10 thousand harmless Web pages) as training datasets and another 20 thousand Web pages (10 thousand malicious Web pages and 10 thousand harmless Web pages) as evaluation datasets. Figures 5.3 and 5.4 show the number and the ratio of Web pages detected by (1) our algorithms alone, (2) existing text-based algorithms alone, and (3) a combination of our algorithms and existing text-based algorithms, respectively, when the recall rate of each algorithm is 10, 20, 30, $\cdots$, 90 (%). They show that, although the Web pages that are detected by both algorithms increase as the recall rate increases, even with a recall rate of 90%, there are some Web pages that cannot be detected by either of the two algorithms alone. These results show that combining our algorithms and existing text-based algorithms enables us to detect more malicious Web pages. In our experiment in Section 5.4, we evaluate the performance of our algorithms alone and a combination of algorithms where first, our proposed algorithms clearly

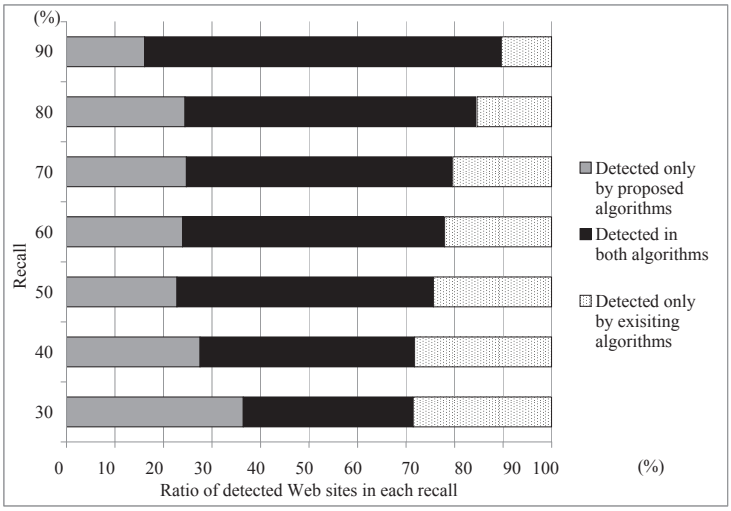


Figure 5.4: Ratio of Web Pages Detected as Malicious by the Proposed and Exsiting Algorithms

detect malicious Web pages with a high degree of accuracy and where apply text-based algorithms to the rest.

5.4 Performance Evaluation

5.4.1 Experimental Scenario and Environments

We implement our algorithms and compare their performance with existing text-based algorithms [52]. The experimental scenario and environments are as follows.

Experimental environments: Terminal of single-core 2.53-GHz 64-GB RAM Linux OS, Lib SVM [97] as a classifier used in the proposed algorithms and MeCab [23] as a morphological analyzer to parse the training datasets in existing text-based algorithms The programs are written in C language.

Datasets: 40 thousand manually labeled Japanese Web pages, consisting of 20 thousand training datasets and 20 thousand evaluation datasets (10 thousand malicious Web pages and 10 thousand harmless Web pages respectively)

Evaluation standard: We evaluate the recall and the precision of the proposed algorithms and existing text-based algorithms. We also evaluate the

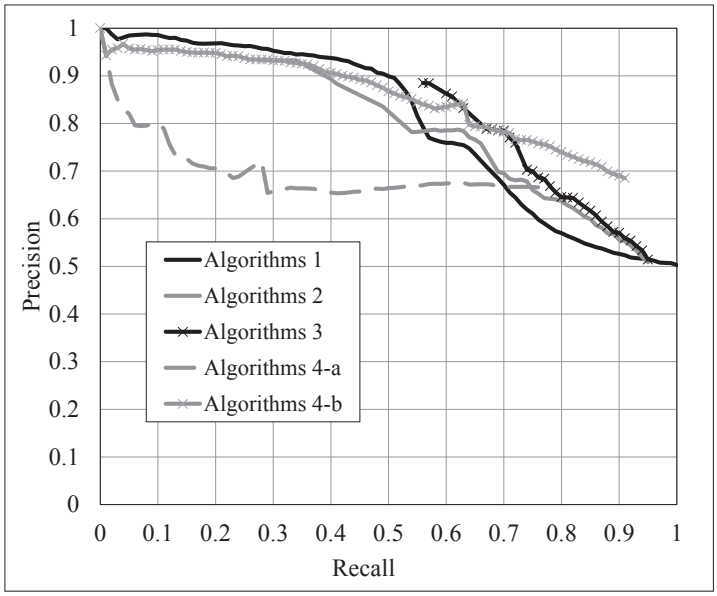


Figure 5.5: Comparison of the Peformance of Each Algorithm

average processing time for a Web page with each algorithm.

Experimental Scenario: Comparison of the performance of the following 5 algorithms: (1) our proposed algorithms alone, (2) existing text-based algorithms alone, (3) combination of algorithms where first, our proposed algorithms clearly detect malicious Web pages with a high degree of accuracy and where we apply the text-based algorithms to the rest (we call these algorithms the **proposed hybrid algorithms**), and (4-a) and (4-b) detecting malicious Web pages in the same way as our algorithms shown in Section 5.3 by using malicious keywords instead of HTML elements as features of SVMs, which are extracted by existing text-based algorithms

5.4.2 Experimental Results

Figure 5.5 shows the recall and the precision of each algorithm. When the performance of our proposed algorithms (1) and existing text-based algorithms (2) are compared, our algorithms have more than 90% precision in the less-than-50% recall range. Although in the high recall range, existing text-based

Table 5.6: Comparison of the Processing Time of Each Algorithm

	Average processing time for classifying a Web page (msec)
Algorithms 1 (proposed alone)	3.85
Algorithms 2 (existing)	3.57
Algorithms 3 (proposed hybrid)	3.65
Algorithms 4-a (existing+SVM, with 26 words)	3.50
Algorithms 4-b (existing+SVM, with 10 thousand words)	12.12
Morphological analysis only (for reference)	6.82

algorithms have high precision, by adding effective strings extracted from HTML elements, our proposed algorithms are expected to increase their recall and precision. In our proposed hybrid algorithms (3), we classify Web pages based on the algorithms (1) until their recall rate reaches 50% and we classify the rest using existing text-based algorithms (2). Our proposed hybrid algorithms show high precision over the entire recall range, by improving the deterioration in precision in the high recall range. Especially at a recall rate of 70%, the hybrid algorithms improve precision by 9.3% from 68.8% to 78.1% compared to existing text-based algorithms. On an F value basis, the hybrid algorithms have a 74.0% F value compared to the 70.6% of existing text-based algorithms. The recall rate and the precision rate are in a trade-off relation. In the proposed HTML-based and hybrid algorithms, this trade-off can be tuned with the parameter of the estimation probability of SVMs, where either of the recall or the precision is prioritized.

Although the algorithm (4-a) uses the same number of keywords as the proposed algorithms (1), the overall performance is lower than that of the proposed algorithms. This means that in our proposed algorithms, each extracted feature performs well in detecting malicious Web pages. The algorithm (4-b), which uses 10 thousand keywords, shows high precision in the high recall range compared to the proposed algorithms (1) and the hybrid algorithms (3).

Finally, we show a comparison of the processing time for a Web page with each algorithm in Table 5.6. The average processing time of our proposed hybrid algorithms (3) and existing text-based algorithms are almost the same, 3.65 msec and 3.57 msec respectively. These processing times are about half that

of the morphological analysis alone. So our proposed algorithms are very fast compared to advanced text-based algorithms such as [53]. The processing time of the algorithm (4-b) is long due to its many features although it has high precision in the high recall range compared to the proposed hybrid algorithms (3). Our proposed algorithms realize high accuracy based on only a few strings, which avoids the problems described above.

5.5 Conclusion

In this chapter, we propose a high-speed, accurate method for detecting malicious Web pages. Our method automatically chooses strings that appear especially in the HTML elements of malicious Web pages based on AIC. We use these strings in combination as features of SVMs in order to detect malicious Web pages. We show that our proposed algorithms can detect those Web pages that existing text-based algorithms have difficulty in detecting by confirming the differences between our proposed algorithms and existing text-based algorithms in terms of their tendency to detect malicious Web pages by experiments. In our experimental environments with 40 thousand pages manually labeled, we confirm that the performance of the hybrid method of combining our algorithms with existing text-based algorithms is a recall rate of 70.0% with a precision rate of 78.1%, which is 9.3% better precision than the same recall of existing text-based algorithms when used on their own.

Chapter 6

User Profiling Based on Text and Community Mining

6.1 Introduction

In this chapter, we propose a hybrid of a text-based method and a community-based method for demographic estimation of users. The text-based method estimates the users who have plentiful text features on their posts. For the rest of users, the community-based method analyzes their related users such as followers and followees who have plentiful text features. The hybrid method covers almost all users by making the most of the information of users and their related users which includes both text information and community information. In the text-based method, characteristic terms used by each demographic segment are automatically detected based on linguistic and statistical analysis by tracking the content of users' past posts. In the community-based method, demographic information is estimated from the relation of the target user and other users. We conduct a large-scale performance evaluation using Twitter platform. The experimental results show that the estimation accuracy of user profiles is high enough for practical use. In this chapter, we evaluate the performance of the proposed method on Twitter platform; however, the essential algorithm of the hybrid of text-based and community-based method is applicable to other media such as blogs and Web pages, where community information is included as hyper

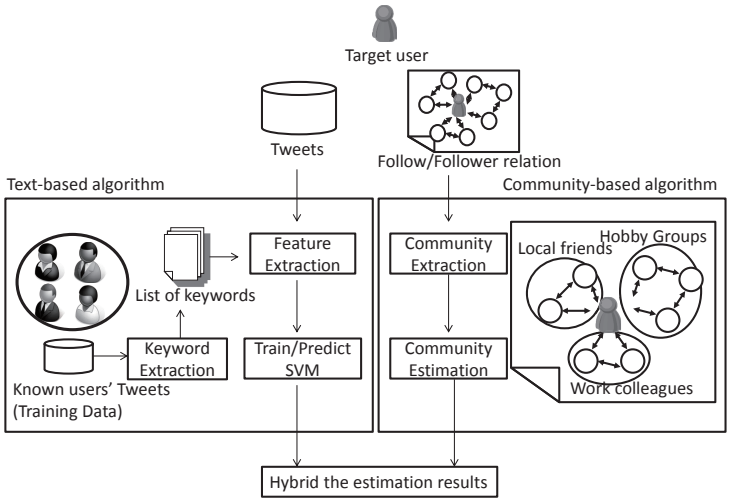


Figure 6.1: Overview of the Proposed Hybrid Method for Demographic Estimation

links between them. In the rest of this chapter, we describe the detail algorithm of the proposed method in Section 6.2, the results of performance evaluations in Section 6.3, and conclude this chapter in Section 6.4.

6.2 Proposed Method

The proposed hybrid method consists of the text-based method and the community-based method (Figure 6.1). The text-based method analyzes the past tweets of target users. The community-based method analyzes the follow/follower relations of target users. In the following section, we describe the detail of the text-based method in Section 6.2.1, the detail of the community-based method in Section 6.2.2, and the hybrid method in Section 6.2.3.

6.2.1 Text-based Method

The proposed text-based method described in Figure 6.1 consists of the following three steps. Step1: extraction of the list of characteristic terms from the training data. Step2: Feature extraction from Twitter users. Step3: Training and prediction of SVMs (Support Vector Machines) This process can be

regarded as a kind of text classification, where the documents written by users are classified into any demographic categories.

In the Step1, in order to realize a rapid demographic estimation for the practical use, only the limited characteristic terms are listed based on statistical criteria. Following are the definition of AIC (Akaike's Information criteria) [87] which we used. Terms are identified that appear especially often in the comments of specific demographic groups. We used the comments of these demographically known users as training datasets.

Definitions are as follows. A document is a set of comments from each user. D_p is a set of documents submitted by each user within a specific demographic. D_n is a set of documents submitted from other users. $E(t)$ is the degree of bias in the distribution of term t between D_p and D_n, respectively. As shown in Table 6.1, N_{11} and N_{21} are the numbers of documents in D_p and D_n, respectively where term t appears. N_{21} and N_{22} are the numbers of documents in D_p and D_n, where term t does not appear. N_{11}, N_{12}, N_{21}, and N_{22} represent the frequency of the appearance of each term t in the training datasets, respectively.

We derive the value of $E(t)$ from both independent model AIC_IM and dependent model AIC_DM as follows according to the findings for handling negative values [88] (Expressions 6.1 and 6.2).

$$\begin{aligned} &When, N_{11}(t)/N(t) > N_{12}(t)/N(\neg t) \\ &E(t) = AIC\_IM(t) - AIC\_DM(t) \\ &When, N_{11}(t)/N(t) \leqq N_{12}(t)/N(\neg t) \\ &E(t) = AIC\_DM(t) - AIC\_IM(t) \end{aligned} \tag{6.1}$$

$$\begin{aligned} AIC\_IM(t) &= -2 \times MLL\_IM + 2 \times 2 \\ MLL\_IM &= N_p(t)logN_p(t) + N(t)logN(t) \\ &+ N_n(t)logN_n(t) + N(\neg t)logN(\neg t) - 2NlogN \\ AIC\_DM(t) &= -2 \times MLL\_DM + 2 \times 3 \\ MLL\_DM &= N_{11}(t)logN_{11}(t) + N_{12}(t)logN_{12}(t) \\ &+ N_{21}(t)logN_{21}(t) + N_{22}(t)logN_{22}(t) - NlogN \end{aligned} \tag{6.2}$$

Table 6.1: Criteria for Calculation of E(t)

	t appears	t does not appear	Total
# of D_p	$N_{11}(t)$	$N_{12}(t)$	N_p
# of D_n	$N_{21}(t)$	$N_{22}(t)$	N_n
Total	$N(t)$	$N(\neg t)$	N

Table 6.2: Examples of Term Frequencies and E(t) Values

Term	$N_{11}(t)$	$N_{12}(t)$	$N_{21}(t)$	$N_{22}(t)$	$E(t)$
Examination	261	61	639	2640	2640
Beer	70	819	830	1882	-216.6
Bucket	17	51	883	2650	-2.0

Here, $AIC\_IM(t)$ and $AIC\_DM(t)$ are defined as follows according to the definition in [87] respectively.

As an example, Table 6.2 shows the frequencies of appearance of the terms "examination," which is characteristic of teens, "beer," which is characteristic of non-teens, and "bucket," which uniformly appears in both teens and non-teens. "Examination" has a positive $E(t)$ value due to its biased distribution in teens documents. "Beer" has a negative $E(t)$ value due to its biased distribution in non-teens documents. "Bucket" has a low $E(t)$ value due to its uniform distribution between *teens* and *non-teens*.

In Step2 and Step3, we extract the feature of user tweets and classify them by the classifier. SVMs are popular classifiers in the field of text categorization problems; they learn the features from training datasets and classify unknown datasets. We trained SVMs for the each demographic segment, such as teens, twenties, thirties, and so on. SVMs are given to the appearance of terms extracted as the features. Table 6.3 shows an example of the input matrix of SVMs, which consists of terms T_1, T_2, T_3, ..., T_x and M_1, M_2, M_3, ..., M_x representing whether each term appears or not (0 or 1). In the training phase, the field "label" was used to represent whether each user was associated with a specific demographic or not.

The estimation accuracy is high when SVMs use all the terms that appear in the training datasets as features; however, a large number of features increases

Table 6.3: Example of Input of SVMs

	T_1	T_2	T_3	...	T_x	Label
User 1	M_{11}	M_{12}	M_{13}	...	M_{1x}	1
User 2	M_{21}	M_{22}	M_{23}	...	M_{2x}	0
...	...	...	...	...	...	...
User Z	M_{z1}	M_{z2}	M_{z3}	...	M_{zx}	0

the time required for training SVMs. By using statistical criteria, we focus only on effective terms for demographic estimation, which can reduce the degree of terms (the value of x in Table 6.3), and enables rapid demographic estimation. We regard the estimated probability in SVM as the degree of membership in each demographic segment, which is used for the purpose of combining the text-based method and the community-based method in Section 6.2.3 We normalize the degree of membership so that the sum for each user segment is 1.

6.2.2 Community-based Method

In the community-based method, demographic information is estimated from the follower/followee relations of the target user. Follower and followee relationships can be regarded as directed links between two users. Some existing approaches have classified Web documents, including hyperlinks, into a single category. However, a twitter user has multiple demographic categories at the same time, such as age, gender, and area of residence. In the proposed method, characteristic biases in the demographic segments of users are detected from the community groups constructed by clustering their followers and followees. For example, a user may belong to several community groups, such as local friends, work colleagues, and hobby groups, whose members have something in common such as age, gender, and regional area. Figure 6.2 shows the detail of the proposed community-based method. The followers and followees of the target user are clustered into groups, then, the relation of each community group is estimated by analyzing the demographic distribution of the group members by the text-based method since the followers and followees have plentiful text features in their tweets.

First, we show the detail of the user clustering method. Since our goal is to construct clusters from the community networks of follower/followee relation-

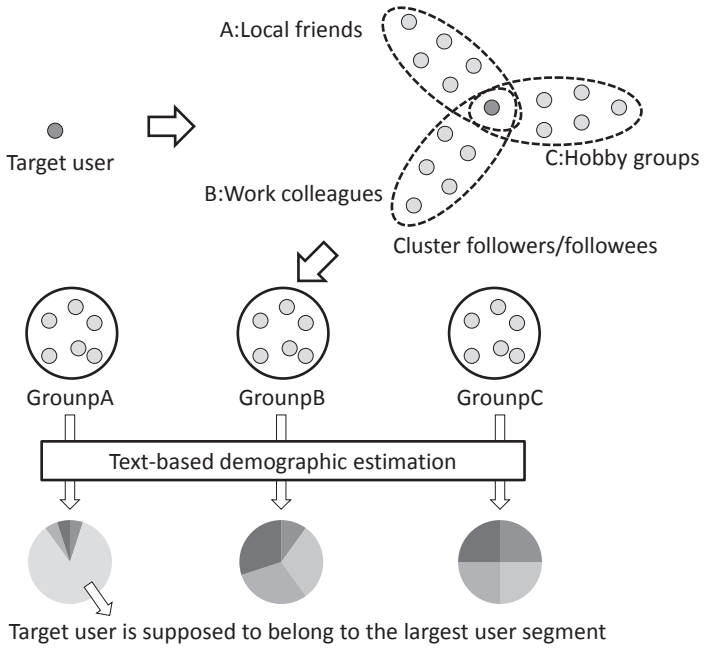


Figure 6.2: Overview of the Community-based Demographic Estimation Method

ships, clustering methods for community networks are expected to work well. For analyzing large-scale networks, several effective methods have been proposed [98, 99, 100]. We used a tool called NodeXL [101], which can automatically import the social network of a Twitter user and divide the network into clusters using the Clauset-Newman-Moore (CNM) method [98]. NodeXL creates approximately up to five user clusters with a central focus on the target user. We create user clusters based on the following heuristic method (Expression 6.3).

$$\begin{aligned} &Community\ c\ is\ initialized\ with\ the\ set\ of\ the\ target\ user \\ &While\ the\ size\ of\ c < Threshold\ T \\ &\quad Add\ I\ hop\ neighbors\ to\ the\ community \\ &Create\ community\ clusters\ for\ c\ based\ on\ the\ CNM method \end{aligned} \tag{6.3}$$

Threshold T decides the size of the community network, which is in the

Table 6.4: Example of the Degree of Membership of the Community-based Method for Teens, Twenties, Thirties, and Over Forties

	10s	20s	30s	Over 40s	Sum
Distribution of Group A	5%	85%	5%	5%	100%
Distribution of Group B	5%	35%	40%	30%	100%
Distribution of Group C	25%	25%	25%	25%	100%
Maximum ratio	25%	85%	40%	30%	-
Estimated degree of membership	0.14	0.47	0.22	0.17	1

trade-off relationship of the accuracy and the complexity of community-based estimation. We determined the value of T to be 300 from preliminary experiment. Follow/Follower relationship often includes news groups and celebrities, which can distort the clusters due to their large scale networks. We eliminated those who had a larger number of followers than 1,000.

Second, we describe the community estimation method. In the community-based method, we focus on the difference in the distributions of demographic segments of the clustered groups, which reflects the characteristics of members of each cluster. For example, group members of local friends probably have similar ages and areas. Work colleagues and hobby groups have the same occupations and hobbies. We apply the text-based demographic estimation method to the users of each cluster constructed by the CNM method. For each demographic segment, the maximum ratio among each community is selected as the degree of membership. The proposed community-based method is as follows (Expression 6.4). Here, G_i is the community extracted by the CNM method. $Ratio(s, G_i)$ is the ratio of demographic segment s in community G_i. We defined *Deg_mem(s)* as the degree of membership of demographic segments.

For example, for age, a user may have the ratio of the demographic segments for each community shown in Table 6.4. In this case, the maximum ratio is 25% of Group C for teens, 85% of Group A for twenties, 40% of Group B for thirties, and 30% of Group B for over-forties. The degree of membership is defined by normalizing the maximum ratio of each demographic segment.

$$\begin{aligned} Max\_ratio(s) &= max_{i=1..n} Ratio(s, G_i) \\ Deg\_mem(s) &= Max\_ratio(s) / \Sigma_{j=1..m} Max\_ratio(s_j) \end{aligned} \tag{6.4}$$

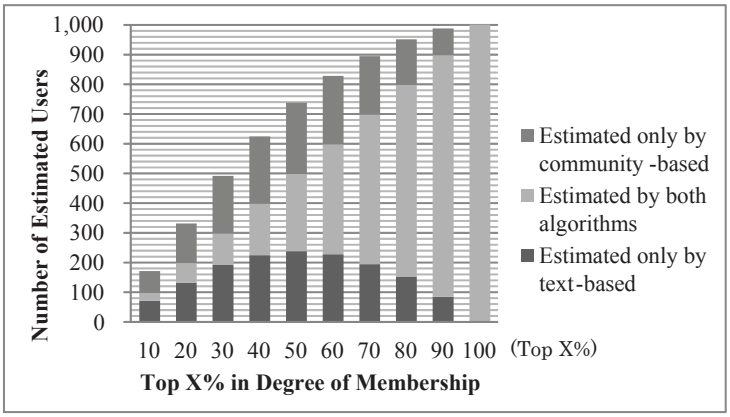


Figure 6.3: Differences in the Estimations from the Text-based Method and the Community-based Method

6.2.3 Hybrid of Text-based and Community-based Methods

Since the community-based method focuses on followers/followees independent of the target users' tweets, the hybrid of the text-based method and community-based method is expected to be effective for the user who has few text features in their tweets. We found that the degree of membership in text-based method is high for the user with plentiful text features and low for the user with few text features. The community-based method can improve the estimation accuracy for these users with low degree of membership. The hybrid method that incorporates the advantages of both methods is expected to work effectively.

We collected 1,000 Twitter users randomly by using the streaming API. Figure 6.3 shows the number of users who are estimated by (1) the text-based method alone, (2) the community-based method alone, and (3) both methods, when the users are estimated in descending order of degree of membership described in Sections 6.2.2 and 6.2.3 For example, when the top 10% users (100 users of the highest degree) are estimated, the number of users estimated by both methods was 26, when either the text-based or community-based method alone yielded 74. Figure 6.3 shows that the users who were estimated with high degree differed between the text-based method and the community-based

method, which means that combining these methods improves the estimation accuracy.

We define the hybrid of the text-based method and the community-based method as follows (Expression 6.5). R is a threshold for degree of membership, above which ratio the degree of membership of the community-based method *Deg_mem_com* is added to the text-based method.

$$\begin{aligned} If\ (Deg\_mem\_com &> Threshold\ R) \\ Deg\_mem\_hybrid &= Deg\_mem\_text + Deg\_mem\_com \\ Else & \\ Deg\_mem\_hybrid &= Deg\_mem\_text \end{aligned} \tag{6.5}$$

6.3 Performance Evaluation

We evaluate the performance of the proposed methods in the following datasets, evaluation metrics, and experimental environments.

6.3.1 Collection of Demographic Information

We describe a way of collecting demographic information on Twitter users. On Twitter, although the number is small, some users state their age and gender in their profile and their location information in the location field. We automatically collect these demographically known users. First, we collect tweets at random using the streaming API of Twitter, and filter only Japanese tweets by eliminating tweets that contain no Japanese characters.

There are certain writing styles relevant as demographic information. As for age, some directly state their age, such as “37-year-old,” some state their year of birth, such as “born in 1974,” and others simply state their age segment, such as “thirties.” As for gender, “man,” “gentleman,” “boy,” etc., are keywords for male users, while “woman,” “lady,” “girl,” etc., are keywords for female users. We detect this kind of information using dictionary-based keyword matching in consideration of spelling inconsistencies.

Automatically collected demographic information is not always correct in such cases as “mom of a 15-year-old boy.” Therefore, the demographic information entry is used only if two people manually clean it, and both of them

Table 6.5: Characteristics of Collected User Ages

	Number of users (Ratio of users %)			
Digit	10s	20s	30s	over 40s
0	4 (0.4)	306 (8.4)	306 (10.2)	80 (7.8)
1	4 (0.4)	228 (6.3)	131 (8.3)	85 (8.3)
2	4 (0.4)	354 (9.7)	118 (7.5)	60 (5.9)
3	18 (1.6)	320 (8.8)	114 (7.2)	62 (6.1)
4	18 (1.6)	330 (9.1)	88 (5.6)	27 (2.6)
5	57 (5.1)	332 (9.1)	104 (6.6)	49 (4.8)
6	123 (11.1)	306 (8.4)	77 (4.9)	31 (3.0)
7	167 (15.1)	337 (9.3)	67 (4.3)	45 (4.4)
8	326 (29.4)	209 (5.7)	54 (3.4)	24 (2.4)
9	300 (27.1)	187 (5.1)	46 (2.9)	27 (2.6)
age segment	88 (7.9)	726 (20.0)	615 (39.1)	530 (52.0)
total	1109	3635	1574	1020

judge the information to be correct. In order to avoid bias in the collected demographic information, we do not use keywords that would produce indirect demographic information such as “high school” for teens or “husband/wife” for males/females.

We collect users who state their age, gender, residence area, occupation, hobby and marital status from 100,000 user profiles. Table 6.5 shows the volume of users we collect for each age segment. For example, the total number of teens is 1,109 and the number of 18-year-old users is 326, which accounts for 29.4% of teens. Although the numbers of younger people (less than 15 years old) and elderly people (over fifties, whose amount 122 is included in age segment of over 40s) are small, the rest are collected uniformly. Six hundred users for each demographic segment according to the distribution are used in the experiment. Table 6.6 shows a summary of the volume of user profiles that we used for the experiment, where genders and areas are collected in the same manner as age. Users are separated into halves in each demographic, one a for training dataset and the other for the estimation target. By separating users, the monitored estimation accuracy is expected to be the same in practical use.

6.3.2 Experimental Environment and Evaluation Metrics

Experimental environment: Terminal with a single-core 3.2-GHz processor, 8 GB of RAM, CentOS, Lib SVM [97] as a classifier, MeCab [23] as a morphological analyzer to parse terms from Japanese tweets, and implementation in

Table 6.6: Number of Users in the Experiment

Demographic Types	Demographic Segments	# of Users in each Segment
Gender	Male or Female	600
Age Group	10s, 20s, 30s, or 40s and older	600
Area	Hokkaido/Tohoku, Kanto, Hokushinetsu, Tokai, Kinki, Chugoku/Shikoku, or Kyushu/Okinawa	600
Occupation	Employee, Part-time, Self Employed, Civil Servant, Homemaker, Student, or Without occupation	600
Hobby	Reading, Gourmet, Vehicle, IT & Electronics, Games, Pets & Plants, Sports, Travel, Fashion, Music, TV & Movie, or Arts	600
Marital status	Married or single	600

the C programming language.

Evaluation metric: We evaluate the text-based method, community-based method, and hybrid method with the following two evaluation metrics. (1) Recall, Precision and F-value are general metrics for the classification methods, which are defined as follows (Expression 6.6). Recall and Precision are in a trade-off relationship with each other, where the degree of membership described in Sections 6.2.1 and 6.2.2 is the parameter for tuning the trade-off. High-precision estimation obtained by tuning the parameter is effective for the target ad, such as sending specific information to a specific user segment. (2) Accuracy of distribution ratio in each segment is a metric to clarify that the estimation errors are not biased in particular demographic segments. For marketing use, the distribution ratio of user segment is important such as follower analysis of a company. The estimated distribution ratio of users may not be correct despite the high recall and precision when the methods have a tendency to place users into a specific demographic segment. The error E for the distribution ratio of each demographic segment and the average error E_{avg} are defined as follows (Expression 6.6). Here, T denotes the demographic, such as age, gender, or area. T_1,T_2,...T_n are segments of T such as teens, twenties, or thirties, while n is the number of segments for each demographic type. $U_t(T_i)$ and $U_e(T_i)$ are the number of target users and the number of estimated users in each demographic segment T_i.

$$\begin{aligned} Recall &= number\ of\ correctly\ estimated\ users/all\ estimation\ target\ users \\ Precision &= number\ of\ correctly\ estimated\ users/ \\ &\quad number\ of\ users\ judged\ as\ a\ specific\ demographic\ segment \\ F &= 2/(1/Recall + 1/Precision) \end{aligned} \tag{6.6}$$

$$\begin{aligned} E(T_i) &= U_e(T_i)/U_t(T_i) - 1 \\ E_{avg}(T) &= \sum_{1}^{n}(|U_t(T_i) - U_e(T_i)|)/\sum_{1}^{n} U_t(T_i) \end{aligned} \tag{6.7}$$

6.3.3 Experimental Results

Tables 6.7 and 6.8 show examples of terms extracted by the text-based method. Table 6.9 shows the trade-off of recall and precision. Table 6.10 shows the general performance of each demographic segment based on F-value. Table 6.9 shows the value of E.

Table 6.7 shows the terms extracted in the demographic segments of age, gender, and area of residence. Terms related to school are extracted from teens' documents. Terms related to college and the job-hunting process are included in twenties' documents. Terms for thirties are related to work and home life. Terms for forties are related to home life, politics, and personal health. Terms for their partners are characteristic of both males and females. In addition, terms related to work, politics, and home electronics are also characteristic for males. Terms related to females include housework and foods. Terms related to area of residence include names of places, local transport facilities, names of local TV stations, and dialects. Table 6.8 shows the terms extracted in the demographic segments of occupation, hobby, and marital status. Terms related to occupations are clearly characteristic of the activities of each job. As for hobbies, IT & Electronics contains the names of devices and operating systems. Fashion contains terms such as make-up and clothes. Music contains terms related to music. Terms for married users are related to family affairs. In contrast, terms for unmarried users are related to love affairs and places to go on a date.

Table 6.7: Example of Extracted Terms (age, gender, and area)

10s	20s	30s	40s over	Male	Female	Kanto	Kinki
Mathematics	University	Work	Son	Government	Husband	Shinjuku	Osaka
School	Part-time	Company	Holiday	Android	Mother	Ikebukuro	Umeda
Examination	Seminar	Business	Golf	Wife	Bath	Shibuya	Kyoto
Test	Job-hunting	Boss	Diplomacy	Company	Laundry	Yamanote-line	Yakedo (dialect)
Physical Education	Lecture	Beer	Backache	Google	Lunch	Akihabara	Hanshin

Table 6.8: Example of Extracted Terms (occupation, hobby, marital status)

Employee	Homemaker	Student	IT & Electronics	Fashion	Music	Married	Unmarried
Office	Husband	Class	IPad	Manicure	Song	Son	Boyfriend
Work	Housework	School	IPhone	Denim	Live	Husband	Lover
Attendance	Laundry	College	Mac	Fashion	Album	Daughter	Love
Boss	Mother-in-law	Examination	OS	Makeup	Band	Home	Bored
Commuting	Mam	Anime	Android	Clothes	Music	Mam	Karaoke

Figure 6.4 shows the trade-off of recall and precision for teens when the users are estimated in descending order of degree of membership such as the top 10%, 20%, 30% $\cdots$, 100%. The precisions of the text-based method and community-based method are high in the low-recall range. As described in Section 6.2.3, the hybrid method integrates these two methods at the threshold of degree of membership. In the case of Figure 6.4, the precision of the top 30% of text-based method is higher than the top 10% of community-based method. Since the community-based method works effectively in the range where the degree of membership of the text-based method is low, the threshold is set at 30% in this case. In the same manner, the threshold is set at the point where the precision of the text-based method is higher than the precision of the top 10% of the community-based method. Looking at the column of the top 100%, the hybrid method surpasses both the text-based method and community-based method. In the hybrid method, the precision is higher than 90%, which we assume enough credible, in the range of less than 50% of the recall value. For example, people in the advertising department can send a half of teen Twitter users a commercial message about products targeting teens.

Table 6.9 shows the F-value of each demographic segment estimated by the

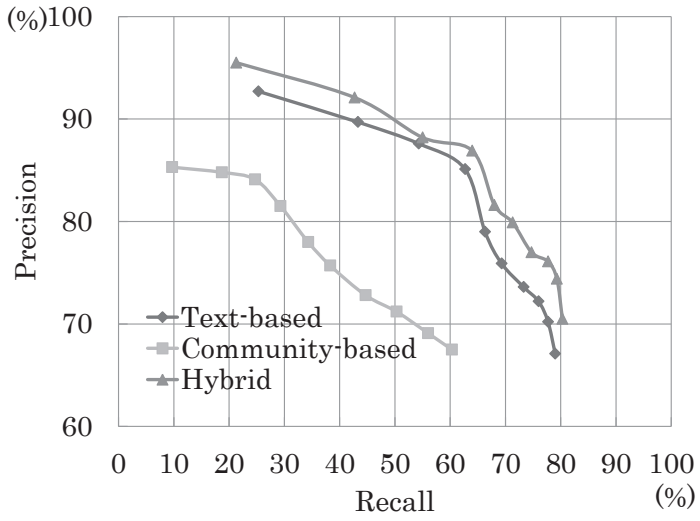


Figure 6.4: Comparison of the Recall and the Precision (%) of the Proposed Methods for the Estimation of Teens

text-based method, the community-based method and the hybrid method as general metrics to understand the features of each method. As for the text-based method, the estimation accuracy for teens is relatively high, and the estimation accuracy for twenties and thirties is relatively low. Most teens may have similar characteristics, such as being students. On the other hand, correct detection of people in their twenties and thirties who may have similar lifestyles is assumed to be difficult. The estimation accuracy for gender is high for both males and females. The estimation accuracy for area is low in Kanto, which is the capital region of Japan. People from other regions may often visit and tweet about the Kanto area. The estimation accuracy for occupation is especially high for students and homemakers, where the terms are characteristic and the variety within communities relatively small. The average estimation accuracy for hobbies is limited at 34.8%. Considering that the number of hobby categories is twelve, the estimation method is effective. The average estimation accuracy for material status is as high as 82.5% in the hybrid method.

As for the community-based method, although it produces F-values lower than those from the text-based method, the precision is high in the low-recall

Table 6.9: Estimation Accuracy of the Proposed Methods in F-value (%)

Type	Segment	Text-based	Community-based	Hybrid
Age	10s	72.6	63.7	75.1
	20s	53.5	41.3	57.5
	30s	53.8	48.6	55.8
	40s over	62.2	43.1	63.0
	Average	60.5	49.2	62.9
Gender	Male	82.7	74.5	83.3
	Female	83.3	61.1	83.7
	Average	83.0	67.8	83.5
Area	Hokkaido/Tohoku	72.2	51.6	77.8
	Kanto	55.8	68.3	57.4
	Hokushinetsu	76.6	43.9	81.6
	Tokai	71.9	48.9	75.1
	Kinki	72.6	65.9	75.9
	Chugoku/Shikoku	70.8	54.0	77.5
	Kyushu/Okinawa	74.6	58.7	79.1
	Average	70.6	55.9	74.9
Occupation	Civil Servant	70.8	66.2	73.0
	Employee	62.3	54.1	64.6
	Homemaker	69.6	68.4	77.7
	Self Employed	57.6	57.8	73.7
	Part-time	65.1	55.8	73.7
	Student	67.6	59.6	81.8
	Without occupation	60.6	62.1	51.6
	Average	64.8	60.6	70.9
Hobby	Music	34.6	34.5	34.6
	Reading	32.0	27.1	38.2
	Games	41.2	38.3	38.2
	TV & Movie	36.8	33.9	37.7
	IT & Electronics	43.4	39.4	48.0
	Travel	30.1	32.9	32.4
	Gourmet	28.9	31.1	32.2
	Fashion	28.7	28.8	29.8
	Sports	26.8	30.4	32.5
	Pets & Plants	37.5	34.1	42.1
	Vehicle	41.8	39.0	43.4
	Arts	35.7	36.2	38.1
	Average	34.8	33.8	37.3
Marital Status	Married	82.7	77.4	83.5
	Unmarried	82.4	75.7	83.5
	Average	82.5	76.6	83.5

range. The estimation tendency is similar for age, where its estimation accuracy for teens is higher than for other segments. As for area, unlike the text-based method, its estimation accuracy for the Kanto area is high. This is probably because the performance reduction due to visitors from other areas observed in the text-based method is avoided by using community information.

Application of the hybrid method increased the estimation accuracy for each

demographic type. In particular, the average estimation accuracy for occupation is increased by 6.1 points in F-value compared to the text-based method. The average estimation accuracy for age, area, and hobby also increased in F-value by 2.4 points, 4.3 points, and 2.5 points, respectively. Detecting communities consisting of users in the same demographic may increase the accuracy for users whose estimation is difficult when based only on text information. The large improvement observed in the occupation derives from work colleagues, school friends, and networks of homemakers, who are assumed to affect online communities. The accuracy for gender is not improved because those that have a significantly different ratio of males to females in the community are often estimated correctly.

Table 6.10 shows the error for the distribution ratio of each user segment in both the text-based method and the hybrid method. The gaps between 300 target users for each demographic segment and the number of users estimated as each demographic segment are evaluated. Considering age for example, the number of total target users is 1,200, consisting of 300 per segment. The text-based method estimated 353 users as teens, 253 users as twenties, 325 users as thirties, and 269 users as over-forties. The error E calculated by expression 6.6 in Section 6.3.2 is 17.7% for teens, -15.7% for twenties, 8.3% for thirties, and -10.3% for over-forties. The average error E_{avg} of the hybrid method is smaller than that of the text-based method by 3.3 points for age, 1.7 points for gender, 2.4 points for area, 9.1 points for occupation, 6.9 points for hobby, and 2.0 points for marital status. The result shows that the hybrid method improves the estimation accuracy for the distribution of user demographics. The hybrid method is particularly effective for the categories occupation and hobby, for which the community-based method works well. With the hybrid method, the E_{avg} decreased to less than the 10% for each demographic.

Although we compare the performance of the hybrid of the text-based and the community-based methods, the hybrid of any other text-based methods such as [58, 59, 60] and the community-based method would be assumed to deliver improved performance in the same manner because the sources used by text-based and community-based methods are independent of each other. The hybrid method requires less than one second of processing time per CPU for estimation of all demographics of a user. This means only one month is

Table 6.10: Error for Distribution Ratio in Each Demographic

Demographic		# of target users	Text-based		Hybrid	
Types	Segment		# of estimated	Error(%)	# of estimated	Error(%)
Age	10s	300	353	17.7	342	14.0
	20s	300	253	-15.7	274	-8.7
	30s	300	325	8.3	316	5.3
	40s and over	300	269	-10.3	268	-10.7
	Average			13.0		9.7
Gender	Male	300	288	-4.0	293	-2.3
	Female	300	312	4.0	307	2.3
	Average			4.0		2.3
Area	Hokkaido/Tohoku	300	293	-2.3	299	-0.3
	Kanto	300	360	20.0	331	10.3
	Hokushinetsu	300	282	-6.0	298	-0.7
	Tokai	300	281	-6.3	286	-4.7
	Kinki	300	303	1.0	285	-5.0
	Chugoku/Shikoku	300	302	0.7	309	3.0
	Kyushu/Okinawa	300	279	-7.0	292	-2.7
	Average			6.2		3.8
Occupation	Civil Servant	300	241	-19.5	271	-9.7
	Employee	300	402	34.1	290	-3.2
	Homemaker	300	340	13.4	343	14.4
	Self Employed	300	326	8.5	329	9.7
	Part-time	300	256	-14.6	288	-4.2
	Student	300	282	-6.1	289	-3.7
	Without occupation	300	252	-15.9	290	-3.2
	Average			16.0		6.9
Hobby	Music	300	266	-11.5	308	2.7
	Reading	300	304	1.5	254	-15.3
	Games	300	334	11.5	346	15.3
	TV & Movie	300	221	-26.3	263	-12.3
	IT & Electronics	300	252	-15.9	264	-12.0
	Travel	300	312	4.1	301	0.3
	Gourmet	300	261	-13.0	277	-7.7
	Fashion	300	428	42.6	345	15.0
	Sports	300	243	-18.9	267	-11.0
	Pets & Plants	300	389	29.6	344	14.7
	Vehicle	300	258	-14.1	301	0.3
	Arts	300	331	10.4	330	10.0
	Average			16.6		9.7
Marital Status	Male	300	306	2.0	300	0.0
	Female	300	294	-2.0	300	0.0
	Average			2.0		0.0

required for estimating the demographics of the twelve million Twitter users in Japan with a quad core computer. For real-time applications, such as analyzing opinions about on-air television programs, it is also possible to estimate non-registered users in real time.

6.4 Conclusion

In this chapter, we have propose a hybrid demographic estimation method of Twitter users based on their past tweets and communities constructed from follower/followee relationships. There have been no previous proposals for large-scale and practical marketing analysis methods of such demographic estimation due to the difficulty of producing a method with sufficient effectiveness and accuracy for practical use.

The proposed hybrid method is applicable to multiple user demographics and to users who make few tweets by making the best use of the Twitter platform, which includes tweets as text information and followers/followees as community information. Our experimental results show that the estimation accuracy (in F-value) of the proposed hybrid method is 83.5% for gender, 62.9% for age, and 74.9% for area. The estimation accuracy of the proposed method is high enough to allow practical utilization of most of the demographic information. The processing time of the method is low enough to analyze all Japanese Twitter users in a month. Demographic information attracts businesses, such as product planners, advertisement agencies, and customer support services, all of which are interested in the demographic distribution of users and their opinions.

In order to deploy the technology widely, we also realize novel applications for browsing online opinions in real-time, categorized by the estimated demographics and the comments' positive/negative character. We have already deployed the application in several departments and companies. In the product planning department, when they draw up a new product proposal, the application is used to include customer opinions about the previous model as feedback into the proposal. In the customer service management department, they use the application to monitor customer opinion about their released products and services. We are also working with TV stations, and the proposed technology has been used on a live debate television program broadcast in Japan, where the question "What shrank Japan?" was announced in the official Twitter account of the program in advance and audiences replied to the question. The estimated distribution of age and gender of twitter users who answered the question was presented on the program. Remarkably, "limitation of freedom of expression shrank Japan" was a characteristic answer among young people, and this was discussed as the topic of the program. We are also providing the demographic

estimation APIs for the company who has their own online opinion analysis services.

In this chapter, we introduce the evaluation results of Japanese Twitter users. However, the approach described in this chapter is applicable to other languages simply by replacing the linguistic analysis tools and collecting the profiles of Twitter users in the manner described in Section 6.2.1. We have also implemented and been deploying an English version of the proposed method and its application in the United States.

Chapter 7

Conclusion

In this thesis, the following three research topics are studied to utilize real-time and huge amounts of online opinions on both businesses and consumers: (1) linguistic analysis techniques for morphological analysis of peculiar expressions, (2) highly accurate detection techniques for malicious information, and (3) user profiling techniques for online opinions.

The existing work related with those topics is summarized in Chapter 2. We refer to related work concerned with linguistic analysis techniques, techniques for malicious document detection, and user profiling techniques.

In Chapter 3, we propose a method for reducing the number of unknown words by replacing peculiar expressions seen in online opinions with formal expressions. In our algorithm, candidates for the substitution of peculiar expressions are automatically retrieved from formally written documents and stored as substitution rules. In order to replace a peculiar expression with the most suitable expression for the context, a substitution rule is selected based on three criteria; its appearance frequency in the retrieval process, the edit distance between substituted sequences and the original text, and the estimated accuracy improvements of word segmentation after the substitution. The experimental results show our algorithm reduces 30.3% of unknown words in original blog documents at the same segmentation accuracy as conventional ones. This reduction rate is higher than twice of the rate of the conventional algorithm.

In Chapter 4, we propose a method to increase the accuracy of malicious Web page detection by correcting the classification of a conventional text-based method based on the dependency relations of the malicious keywords and their

neighboring segments. In addition, we propose a practical algorithm to increase performance by expanding the malicious segment pairs using a thesaurus. In our experiments with a large scale Web pages, the performance of the proposed base-line method improves the performance of the conventional method by up to 6.6% in F value. Removing noisy segment pairs based on their expanded path is also effective which increases the peak performance of the base-line method by 0.75% and the improvement from the conventional method is 7.3% in F value.

In Chapter 5, we propose a high-speed, accurate method for detecting malicious Web pages. Our algorithm automatically chooses strings that appear especially in the HTML elements of malicious Web pages based on AIC. We use these strings in combination as features of SVMs in order to detect malicious Web pages. We show that our proposed method can detect those Web pages that existing text-based methods have difficulty in detecting by confirming the differences between our proposed method and an existing text-based method in terms of their tendency to detect malicious Web pages by experiments. In our experimental environments with manually labeled 40,000 Web pages, we confirm that the performance of the hybrid method of combining our method with existing text-based method is a recall rate of 70.0% with a precision rate of 78.1%, which is 9.3% better precision than the same recall of existing text-based method when used on their own.

Finally, in Chapter 6, we propose a hybrid demographic estimation method of Twitter users based on their past tweets and communities constructed from follower/followee relationships. There is no previous proposals for large-scale and practical marketing analysis methods of such demographic estimation due to the difficulty of producing a method with sufficient effectiveness and accuracy for practical use. The proposed hybrid method is applicable to multiple user demographics and to users who make few tweets by making the best use of the Twitter platform, which includes tweets as text information and followers/followees as community information. Our experimental results show that the estimation accuracy (in F-value) of the proposed hybrid method is 83.5% for gender, 62.9% for age, and 74.9% for area. The estimation accuracy of the proposed method is high enough to allow practical utilization of most of the demographic information. The processing time of the method is low enough to analyze all Japanese Twitter users in a month. Demographic information attracts

businesses, such as product planners, advertisement agencies, and customer support services, all of which are interested in the demographic distribution of users and their opinions.

As future work, we are planning to study advanced analysis techniques for realizing a greater variety of additional services. Specifically, we need to study techniques for sentiment analysis in order to meet the demands of accurate sentiment analysis. In order to maximize the advantages of online opinion analysis, we need mechanisms which lead the findings from online opinions to actions in the real world. For example, analysis techniques for predicting future trends in the real world from online opinions can realize innovative services. Although we focus on techniques for analysis of online opinions in this thesis, techniques for collection and distribution of online opinions are also important. Resolving ambiguity of search terms and completion or suggestion of queries are essential techniques for improving user convenience. In addition, platforms for storing and searching these huge amount of online opinions in real-time are also required. From the viewpoint of distribution of analyzed information, techniques for visualization of text information and network management, which allow real-time distribution of information on various devices such as smartphones and tablets are important.

Acknowledgements

I would like to express my sincerely appreciation to continued support of my supervisor Professor Teruo Higashino of Osaka University through trials and tribulations of this Ph.D thesis. I would like also thank his encouragement and invaluable comments in preparing this thesis.

I am very grateful to Professor Masayuki Murata, Professor Koso Murakami and Professor Hirotaka Nakano of Osaka University for their invaluable comments and helpful suggestions concerning this thesis.

I am also very grateful to Associate Professor Hirozumi Yamaguchi, Assistant Professor Takaaki Umedu, Assistant Professor Akihito Hiromori, Assistant Professor Akira Uchiyama, of Osaka University, and Thilmee M. Baduge of Panasonic Corporation for their valuable comments.

I appreciate the executives of KDDI R&D Laboratories, Inc. for their kind support to this research, namely: Dr. Yutaka Yasuda, Chairman of the Board of Directors, Dr. Yasuyuki Nakajima, President, CEO, Dr. Masatoshi Suzuki, Executive Vice President, Dr. Shigeyuki Akiba, Senior Vice President, Executive Principal Research Engineer, and Dr. Shuichi Matsumoto, Vice President.

I also deeply appreciate Dr. Yasuhiro Takishima, Exective Director of Human Communication Division of KDDI R&D Laboratories, Inc. for the continuous guidance since I entered the company. I would like to express my thanks to Dr. Chihiro Ono, Group Leader of Intelligent Media Processing Laboratory of KDDI R&D Laboratories, Inc. for taking the opportunity of writing this thesis. I am also very grateful to Dr. Kazunori Matsumoto, Dr. Gen Hattori, of Intelligent Media Processing Laboratory of KDDI R&D Laboratories, Inc., Mr. Tadashi Yanagihara of TOYOTA InfoTechnology Center Co., Ltd., and Dr. Hideki Asoh of National Institute of Advanced Industrial Science and Technology for their kind support throughout the process of the research on this

thesis. I am deeply grateful to Dr. Masami Suzuki, Mr. Hiromi Ishizaki, Ms. Masami Nakazawa, Dr. Maike Erdmann, of Intelligent Media Processing Laboratory of KDDI R&D Laboratories, Inc., Dr. Keiichiro Hoashi, Group Leader of Application Platform Group of KDDI R&D Laboratories, Ms. Yukiko Habu, Dr. Akihiro Kobayashi, and Mr. Tomoya Takeyoshi, of Application Platform Group of KDDI R&D Laboratories for their fruitful discussion.

Last but not least, I express special thanks to my family and especially my wife for her support and understanding.

Bibliography

[1] Serge Abiteboul, "Querying Semi-Structured Data", in *Proceedings of the 6th International Conference on Database Theory (ICDT)*, 1997, pp. 1–18.

[2] Sergey Brin and Lawrence Page, "The Anatomy of a Large-Scale Hypertextual Web Search Engine", in *Proceedings of the 7th International Conference on World Wide Web (WWW)*, 1998, pp. 107–117.

[3] Peter Buneman, Susan Davidson, Gerd Hillebrand, and Dan Suciu, "A Query Language and Optimization Techniques for Unstructured Data", in *Proceedings of the 1996 ACM SIGMOD International Conference on Management of Data (SIGMOD)*, 1996, pp. 505–516.

[4] Soumen Chakrabarti, Byron E. Dom, Ravi Kumar, Prabhakar Raghavan, Sridhar Rajagopalan, and Andrew Tomkins, "Mining the Link Structure of the World Wide Web", *Journal of IEEE Computer*, vol. 32, no. 8, pp. 60–67, 1999.

[5] Francis Crimmins, Alan F. Smeaton, Taoufiq Dkaki, and Josiane Mothe, "TétraFusion: information discovery on the Internet", *Journal of IEEE Intelligent Systems & Their Applications*, vol. 14, no. 4, pp. 55–62, 1999.

[6] Doug Beeferman and Adam Berger, "Agglomerative Clustering of a Search Engine Query Log", in *Proceedings of the 6th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining (SIGKDD)*, 2000, pp. 407–416.

[7] Jagdev Bhogal, Andy Macfarlane, and Peter Smith, "A Review of Ontology Based Query Expansion", *Journal of Information Processing and Management*, vol. 43, no. 4, pp. 866–886, 2007.

[8] Bodo Billerbeck and Justin Zobel, "Techniques for Efficient Query Expansion", in *Proceedings of the 11th String Processing and Information Retrieval Symposium (SPIRE)*, 2004, pp. 30–42.

[9] Georg Buscher, Andreas Dengel, and Ludger V. Elst, "Query Expansion Using Gaze-based Feedback on the Subdocument Level", in *Proceedings of the 31st Annual International ACM SIGIR Conference on Research and Development in Information Retrieval (SIGIR)*, 2008, pp. 387–394.

[10] Claudio Carpineto, Giovanni Romano, and Vittorio Giannini, "Improving Retrieval Feedback with Multiple Term-Ranking Function Combination", *Journal of ACM Transactions on Information Systems (TOIS)*, vol. 20, no. 3, pp. 259–290, 2002.

[11] Kevyn C. Thompson, "Reducing the Risk of Query Expansion via Robust Constrained Optimization", in *Proceedings of the 18th ACM Conference on Information and Knowledge Management (CIKM)*, 2009, pp. 837–846.

[12] Jens Graupmann, Jun Cai, and Ralf Schenkel, "Automatic Query Refinement Using Mined Semantic Relations", in *Proceedings of the 2005 International Workshop on Challenges in Web Information Retrieval and Integration (WIRI)*, 2005, pp. 205–213.

[13] Hugh E. Williams, Justin Zobel, and Dirk Bahle, "Fast Phrase Querying with Combined Indexes", *Journal of ACM Transactions on Information Systems (TOIS)*, vol. 22, no. 4, pp. 573–594, 2004.

[14] David Hawking, "Scalable Text Retrieval for Large Digital Libraries", in *Proceedings of the 1st European Conference on Research and Advanced Technology for Digital Libraries (ECDL)*, 1997, pp. 127–145.

[15] Tomek Strzalkowski and Barabara Vauthey, "Fast Text Processing for Information Retrieval", in *Proceedings of the Workshop on Speech and Natural Language*, 1991, pp. 346–352.

[16] Gerard Salton, "Developments in Automatic Text Retrieval", *Journal of Science*, vol. 253, no. 5023, pp. 974–980, 1991.

[17] David C. Blair and Melvin E. Maron, "An Evaluation of Retrieval Effectiveness for a Full-Text Document-Retrieval System", *Journal of Communications of the ACM (CACM)*, vol. 28, no. 3, pp. 289–299, 1985.

[18] Stefan Büttcher, Charles L. A. Clarke, and Brad Lushman, "Term Proximity Scoring for Ad-Hoc Retrieval on Very Large Text Collections", in *Proceedings of the 29th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval (SIGIR)*, 2006, pp. 621–622.

[19] Jonathan L. Elsas, Jaime Arguello, Jamie Callan, and Jaime G. Carbonell, "Retrieval and Feedback Models for Blog Feed Search", in *Proceedings of the 31st Annual International ACM SIGIR Conference on Research and Development in Information Retrieval (SIGIR)*, 2008, pp. 347–354.

[20] Wei Zhanga, Clement Yu, and Weiyi Meng, "Opinion Retrieval from Blogs", in *Proceedings of the 16th ACM Conference on Conference on Information and Knowledge Management (CIKM)*, 2007, pp. 831–840.

[21] Andreas Hotho, Robert Jäschke, Christoph Schmitz, and Gerd Stumme, "Information Retrieval in Folksonomies: Search and Ranking", in *Proceedings of the 3rd European Conference on The Semantic Web: Research and Applications (ESWC)*, 2006, pp. 411–426.

[22] Erik Ward, Kazushi Ikeda, Maike Erdmann, Masami Nakazawa, Gen Hattori, and Chihiro Ono, "Automatic Query Expansion and Classification for Television Related Tweet Collection", *Journal of Information Processing Society of Japan (IPSJ) SIG Technical Reports*, vol. 2012, no. 10, pp. 1–8, 2012.

[23] Taku Kudo, Kaoru Yamamoto, and Yuji Matsumoto, "Applying Conditional Random Fields to Japanese Morphological Analysis", in *Proceedings of the 2004 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, 2004, pp. 230–237, `http://mecab.sourceforge.net/`.

[24] Chris Davatzikos, Marc Vaillant, Susan M. Resnick, Jerry L. Prince, Stanley Letovsky, and Bryan R. Nick, "A Computerized Approach for Morphological Analysis of the Corpus Callosum", *Journal of Computed Assisted Tomography*, vol. 20, no. 1, pp. 88–97, 1996.

[25] Taku Kudo and Yuji Matsumoto, "Japanese Dependency Analysis using Cascaded Chunking", in *Proceedings of the 6th Conference on Natural Language Learning (CONLL)*, 2002, pp. 63–69.

[26] Joakim Nivre and Johan Hall, "Maltparser: A Language-Independent System for Data-Driven Dependency Parsing", in *Proceedings of the 4th Workshop on Treebanks and Linguistic Theories (TLT)*, 2005, pp. 13–95.

[27] Daisuke Kawahara and Sadao Kurohashi, "Fertilization of Case Frame Dictionary for Robust Japanese Case Analysis", in *Proceedings of the 19th International Conference on Computational Linguistics (COLING)*, 2002, pp. 425–431.

[28] Daisuke Kawahara and Sadao Kurohashi, "Case Frame Construction by Coupling the Predicate and its Closest Case Component", *Journal of Natural Language Processing (NLP)*, vol. 9, no. 1, pp. 3–19, 2002.

[29] Janyce Wiebe and Ellen Riloff, "Finding Mutual Benefit between Subjectivity Analysis and Information Extraction", *Journal of IEEE Transactions on Affective Computing*, vol. 2, no. 4, pp. 175–191, 2011.

[30] Kushal Dave, Steve Lawrence, and David M. Pennock, "Mining the Peanut Gallery: Opinion Extraction and Semantic Classification of Product Reviews", in *Proceedings of the 12th International Conference on World Wide Web (WWW)*, 2003, pp. 519–528.

[31] Namrata Godbole, Manjunath Srinivasaiah, and Steven Skiena, "Large-Scale Sentiment Analysis for News and Blogs", in *Proceedings of the 1st International Conference on Weblogs and Social Media (ICWSM)*, 2007.

[32] Fabrizio Sebastiani, "Machine Learning in Automated Text Categorization", *Journal of ACM Computing Surveys*, vol. 34, no. 1, pp. 1–47, 2002.

[33] Thorsten Joachims, "Text Categorization with Support Vector Machines: Learning with Many Relevant Features", in *Proceedings of the 10th European Conference on Machine Learning (ECML)*, 1998, pp. 137–142.

[34] Alexander Genkin, David D. Lewis, and David Madigan, "Large-Scale Bayesian Logistic Regression for Text Categorization", *Journal of Technometrics*, vol. 49, no. 3, pp. 291–304, 2007.

[35] Farshad Kyoomarsi, Hamid Khosravi, Esfandiar Eslami, Pooya K. Dehkordy, and Asghar Tajoddin, "Optimizing Text Summarization Based on Fuzzy Logic", in *Proceedings of the 7th IEEE/ACIS International Conference on Computer and Information Science (ACIS-ICIS)*, 2008, pp. 347–352.

[36] Fang Chen, Kesong Han, and Guilin Chen, "An Approach to Sentence-Selection-Based Text Summarization", in *Proceedings of the 2002 IEEE Region 10 Conference on Computers, Communications, Control and Power Engineering (TENCON)*, 2002, vol. 1, pp. 489–493.

[37] Yihong Gong and Xin Liu, "Generic Text Summarization Using Relevance Measure and Latent Semantic Analysis", in *Proceedings of the 24th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval (SIGIR)*, 2001, pp. 19–25.

[38] Kevin Knight and Daniel Marcu, "Summarization Beyond Sentence Extraction: A Probabilistic Approach to Sentence Compression", *Journal of Artificial Intelligence*, vol. 139, no. 1, pp. 91–107, 2002.

[39] James A. Wise, "The Ecological Approach to Text Visualization", *Journal of the American Society for Information Science (JASIS)*, vol. 50, no. 13, pp. 1224–1233, 1999.

[40] James A. Wise, James J. Thomas, Kelly Pennock, David Lantrip, Marc Pottier, Anne Schur, and Vern Crow, "Visualizing the Non-Visual: Spatial Analysis and Interaction with Information from Text Documents", in *Proceedings of the IEEE Symposium on Information Visualization 1995 (INFOVIS)*, 1995, pp. 51–58.

[41] Nancy E. Miller, Pak C. Wong, Mary Brewster, and Harlan Foote, "Topic Islands - A Wavelet-Based Text Visualization System", in *Proceedings of the Conference on Visualization (VIS)*, 1998, pp. 189–196.

[42] Marko Grobelnik and Dunja Mladenić, "Efficient Visualization of Large Text Corpora", in *Proceedings of the 7th Trans-European Language Resources Infrastructure (TELRI) seminar*, 2002.

[43] Greg Linden, Brent Smith, and Jeremy York, "Amazon.com Recommendations: Item-to-Item Collaborative Filtering", *Journal of IEEE Internet Computing*, vol. 7, no. 1, pp. 76–80, 2003.

[44] John S. Breese, David Heckerman, and Carl Kadie, "Empirical Analysis of Predictive Algorithm for Collaborative Filtering", in *Proceedings of the 14th Conference on Uncertainty in Artificial Intelligence (UAI)*, 1998, pp. 43–52.

[45] Jonathan L. Herlocker, Joseph A. Konstan, Loren G. Terveen, and John T. Riedl, "Evaluating Collaborative Filtering Recommender Systems", *Journal of ACM Transactions on Information Systems (TOIS)*, vol. 22, no. 1, pp. 5–53, 2004.

[46] Michael J. Pazzani and Daniel Billsus, "Content-Based Recommendation Systems", *Journal of Springer-Verlag the Adaptive Web*, vol. 4321, pp. 325–341, 2007.

[47] Chumki Basu, Haym Hirsh, and William Cohen, "Recommendation as Classification: Using Social and Content-Based Information in Recommendation", in *Proceedings of the 15th National Conference on Artificial Intelligence (AAAI)*, 1998, pp. 714–720.

[48] Raymond J. Mooney and Loriene Roy, "Content-Based Book Recommending Using Learning for Text Categorization", in *Proceedings of the 5th ACM Conference on Digital Libraries (DL)*, 2000, pp. 195–204.

[49] Chihiro Ono, Kazushi Ikeda, Gen Hattori, Kazunori Matsumoto, and Yasuhiro Takishima, "TV Viewing Support System Considering Both Individual and Family Preferences", in *Proceedings of the 18th User Modeling, Adaptation and Personalization (UMAP)*, 2010, pp. 31–33.

[50] Junichi Kazama, Yutaka Mitsuishi, Takaki Makino, Torisawa Kentaro, Kouichi Matsuda, and Junichi Tsujii, "Morphological Analysis for Japanese Web Chat", in *Proceedings of Natural Language Processing (NLP)*, 1999, pp. 509–512.

[51] Kazushi Ikeda, Tadashi Yanagihara, Kazunori Matsumoto, and Yasuhiro Takishima, "An Automatic Rule Generation Method for Modifying Infor-

mal Expression in Blog Documents", *Journal of the Database Society of Japan (DBSJ)*, vol. 8, no. 1, pp. 23–28, 2009.

[52] Tadashi Yanagihara, Kazushi Ikeda, Kazunori Matsumoto, and Yasuhiro Takishima, "Fast n-gram Assortment Construction for Filtering Hazardous Information", *Journal of Information Processing Society of Japan (IPSJ) SIG Technical Reports*, vol. 3, pp. 1–5, 2009.

[53] Keiichiro Hoashi, Kazunori Matsumoto, Naomi Inoue, and Kazuo Hashimoto, "Document Filtering Method Using Non-Relevant Information Profile", in *Proceedings of the 23rd Annual International ACM SIGIR Conference on Research and Development in Information Retrieval (SIGIR)*, 2000, pp. 176–183.

[54] Atsushi Matsumura, Atsuhiro Takasu, and Jun Adachi, "The Effect of Information Retrieval Method Using Dependency Relationship Between Words", in *Proceedings of Recherche d'Information Assistée par Ordinateur (RIAO)*, 2000, pp. 1043–1058.

[55] Wai H. Ho and Paul A. Watters, "Statistical and Structural Approaches to Filtering Internet Pornography", in *Proceedings of 2004 IEEE International Conference on Systems, Man and Cybernetics (SMC)*, 2004, pp. 4792–4798.

[56] Yoshikiyo Kato, Daisuke Kawahara, Kentaro Inui, Sadao Kurohashi, and Tomohide Shibata, "Identifying Information Sender Configuration of Web Pages", in *Proceedings of the 2009 IEEE/WIC/ACM International Conference on Web Intelligence and Intelligent Agent Technologies (WI-IAT)*, 2009, vol. 1, pp. 335–340.

[57] Moshe Koppel, Jonathan Schler, and Shlomo Argamon, "Computational Methods in Authorship Attribution", *Journal of the American Society for Information Science and Technology (ASIS&T)*, vol. 60, no. 1, pp. 9–26, 2009.

[58] Shlomo Argamon, Moshe Koppel, James W. Pennebaker, and Jonathan Schler, "Automatically Profiling the Author of an Anonymous Text", *Journal of Communications of the ACM (CACM)*, vol. 52, no. 2, pp. 119–123, 2009.

[59] Dominique Estival, Tanja Gaustad, Son B. Pham, Will Radford, and Ben Hutchinson, "Author Profiling for English Emails", in *Proceedings of the 10th Conference of the Pacific Association for Computational Linguistics (PACLING)*, 2007, pp. 262–272.

[60] Dang D. Pham, Giang B. Tran, and Son B. Pham, "Author Profiling for Vietnamese Blogs", in *Proceedings of the 2009 International Conference on Asian Language Processing (IALP)*, 2009, pp. 190–194.

[61] Ahmed Abbasi and Hsinchun Chen, "Writeprints: A Stylometric Approach to Identity-Level Identification and Similarity Detection in Cyberspace", *Journal of ACM Transactions on Information Systems (TOIS)*, vol. 26, no. 2, 2008.

[62] Moshe Koppel, Jonathan Schler, and Elisheva B. Dokow, "Measuring Differentiability: Unmasking Pseudonymous Authors", *Journal of Machine Learning Research*, vol. 8, pp. 1261–1276, 2007.

[63] Yoshikazu Takemoto and Shunichi Fukushima, "Implementation and Evaluation of a Morphological Analysis Method for Colloquial Japanese Text", *Journal of The Special Interest Group Notes of IPSJ (IPSJ SIG Notes)*, vol. 94, no. 77, pp. 105–112, 1994.

[64] Atsushi Takeshita and Hironobu Fukunaga, "Morphological Analysis for Spoken Language", in *Proceedings of the 42nd National Convention of Information Processing Society of Japan (IPSJ) 1C-3*, 1991.

[65] Yuji Matsumoto and Yasuharu Den, "Morphological Analysis of Spoken Japanese", *Journal of The Special Interest Group Notes of IPSJ (IPSJ SIG Notes)*, vol. 2001, no. 55, pp. 9–14, 2001.

[66] Takeshi Masuyama, Satoshi Sekine, and Hiroshi Nakagawa, "Automatic Construction of Japanese KATAKANA Variant List from Large Corpus", in *Proceedings of the 20th International Conference on Computational Linguistics (COLING)*, 2004, pp. 1214–1219.

[67] Yugo Murawaki and Sadao Kurohashi, "Online Acquisition of Japanese Unknown Morphemes using Morphological Constraints", in *Proceedings*

of the 2008 Conference on Empirical Methods in Natural Language Processing (EMNLP), 2008, pp. 429–437.

[68] Shinsuke Mori and Makoto Nagao, "Word Extraction from Corpora and Its Part-of-Speech Estimation Using Distributional Analysis", in *Proceedings of the 11th International Conference on Computational Linguistics (COLING)*, 1996, pp. 1119–1122.

[69] Tadashi Yanagihara, Kazunori Matsumoto, Kazushi Ikeda, and Yasuhiro Takishima, "Word Segmentation Estimation using Information Criteria", in *Proceedings of Information Processing Society of Japan (IPSJ) NLP 190*, 2009, pp. 43–47.

[70] Renxu Sun, Chai H. Ong, and Tat S. Chua, "Mining Dependency Relations for Query Expansion in Passage Retrieval", in *Proceedings of the 29th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval (SIGIR)*, 2006, pp. 382–389.

[71] Ying Liu, Peter Scheuermann, Xingsen Li, and Xingquan Zhu, "Using WordNet to Disambiguate Word Senses for Text Classification", in *Proceedings of 7th International Conference on Computational Science (ICCS)*, 2007, vol. 4489, pp. 780–788.

[72] Ming H. Hsu, Ming F. Tsai, and Hsin H. Chen, "Combining WordNet and ConceptNet for Automatic Query Expansion: A Learning Approach", in *Proceedings of the 4th Asia Information Retrieval Symposium (AIRS)*, 2008, pp. 213–224.

[73] Masaharu Yoshioka and Makoto Haraguchi, "On a Combination of Probabilistic and Boolean IR Models for WWW Document Retrieval", *Journal of ACM Transactions on Asian Language Information Processing (TALIP)*, vol. 4, no. 4, pp. 340–356, 2005.

[74] Lin Li, Shingo Otsuka, and Masaru Kitsuregawa, "Finding Related Search Engine Queries by Web Community Based Query Enrichment", *Journal of World Wide Web*, vol. 13, no. 1–2, pp. 121–142, 2010.

[75] Kazushi Ikeda, Tadashi Yanagihara, Kazunori Matsumoto, and Yasuhiro Takisima, "Detection of Illegal and Hazardous Information Using Depen-

dency Relations and Keyword Abstraction", in *Proceedings of the 2nd Forum on Data Engineering and Information Management (DEIM) Conference C9-5*, 2010.

[76] Robert Layton, Paul Watters, and Richard Dazeley, "Authorship Attribution for Twitter in 140 Characters or Less", in *Proceedings of the 2nd Cybercrime and Trustworthy Computing Workshop (CTC 2010)*, 2010, pp. 1–8.

[77] Rui S. Silva, Gustavo Laboreiro, Luís Sarmento, Tim Grant, Eugénio Oliveira, and Belinda Maia, "'twazn me!!! ;(' Automatic Authorship Analysis of Micro-Blogging Messages", in *Proceedings of the 16th International Conference on Applications of Natural Language to Information Systems (NLDB)*, 2011, pp. 161–168.

[78] Henry Small, "Co-citation in the Scientific Literature: A New Measure of the Relationship Between Two Documents", *Journal of the American Society for Information Science (JASIS)*, vol. 24, no. 4, pp. 265–269, 1973.

[79] Natthakan I. On, Tossapon Boongeon, Simon Garrett, and Chris Price, "A Link-Based Cluster Ensemble Approach for Categorical Data Clustering", *Journal of IEEE Transactions on Knowledge and Data Engineering (TKDE)*, vol. 24, no. 3, pp. 413–425, 2010.

[80] Pável Calado, Marco Cristo, Marcos A. Gonçalves, Edleno S. Moura, Berthier R. Neto, and Nivio Ziviani, "Link-based similarity measures for the classification of Web documents", *Journal of the American Society for Information Science and Technology (ASIS&T)*, vol. 57, no. 2, pp. 208–221, 2006.

[81] Xiaoguang Qi and Brian D. Davison, "Knowing a Web Page by the Company It Keeps", in *Proceedings of the 15th ACM International Conference on Information and Knowledge Management (CIKM)*, 2006, pp. 228–237.

[82] Tong Zhang, Alexandrin Popescul, and Byron Dom, "Linear Prediction Models with Graph Regularization for Web-page Categorization", in *Proceedings of the 12th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining (SIGKDD)*, 2006, pp. 821–826.

[83] Xiaoguang Qi and Brian D. Davison, "Web Page Classification: Features and Algorithms", *Journal of the ACM Computing Surveys (CSUR)*, vol. 41, no. 2, pp. 208–221, 2009.

[84] Kazushi Ikeda, Gen Hattori, Kazunori Matsumoto, Chihiro Ono, and Yasuhiro Takishima, "Social Media Visualization for TV", in *Proceedings of the International Broadcasting Convention (IBC) Conference D-243*, 2011.

[85] Vladimir I. Levenshtein, "Binary Codes Capable of Correcting Deletions, Insertions and Reversals", *Journal of Soviet Physics Doklady*, vol. 10, pp. 707–710, 1966.

[86] Masaaki Nagata, "A Stochastic Japanese Morphological Analyzer Using a Forward-DP Backward-A* N-Best Search Algorithm", in *Proceedings of the 15th International Conference on Computational Linguistics (COLING)*, 1994, vol. 1, pp. 201–207.

[87] Hirotugu Akaike, "A New Look at the Statistical Model Identification", *Journal of IEEE Transactions on Automatic Control*, vol. 19, no. 6, pp. 716–723, 2003.

[88] Kazunori Matsumoto and Kazuo Hashimoto, "Schema Design for Causal Law Mining from Incomplete Database", in *Proceedings of the 2nd International Conference on Discovery Science (DS)*, 1999, pp. 92–102.

[89] National Institute of Information and Communications Technology, "EDR Thesaurus", `http://www2.nict.go.jp/r/r312/EDR/index.html`.

[90] Daisuke Kawahara and Sadao Kurohashi, "A Fully-Lexicalized Probabilistic Model for Japanese Syntactic and Case Structure Analysis", in *Proceedings of the main conference on Human Language Technology Conference of the North American Chapter of the Association of Computational Linguistics (HLT-NAACL)*, 2006, pp. 176–183.

[91] Mitsuo Yoshida and Mikio Yamamoto, "Primary Content Extraction from News Pages without Training Data", *Journal of Database Society of Japan (DBSJ)*, vol. 8, no. 1, pp. 29–34, 2009.

[92] Shian H. Lin and Jan M. Ho, "Discovering Informative Content Blocks from Web Documents", in *Proceedings of the 8th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining (SIGKDD)*, 2002, pp. 588–593.

[93] Corinna Cortes and Vladimir Vapnik, "Support-Vector Networks", *Journal of Machine Learning*, vol. 20, no. 3, pp. 273–297, 1995.

[94] Quinlan J. Ross, *C4.5: Programs for Machine Learning*, Morgan Kaufmann Publishers Inc., 1993.

[95] Simon Haykin, *Neural Networks: A Comprehensive Foundation*, Prentice Hall PTR, 1998.

[96] David J. Hand, Heikki Mannila, and Padhraic Smyth, *Principles of Data Mining*, The MIT Press, 2001.

[97] Rong E. Fan, Pai H. Chen, and Chih J. Lin, "Working Set Selection Using Second Order Information for Training Support Vector Machines", *Journal of Machine Learning Research*, vol. 6, pp. 1889–1918, 2005, `http://www.csie.ntu.edu.tw/~cjlin/libsvm/`.

[98] Aaron Clauset, Mark E. J. Newman, and Cristopher Moore, "Finding Community Structure in Very Large Networks", *Journal of the Physical Review E*, vol. 70, no. 6, pp. 066111, 2004.

[99] Jure Leskovec, Kevin J. Lang, Anirban Dasgupta, and Michael W. Mahoney, "Statistical Properties of Community Structure in Large Social and Information Networks", in *Proceedings of the 17th International Conference on World Wide Web (WWW)*, 2008, pp. 695–704.

[100] Wei Dong, Moses Charikar, and Kai Li, "Efficient K-Nearest Neighbor Graph Construction for Generic Similarity Measures", in *Proceedings of the 20th International Conference on World Wide Web (WWW)*, 2011, pp. 577–586.

[101] Marc Smith, Natasa M. Frayling, Ben Shneiderman, Eduarda M. Rodrigues, Jure Leskovec, and Cody Dunne, "NodeXL: A Free and Open Network Overview, Discovery and Exploration Add-in for Excel

2007/2010", `http://nodexl.codeplex.com/` from the Social Media Research Foundation, `http://www.smrfoundation.org` (2010).