Text Classification in Practice:Formalizing the structure of real problems

石田, 栄美

Text Classification in Practice:
Formalizing the structure of real problems

January 2019

Emi Ishita

Contents

Chapter 1

Introduction

Various machine learning classifiers such as Naïve Bayes, Decision Trees and Support Vector Machines have been developed. In addition, classifiers using Deep Learning methods have also recently been proposed. As the amount of digitized and born-digital data continues to increase, attention to the design and application of classifiers has continued to increase as well.

In text classification research, with the test collection for automated text classification (e.g., Reuters-21578 [1]) provided, theoretical research and the development and improvement of classifiers have been actively conducted. Classification using a support vector machine (SVM) classifier with use of WordNet information, for example, has achieved high classification performance by obtaining an F value of 0.94 or higher for the classification of topic categories such as *earn* and *grain* in Reuters-21578 [2]. The relationship between a document topic and words or phrases is clear because the classification of a document topic is conducted based on the content of the document. Therefore, a classifier such as the SVM can classify documents into related topic categories, using words and phrases characterized as the document topic. However, for classifying documents into non-topical categories, such as the detection of academic papers, the types of information that are effective for characterizing documents are unclear.

For a machine-learning classifier, training data are required. Training data are a set of documents with assigned categories. In general, assigning categories to documents is conducted manually by human coders. When newspaper articles are assigned into topic categories such as corporate/industrial, economics, government/social, and market [3], it is relatively easy to determine the articles that should be assigned to a particular category by reading the contents of the articles. However, to assess whether, for example, a given set of articles primarily discuss the survival of nuclear power plants after the Great East Japan Earthquake, or to infer human values reflected in opinion sentences, the construction of training data is time consuming and expensive. Therefore, in research focusing on the classification tasks to categories where it is difficult to determine whether the documents should be assigned to a given category, the following problems should be considered: How much training data should be prepared for the classifiers? Which should be prioritized when the budget to construct training data is limited: quantity or quality of the training data?

In this dissertation, three real problems in text classification that arise when introducing the classifiers in practice are explored.

The first part of this dissertation discusses the detection of academic papers. In web search, search engines respond to queries by providing documents that satisfy specific information needs. Search engines that focus only on topical matching, however, cannot perform distinctions based on the source credibility or information quality. Hence, it can be useful to augment a search engine with the results of a classifier for a document genre, so that academic papers, press releases, white papers, etc. might be recognized. In the current study, the task is to detect academic papers. Recently, open access journals and self-archiving of academic papers have become more common. Thus, it is now easy to obtain full-text papers, often in the PDF form, from the web. However, words that are used in other less reputable genres (e.g., blogs, personal web pages, or conspiracy sites) are often used in academic papers. Therefore, it is necessary to detect academic papers from documents of various genres (document types). In this study, information characterizing academic papers are identified and included into the feature set; subsequently, academic papers are detected using the trained classifiers. For the information, the section types, other elements of academic papers, and descriptive attributes of files such as file size, page number, and length were considered, in addition to words occurring in academic papers.

The second task is coding content in computational social science. Classifiers have been applied for quantitative analysis in social science research in recent years, notably (but not exclusively) for large-scale sentiment analysis. For example, sentiment classification has provided an effective and efficient method to measure positive or negative attitudes toward policies, politicians, brands, products, and services at the web scale [4]. However, a much deeper and more interesting question is why people hold certain sentiments. Human values, which can be defined as what people consider important in life [5], can be used to predict individuals' attitudes [6] and behaviors [7]. A method in which social scientists study human values is by using content analysis. Hence, social science researchers determine the scope of a set of documents to be analyzed (target documents), develop a set of categories (a "coding frame") for analyzing target documents, and manually assign categories to them (a "coding" process). Finally, they analyze statistically based on the number of categories given and then draw conclusions according to their research questions. Although a set of target documents is finite, the number of target documents becomes large when social scientists focus on socially significant topics for contents analysis. Social scientists typically conduct coding by hand for all target documents; however, this is difficult when coding at a large scale. Therefore, the method to code the entire set of target documents is challenging. A solution is by using classifiers, in which a human coder codes part of a document set (training set), a classifier is learned by the training set, and subsequently the trained classifier assigns categories to the remaining target documents. In this research, classifiers that estimate human values reflected in opinion sentences are explored first. Next, the rates of training data for classifiers are studied. The trained classifiers are trained by part of the human-coded documents (training data). Subsequently, the performances of the trained classifiers with

different numbers of training data are examined. The results of this experiment indicate the possibility of reducing a human coder's effort in the task of coding human values.

In content analysis for human values, the selection of documents to be analyzed is important as well as the coding of human values to sentences. The third task in this dissertation is to identify whether documents collected by the keyword search are target documents to be analyzed. The set of target documents consists of documents that primarily describe or discuss a topic of primary focus for content analysis. Even words related to a topic are included in documents; documents that do not involve the topic of content analysis are out of scope of the target documents. For instance, in the content analysis of newspaper articles on the discussion of nuclear power plants, an article that primarily describes a political situation such as a governor's election in Tokyo is out of target (off topic) even when the words "nuclear power plant" are included in that article. The topic identification task that assesses whether a document is to be analyzed (on-topic/off-topic) is a qualitative task of determining the primary topic of the document. The construction of training data is time consuming and expensive, as it involves the coding of human values. When constructing training data for classifiers, it is best to obtain a large amount of high-quality training data. However, the budgets for research projects are often limited. Within a certain budget, training data are to be constructed with the most suitable balance of quality and quantity that yield the best performance of the classifiers. In this study, the quality and number of documents required for achieving the best performance by classifiers under a certain cost are examined. In other words, the tradeoff between the quality and quantity of training data is explored.

1.1 Contributions of the Dissertation

In this section, the principal contributions of the dissertation are summarized.

First, the experiment results regarding the detection of academic papers indicate that features collected based on the characteristics of academic paper suffice for this task. These features include structural information such as section headings and component elements (such as the presence of references) and the descriptive attributes of a file of a particular genre such as file size, page number, and length, in addition to a tailored set of expressions as word-level content features. The benefits of this approach are demonstrated in two languages (Japanese and English) for an important genre (academic papers).

Next, the experiments indicate that machine-learning techniques for text classification can be used to infer human values in written text, and can yield results similar to that which human coders would produce. These were the first experiments to demonstrate that text classification could be used in such a manner. Moreover, these experiments demonstrate that strong results can be obtained using only word-level content features, and that relatively modest quantities of training data may suffice. These results have important practical implications for the scalability of this type of content analysis in social science research. In other words, in this study, classifiers are trained using training data consisting of sentences assigned to categories by a human coder;

subsequently, the trained classifiers that code the remaining sentences were employed. These experimental results indicate that the approach can be used to conduct content analysis for a large number of document sets.

Finally, experiments focusing on the quality–quantity tradeoff for the training data indicate that an adaptive approach to managing data quality can facilitate in optimizing the cost–benefit ratio. In particular, the experiment results indicate that early in the training process, the diversity of the content being coded trumps the quality of those coding, suggesting that when multiple coders are available, the traditional practice of adjudicating conflicting coding of the same documents may not be as cost effective as an alternative in which each coder labels different documents. More generally, the results indicate that larger quantities of lower quality training data are useful early in the training process.

1.2 Structure of the Dissertation

This section describes the structure of the remainder of this dissertation.

Chapter 2, which is based on a journal article published by the Japanese Society of Library and Information Science, presents the design, results, analysis, and experiments for academic paper genre classification [8]. This article received the best paper award for FY 2014 in that journal.

Chapter 3 presents the design, results, and analysis of two sets of experiments for using text classification to infer the direct or indirect expression of human values in text. This chapter draws together two related publications, the first at the Annual Conference of the American Society for Information Science and Technology (ASIST) [9] and the second at International Conference on E-Service and Knowledge Management (ESKM) [10]. The ASIST paper was the first to apply text classification to human values, and the ESKM paper built on those early results to explore learning rates and to characterize classifier performance with limited quantities of training data.

Chapter 4 introduces a more nuanced three-stage coding process that combines topical and non-topical classification for inferring the expression of human values with respect to a specific topic. The topical classification experiments shed light on the quality-quantity trade-off by explicitly modeling the cost of achieving higher coding quality. This chapter is based on a paper that has been accepted for publication at the Fourteenth Annual Conference of the iSchools in April, 2019 [11].

Finally, Chapter 5 concludes the dissertation with some remarks on several directions for future research.

References

[1] Lewis, D. D. Reuters-21578 Text Categorization Collection Data Set. http://www.daviddlewis.com/resources/testcollections/reuters21578/ (accessed 2019-01-26)

[2] 福本文代, 鈴木良弥. (2002). 「WordNetの同義語クラストその上位関係を利用した文書の自動分類」『情報処理学会論文誌』, 43(6), 1852-1865.

[3] Lewis, D. D., Yang, Y., Rose, T. G., & Li, F. (2004). RCV1: A New Benchmark Collection for Text Categorization Research. *Journal of Machine Learning Research, 5*, 361-397.

[4] Pang, B. & Lee, L. (2008). Opinion mining and sentiment analysis. *Journal of foundations and trends in information retrieval, 2(1-2)*, 1-135. DOI:10.1561/1500000011

[5] Friedman, B., Kahn, P. H. Jr., & Borning, A. (2006). Value sensitive design and information systems. *Human-computer interaction and management information systems: Foundations*, 348-372. DOI: 10.1002/9780470281819.ch4

[6] Templeton, T. C., & Fleischmann, K.R., (2011). The relationship between human values and attitudes toward the Park51 and nuclear power controversies. *Proceedings of the 74th Annual Meeting of the American Society for Information Science and Technology (ASIST2011)*, 10p. DOI: 10.1002/ meet.2011.14504801172

[7] Schwartz, S. H. (2007). Value orientations: Measurement, antecedents and consequences across nations. *Measuring Attitudes Cross-nationally: Lessons from the European Social Survey*, 169-203. DOI: 10.4135/9781849209458. n9.

[8] 石田栄美, 安形輝, 宮田洋輔, 池内淳, 上田修一 (2014). 「構造と構成要素に基づく学術論文の自動判定」『日本図書館情報学会誌.』, 60(1), 18-34.

[9] Ishita, E., Oard, D. W., Fleischmann, K. R., Cheng, A.-S., & Templeton, T. C. (2010). Investigating multi-label classification for human values. *Proceedings of the 73rd Annual Meeting of the American Society for Information Science and Technology(ASIST2010), 47(1)*, 1-4.

[10] Ishita, E., Oard, D. W., Fleischmann, K. R., Tomiura, Y., Takayama, Y., & Cheng, A.-S. (2015). Learning curves for automating content analysis: How much human annotation is needed? *Proceedings - 2015 IIAI 4th International Conference on Advanced Applied Informatics (IIAI-AAI 2015)*, 171-176. DOI:10.1109/IIAI-AAI.2015.295, © [2015] IEEE. Reprinted, with permission, from [Ishita, E., Oard, D. W., Fleischmann, K. R., Tomiura, Y., Takayama, Y., & Cheng, A.-S, Learning curves for automating content analysis: How much human annotation is needed?, *Proceedings - 2015 IIAI 4th International Conference on Advanced Applied Informatics*, July/2015]

[11] Ishita, E., Fukuda, S., Oga, T., Oard, D. W., Fleischmann, K. R., Tomiura, Y., & Cheng, A.-S. (2019). Toward three-stage automation of annotation for human values. *Proceedings of the iConference 2019* (to appear).

Chapter 2

Automatic Detection of Academic Papers based on Their Structure and Elements

In this chapter, we attempted to select features for detecting whether or not texts are academic papers. Generally, when features are selected for classifiers, the methods of selection use linguistic features such as the parts of speech of words occurring in the texts. By contrast, the approach in this research is that the structures and elements of academic papers, and the specific words and phrases occurring in academic papers, were used as features.

In general, a document has a specific form and structure depending on its purpose and role. Its elements and order of description are prescribed by social customs and laws and regulations. For example, legal documents, the text of judicial judgments, office documents or business documents each have a specific format. In many cases, a form of document is established by certain agreements among people who create and use those documents, then the format is discussed, and this becomes the standard among those people. Finally, the form of the document becomes established over time. Conversely, if a document has certain elements and a specific form, it is becoming accepted that it is a particular type of document. Currently, many documents are digitized. Therefore, the analysis of the format and structures of documents and the detection of specific types of document from a high volume of digitized document files have been addressed as a research topic [1].

In this chapter, the focus is on the academic paper as a type of document form. Academic research outcomes in any field are made public through certain procedures. It is thought that academic papers have come to have a specific form with the specialization of academic fields and the dissemination of academic journals. In recent years, open access journals and the self-archiving of academic papers have become more common and widespread, and there are many academic papers openly accessible on the web. Academic papers on the web are written in multiple languages and across various academic fields. However, English is still a common international language in the academic world. Therefore, the purpose of this research is to detect academic papers written in English. In

addition, there is a focus on Japanese academic papers, because building the method needs to be independent of language.

The structure of this chapter is as follows. In Section 2.1, the format and elements of academic papers were extracted from guidelines and textbooks concerning the writing of academic papers. In Section 2.2, the format and elements of papers on the web were inspected and the specific structure and elements of papers actually used were confirmed. In Section 2.3, we selected features based on the structure and elements of academic papers confirmed in Section 2.2, and then we conducted an experiment using selected features to detect academic papers from the English and Japanese portable document format (PDF) collections collected by web crawling.

2.1 Elements and Structure of Academic Papers

The format and elements of academic papers are defined in the submission rules (author or submission guidelines) of each academic journal, and authors submit papers according to these rules. The rules differ depending on the academic journal, but there are common rules among them. For example, individual journals determine whether the title of a figure is "FIG. 1" or "Figure 1," but it is common rules that put the title of a figure below the figure and the title of the table above the table.

2.1.1 Establishment of Academic Papers

John Ziman pronounced that the most important medium of scientific communication was the "primary paper" in academic journals. This was a completely new concept in the late 17th century [2]. Prior to this, the means of communicating scientific information was through a "letter" and/or books. Scientific (academic) papers were extremely short having a few pages and did not assemble new principles, but they had taken a more drastic step from the scaffolds established in previous studies by cited other papers [2]. They were also published within a short period and worked for promotion of debates and at the same time achieving priority for the findings. Moreover, scientific papers became public documents and were organized for retrieval and quotation [2].

With the launch of academic journals such as Philosophical Transactions in 1665, the "letter" format had been mainly used for quite some time, and the present "paper" format was not used. The proportion of "letters" in articles published in Philosophical Transactions from 1740 to 1859 was examined [3]. The author reported that the proportion of "letters" was around 50% in the 1760s, and then the proportion of “letters” gradually decreased over the following century, until "letters" had almost disappeared by the 1850s [3].

In the latter half of the 19th century, the number of academic journals and academic papers increased, and two million scientific papers were published during the 19th century [4]. Joseph Harmon declared that the 19th century brought about significant advances in experimental design, innovative statistical methods for analyzing experimental results, and new theories explaining experimental results and observations

[5]. These were expected to be exhibited in articles. The new form of technical paper was required by professional scientists when editing journals [5].

As a result, the paper form changed in the 20th century to the form consisting of topic, abstract, introduction, method or experimentation details, result, discussion, conclusion, acknowledgment, and reference, which combined was called the "topical structure." According to Harmon, this was an efficient way to organize the necessary information logically and efficiently when reporting on experiments [5]. Sam F. Trelease encouraged scientists employing this structure in the book "Preparation of Scientific and Technical Papers" (1927), the first book written in English on the subject of academic paper writing [6].

In Japan, three books about how to write academic papers were published from 1929 to 1934. In the oldest book "How to write medical papers second part No.1 (『医学論文の書き方　後篇　第１』)" (1929) written by Inokichi Kubo [7], he established the following as structures of academic papers: "1. Title and table of contents(表題及目次), 2. Introduction (緒言), 3. History(歴史), 4. Materials and Methods (材料及方法), 5. Result and Scores (結果成績), 6. Summary and Conclusion (統括結論), 7. Acknowledgments (謝辞), 8. Reference(文献), and 9. Abstract (抄録)." Structure "3. History" is "the historical description about the same problem as conducted by analyzing". It is about the description of related and prior research. We could see that those elements of the paper were similar to the modern style of academic paper. The elements of the paper given by Kubo were very similar to the "topical structure" defined by Harmon. However, the influence relationship between them is unclear.

2.1.2 Diffusion of the IMRAD Format

Conversely, there was another explanation concerning the structure of academic papers that was developed based on the IMRAD format consisting of introduction, method, result, and discussion. In 1989, Robert A. Day revealed that a paper written by Louis Pasteur was in the IMRAD format [8]. Day and Barbara Gastel described the history of scientific papers in the guide detailing how to write a paper, and it differed from Harmon's explanation [9]. Papers of academic journals originating in the middle of the 17th century were "descriptive" [9]. Typically, a scientist would report that "First, I saw this, and then I saw that" or "First, I did this, and then I did that." The observations were displayed in a simple chronological sequence. In the 19th century, science had developed and the method became more important in papers [9]. For example, Louis Pasteur, who developed pure-culture methods of studying microorganisms, described his experimental method in exquisite detail. After that, "the reproducibility of experiments became a fundamental tenet of the philosophy sciences, and a segregated methods section led the way toward the highly structured IMRAD format" [9].

Day also indicated that the logic of IMRAD could be defined in question form [9]:

> What question (problem) was studied? The answer is the Introduction. How was the problem studied? The answer is the Methods. What were the findings? The answer is the Results. What do these findings mean? The answer is the Discussion.

Furthermore, the IMRAD format helped authors to organize and write the manuscript and it provided common background for editors, referees, and readers [9].

Sollaci et al. examined the IMRAD format of papers published in the British Medical Journal, JAMA, Lancet, and the New England Journal of Medicine, which were core journals in the medical field from 1935 to 1985 [10]. In 1935, there were no papers with the IMRAD format in any journals, but their presence increased rapidly from the 1960s onwards. All papers in The New England Journal of Medicine in 1975 had the IMRAD format, and all papers in the other three journals also had the format in 1985. In further detail, only 20% of papers in the British Medical Journal had the IMRAD format in 1935, even including those partially in that format. The proportion increased up to the 1960s. The transition to the IMRAD format in the medical field was a gradual process, which took a long time.

Currently, the term "topical structure" is not used, and the term IMRAD had been commonly used. "Topical structure" included not only the body of the paper but also elements such as the title, corresponding text, authors, and references, whereas the IMRAD format included only the structure of the body in the paper.

The IMRAD format developed mainly in the medical and biological fields but then it became commonly used in broader fields from the natural sciences to the social sciences. The book "How to write academic papers" was published in 1977 by Umeberto Eco, and it provides guidelines for writing academic papers in the humanities. The Japanese translation of the book was published in 1991 and was then repeatedly published thereafter. However, there was no description of the structure and elements of the papers [11]. In the humanities, awareness of the structure of papers was limited.

In 1977, Akio Sawada understood the structure of papers in the natural sciences field and stated the following [12] while not emphasizing the differences by field.

> “The process, which consists of looking at various data and facts, thinking about questions of inexplicable phenomena, making a hypothesis to answer that question, verifying whether the hypothesis to explain the reality comparing with facts, was common to all academic fields. Academic papers in natural science mostly consisted of "Introduction showing questions", "Method of experiment", "Experimental results, Evaluation, Discussion, and Conclusion.” In the humanities and social sciences, researchers could not conduct experiments that reproduce the

> results, but the procedure to verify the hypothesis by fact was equivalent to the process of verification by natural science experiments. The whole procedure was fundamentally no differences between both fields".

In addition, Sawada stated that there was a "temporal approach" to explaining the occurrence and formation of phenomena systematically, and a "logical approach" to explaining it in a logical and analytical manner, and that 5W of 5W1H was helpful as a clue in discussions within the social science and human science fields [13]. This was similar to the description of the IMRAD format referenced by Day. Consequently, awareness of the structure of the paper rose in the human sciences as well, and proposals for structures were made.

In the natural science field, the structure of academic papers, consisting of introduction, method, result, and conclusion had been established for decades. However, it was only recently named as the IMRAD. The consensus that papers in natural science fields should have the IMRAD format, has become gradually recognized by most researchers in the social sciences and humanities in Japan, although they had no knowledge of the acronym "IMRAD."

In addition, it had been repeatedly referenced in the guidelines and books about writing academic papers, that title, author name, affiliation, abstract, keywords, acknowledgments, and references to papers require inclusion as elements of papers. For example, in "Presentation of Scientific Papers (SIST08)" ("Preparation and Components of Scholarly Papers" in 2010 edition [14]) of Standard for Information of Science and Technology established in 1986, the elements of academic papers are shown.

The development of electronic journals has caused potential changes that need to be identified. For example, a DOI (Digital Object Identifier System), the identifier of each paper, has been frequently shown on papers. It could be used for automatically detecting academic papers. The specific expression and words used in papers could be important identifiers, although language may affect their usefulness.

As described above, academic papers had specific structures and elements. Many guidelines and textbooks recommended using the format. In practice, if academic papers use common structure and elements, these could be important identifiers for automatically detecting academic papers.

2.2 Survey on Structure and Elements of Academic Papers

We investigated the proportion of academic papers with the IMRAD format and the elements constituting them. The intent of this chapter is to delineate the automatic

detection of academic papers from PDFs on the web, and therefore in this section we inspected the structure and elements of academic papers on the web.

2.2.1 Related Work on the Structure and Elements of Papers

There are several studies on the subject of trying to determine the structure and elements of academic papers. In the field of medicine, Sollaci et al. examined 1,297 original papers published in the British Medical Journal, JAMA, Lancet, and the New England Journal of Medicine from 1935 to 1985 [10]. They found that all papers have the IMRAD format in 1985 and the IMRAD format had become widelyspread within this field. The number of papers in the IMRAD format grew by a factor of 4 in the 20 years from 1955 to 1975. In the field of physics, Charles Bazerman investigated various characteristics of academic papers published in the Physical Review from 1893 to 1980 [15]. In the papers in Physical Review, he pointed out that the papers had been structured using post-1950 headings and then more abstract headings, such as "Experiment," had been used, not using a proper noun as in past years. In the field of experimental psychology, Keiko Kurata and Takayuki Sakagami examined all papers concerning certain experiments published in 1975 and 1998 in the Journal of Experimental Analysis of Behavior, the Journal of Experimental Psychology: Animal Behavior Processes, the Journal of Experimental Psychology: Human Learning and Memory, and Memory and Cognition [16]. They showed that the proportion of both introduction and discussion increased, and discussion and interpretation sections were considered as important as just showing results. Ling Lin et al. surveyed 433 papers in 39 fields of engineering, applied chemistry, social science, and humanities, published in 2007 [17]. Many of the empirical papers had the IMRAD format, but the structure "introduction - literature review - method - results and discussion - conclusion (ILM [RD])" was the most frequent pattern among those papers.

As shown above, there were several research papers examined in specific fields. In this research, we conducted the survey for academic papers from a broad range of fields.

2.2.2 Survey of the Structure and Elements of the Papers

(1) Collection of Journals for the Survey

To obtain academic papers from widely distributed and different fields and independent of the academic journal's rating, academic journals published in Japan were randomly selected from the "CiNii Journal Directory including full-text versions (CiNii 本文収録刊行物ディレクトリ)". The directory included English journals, but those were excluded from the selected journals. It was referred to as the Japanese journal set. Foreign journals were randomly selected from the journal list of Web of Science, Journal Citation Reports, and the Arts & Humanities Citation Index. The selected foreign journals included journals in which the body of the articles was written in a non-English language, while title and abstract were expressed in English. However, as all the papers selected from these

journals by the following method were written in English, this was referred to as the English journal set.

We checked manually whether papers with electronic files (full-text version) of each journal in both the journal sets were provided using database and search engines. When PDF formats were available, for the survey we downloaded the second article shown in the table of contents in the first issue of 2010, because they were relatively new publications and had similar publication dates. The reason for the selection of the second article was that first articles were often the preface or the outline. In this survey, we focused only on papers (academic articles) that existed on the web and were available in PDF format, but we also included papers for which a charge made to gather as many academic papers as possible. When papers not taking the form of academic papers such as bibliographies and poetry were found, these journals were excluded from the targeted items. As a result, a total 1,172 files were obtained including 490 files for the Japanese set and 682 files for the English set.

(2) Survey Items

Based on the elements of academic papers described in Section 2.2, presence of these elements "title of papers," "author name," "affiliation," "abstract," "keywords," "reference," "ISSN," and "DOI" were checked in each paper, as well as checking "headings" as a structure. The writing direction, vertical or horizontal, was also checked.

If a paper had "headings," we recorded the characters of those headings. We labeled the structure only when the recorded heading was explicitly represented by the words shown in Table 2-1. For example, if the heading was "discussion," it was considered as the structure "discussion (D)". If the heading was "Effectiveness of the proposed method," it was not considered as "discussion (D)" because it was not explicitly expressed. For this survey, six labels were used "Introduction (I)," "Method (M)," "Result (R)," and "Discussion (D)" comprising the IMRAD format, in addition to the structures "Literature Review (L)" and "Conclusion (C)". If the same labels were assigned sequentially in a paper, they were combined into one. For example, if a paper consisting of five headings of "Introduction," "Materials," "Experimental Method," "Result," and "Discussion" was provided, it was labeled as "IMMRD," but it was modified to the "IMRD" after removing the duplicate "M".

Table 2-1. Examples of words representing academic paper's structure

	English	Japanese
I	Introduction Background	はじめに 序論
M	Material and Method Methodology	試料と方法 実験
R	Results Finding	実験結果 結果
D	Discussion Implication	考察 議論
L	Literature Review Related Research	文献レビュー 先行研究
C	Conclusion Summary	おわりに 結論

2.2.3 Results

(1) Elements

The elements for detecting academic papers such as titles (English: 100%, Japanese: 100%), author name(s) (English: 99.9%, Japanese: 100%), affiliation (English: 97.2%, Japanese: 100%), journal title (English: 98.5%, Japanese: 97.8%) were shown explicitly in almost all academic papers. There was no major difference between the Japanese set and the English set. Other elements are shown in Table 2-2. Regarding abstracts (English: 89.0%, Japanese: 89.2%) and references (English: 98.4% and Japanese: 98.8%), their ratios of presence were lower than titles and author names, but both elements still occurred in papers with relatively high ratios. There was no major difference between the English and Japanese sets. Although the ratio of keyword was lower than that of abstracts and reference, 75.5% of papers in the Japanese set included them, and 63.2% in the English set also did. The writing direction of all papers was horizontal.

Regarding the presence or absence of headings as the index of the structure of academic papers, 0.8% of papers in the Japanese set and 7.3% of the English set had no headings. The ratio of papers without headings in the English set was higher than in the Japanese set. In 50 papers without headings, 44 papers (88.0%) were from the Arts & Humanities Citation Index. The English set included many articles from the humanities such as literature and religious studies. This may have had an effect on the result.

For the ratios of ISSN and DOI, there was a big difference between both sets. ISSN was not included in any of papers in the Japanese set, whereas 40% of papers in the

English set included it. As for the DOI, it was rarely included in papers in the Japanese set. In Japan, it was difficult to assign DOI because the registration agency of DOI in Japan did not exist until the "Japan Link Center" was approved on March 15, 2012.

Table 2-2. Ratio of elements of academic papers.

	Japanese (N = 490)		English (N = 682)		Total (N = 1,172)	
Abstract	437	89.2%	607	89.0%	1,044	89.1%
Keyword	370	75.5%	431	63.2%	801	68.3%
Reference	482	98.4%	678	98.8%	1,156	98.6%
Headings	486	99.2%	632	92.7%	1,118	95.4%
ISSN	0	0%	293	43.0%	293	25.0%
DOI	1	0.2%	508	74.5%	509	43.4%

(2) Structure

In this survey, review articles were excluded. We examined 1,087 papers with headings from 1,118 articles. After the survey, 35 structure patterns were discovered. Table 2-3 shows the top ten patterns. “L” stands for literature review or related research. Three hundred and eighty of all of them (35%) had no explicit structure except for “Introduction.” The patterns “IMRD” and “IMRDC” were second and third place respectively. In the IMRAD format, 181 papers (17%) had “Conclusion” at the end of paper. Moreover, the pattern “MRD” also was found, which follows the IMRAD format, but these papers had no “Introduction.” Although not the complete IMRAD format, 40% of papers in PDF format on the web had the IMRAD format including the structure "Conclusion" or without "Introduction.”

Table 2-3. Top ten structures patterns of academic papers.

Structure	Number of documents	(%)
I	380	35.0%
IMRD	191	17.6%
IMRDC	187	17.2%
No structure	136	12.5%
MRD	45	4.1%
IM	35	3.2%
IMR C	19	1.7%
I L	16	1.5%
MRDC	14	1.3%
IMR	10	0.9%
Others	54	5.0%
Total	1,087	100%

2.2.4 Structure and Elements of Academic Papers

We investigated how extensively the elements and structure identified in Section 2.1 were included in academic papers on the web. The results concluded that titles, author name, affiliation, journal title, abstract, reference, and keywords were found and that they were considered as common elements of academic papers. Regarding the structure, about 40% of papers explicitly had the IMRAD format or a similar format. In Table 2-3, ratio of the IMRAD format and similar ones was 87.5%, excluding the ratio of papers without structures. The results showed that papers published in most English and Japanese journals used common elements and structures. They were also independent of languages.

When detecting academic papers from within a group of documents mixed together with other types of documents, these common elements and structures were considered useful. Even if a paper did not have all elements and structures, it could be considered as an academic paper if it contained part or parts of them. In addition, this survey covered only academic papers published in academic journals, but it could also be applied to papers of bulletins by universities and conference proceedings, as they also used a similar format. In the next section, we selected features to detect academic papers automatically based on common elements and structures of academic papers obtained from this survey.

2.3 Detecting Academic Papers

For automatically detecting academic papers, features were selected based on specific structure and elements of academic papers. Furthermore, to verify the effectiveness of selected features, a large PDF sets were created. We conducted experiments to detect academic papers using the selected features. In addition, the "academic paper" in this section included the following articles; papers published in academic journals, papers in bulletins published by universities, and papers in conference proceedings.

2.3.1 Related Work on Text Genre Classification

For the method of detecting academic papers, the approach was similar to that of identifying the genre and type of texts.

Shlomo Argamon et al. conducted research to identify 12 separate journals in six fields of experimental and historical sciences. They used frequency information of function words such as "the" and "as" and features such as "and," "moreover," and "in general" to express an expansion, a comment, and modality by analyzing academic journal papers as feature, for machine learning methods [18]. Efstathios Stamatatos et al. conducted experiments to classify texts into genres such as Review & outlook (Editorial) and Letters to the editor, based on the frequency information of words appearing in the texts, using a corpus of The Wall Street Journal newspaper [19]. They selected frequently used words, but pointed out that the positioning of punctuation marks was also important information. Pranitha K. Kumari et al. classified web pages (the 7genre corpus) into seven genres such as a blog, an online shop, and a list, using a Random Forest classifier [20]. They selected features to classify genre based on stems obtained from several stemming methods. Ioannis Kanaris et al. conducted the classification experiments using two test collections: 7genre corpus and KI-04 consisting of eight genres such as arts, links, help, and online shops [21]. In this research, the features used were the character n-gram and HTML tags included in the web page, using the support vector machine (SVM) classifier. Chul Su Lim et al. attempted to classify web pages into eleven categories, such as personal/public/commercial homepages, link collections, research reports, and FAQs [22]. They used the following items as features: (1) information on URL such as URL depth, file name, domain name, and words included in the URL such as FAQ, paper, research, (2) HTML tag, (3) number of characters, number of words, average number of words per sentence, number of POS (part-of-speech) tags, total number of words for 9 POSs tags etc., (4) the number of content words and function words, (5) structural information such as the number of declarative sentences and the number of imperative sentences. Michael Shepherd et al. divided texts into three attributes of content, form, and functionality [23]. The content included the number of uses of meta tag and common words, and the form included number of images, definition of CSS, domain and subdomain information of the web pages, the number of words in a page. The functionality included the number of links in the web pages, the proportion of links that were links to locations within the same page,

and the proportion of them that were links to other pages on other sites. They classified web pages into genres such as personal/corporate/organization home pages using these features. Chaman Thapa studied the problem of classifying websites into four non-topical categories: public, private, non-profit, and commercial franchise [24]. They used two types of features; the textual features including terms, part-of-speech bigrams and named entities, and the structural features including the link structure of the site and URL patterns.

To identify genres and types of texts, test collections including texts were classified to some genres or categories. Classifiers were trained to extract features in texts or attribute information of texts from the test collection. There were many research cases to develop classifiers, but there were also many researches focusing on a feature selection to be input to classifiers. In this research, we also focus on feature selection.

For non-topical classification, hundreds to thousands of words and their frequency information in the text were used as features, and features of genres and types were also used sometimes. In this research, we used words representing the characteristics of academic papers as features. As shown in Section 2.2, academic papers had common structures and elements, regardless of field or language. These characteristics did not appear many times in texts, but if some structures and elements were identified, the text could be an academic paper. To detect an academic paper, it proved effective to use these features. Classifiers generally require a large amount of training data because they used the statistical information of words. However, there was the advantage that classifiers did not require a large volume of data to extract features based on structures and elements because classifiers use only the information of presence or absence of those features. Furthermore, HTML tags, information of URL, link structures, and file attribute information were used as features for classifying web pages in addition to words [22, 23, 24]. In this research, because the PDF file was a target, the domain of the URL of the file, the words used in the URL, and the attribute information of the file were used as features.

2.3.2 Feature Selection

In the work of an automatic detection of Japanese academic papers [25], file size, number of pages, page orientation, URL domain, writing style, and keywords were used as features to detect Japanese academic papers. In this paper, we selected new features based on elements and structures of papers identified in Section 2.2. These features belonged to the four basic categories such as file attributes, structure of academic papers, elements of academic papers, and expressions in academic/non-academic papers. In addition, we aimed at selecting features independent of language. Table 2-4 showed the list of selected features.

In terms of file attributes, structure of files, domain of files, and words in the URL were selected as features. In the structure of file category, file size, number of pages, layout, and encrypted/unencrypted were selected because the average size or volume of

those differed between academic papers and non-academic papers (it will be described in a later section). The domain of files was extracted from the domain of the URL.

Conforming to the IMRAD format, introduction, method, result, discussion, and conclusion were selected as features of the basic structure of a paper. As elements, abstract, keywords, references, acknowledgments, tables, figures, and appendix were selected. When these words appeared as headings and had a single space before them, and with a new line break following the words, they were considered as headings. Furthermore, in some cases, those elements had the same meanings, but may have used different expressions. For example, the structure "abstract" might be expressed by words such as "abstract" or "summary" in English papers, and "abstract", "summary", ”抄録", "要約", "要旨", or "概要" in Japanese papers. Therefore, in this study, in the case of an English article, if a paper had "abstract,” "summary," or "synopsis," it was considered that it had structure "abstract."

For expressions in each of academic and non-academic papers, words likely to be used frequently in those types of texts were selected. The selection was based on the expressions used in the previous report [25] and words were suggested through the assessment procedures by five assessors. The selected expressions were shown in Table 2-4. The codes DOI and ISSN were often shown in footers and headers, not in the body of the paper. However, they were selected through the investigation in the previous section because they were unique items for academic papers. "Journal title in margins" was the journal title shown in margins.

These features were input to classifiers for training and then classifiers detected whether or not a paper was an academic paper. For the file size and the number of pages, the numerical values were used as features. For words in the URL, a Boolean variable was set to 1 if there was "paper" or "research" included in the URL. For the orientation, a Boolean variable was set to 1 if it was portrait and 0 if it was landscape. For the basic structure, elements, and expressions, Boolean variables were set to 1 if they were present and 0 if they were absent.

Table 2-4. Selected features.

		English	Japanese
File Attributes	Structure	File size	
		Number of pages	
		Layout (portrait・landscape)	
		With or without encrypted	
	domain	.edu	ac.jp
		.com	.com|co.jp
		.gov	go.jp
	Words in URL	paper|article|research	paper|article|research
Basic Structure	Introduction	introduction|background	はじめに|序論|緒言|目的|背景
	Method	method|methodlogy	手法|方法|分析
	Result	finding|result	(実験|調査)結果
	Discussion	discussion|implication	考察|議論
	Review	literature review |related research	(先行|既往)(文献|研究|調査)
	Conclusion	conclusion|summary	(結論|結語|まとめ|おわりに)
Elements	Abstract	abstract|summary|synopsis	抄録|要約|要旨|概要 |abstract|summary
	Keyword	keyword|key word	keyword|key word|キーワード
	Reference	bibliography|reference	((引用|参考|参照)文献| reference|参考資料|注|註)
	Acknowledgment	acknowledgment	謝辞|acknowledgment
	Figure and table	(figure| fig.), table	図|表
	Appendix	appendix	appendix |付録|附録
	Affiliation	university|institute	大学|研究所|†|‡|University
	Journal titles in margins	journal|bulletin	紀要|学会誌|論文誌
Expressions for academic	Research	research	研究
	Paper	paper|article	研究報告｜論文
	Code	doi |issn	doi:|doi |issn
	In this paper	this (article|paper|research)	本(論文|研究)
	Subject	subject	被験者
	Research method	survey|experiment |analysis|theory|hypothesis	実験|調査|アンケート|理論 |仮説
	Reviewer	referee ｜ reviewer	査読者
Expression for non-aca	Press release	press release	プレスリリース|報道資料

2.3.3 Building PDF Files Test Collection

In the experiment, each set of 20,000 PDF files sets, of English papers and of Japanese papers was constructed using the following procedure: (1) collection of the URLs of the PDFs, (2) downloading of the PDF files based on the URLs, and (3) manual identification of academic papers.

(1) Collection of URLs of PDFs

To construct the test collection with no bias toward specific fields, the search engine API was used to collect the URLs of the PDFs. As the API, Yahoo! Search BOSS 28 [26] of the search engine Yahoo! of the United States was used. The reason for choosing the Yahoo! API was that (1) the API was publicly available, (2) it allowed searching for the file type "PDF," (3) there was little restriction on the number of searches, (4) there was no limitation on obtaining search results.

As search terms, general nouns were used. To collect English PDFs, 117,797 phrases registered in the noun phrase dictionary (index.noun) in WordNet 3.0 [27] were used. Phrases consisting of multiple words were also used as search terms. URLs were collected from the top 500 rankes in the search results per search. If the number of search results was less than 500, URLs from all search results were collected. Search by API and the collection of URLs were conducted in July 2010. After the removal of duplicates of collected URLs, 22,591,139 different URLs were obtained. In the preliminary survey, the language filter was applied to a search. We mainly obtained PDF files in English, but we also collected a small number of PDF files in other languages. Thus, the language filter was not applied to collect English PDF files.

For collection of Japanese PDF files, 27,384 nouns registered in both Japanese WordNet [28] and IPAdic [29] were used as search terms. In December 2010, URLs were collected using those search terms with "PDF" file type option and the "Japanese" language filter option. Up to 1,000 URLs were collected, ranked by search results. Finally, 6,602,504 URLs were obtained after the removal of duplicates.

Next, 30,000 URLs were selected randomly from both collected URL sets. The English PDF files were downloaded in August 2010, and the Japanese PDF files in January 2011, based on those collected URLs. For the Japanese PDF file set, PDF files with subdomains of URL ".cn" ". tw" ".hk" ".kr" ". sg" were excluded because they were written in Chinese.

Texts were extracted from the downloaded PDF files sets using Apache PDFBox 1.2.1 [30]. 27,848 files in the English PDFs and 27,158 files in the Japanese PDFs were obtained, including files which were extracted only with partial text. Then 20,000 files were randomly selected from the English files and the Japanese files. Hereinafter, they are referred to as the English PDF file set and the Japanese PDF file set.

(2) Manually Identifying Academic Papers

For both English and Japanese PDF sets, five assessors checked individual PDF files to determine whether they were academic papers as defined in this chapter, based on the following criteria.

(a) The file was in the form of an academic paper

(b) Title, author name(s), affiliation(s) were shown

(c) Citations and references were included

(d) One article comprises one file

(e) The file had two pages or more.

To unify criteria among assessors, training sessions were conducted using several samples. Even during the assessment period, they examined files with different assessments and tried to maintain the consistency of the assessments. In addition, a second assessor repeated an assessment on a file assessed as an academic paper by the first assessor. For the English PDF file set, files where abstracts and titles were written in English but the body written in another language were excluded.

(3) Characteristics of the English and Japanese PDF File Sets

Table 2-5 shows the number of academic and non-academic papers, the file size, the average number of pages, and the ratio of portrait oriented papers in both sets. Although 2,011 (10.1%) of files were academic papers in the English PDF file set, 587 (2.9%) of the files in the Japanese PDF file set. When the Japanese PDF file set was put together, different search terms were used and files from the top 1,000 URLs ranked from search results were collected, and this was different from the method of construction of the English PDF file set. However, the results indicated that the proportion of academic papers in the Japanese PDF file set was lower than the English PDF file set.

The portrait orientation was defined as a file whose vertical length was longer when comparing the vertical length and the horizontal length of the first page of the PDF file. The ratios of the portrait layout in academic papers in both the English and Japanese sets were 99.9%, which meant most of academic papers were portrait orientation. Furthermore, the number of characters was greater in the academic papers of both sets, but the average number of pages for was larger for non-academic papers in the English set, but larger for academic papers in the Japanese set.

Table 2-5. Statistics of English and Japanese PDF file sets.

	English			Japanese		
	Academic	Non-academic	Total	Academic	Non-academic	Total
Number of Files	2,011	17,989	20,000	587	19,413	20,000
Average size of characters	50,152	48,220	48,414	37,821	24,666	25,052
Average number of pages	13.1	17.9	17.4	11.1	10.0	10.0
Ratio of portrait	99.9%	93.7%	94.3%	99.5%	92.2%	92.4%

Table 2-6 shows the number of files and the ratio of the top five domains of the URL in the English PDF set. ".edu" domain in the academic papers accounts for 25% of the total. ".org" and ".com" followed in second and third places respectively, and these top three domains accounted for 53.3% of the academic papers. Conversely, ".com" was the most dominant domain for non-academic papers. ".org" and ".edu" were in second and third places respectively. Although these three domains were the dominant domains of the English PDF set, it can be seen that the ".edu" domain was the dominant domain in academic papers, and the ".com" domain was dominant in the non-academic papers. Table 2-7 shows the number of the top five second level domains of the URLs in the Japanese PDF set (".com" was only shown as the top level domain). 54.5% of the domains of the academic papers were ".ac.jp." For the domain of the non-academic papers, the highest percentage was ".com" at 14.3%. However, there was no significant difference with other domains.

There were differences in the attributes of files and distribution of domains among academic papers and non-academic papers. Those characteristics were used for feature selection because they were useful in identifying academic papers.

Table 2-6. Domain distribution of the English PDF file set.

Academic paper			Non-academic paper		
Domain	Number of files	(%)	Domain	Number of files	(%)
.edu	502	25.0%	.com	5,411	30.1%
.org	360	17.9%	.org	3,658	20.3%
.com	209	10.4%	.edu	1,967	10.9%
.uk	101	5.0%	.uk	1,070	6.0%
.gov	37	1.8%	.gov	939	5.2%
Others	802	39.9%	Others	4,936	27.5%
Total	2,011	100%	Total	17,981	100%

Table 2-7. Domain distribution of the Japanese PDF file set.

Academic paper			Non-academic paper		
Domain	Number of files	(%)	Domain	Number of files	(%)
ac.jp	320	54.5%	.com	2,775	14.3%
go.jp	47	8.0%	co.jp	2,434	12.5%
or.jp	40	6.8%	ac.jp	1,913	9.9%
co.jp	28	4.8%	or.jp	1,505	7.8%
.com	12	2.0%	go.jp	1,299	6.7%
Others	140	23.9%	Others	9,487	48.9%
Total	587	100%	Total	19,413	100%

2.3.4 Classifiers and Evaluation Measures

(1) Classifiers

In this research, we conducted the experiments using multiple classifiers and then compared their performance. As classifiers, SVM, AdaBoost, Random Forest, Naive Bayes, and Decision Tree (C 4.5) in Weka 3.5.6 [31] were used.

(a) Support Vector Machine (SVM)

SVM is a type of binary classifier proposed by Vladimir N. Vapnik [32]. It has high generalization capability and is able to handle high dimensional features by using kernel functions. There have been many applications in text classification research using this because they required a large number of features. It has been said that SVM achieves high performance for these cases. The polynomial kernel was used for the experiments.

(b) AdaBoost

The boosting method is a way to improve performance by learning the weighting of multiple weak classifiers whose accuracy is relatively low. AdaBoost was an improvement of the initial boosting method. Using AdaBoost, a classifier, combining weak classifiers based on the presence or absence of individual words, showed higher performance than the classifier using the nearest neighbor method (k-NN method) or the Naïve Bayes method [33]. The decision stumps were used as a weak classifier in this experiment. The number of iterations was 100.

(c) Random Forest

Random Forest is an ensemble learning method of combining multiple weak learners whose performance is not very high and improving that performance by learning weights [34]. In this experiment, the number of decision trees used as weak classifiers was 100, and the number of features selected randomly by each decision tree was five.

(d) Naïve Bayes

Naive Bayes is a classifier based on Bayes probability. In previous studies using this classifier, it showed a low precision but a high recall. This is different performance when compared with other classifiers.

(e) Decision Tree (C4.5) [35]

The decision tree is a highly readable classifier and is often used as a weak learner of group learning such as AdaBoost. We used C4.5 (module name is J48) implemented in Weka for the experiments.

(2) Evaluation Measure

As evaluation measures, precision (P), recall (R), and F measure (F) were used. F measure is the harmonic measure of precision and recall. F score can change the weight of them depending on the parameter α. Here, the F_1 score with the most common value of α = 0.5 was used, meaning that precision and recall were the same weight. In this experiment, the scores of each measure were computed with ten-fold cross-validation and macro-averaging.

2.3.5 Results

(1) Results with the English and Japanese PDF files sets

We verified selected features by experiments using the English PDF file set and the Japanese PDF file set. We conducted experiments to detect academic papers among PDF files using the five classifiers described above. Ten cross-validation sets were used for the experiments. The results in each set are shown in Table 2-8. In the English PDF file set, the highest precision was 0.79 by the Random Forest classifier and the highest recall was 0.84 by the Naive Bayes classifier. In terms of F_1 scores, Random Forest achieved the best performance of 0.74. For the Japanese PDF file set, all scores were lower in each classifier than that in the English PDF file set. The highest precision was 0.72 by the SVM classifier, the best recall was 0.78 by the Naive Bayes classifier, and the best F_1 score was 0.53 by Random Forest.

In similar work to detect Japanese academic papers [25], the highest performance was the voting classifier (Vote), and the F_1 score was 0.49 (accuracy was 0.44 and recall was 0.54). Compared with these figures, performance was improved across all evaluation measures. With regard to the English PDF file set, it was possible to detect academic papers with a probability exceeding 70%.

Table 2-8. Precision (P), recall (R) and F_1 scores (F_1) with the English and Japanese PDF file sets.

	English PDF file set			Japanese PDF file set		
Classifiers	P	R	F_1	P	R	F_1
AdaBoost	0.72	0.53	0.61	0.67	0.35	0.46
Decision Tree	0.76	0.65	0.70	0.65	0.38	0.48
Naïve Bayes	0.41	0.84	0.55	0.30	0.78	0.43
Random Forest	0.79	0.70	0.74	0.68	0.44	0.53
SVM	0.73	0.54	0.62	0.72	0.10	0.18

(2) Results using the Corrected Japanese PDF file Set

The performance measures using the Japanese PDF file set were lower than that using the English PDF file set for all classifiers. There may have been multiple reasons for this and the causes may need to be investigated. The proportion of academic papers in the English PDF file set was 10.1% but only 2.9% in the Japanese PDF file set. In the previous experiments, we did not adjust the ratio of academic papers in each set to replicate the real situation on the web. For these experiments, however, we used the corrected Japanese PDF file set, which had same ratio of academic papers as in the English PDF

file set. The corrected Japanese PDF file set consisted of 10% academic papers and 90% non-academic papers. Non-academic papers were randomly selected from the Japanese PDF file set.

Table 2-9 shows the results using the corrected Japanese PDF file set. The F_1 score by Random Forest classifier was 0.71, which was a greatly improvement over the original Japanese PDF file set. Comparing the results of Table 2-9 with those of Table 2-8, the scores of all measures improved in all classifiers. This result indicates that the ratio of academic papers in the Japanese PDF file set did affect the performance of each classifier. In addition, because similar results were obtained in the English file set and the corrected Japanese file set, it also indicated that this feature selection was language independent.

Table 2-9. Precision (P), recall (R) and F_1 scores (F_1) with the Japanese corrected PDF file set.

Classifiers	P	R	F_1
AdaBoost	0.80	0.60	0.69
Decision Tree	0.73	0.64	0.69
Naïve Bayes	0.60	0.81	0.69
Random Forest	0.77	0.65	0.71
SVM	0.79	0.60	0.69

2.3.6 Discussion

We conducted experiments on detecting academic papers among the PDF file sets collected off the web, using selected features based on the structure and elements of academic papers. Using the English PDF file set, the Random Forest achieved the highest performance with over 70% of precision and recall. The experiments using the corrected Japanese PDF file set, which had similar ratio of academic papers as in the English set, showed similar performance as for the English set. This result showed that the selected features in this study were independent of language.

The purpose of the research was to select features and we focused on efficiently selecting features. However, to improve performance, it was also important to analyze failures not caused by features. For example, there were cases where text extraction from a PDF file failed. There were cases where single byte alphabetic or numeric characters were extracted, but extraction of Japanese characters failed, and if the file was written vertically, each single character was extracted as a new line. These text extraction failures were largely dependent on the nature of the PDF files and the software used to extract the text.

2.4 Summary

In this chapter, we discussed features for automatically detecting academic papers. Firstly, we extracted the structure and elements of academic papers from literatures

describing the form of academic papers, and the guidelines and textbooks about writing papers. The result of our analysis showed that the IMRAD format was the most common format as the structure, and titles, author name, affiliation, abstracts, and references were identified to be elements of academic papers. Secondly, we inspected the structures and elements of academic papers present on the web and found that many papers written in English or Japanese used the structures and elements we extracted. Thirdly, we selected features based on the extracted structure and elements to automatically detect academic papers. In the selection process, we selected features independent of field and language. We conducted experiments using those selected features to detect academic papers from 20,000 PDF files collections (English and Japanese sets) collected by web crawling. The results from the Random Forest classifier showed an F_1 score of 0.74 obtained from the English PDF set and 0.53 from the Japanese PDF set.

Based on the above results, we showed the potential for automatically detecting academic papers using a small number of features, provided we can find structures and elements representing the form and characteristics of academic papers. In many cases, features for machine learning classifiers are selected using linguistic or notational characteristics, or the statistical information of texts. However, our research approach is different. In other words, the approach of capturing characteristics such as the form and structures of academic papers and then selecting features based on these characteristics, is also an effective approach. These features could be applied not only to English and Japanese but also to other languages. Furthermore, the approach of this research may be applied to detect other types of texts.

References

[1] Argamon, S., Whitelaw, C., Chase, P., Hota, S. R., Garg, N., & Levi-tan, S. (2007). Stylistic text classification using functional lexical features. *Journal of the American Society of Information Science, 58(6)*, 802-822.

[2] Ziman, J. M. (1981). 『社会における科学（上）』 松井巻之助訳, 草思社, 206p.

[3] 飯田崇文 (2000). 「「学術論文の社会学」試論--「書簡」から「論文」へ:Philosophical Transactions(1740～1859)」『早稲田大学大学院文学研究科紀要第 1 分冊』46, 59-65.

[4] Vickery, Brian. C. (2002). 『歴史のなかの科学コミュニケーション』村主朋英訳, 勁草書房, 268p.

[5] Harmon, J. E. (1989). The structure of scientific and engineering papers: A historical perspective. *IEEE Transactions on Professional Communication,* 32(3), 132-138.

[6] Trelease, S. F. & Yule, E. S. (1927). *Preparation of Scientific and Technical Papers*. Williams & Wilkins, 117p.

[7] 久保猪之吉 (1929). 『医学論文の書き方　後篇　第１』久保猪之吉, 1929, [69]p.
[8] Day, R. A. (1989). The origins of the scientific paper: The IMRAD format. *American Medical Writers Association Journal, 4(2),* 16-18.
[9] Day, R. A. & Gastel, B. (2010). 『世界に通じる科学英語論文の書き方』美宅成樹訳, 丸善, 321p.
[10] Sollaci, L. B. & Pereira, M. G. (2004). The introduction, methods, results, and discussion (IMRAD) structure: A fifty-year survey. *Journal of Medical Library Association, 92(3)*, 364-367.
[11] Eco, U. (1991).『論文作法 調査・研究・執筆の技術と手順』谷口勇訳, 而立書房, 274p.
[12] 澤田昭夫 (1977). 『論文の書き方』講談社, 259p.
[13] 澤田昭夫 (1986). 『論文のレトリック』講談社, 330p.
[14] 科学技術情報流通技術基準(SIST)SIST08：2010. 「学術論文の執筆と構成」http://sist-jst.jp/pdf/SIST08_2010.pdf, (参照 2012-12-25).
[15] Bazerman, C. (1984). Modern evolution of the experimental report in physics: Spectroscopic articles in Physical Review, 1893-1980. *Social Studies of Science, 14(2), 163-196.*
[16] 倉田敬子 (2007). 『学術情報流通とオープン・アクセス』 勁草書房, 196p.
[17] Lin, L. & Evans, S. (2012). Structural patterns in empirical research articles: A cross-disciplinary study. *English for Specific Purposes. 31(3)*, 150-160.
[18] Argamon, S., Dodick, J. & Chase, P. (2008). Language use reflects scientific methodology: A corpus-based study of peer-reviewed journal articles. *Scientometrics, 75(2)*, 203-238.
[19] Stamatatos, E., Fakotakis, N. D., & Kokkinakis, G. K. (2000). Text genre detection using common word frequencies. *Proceedings of the 18th conference on Computational linguistics (COLING '00), 2,* 808-814.
[20] Kumari, P. K. & Reddy, A. V. (2012). Performance improvement of web page genre classification. *International Journal of Computer Applications, 53(10)*, 24-27.
[21] Kanaris, I. & Stamatatos, E. (2009). Learning to recognize webpage genres. Information Processing and Management, 45(5), 499‐512.
[22] Lim, C. S., Leeb, K. J. & Kima, G. C. (2005). Multiple sets of features for automatic genre classification of web documents. *Information Processing & Management, 41(5)*, 1263‐1276.
[23] Shepherd, M., Watters, C. & Kennedy, A. (2004). Cybergenre: automatic identification of home pages on the web. *Journal of Web Engineering, 3(3)*, 236-251.
[24] Thapa, C., Zaiane, O., Rafiei, D., and Sharma, A. M. (2012). Classifying websites into non-topical categories. *Proceedings of the 14th international conference on Data Warehousing and Knowledge Discovery(DaWaK'12)*, 364-377.
[25] 安形輝, 池内淳, 石田栄美, 野末道子, 久野高志, 上田修一 (2006). 「日本語学術論文 PDF ファイルの自動判定」『Library and Information Science』, 56, 43-63.
[26] Yahoo! Developer Network. Yahoo! BOSS Search Services (Build your Own Search Service), http://developer.yahoo.com/boss /search/ (参照 2012-12-28).

[27] Princeton University. WordNet; A lexical database for English. http://wordnet. princeton.edu/ (参照 2012-12-28).
[28] 独立行政法人情報通信研究機構. 「日本語 WordNet」, http://nlpwww.nict.go.jp/ wn-ja/ (参照 2012-12-28)
[29] Sourceforge.jp. IPAdic legacy, http:// sourceforge.jp/projects/ipadic/(参照 2012- 12-28).
[30] The Apache Software Foundation. Apache PDFBox - Java PDF Library. http:// pdfbox.apache.org/ (accessed2012-12-28).
[31] The University of Waikato. Weka; Waikato Environment for Knowledge Analysis. http://www.cs.waikato.ac.nz/ml/weka/ (参照 2012-12-28).
[32] Vladimir N. V. (2000). *The nature of statistical learning theory*, 2nd ed. New York, Springer, xix, 314p.
[33] Schapire, R. E. & Singer, Y. (2000). BoosTexter: A boosting-based system for text categorization. *Machine Learning, 39(2/3)*, 135-168.
[34] Breiman, L. (2001). Random forests. Machine Learning, 45(1), 5-23. DOI: 10.1023/A:1010933404324
[35] Quinlan, J. R. (1993). *C4.5: Programs for Machine Learning*. San Francisco, CA, Morgan Kaufmann Publishers, 302p.

Chapter 3

The Applicability of Classifiers to Content Analysis and Quantity of Training Data

In this chapter, coding content for human value categories is the focus. Content analysis is a widely used method among social scientists. The typical social science research process consists of the following steps; (1) theorizing, including identifying research questions and collecting a corpus, (2) creating a typology of the phenomena to be studied and coding guidelines for training additional coders, (3) a pilot study to refine both the typology and the coding guidelines, (4) coding the entire corpus, and (5) quantitative analysis using appropriate statistical techniques. Human effort is required for all steps, although it may in many cases be augmented by software, such as the use of qualitative data analysis software for steps (3) and (4) and statistical software packages for step (5). The process is often iterative in the early stages, with the coding frame evolving as new phenomena are encountered. The process typically ultimately converges, so after some point the human effort is principally devoted to examining content and assigning codes from an existing coding frame. It is this later phase in step (4), the assignment of existing codes to existing content, following patterns that have already been established and for which numerous examples exist from early coding, that may in some cases be amenable to automation. In particular, when the amount of texts to be coded is more than the amount of manual coding that can be done, we can consider a method for automatic coding. One useful way is for coders to code a certain amount of data manually, then a classifier is trained using that data, and then the trained classifier automatically codes the remainder of the data. In this chapter, this method is applied to infer human values in sentences using the words in those sentences. Additionally, how much training data is needed for classifiers to obtain similar results by human coders is examined.

In Section 3.2, we build classifiers to infer human values in 2,294 sentences. These sentences were drawn from 28 written prepared testimonies from public hearings on Net neutrality which were manually assigned human value categories. In Section 3.3, we use several classifiers to examine how much training data is required, using more training data and more refined categories compared with that used in Section 3.2.

3.1 Related Work

Although the use of machine learning for text classification has been well studied by computational linguistics and information retrieval researchers, there has to date not been a great deal of uptake of these techniques in the social sciences. Notably, Scott and Smith [1] used Leximancer to automate the coding of newspaper articles based on hand coding of some short segments. Yan, McCracken, and Crowston [2] built a software tool to assist social scientists performing content analysis. Their semi-automatic system leveraged natural language processing and machine learning techniques (specifically a SVM classifier) for initial automatic coding. They used a gold standard corpus that includes 408 email messages, in which sentences may by assigned more than one code. There are a total of 39 codes in their coding scheme. The average recall they achieved over all 39 codes is 0.702. In contrast, the average overall precision is 0.078. Wallace et al. [3] proposed automatically coding transcripts of patient-provider interactions with topic codes via machine learning. They used a Conditional Random Field (CRF) to model utterance topic probabilities. They evaluate their approach using kappa, accuracy, precision, recall, and F-measure. As these studies illustrate, for studies involving automatic content analysis it is common to evaluate classifiers using precision, recall and the F-measure (the harmonic mean of precision and recall). In this paper, we focus principally on the F_1 measure. Evans et al. [4] reported results of an experiment designed to test the strengths and weakness of alternative approaches for classifying the positions and interpreting the content of advocacy briefs submitted to the U.S. Supreme Court. They found that the Wordscores introduced by Laver, Benoit, and Garry [5] and variants of a Naïve Bayes classifier performed well. They evaluated these classifiers based on precision, recall and accuracy as measures. As in computational linguistics, variants of item-level accuracy are the most often reported intrinsic measures of classifier accuracy, but Hopkins and King [6] developed measures based on measured bias in aggregate results for an opinion analysis system that they applied to create aggregate data on opinions expressed in blogs about presidential candidates. Some computational linguistics research has focused on coding sentiment as a form of subjectivity (e.g., Wiebe, Wilson, and Cardie [7]).

3.2 Applicability of Classifiers to Content Analysis

In this section, we attempt to introduce classifiers for coding in content analysis. We use the test collection for this purpose.

3.2.1 Test Collection

We have built a test collection from 28 written statements prepared in advance by witnesses invited to testify before “Net neutrality” hearings held by the U.S. Senate Committee on Commerce, Science, and Transportation on February 7, 2006 [8], and by the Federal Communications Commission (FCC) on April 17, 2008 [9]. Initially, we tried

using the Schwartz [10] Value Inventory (SVI), which forms the basis for a widely used survey instrument, as a coding frame [11]. The SVI's 56 categories proved too fine-grained for our analysis, and using the higher-level categories provided by the SVI's three-level ontology did not substantially improve inter-coder agreement. We therefore developed a specialized coding frame, one informed by the SVI, which contains 10 categories: effectiveness, human welfare, importance, independence, innovation, law and order, nature, personal welfare, power, and wealth. To simplify automatic classification and calculation of inter-coder agreement, we chose to code sentences rather than arbitrary passages. A sentence might reflect one or more values, or it might reflect no values. Human coders view sentences in context, while (for now) our automated classifiers do not.

Table 3-1. Examples of values coding.

Labels	Sentence
effectiveness, independence	Its open nature has enabled those with unique interests and needs to meet and form virtual communities like no tool before it.
human welfare, independence, power, wealth	It has also empowered consumers as citizens and as entrepreneurs.
effectiveness, innovation	Consumers are increasingly creative in the way that they use these new technologies - nowhere more so than here in Silicon Valley.

The principal coder manually coded all 2,294 sentences in the 28 documents. Table 3-1 shows some examples. This yielded a total of 3,403 category coding over 1,783 sentences that were coded with at least one category; the 511 sentences with no assigned categories were removed. The number of codes per sentence in the resulting corpus ranges from 1 to 7, with a median of 2 and a mean of 1.91. The average sentence length is 24.5 words. We selected 4 documents for coding by four additional coders. There are a number of cases in which one coder coded no values for a sentence while the other coder coded one or more values. There are, of course, also cases in which both coders coded different values. These differences in coding may be due to differences in interpretation, or to simple errors. As Artstein and Poesio [12] recommend, we use multiple-π (computed on binary agreement for each category and then macro-averaged over categories) for characterizing the chance-corrected agreement between multiple coders:

$$\pi = \frac{A_0 - A_e}{1 - A_e}$$

Let n_{ik} be the number of times item i is classified in category k. Each category k contributes $\binom{n_{ik}}{2}$ pairs of agreeing judgments for item i; the amount of agreement agr_i for item i is therefore the sum of $\binom{n_{ik}}{2}$ over categories $k \in K$, divided by $\binom{c}{2}$, the total

number of judgment pairs per item. The overall observed agreement is the mean of agr_i for all items $i \in I$

$$A_0 = \frac{1}{I}\sum_{i \in Items} \text{agr}_i = \frac{1}{Ic(c-1)}\sum_{i \in Iitems} \sum_{k \in K} n_{ik}\,(n_{ik} - 1),$$

$$A_e^{\pi} = \sum_{k \in K} \left(\hat{P}(k)\right)^2 = \sum_{k \in K} \left(\frac{1}{Ic} n_k\right)^2 = \frac{1}{(Ic)^2} \sum_{k \in K} {n_k}^2,$$

where i is the number of items (i.e., the number of sentences), c is the number of coders, and n_k is the total assignments for category k (i.e., yes or no). For our five coders and the four documents of 227 sentences, multiple-π is 0.306, which corresponds to "fair" agreement according to Landis and Koch [13], and which is well below what is normally considered satisfactory for training and evaluating automated systems in computational linguistics. As presently formulated, this is a difficult task for humans. We will in Section 3.3, develop more carefully specified coding guidelines. For the remainder of this section we use the principal coder's coding as ground truth.

3.2.2 Multi-Label Classification

Multi-label classification raises two principal challenges: (1) how should multiple labels be used during training? and (2) how many labels should a classifier assign? Tsoumakas and Katakis [14] suggest five methods for selecting training instances when multiple labels are present. We tried two: (Train 1) replicating each sentence that has more than one label and assigning a unique label to each replication, and (Train 2) selecting only the most selective label (in our case, the label with the lowest frequency in the training set) for each training sentence. This distinction affects only training; in both cases, the machine's task is to assign the right set of labels to each sentence in the evaluation set. We had tried two other methods from Tsoumakas and Katakis [14] previously with disappointing results (on the SVI category set): creating an aggregate for each label combination in the training set, or limiting the training set to sentences with a single label.

We used *k*-Nearest-Neighbor (*k*NN) classifiers (with *k*=1, 3, 5, 10, 15, …, 40) from the University of Waikato's Weka toolkit [15]. Preliminary experiments showed stemming to be helpful so we used the Snowball implementation of the Porter stemmer [16]. Terms occurring four or fewer times in the entire collection were removed. We tried two ways of weighting the k examples: equal-weighted (voting) and inverse-distance weighted (w=1/distance). In the tables below, we refer to these as "vote" and "iw." Weka's *k*NN classifiers rank candidate labels. We used two methods for deciding how many labels to select: Oracle and Threshold. For the (unfair) Oracle condition, we assigned the same number of labels as in the evaluation data, thus producing an approximate upper bound on the accuracy of our classifier. If sentence s has i labels in the ground truth, we simply select Weka's i most probable labels. In case of ties, all labels with the same classifier-assigned score are chosen.

For the Threshold condition, we learned a threshold on the score assigned by the *k*NN classifier. To set this threshold, we divided the test collection into three sets; 80% for training the *k*NN classifier (the "training set"), 10% for learning the threshold (the "devtest" set), and the remaining 10% for evaluation (the "evaluation" set). We set the threshold using the following steps; 1) learn a classifier using the training set, 2) automatically assign label probabilities to each devtest sentence, 3) select the threshold to optimize F_1 on devtest, 4) automatically assign label scores to each sentence in the evaluation set, and then 5) select all labels with a score higher than the threshold. We repeat both (Oracle and Threshold) with 10 disjoint evaluation sets, reporting 10-fold cross-validation results.

We are interested in both false negatives and false positives, so we elected to compare the performance of each method by first computing macro-averaged precision (number of correctly assigned categories over number of assigned categories) and recall (number of correctly assigned categories over number of human-coded categories) and then computing the balanced F_1 measure (the harmonic mean of precision and recall). Table 3-2 shows the macro-averaged F_1 for automatic classification by Train 1 and Train 2 on the Oracle condition; Train 1 does slightly better than Train 2. "Threshold" is the macro-averaged F_1 for classification using a score threshold to select the number of categories to be assigned. For these experiments, we swept the threshold between 0.01 and 0.25 in increments of 0.01 separately for each fold and selected the threshold that yielded the best F_1 on the devtest set for that fold. The best macro-averaged F_1 is 0.45 (for k=25, vote). This result is quite close to the best comparable result for the Oracle condition (0.48 for k=25, vote). We therefore conclude that our simple threshold selection method is reasonably effective. The results are relatively insensitive to k, and both weighting schemes yield similar results.

Table 3-2. Classification accuracy using *k*NN (F_1).

	Oracle		Oracle		Threshold	
	Train 1		Train 2		Train 1	
k	vote	iw	vote	iw	vote	iw
15	0.46	0.46	0.43	0.43	0.43	0.44
20	0.48	0.48	0.45	0.44	0.44	0.44
25	**0.48**	0.47	**0.46**	0.45	**0.45**	0.45
30	0.48	0.47	0.45	0.45	0.45	0.45
35	0.47	0.47	0.45	0.45	0.45	0.45

3.2.3 Comparison of Human and System Coding

Although F_1 is a widely reported measure for classifier accuracy, it is rather opaque as an absolute measure; the better use of F_1 is as a basis for comparing alternative classification techniques. Our ultimate goal suggests a natural comparison: what F_1 would another human performing the same task achieve? The results in Table 3-2 suffer from two artificialities that are useful during development, but would make such a comparison less informative: (1) they focus only on sentences to which at least one label was assigned in the ground truth, and (2) the cross-validation was performed without regard to which document a sentence came from (since our system presently takes no advantage of any context beyond the sentence).

In order to establish comparable conditions, we selected the same four witten statements that were coded by the four additional human coders; this yields 227 sentences in the evaluation set, including sentences with no assigned values. As before, we use the principal coder's codes as ground truth, and we compute F_1 for our system and for each other assessor (again, computing F_1 on binary agreement for each category and then macro-averaging over categories). For the classification system, the remaining 24 prepared statements constitute the training data. For training, we use only sentences with at least one coded value, and we use the same threshold learned in the cross-validation experiment. For the results in Table 3-2 we had always selected at least one value, but here we would allow the threshold to exclude all categories.

Table 3-3 shows the macro-averaged F_1 for each coder (numbered 2 … 5) and our best *k*NN threshold classifier from the earlier experiments (k=25, vote). Clearly there is considerable room for improvement, with the best human coders achieving F_1 values more than twice as high as our present system. Our preliminary analysis points to problems resulting from the use of a single threshold. As Table 3-4 shows, our automated system is overly sensitive, as the system assigned at least one label to every sentence.

Table 3-3. Comparing human and system coding (F_1).

Second Coder				25-NN
2	3	4	5	Vote
0.64	0.66	0.70	0.39	0.31

Table 3-4. Number of sentences per category.

	Ground Truth	2	3	4	5	25-NN (Vote)
Effectiveness	24	31	13	27	0	222
Human Welfare	51	36	40	60	17	111
Importance	67	48	54	54	6	196
Independence	49	46	22	63	36	124
Innovation	16	39	17	31	13	38
Law and Order	24	42	15	40	19	82
Power	49	37	28	42	7	135
Wealth	23	65	12	25	1	212
None	58	49	102	59	138	0

3.2.4 Discussion

In this section, we demonstrated that classifiers can be applied to coding in a content analysis widely that is used by social scientists, and we evaluated this using a collection which human values have been manually assigned.

The results indicate that using word as features can be effective. In addition, we obtained some insight into directions for future research. For example, if the presence or absence of each value were independent, a classifier might instead be built for each value.

In the next section, we build binary classifiers for each category. Also, the human value categories are refined, and we use a large test collection with 102 testimonies. With this larger collection we can ask how much training data would actually be needed.

3.3 Quantity of Training Data vs. Human Effort

We examined the role of human values in shaping the Net neutrality debate through a content analysis of testimonies from U. S. Senate, House, and FCC hearings on Net neutrality [8, 9]. A professional coder coded sentences in 102 prepared statements for six human values [17].

We explore the potential for reducing human effort when coding text segments for use in content analysis. The key idea is to do some coding by hand, to use the results of that initial effort as training data, and then to code the remainder of the content automatically. Alternatively, we might ask how many documents must be manually coded to achieve at least 90% of the best F_1 value that we could achieve (with 101 documents). In this section we describe the test collection and introduce our experiment design and evaluation measures.

3.3.1 Test Collection

The collection consists of 102 written prepared statements about Net neutrality from public hearings held by the U.S. Congress [8] and the U.S. Federal Communications Commission (FCC) [9]. The categories used in the content analysis were selected from the Meta-Inventory of Human Values (MIHV) [18]. After four rounds of developmental coding, we selected six MIHV categories; wealth, social order, justice, freedom, innovation, and honor, all of which were fairly common in our collection (with a prevalence above 4%). One of the co-researchers of this research, a social scientist, then coded each of the sentences in the 102 documents with zero or more MIHV categories. In 102 documents, a total 9,890 sentences were manually coded. Table 3-5 shows examples of codes for some sentences.

We use a classifier to assign labels representing human values to each sentence. For our experiments, we identified each sentence using TreeTagger [19]. Sentences including more than 40 words or no non-stopwords were removed. After removing words in the SMART stopword list [20] from those sentences, those sentences were stemmed by the Porter stemmer [16]. No other feature selection techniques were applied, both because full-vocabulary SVM's have been shown to be effective for text classification and because the relatively short sentence lengths already raise some concerns about sparsity. After pre-processing, 8,660 sentences remain in the test collection. Table 3-6 shows the how many of the 8,660 sentences were manually coded with each of the six values. There are 1,545 sentences that were not coded for any value, and all but two of the 102 documents have at least one sentence with no value. A total of 20 documents were coded by a second coder, yielding the kappa values shown in Table 3-6 [20].

Table 3-5. Example sentences.

Categories	Sentence
freedom, social order	Consumers are entitled to access the lawful Internet content of their choice
honor	I am one of the network engineers involved for many years in designing, implementing and standardizing the software protocols that underpin the Internet.
innovation, freedom	Part of the reason why the Internet is such a creative forum for new ideas is that there are very few barriers to using the Internet to deliver products, information and services.
justice	Under these circumstances, requiring those most responsible for congestion to bear a greater percentage of the costs would be both good network management and fair from a consumer standpoint.
social order	The Commission, under Title I of the Communications Act, has the ability to adopt and enforce the net neutrality principles it announced in the Internet Policy Statement.

Table 3-6. Coder agreement (Kappa) and prevalence for each MIHV category.

Categories	*Kappa*	*#sentences*	*#docs*
wealth	0.621	3,156	102
social order	0.688	2,503	102
justice	0.423	2,267	99
freedom	0.628	2,155	101
innovation	0.714	1,018	94
honor	0.437	317	80

3.3.2 Experimental Design

In our earlier work, we have sought to characterize the best possible classification effectiveness for a variety of widely used classifier designs [22]. To do this, we built classifiers for each value, each trained with binary category coding for all sentences in some 101-document subset of the 102 documents. We then tested those classifiers on the sentences from the one remaining document. This process was repeated 102 times, once with each of the 102 documents as the held out test document. This approach, 102-fold

cross-validation, produces one classification result for each document. We tried Support Vector Machine (SVM), Naïve Bayes (NB) and k-Nearest Neighbor classifiers from University of Waikato's Weka toolkit [15], obtaining the best results (F_1=0.7068, averaged over all six categories) from a linear kernel SVM classifier and the second best results (F_1=0.6333) from the NB classifier. We therefore report SVM and NB cross-validation results as upper baselines for the learning curve experiments in this paper.

The purpose of this paper is to determine how much training data is actually needed. Specifically, we examine how many documents are required to obtain classifier performance similar to that which could be obtained using 102-fold cross-validation. The classifiers are trained using different numbers of training documents (one to 51 documents), and the trained classifiers are then used to assign value labels to each sentence in the remaining 51 documents. For each classifier (i.e., for each number of training documents) we compute precision, recall and F_1 to make a learning curve for F_1.

We ran a linear kernel SVM and NB in Weka for some number *i* of training documents, as follows:

(a) The collection was divided into two disjoint groups of 51 documents (Group A and B).
(b) i documents from Group A were randomly selected and used to train a classifier.
(c) That trained classifier then assigned values to each sentence in the 51 documents of Group B.
(d) Groups A and B were swapped (Group A becoming testing) and steps b) and c) were repeated.
(e) The 102 resulting classification decisions were used to compute F_1.

In step a), interval sampling was applied to divide the two groups. The documents were arranged as the order in which they were manually coded. Group A always includes documents numbered 1, 3, 5, ... 101 and Group B includes documents numbered 2, 4, 6, ..., 102. Steps b) through e) were repeated 10 times, and for tabular display the ten F_1 values are averaged. Steps b) through e) were repeated for i = {1, 2, 3, …., 51} training documents.

3.3.3 Evaluation Measure

We compute precision as number of correctly assigned categories divided by the number of assigned categories, recall as the number of correctly assigned categories divided by the number of human-coded categories, and F_1 as the balanced harmonic mean of precision and recall. Table 3-7 shows the contingency table for binary classification.

Table 3-7. Contingency table for evaluation measure.

		Human	
		Positive	Negative
Classifier	Positive	a	b
	Negative	b	d

These evaluation measures are computed as follows:

$$Precision = \frac{a}{a+b}$$

$$Recall = \frac{a}{a+c}$$

$$F_1 = \frac{1}{(\frac{1}{Precision} + \frac{1}{Recall})/2}$$

3.4 Results

Figure 3-1 shows F_1 scores for the SVM classifier with different numbers of training documents for each of the six values. For example, the F_1 scores for wealth produced by an SVM with one randomly selected training document vary between 0.011 and 0.463, depending on which training document was randomly selected. As expected, F_1 scores with 10 randomly selected training documents are more stable, varying between 0.529 and 0.606.

Figure 3-2 and 3-3 show the resulting learning curve for each of the six categories from the SVM and NB classifiers, respectively. Tables 3-8 and 3-9 show the same information for specific numbers of training documents (including the baseline result for 101 training documents), by SVM and NB classifiers. As we have seen in our prior work, honor is a difficult category for automated classifiers [22], perhaps because of the relative sparsity of positive training examples. We therefore focus principally on the other five categories for the remainder of this paper.

As we have previously reported, the SVM classifier archives better F_1 values than the NB classifier for each of the six categories with 102-fold cross-validation. Comparing corresponding values from Tables 3-8 and 3-9, we can now see that the same pattern of dominance by the SVM is evident for each smaller number of training documents at which we computed averages. Interestingly, the same is not true of honor. As Figures 3-2 and 3-3 illustrate, the relative order of the F_1 scores between the six categories become stable early, providing clear evidence for the relative difficulty of the classification tasks. The social order category consistently yields the best results, followed by wealth, then freedom and innovation, then justice, and finally honor. This ordering does not follow the relative frequency of value categories in the collection. It is interesting to note that the F_1 for the

NB classifier rises markedly faster than the F_1 for the SVM classifier, however, for both justice and honor. Although not similar in relative frequency, these two categories do have markedly lower kappa values. We interpret this as perhaps indicating that the NB classifier is able to more easily accommodate inconsistent training data when only a limited amount of training data is available. This observation could be important when applying these techniques in severely cost-constrained or time-constrained settings.

From Table 3-8 we can see that the SVM classifier achieves a mean F_1 (over 10 runs) that is above 90% of the high (102-fold cross-validation) baseline with 50 training documents for all five of the categories other than honor; it achieves that 90% threshold with 30 documents for four of those five categories; and with 20 documents for three of those four categories. From Table 3-9, the NB classifier achieves a mean F_1 that is above the corresponding high (102-fold cross-validation NB) baseline with 20 documents for all five of the categories other than honor. Nonetheless, the NB classifier's high baseline is markedly lower than the high baseline of the SVM classifier, and the SVM classifier is thus the better choice overall.

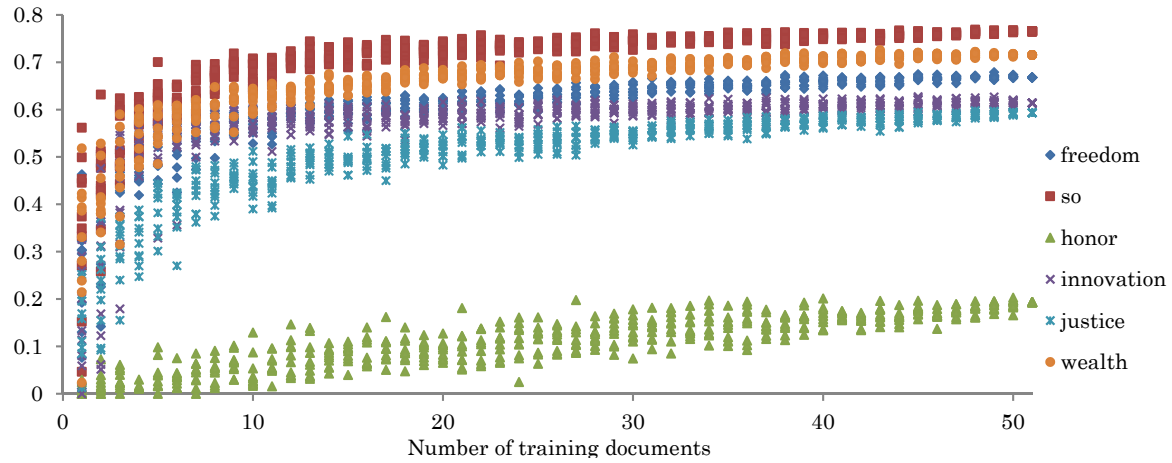


Figure 3-1. F_1 for SVM classifier with different numbers of training documents

Table 3-8. Mean F_1 for different number of training documents, SVM (% is of 101-doc result).

	Number of training documents for SVM				
	10 docs	*20 docs*	*30 docs*	*50 docs*	*101 docs*
wealth	0.620 (84%)	0.673 (91%)	0.689 (93%)	0.716 (96%)	0.742
social order	0.680 (87%)	0.727 (93%)	0.746 (95%)	0.766 (98%)	0.784
justice	0.449 (70%)	0.518 (80%)	0.546 (85%)	0.592 (92%)	0.645
freedom	0.576 (82%)	0.611 (87%)	0.640 (91%)	0.670 (96%)	0.700
innovation	0.572 (89%)	0.589 (91%)	0.600 (93%)	0.615 (95%)	0.645
honor	0.045 (17%)	0.093 (35%)	0.135 (50%)	0.189 (71%)	0.267

Table 3-9. Mean F_1 for different number of training documents, NB (% is of 101-doc result).

	Number of training documents for NB				
	10 docs	*20 docs*	*30 docs*	*50 docs*	*101 docs*
wealth	0.630 (93%)	0.653 (97%)	0.659 (98%)	0.666 (98%)	0.674
social order	0.716 (93%)	0.743 (96%)	0.750 (97%)	0.762 (97%)	0.770
justice	0.485 (90%)	0.512 (95%)	0.519 (96%)	0.523 (96%)	0.541
freedom	0.565 (94%)	0.587 (97%)	0.595 (99%)	0.600 (99%)	0.603
innovation	0.517 (89%)	0.531 (91%)	0.545 (94%)	0.555 (94%)	0.581
honor	0.118 (53%)	0.171 (77%)	0.195 (88%)	0.206 (88%)	0.222

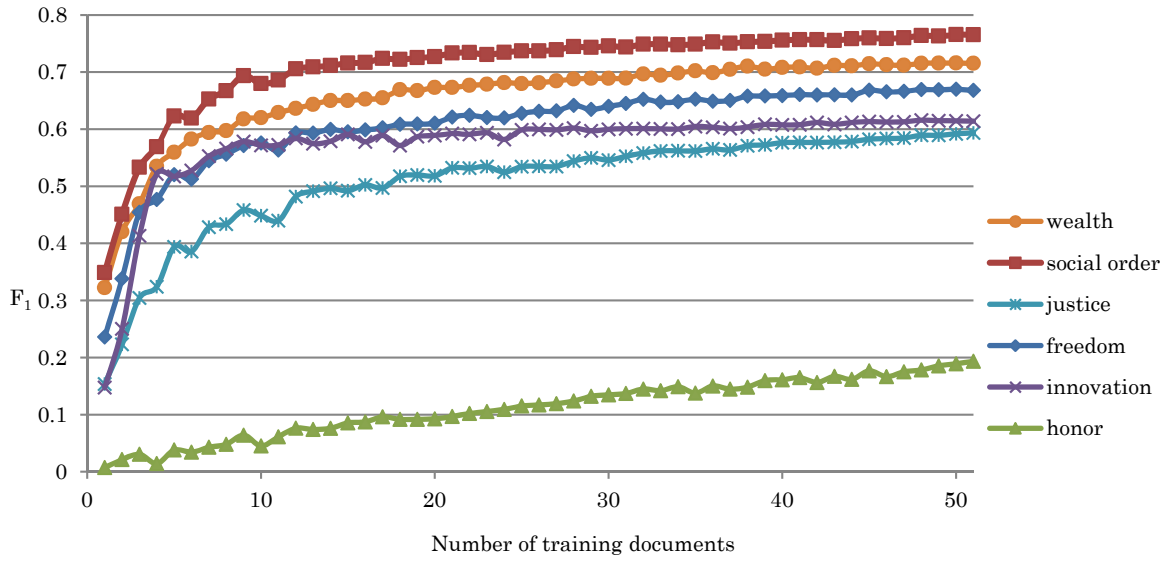


Figure 3-2. Learning curves for SVM classifier

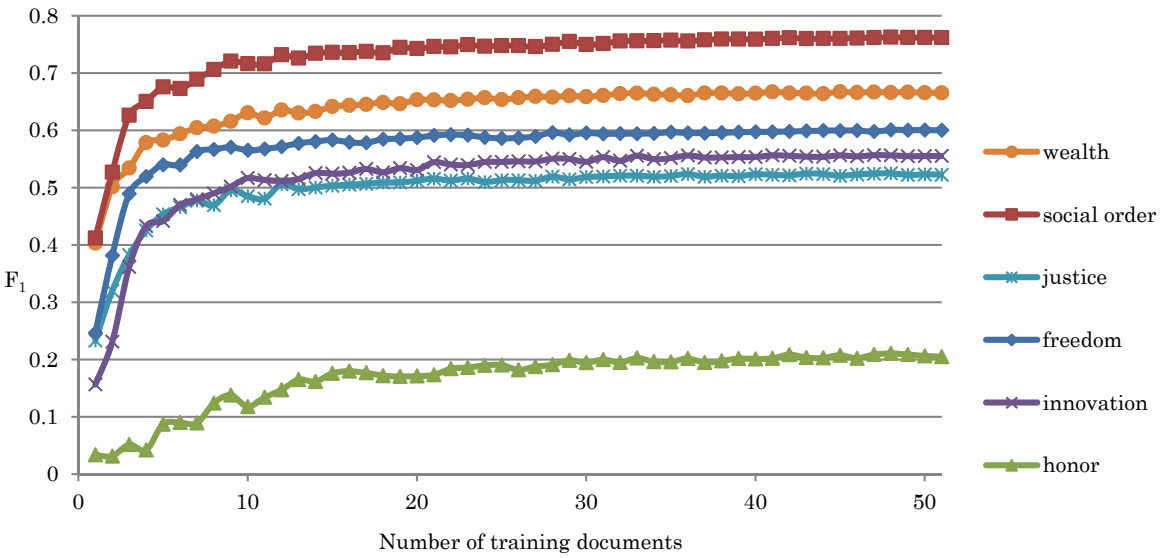


Figure 3-3. Learning curves for NB classifier

For comparison with a human coder, we assume that the primary coder's coding results are correct, and compute F_1 as if the second coder were a classifier using the same approach we reported in [21]. Table 3-10 shows the F_1 computed in this way for a human coder over 20 documents. Although tested on fewer document, than the results in Table 3-9, we note that our classifier results are within about 15% (relative) of the Mean F_1 achieved by human coding for five of the six categories. Only for honor is human coding markedly (173%) better than our best classification results.

Table 3-10. Mean F_1 for Human coder.

	Human coder
wealth	0.797
social order	0.767
justice	0.546
freedom	0.722
innovation	0.741
honor	0.461

3.5 Summary

In this chapter, we proposed a method to introduce classifiers for coding in content analysis to limit of amount of text most be coded manually. A total of 2,005 sentences from

28 prepared testimonies presented before hearings on Net neutrality were manually coded for one or more of ten human values using a coding frame based on experience coding similar content using the Schwartz Values Inventory. *K*-Nearest-Neighbor classifier with several settings were compared. Although there was at the time room for improvement, these early results suggest that scalable approaches that are sufficiently accurate for some social science applications may be achievable using word features.

Next, we examined that how much training data was needed. The test collection for that study included 102 written prepared statements about Net neutrality from public hearings held by the U.S Congress and the U.S. Federal Communications Commission (FCC). Six categories were used in this analysis: wealth, social order, justice, freedom, innovation and honor. A support vector machine (SVM) classifier and a Naïve Bayes (NB) classifier were trained on manually coded sentences from between one and 51 documents and tested on a held out of set of 51 documents. The results show that the inflection point for a standard measure of classifier accuracy (F_1) occurs early, reaching at least 85% of the best achievable result by the SVM classifier with only 30 training documents, and at least 88% of the best achievable result by the NB classifier with only 30 training documents. The results show that machine classification could reasonably be scaled up to larger collections of similar documents without additional human coding effort. In other words, we have shown that human effort can be reduced in content analysis conducted by social scientists.

References

[1] Scott, N., & Smith, A.E. (2005). Use of automated content analysis techniques for event image assessment. *Tourism Recreation Research, 30(2)*, 87-91.

[2] Yan, J. L. S., McCracken, N., & Crowston, K. (2014). Semi-automatic content analysis of qualitative data. *iConference 2014 Proceedings*, 1128-1132. DOI:10.9776/143999

[3] Wallace, B. C., Laws, M. B., Small, K., Wilson, I. B., & Trikalinos, T. A. (2014). Automatically annotating topics in transcripts of patient-provider interactions via machine learning. *Medical Decision Making, 34(4)*, 503-512. DOI: 10.1177/0272989X13514777

[4] Evans, M., McIntosh, W. V., Lin, J., & Cates, C. L. (2006). Recounting the courts? Applying automated content analysis to enhance empirical legal research. *1st Annual Conference on Empirical Legal Studies Paper*, http://dx.doi.org/10.2139/ssrn.914126

[5] Laver, M., Benoit, K., & Garry, J. (2003) Extracting policy positions from political texts using words as data. *The American Political Science Review, 97(2)*, 311-331.

[6] Hopkins, D., & King, G. (2007). Extracting systematic social science meaning from text. available at http://gking.harvard.edu/files/words.pdf

[7] Weibe, J., Wilson, T., & Cardie, C. (2005). Annotating expressions of opinions and emotions in language. *Language Resources and Evaluation, 39(2-3)*, 165-210.

[8] U.S. Senate. Senate Committee on Commerce, Science and Transportation Hearing on Network Neutrality. Feb 7, 2006

[9] FCC. Broadband Network Management Practices Public Hearing. Palo Alto, CA, April 17, 2008

[10] Schwartz, S.H. (1992) Universals in the content and structure of values, *Advances in Experimental Social Psychology*, Acad. Press, 25, 1-66.

[11] Cheng, A.-S., Fleischmann, K.R., Wang, P., Ishita, E., & Oard, D.W. (2010). Values of stakeholders in the net neutrality debate: Applying content analysis to telecommunications policy, *43rd Hawaii International Conference on System Sciences (HICSS),* DOI:10.1109/HICSS.2010.434

[12] Artstein, R., & Poesio, M. (2008) Inter-coder agreement for computational linguistics. *Computational Linguistics, 34(4)*, 555-596.

[13] Landis, J. R., & Koch. G. G. (1977). The measurement of observer agreement on categorical data, *Biometrics, 33*, 159-174.

[14] Tsoumakas, G. & Katakis, I. (2007). Multi label classification: An overview, *International Journal of Data Warehousing and Mining, 3(3)*, 1-13.

[15] Hall, M., Frank, E., Holmes, G., Pfahringer, B., Reutemann, P., & Witten, I. H., (2009). The WEKA data mining software: An update. *SIGKDD Conference on Knowledge Discovery and Data Mining, 12(1), 10-18*.

[16] Poter, M. F. (1980). An algorithm for suffix stripping. *Readings in Information Retrieval (1997)*, Morgan Kaufmann, 313-316.

[17] Cheng, A.-S. (2012). Values in the Net Neutrality debate: applying content analysis to testimonies from public hearings. *University of Maryland Theses and Dissertations / Information Studies Theses and Dissertations*, http://hdl.handle.net/1903/12701

[18] Cheng, A.-S. & Fleischmann, K. R. (2010). Developing a Meta-Inventory of Human Values. *Proceedings of the American Society for Information Science and Technology (ASIST2010), 47(1)*, 1-10.

[19] Schmid, H. (1994). Probabilistic part-of-speech tagging using decision trees. *Proceeding of International Conference on New Methods in Language Processing*,

[20] Salton, G. M. & McGill, M. J. (1980). The SMART and SIRE experimental retrieval systems. *Readings in Information Retrieval (1997)*, Morgan Kaufmann, 381-399.

[21] Takayama, Y., Tomiura, Y., Ishita, E., Oard, D. W., Fleischmann, K. R. & Cheng, A.-S. (2014). A word-scale probabilistic latent variable model for detecting human values. *Proceedings of the 23rd ACM International Conference on Information and Knowledge Management (ACM CIKM 2014)*, 489-1498.

[22] Takayama, Y., Tomiura, Y., Ishita, E., Wang, Z., Oard, D. W., Fleischmann, K. R., & Cheng, A.-S. (2013). Improving automatic sentence-level annotation of human

values using augmented feature vectors. *Proceedings of Conference of the Pacific Association for Computational Linguistics (PACLING 2013)*, in CD-ROM.

Chapter 4

The Quality-Quantity Trade-off for Training Data in Automation of Content Analysis

In Chapter 3, the possibility of introducing classifiers to assign sentences to value categories in content analysis for human values was examined. In this chapter, content analysis for human values is again the focus, but this time with the goal of selecting documents for analysis based on their topic, and coding the presence or absence of human values in sentences. The proposed approach divides the processes into three stages, studying whether the introduction of classifiers is effective in the first two stages.

The approach to automating parts of the content analysis process that we propose in this paper thus consists of three stages; 1) identification of the documents to be labeled, (2) determination of which text spans should be labeled in each document, and (3) coding of human values for those selected text spans. For the second and third stages, we adopt Takayama's approach [1] of using sentences as text spans, which limits the complexity of the text span selection without a serious adverse effect on the utility of the labeling process as a basis for content analysis, as shown by Cheng et al. [2]. As a result, our first stage is binary (on-/off-topic) topic classification at document scale, our second stage is non-topical binary classification (values present/values absent) at sentence scale, and our third stage, as in Chapter 3, is non-topical multi-label classification for each possible human values label, again at sentence scale.

Prior work on automating values classification in Chapter 3 made two implicit assumptions. First, we assumed that the documents to be labeled have already been assembled. While this can be a reasonable assumption for small-scale studies in which content analysis is performed by people, scaling that process up to larger collections requires some level of automation in the determination of which documents should be labeled. Second, we assumed that the process of determining which spans of text should be labeled can be conflated with the process of determining which human values label(s) should be assigned to each span. This conflation is reasonable when most sentences should express or reflect some human value(s), as was the case in the collection in Chapter 3, but there are other settings in which the expression of human values is less frequent.

We also examine the quality-quantity trade-off for training data. Classifiers require significant amounts of coded content for training, data that can be expensive to obtain. In this chapter, we construct a newspaper editorial test collection focused on discussion of nuclear power, and it was expensive to build a test collection manually. We are therefore interested in techniques that make the best use of limited amounts of human coding effort. We consider two ways of creating training data, one in which coders work together to create the best possible codes for each document, and the other in which they work along to code the largest possible number of different documents for a given level of coding effort. We then compare classifiers trained with each approach to coding by plotting learning curves that show how well we can do as the number of coding increases.

In the rest of this chapter, we report findings from automatic on/off topic identification using classifiers, as well as determining whether value invocations would be interpreted by a coder as present or absent within sentences from on-topic articles in Section 4.2. We also show, using a more limited set of experiments, that promising levels of precision and recall can be achieved for the value sentence identification task in Section 4.3. First, however, we begin by introducing the new test collection of newspaper editorials that we have assembled for these experiments.

4.1 Test Collection of Nuclear Power Debate Editorials

Our focus in this work is on automation of content analysis to identify the role of human values in the nuclear power debate in Japan. The Great East Japan Earthquake occurred on March 11, 2011, damaging the Fukushima Daiichi nuclear power plant. After this disaster, the safety of other nuclear power plants received extensive discussion in the Japanese mass media. This event also reignited the global debate over the safety of nuclear power that dates back to the early days of nuclear power, implicating a broad range of human values, and playing out differently in different countries. We have chosen newspaper editorials as our focus for this study, but other work is also looking at the ways in which human values are expressed in newspaper articles. Most newspaper articles adopt a third-person perspective to report on events and the statements of others, but we have chosen to focus here on editorials because we might expect to find a first-person perspective in editorials.

We performed focused collection to obtain a Japanese editorial' corpus. To create the corpus, we searched for editorials (as labeled by metadata) in the commercially available Mainichi Shimbun CD-ROM [3-8] from 2011 to 2016 using the query “原発（abbreviation of nuclear power plant) OR 原子力（nuclear power)”. This query retrieved 750 documents. There are 242 articles from 2011, 166 from 2012, 124 from 2013, 93 from 2014, 62 from 2015, and 63 from 2016. The number of editorials decreased with the passing of years. However, nuclear power has become somewhat of a meme in Japanese society, often invoked tangentially or as a point of comparison, rather than being the focus of the

editorial. It is therefore necessary to separate those documents that we wish to analyze from those that simply happen to use the term. This on-topic/off-topic classification task is our first stage.

To support text classification experiments for this first stage, we performed 10 rounds of coding. In Round 1 we randomly selected 7 editorials from the first year of the corpus. Two coders, both native speakers of Japanese, then independently coded each editorial as on topic or off topic. The two coders then met and created adjudicated on/off topic labels by consensus, formalizing the basis for their decisions as written coding guidelines. As Table 4-2 shows, Coder A, a social science researcher, initially exhibited perfect agreement with the consensus; Coder B, an information science researcher who was new to the task, achieved considerably lower agreement with the consensus adjudicated judgments. As shown in Table 4-1, this process of independently coding, reaching consensus, and updating the coding guidelines was then repeated for a total of 10 rounds of progressively increasing size, in each subsequent case using documents selected from across the time span of the corpus. As can be seen in Table 4-2, Coder B's decisions came to more closely agree with the consensus as iterations proceeded, ultimately achieving noticeably closer agreement with the consensus than Coder A in two of the last three rounds.

Table 4-1. Stratified document samples for the ten-round coding process.

	2011	2012	2013	2014	2015	2016	Total
Round 1	7						7
Round 2		5	3	2	1	2	13
Round 3	8	4	2	2	2	2	20
Round 4	8	4	2	2	2	2	20
Round 5	7	7	5	5	3	3	30
Round 6	7	7	5	5	3	3	30
Round 7	7	7	5	5	3	3	30
Round 8	33	22	17	12	8	8	100
Round 9	33	22	17	10	8	8	98
Round 10	33	22	17	12	8	8	100
Total	143	100	73	55	38	39	448

Table4-2. Cohen's Kappa for each round.

Round	#documents	A vs. B	Adjudicated vs. A	Adjudicated vs. B
1	7	0.588	1.000	0.588
2	13	0.806	0.806	1.000
3	20	0.875	1.000	0.875
4	20	0.494	0.886	0.583
5	30	0.430	0.796	0.605
6	30	0.474	0.735	0.730
7	30	0.437	0.769	0.651
8	100	0.535	0.705	0.816
9	98	0.464	0.730	0.741
10	100	0.477	0.635	0.821

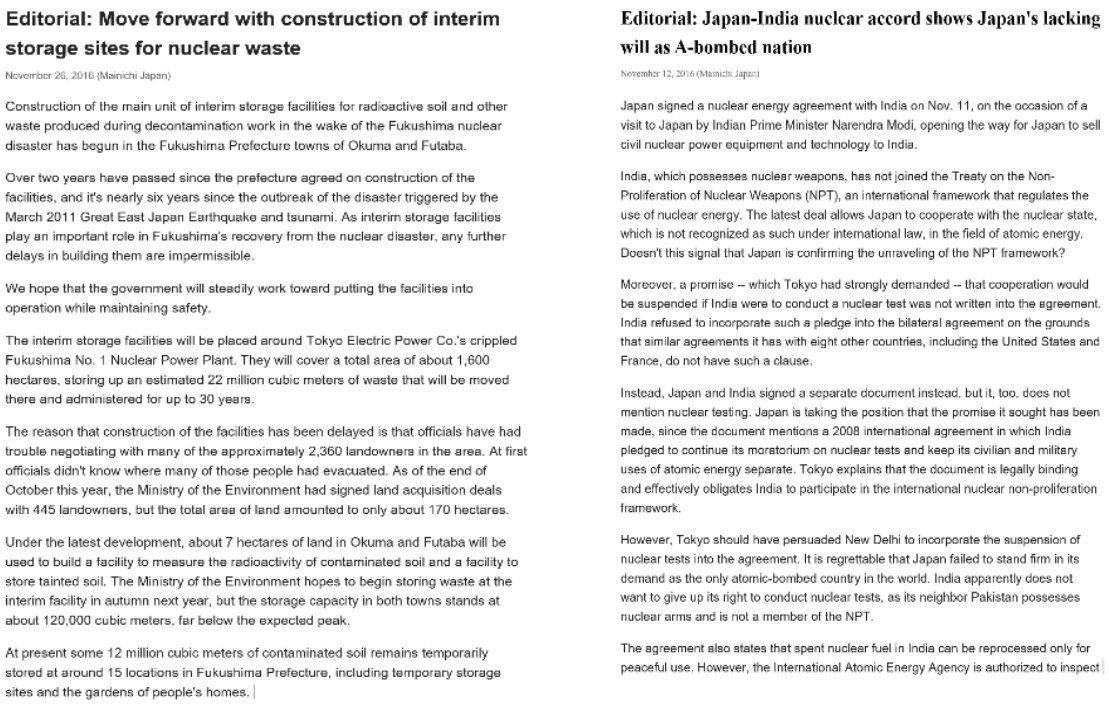
4.2 On/Off Topic Identification

In this section, the experiments for on/off topic identification were described.

4.2.1 Coding Guidelines

The central principle of our coding guidelines is that a circumstance, a current status, or government policy related to the Fukushima Daiichi nuclear power disaster are on topic content. For examples, editorials describing discussions caused by the disaster, or that expressed value judgments regarding one or more events relating the disaster, would be classified as on topic. On the other hand, an editorial about a government or local election would be classified as off topic if it mentioned that issues related to the disaster were important in a campaign, but those issues were not the focus of the editorial. Editorials about restarting some other nuclear power plant, without mentioning the Fukushima Daiichi disaster, would also be off topic.

Figure 4-1 illustrates some of the challenges of topic classification. The editorial on the left [9] is about the contentious issue of storage of nuclear waste that results from the operation of nuclear power plants. The editorial on the right [10], by contrast, is about commercial overseas sales of nuclear power technology. Unquestionably, both address topics related to the commercial production of nuclear power, but according to our coding guidelines, only the editorial on the left is on topic. The focus of the editorial on the right is on limiting the proliferation of nuclear weapons by imposing limitations on how the spent fuel for nuclear power plants can be reprocessed.



Editorial: Move forward with construction of interim storage sites for nuclear waste

November 26, 2016 (Mainichi Japan)

Construction of the main unit of interim storage facilities for radioactive soil and other waste produced during decontamination work in the wake of the Fukushima nuclear disaster has begun in the Fukushima Prefecture towns of Okuma and Futaba.

Over two years have passed since the prefecture agreed on construction of the facilities, and it's nearly six years since the outbreak of the disaster triggered by the March 2011 Great East Japan Earthquake and tsunami. As interim storage facilities play an important role in Fukushima's recovery from the nuclear disaster, any further delays in building them are impermissible.

We hope that the government will steadily work toward putting the facilities into operation while maintaining safety.

The interim storage facilities will be placed around Tokyo Electric Power Co.'s crippled Fukushima No. 1 Nuclear Power Plant. They will cover a total area of about 1,600 hectares, storing up an estimated 22 million cubic meters of waste that will be moved there and administered for up to 30 years.

The reason that construction of the facilities has been delayed is that officials have had trouble negotiating with many of the approximately 2,360 landowners in the area. At first officials didn't know where many of those people had evacuated. As of the end of October this year, the Ministry of the Environment had signed land acquisition deals with 445 landowners, but the total area of land amounted to only about 170 hectares.

Under the latest development, about 7 hectares of land in Okuma and Futaba will be used to build a facility to measure the radioactivity of contaminated soil and a facility to store tainted soil. The Ministry of the Environment hopes to begin storing waste at the interim facility in autumn next year, but the storage capacity in both towns stands at about 120,000 cubic meters, far below the expected peak.

At present some 12 million cubic meters of contaminated soil remains temporarily stored at around 15 locations in Fukushima Prefecture, including temporary storage sites and the gardens of people's homes.

Editorial: Japan-India nuclear accord shows Japan's lacking will as A-bombed nation

November 12, 2016 (Mainichi Japan)

Japan signed a nuclear energy agreement with India on Nov. 11, on the occasion of a visit to Japan by Indian Prime Minister Narendra Modi, opening the way for Japan to sell civil nuclear power equipment and technology to India.

India, which possesses nuclear weapons, has not joined the Treaty on the Non-Proliferation of Nuclear Weapons (NPT), an international framework that regulates the use of nuclear energy. The latest deal allows Japan to cooperate with the nuclear state, which is not recognized as such under international law, in the field of atomic energy. Doesn't this signal that Japan is confirming the unraveling of the NPT framework?

Moreover, a promise -- which Tokyo had strongly demanded -- that cooperation would be suspended if India were to conduct a nuclear test was not written into the agreement. India refused to incorporate such a pledge into the bilateral agreement on the grounds that similar agreements it has with eight other countries, including the United States and France, do not have such a clause.

Instead, Japan and India signed a separate document instead, but it, too, does not mention nuclear testing. Japan is taking the position that the promise it sought has been made, since the document mentions a 2008 international agreement in which India pledged to continue its moratorium on nuclear tests and keep its civilian and military uses of atomic energy separate. Tokyo explains that the document is legally binding and effectively obligates India to participate in the international nuclear non-proliferation framework.

However, Tokyo should have persuaded New Delhi to incorporate the suspension of nuclear tests into the agreement. It is regrettable that Japan failed to stand firm in its demand as the only atomic-bombed country in the world. India apparently does not want to give up its right to conduct nuclear tests, as its neighbor Pakistan possesses nuclear arms and is not a member of the NPT.

The agreement also states that spent nuclear fuel in India can be reprocessed only for peaceful use. However, the International Atomic Energy Agency is authorized to inspect

Figure 4-1. English examples of on-topic (left) and off-topic (right) editorials (trimmed for display). The actual editorials used in the study are in Japanese.

4.2.2 Experimental Design

The second author of this paper used the resulting test collection to train and evaluate automated text classifiers for the on-off topic identification task. We first used the standard Juman package 7.01 [11] to perform tokenization and morphological analysis for Japanese. We performed feature selection by using only nouns, verbs and adjectives that occur two or more times in the corpus as the feature set for each classifier. We used all such terms found in the body of each editorial. The bodies of the editorial contained, on average, 668.2 terms before feature selection, and 323.0 terms after feature selection. We conducted experiments with two types of classifiers. The first was a Support Vector Machine (SVM), a commonly used binary classifier that we implemented using TinySVM [12], for which we tried either a linear or a quadratic kernel (second-order polynomial kernel) function. The second was fastText, a neural network "deep learning" classifier developed by Facebook's AI Research lab [13].

Figure 4-2 illustrates the evaluation design. To create learning curves, we first randomly shuffle the documents within each of the first 9 coding rounds. Specifically, we shuffle the 7 documents in Round 1 into a random order, we then shuffle the 13 documents in Round 2 into another random order, and we continue that process through Round 9. We then iteratively train a classifier by incrementally adding training data. Specifically, we train a classifier with just the first document, evaluate that classifier using F_1 (the harmonic mean of precision and recall) using the 100 coded documents from Round 10 as

a held-out evaluation set, then retrain the classifier on the first two documents from Round 1, then evaluate that classifier in the same way, and so on until all 348 documents from Round 1 through Round 9 have been used for training. We repeat that entire process 100 times, with 100 random shuffles of each Round, and then we plot the average F_1 (or accuracy) over those 100 rounds as a learning curve. The evaluation set, which by construction was randomly sampled from the entire collection, is nearly balanced, with 48 on-topic and 52 off-topic editorials. The learning curves for F_1 and accuracy are therefore similar, so for space reasons we show learning curves only for F_1 in this paper.

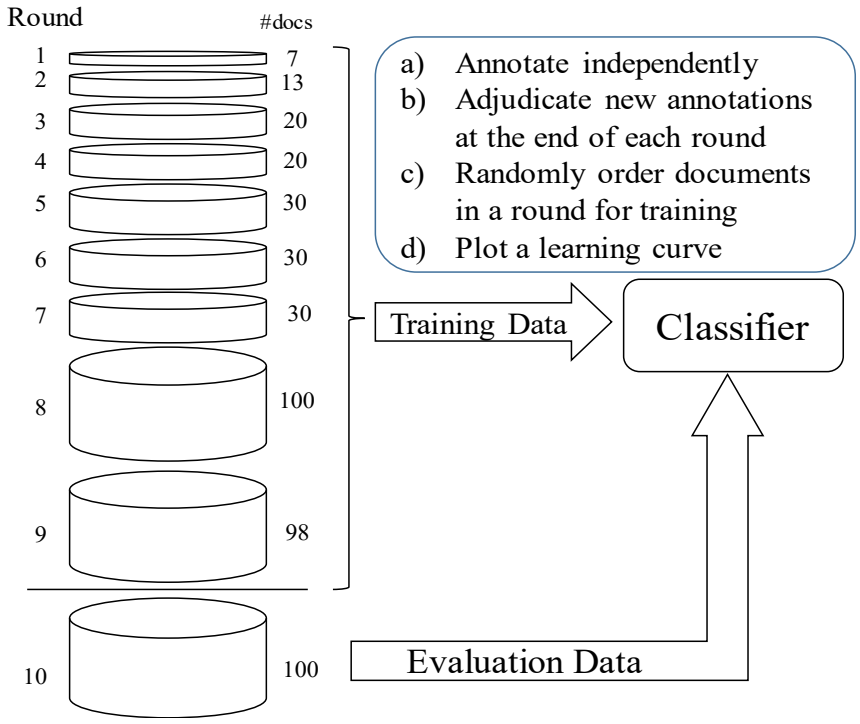


Figure 4-2. Evaluation design.

We plot learning curves by placing the number of coding decisions on the horizontal axis and the F_1 value for that number of coding decisions on the vertical axis. We plot four types of learning curves:

- A: we use Coder A's decisions for training.
- B: we use Coder B's decisions for training.
- Hybrid: we alternate between using Coder A's or Coder B's decisions for training.
- Adjudicated: we use the adjudicated decisions for training.

Note that creating one adjudicated decisions requires that Coder A and Coder B each first independently code the same document. In order to plot all four learning curves consistently, we randomly select half of the documents from each round when training using Adjudicated coding, and we plot the resulting values based on the original number of independent coding. For this reason, the learning curve for Adjudicated coding are plotted only at even numbers of decisions. Note also that it is not possible to meaningfully

train a classifier until at least one on-topic and at least one off-topic editorial are available for training, and with some coding methods and some random shuffles that does not happen until several coded documents have been seen. We therefore suppress the plotting of the learning curve until meaningful F_1 values are available for all 100 random shuffles.

4.2.3 Results

Figure 4-3 shows four learning curves, all of which are evaluated using adjudicated codes from Round 10. These are for a quadratic kernel SVM, which was the better of the two SVM kernel functions that we tried. As can be seen, Coder A's decisions initially provide the best training data, with Coder B's decisions becoming more useful for training in the later rounds, where they more closely approximate the adjudicated decisions. Alternating coding between Coder A and Coder B (the Hybrid learning curve) unsurprisingly comes out between the two. Using the Adjudicated decisions, which we reasonably presume are of higher quality than the decisions from either coder, is not the best choice, at least not until rather late in the process. The reason for this is that the x axis here shows the number of decisions, and to get adjudicated decisions, every document has to be coded twice. It seems clear that the additional diversity introduced by having different coders annotate different documents (as in the Hybrid learning curve) yields better results early on than having different coders annotate the same documents.

Figure 4-4 (a) and (b) show learning curves for the same training conditions, in this case evaluated using either Coder A's independent decisions from Round 10 (a) or Coder B's (b) independent decisions from Round 10. Unsurprisingly, each coder's training data is better matched to their own evaluation data, but note that the Hybrid training learning curve still dominates the Adjudicated training learning curve until quite a substantial number of training decisions have been made. From Figures 4-3 and 4-4 we can therefore conclude that the central message of Figure 4-3, that the Hybrid training strategy works well early in the training process, does not depend on an assumption that our adjudicated decisions are perfect (which they are not).

Figure 4-5 shows the same four learning curves when using the fastText classifier. As in Figure 4-3, these are evaluated using the adjudicated decisions from Round 10. The fastText results exhibit a similar pattern to what we observe with the SVM, although the initial learning rate for fastText is somewhat lower. We do see that fastText ultimately does outperform the SVM with Hybrid training, but that only happens rather late in the learning process. This is consistent with what we would expect from neural methods, which can achieve good results in data-rich settings. Although not shown, learning curves using Coder A's or Coder B's decisions for evaluation show similar patterns.

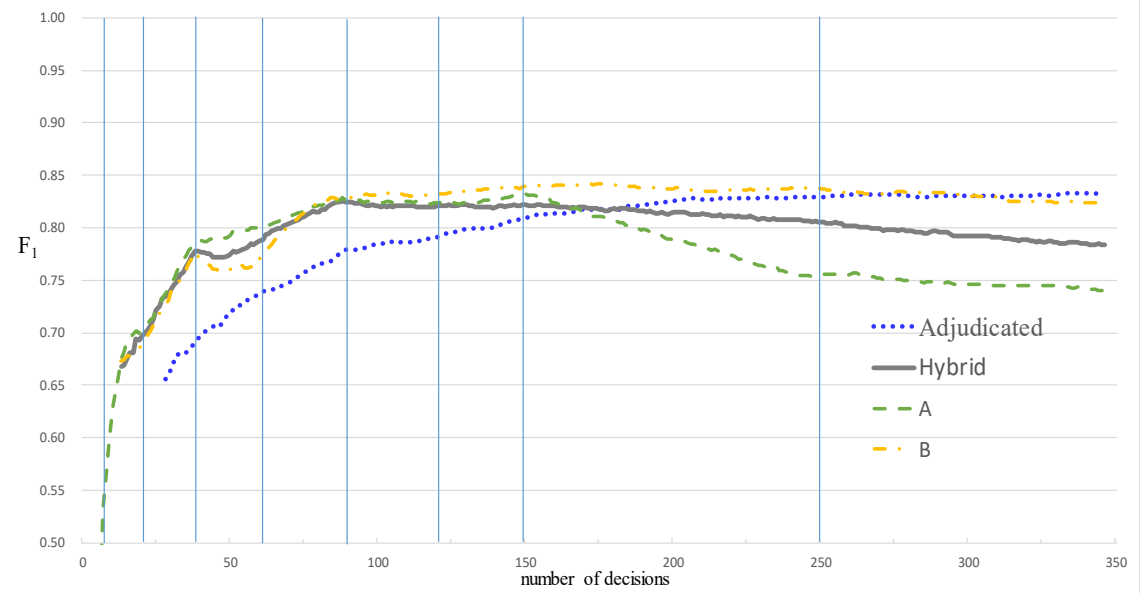


Figure 4-3. F_1 for quadratic kernel SVM, 100 held out adjudicated decisions used for evaluation, average of 100 random shuffles within each round.

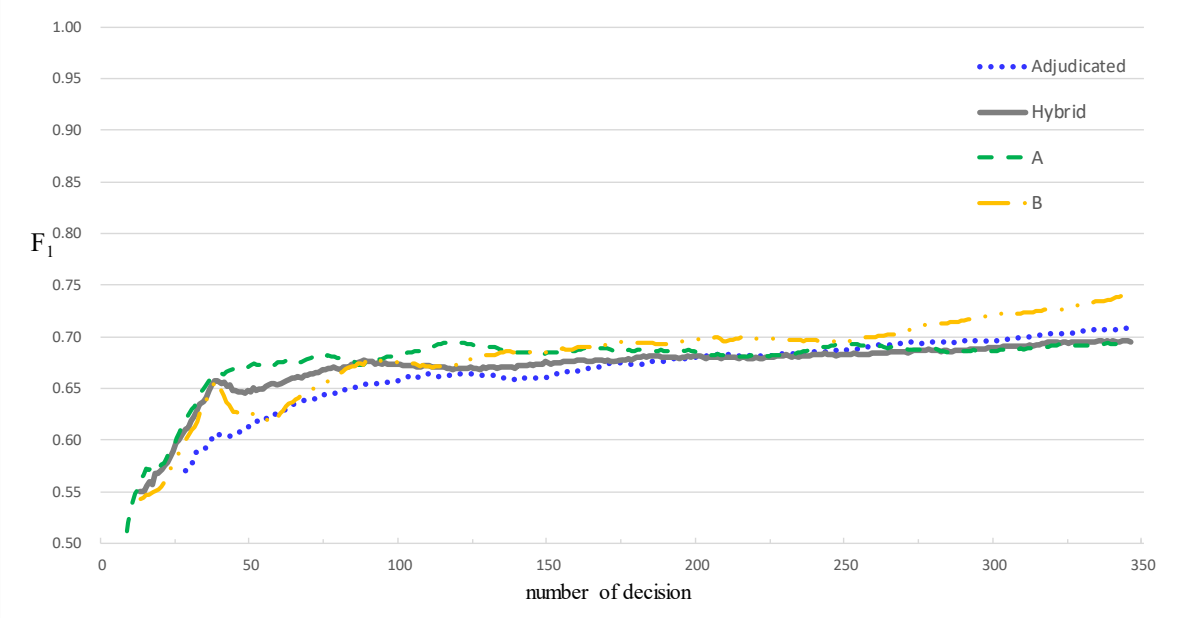


Figure. 4-4 (a). F_1 for quadratic kernel SVM, 100 held out decisions by Coder A used for evaluation; average of 100 random shuffles per round.

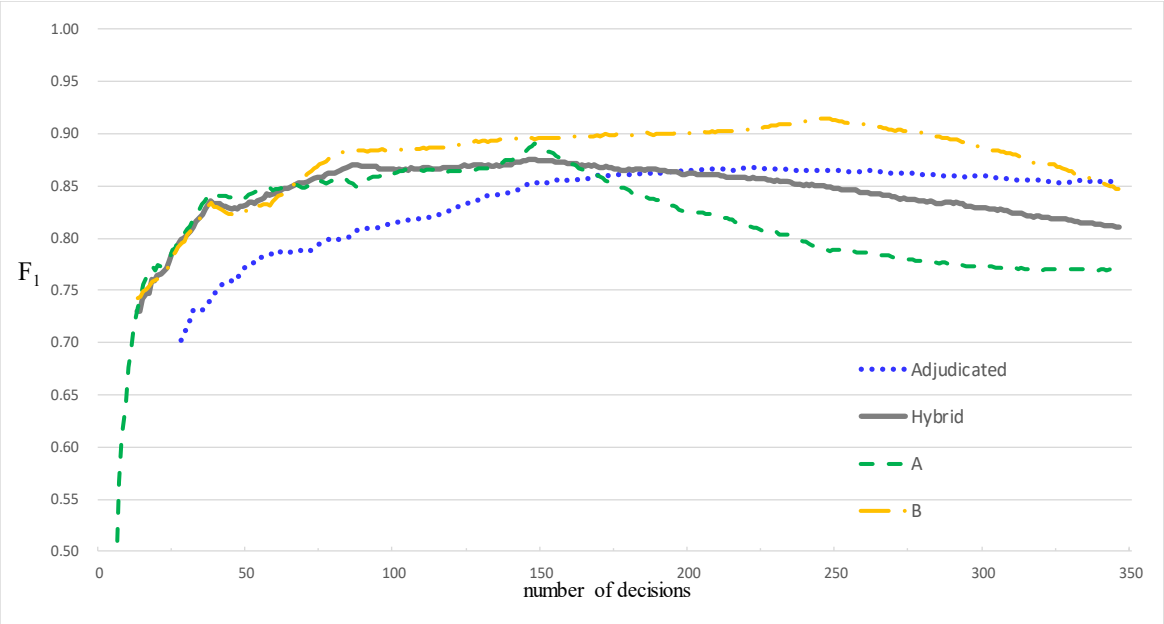


Figure. 4-4 (b). F_1 for quadratic kernel SVM, 100 held out decisions by Coder B used for evaluation; average of 100 random shuffles per round.

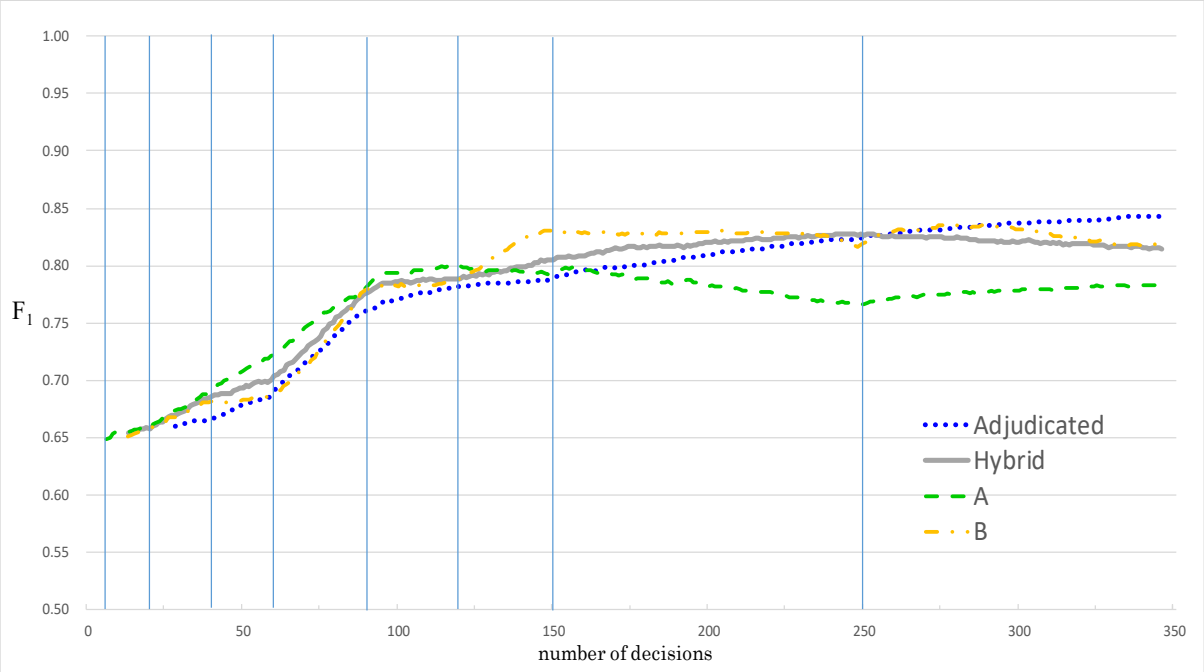


Figure 4-5. F_1 for fastText, 100 held out adjudicated decisions used for evaluation, averaged over 100 random shuffles per round.

4.3 Identifying Value Sentences

Although editorials generally contain more of the author's perspective than news articles, they still include events, facts, and other elements that are present for background. Editorials include, for example, "There are 54 nuclear power plants along the coastline," "In this time, multiple hypocenters interlocked and caused a huge earthquake," and "A basic energy plan in Japan that will guide the country's medium- to long-term energy policy will be decided by the Cabinet within the month" (these are English translations from editorials in the corpus). Our interpretation is that these types of sentences do not express or reflect specific values, so we annotate them as "fact sentences." Here are some examples of what we call value sentences: "It is time to find the appropriate power source for an earthquake-prone country such as Japan and how to live based on it," "It could be said that securing transparency is the most important issue in selecting new policies," and "The government and electric power companies need to deepen their political discussion by accepting citizens' anxiety." For our initial experiments with automatically determining which sentences a coder would interpret as value sentences, we have built a small training set and conducted a pilot study to characterize classification effectiveness. The same two coders as above independently reviewed each sentence in a total of 28 on-topic editorials to distinguish between value and fact sentences. Three rounds of independent coding were used, with adjudication by consensus between the two coders performed at the end of each round. Table 4-3 shows the number of documents and sentences in each round, along with agreement statistics before adjudication. A total of 578 of the 762 sentences were annotated as value sentences in the adjudicated codes.

Table 4-3. Cohen's Kappa for each round.

Round	#documents	#sentences	A : B
1	2	91	0.505
2	12	317	0.639
3	14	354	0.640

We then trained a quadratic-kernel SVM classifier on the adjudicated judgments, evaluating it using 10-fold cross-validation at sentence scale. For the first fold we selected the first 686 sentences for training and the remaining 76 for evaluation, repeating a similar process for ten different sets of 76 (or 77) evaluation sentences. As Table 4-4 shows, this yielded F_1 values near 0.87, regardless of whether we performed feature selection as above or simply used all of the terms in each sentence as features. Because we plan to perform value labeling only on sentences that have been automatically classified as values sentences, a high level of recall from the stage that finds value sentences is needed. As

Table 4-3 shows, recall values near 0.94 are possible (with feature selection) on this collection.

Table 4-4. Precision, Recall, and F_1 by SVM with quadratic kernel for finding value sentences.

Features	Precision	Recall	F_1
noun, verb, adjective	0.805	0.938	0.867
All	0.818	0.924	0.868

4.4 Summary

In this chapter, we have introduced a three-stage model for automating the association of human values with specific passages of text on large-scale collections. In particular, we focused on the quality-quantity trade-off for training data to optimize the cost-benefit ration, in the first stage of the three stage model. We built a test collection of 448 editorials on the nuclear power debate in Japan. We conducted experiments on on/off topic identification, using that collection. The results indicate that coding diversity trumps coding quality, suggesting that when multiple coders are available, the traditional practice of adjudicating conflicting decision for of the same documents is not as cost effective as an alternative in which each coder labels different documents.

Moreover, we have shown that when coding budgets are limited, it can be useful to focus on single-labeled training examples rather than adjudicated training examples that are created by assigning multiple coders to the same document, and that an SVM classifier can be a better choice than a state of the art neural deep learning classifier. Both of these results are consistent with results that have been reported in other settings (e.g., [14]); our contribution has been to bring these insights to bear in the context of a multi-stage coding process for human values.

References

[1] Takayama, Y., Tomiura, Y., Ishita, E., Oard, D. W., Fleischmann, K. R., & Cheng, A.-S. (2014). A word-scale probabilistic latent variable model for detecting human values. *The proceedings on ACM International Conference on Information and Knowledge Management (CIKM 2014)*, 1489-1498. DOI 10.1145/2661829.2661966

[2] Cheng, A.-S., Fleischmann, K.R., Wang, P., Ishita, E., & Oard, D.W. (2012). The role of innovation and wealth in the net neutrality debate: A content analysis of human values in congressional and FCC hearings. *Journal of the American Society for Information Science and Technology, 63*, 1360-1373.

[3] The Mainichi Newspapers Co.,Ltd. CD-Mainichi Shimbun 2011 Data Collection.

[4] The Mainichi Newspapers Co.,Ltd. CD-Mainichi Shimbun 2012 Data Collection.
[5] The Mainichi Newspapers Co.,Ltd. CD-Mainichi Shimbun 2013 Data Collection.
[6] The Mainichi Newspapers Co.,Ltd. CD-Mainichi Shimbun 2014 Data Collection.
[7] The Mainichi Newspapers Co.,Ltd. CD-Mainichi Shimbun 2015 Data Collection.
[8] The Mainichi Newspapers Co.,Ltd. CD-Mainichi Shimbun 2016 Data Collection.
[9] The Mainichi. (2016, November 26). *Editorial: Move forward with construction of interim storage sites for nuclear waste*. The Mainichi, https://mainichi.jp /English/ articles/20161126/p2a/00m/0na/006000c, (last accessed 2019-01-26).
[10] The Mainichi. (2016, November 12). *Editorial: Japan-India nuclear accord shows Japan's lacking will as A-bombed nation.* The Mainichi, https://mainichi.jp/english/articles/20161112/p2a/00m/0na/004000c, (last accessed 2019-01-26).
[11] JUMAN (a user-extensible morphological analyze for Japanese). http://nlp.ist.i.kyoto-u.ac.jp/EN/index.php?JUMAN, (last accessed 2018-9-10).
[12] TinySVM: Support Vector Machines, http://chasen.org/~taku/software/TinySVM/, (last accessed 2018-9-10).
[13] Joulin, A., Grave, E., Bojanowski, P., Mikolov, T.: Bag of tricks for efficient text classification, https://arxiv.org/abs/1607.01759, (last accessed 2018-9-10).
[14] Khetan, A., Lipton, Z. C., & Anandkumar, A. (2018). Learning from noisy singly-labeled data. *Proceedings of Sixth International Conference on Learning Presentations (ICLR) 2018*, 15p. https://arxiv.org/abs/1712.04577, (last accessed 2018-9-10).

Chapter 5

Conclusion

This dissertation reported on the results of research addressing three real problems of automatic text classification.

In chapter 2, we focused on detecting academic papers from PDF files found on the web, which kinds of information are useful as features was the focus of the study. We selected plausibly useful features based on the structure of the documents, the presence of specific elements or specific expressions, and descriptive attributes of the document files. The utility of these features was then characterized using experiments with two newly constructed test collections. For example, a Random Forest classier achieved an F_1 measure of 0.74 on the 20,000-document English PDF collection. In aggregate, our results show that document features designed specifically for this task yield good results.

In chapter 3, we explored the utility of text classification by introducing the first classifier for coding human values as one application for non-topical classification in computational social science. We regarded each human value in the coding frame as a category and measured the ability of existing classifiers to infer the presence of human values at sentence scale. Our test collection included 2,294 sentences from 28 written opening testimonies that were prepared for public hearings on Net neutrality in United States of America. Each sentence was assigned categories from an inventory of 10 human values selected from the Schwartz Value Inventory. We automatically assigned human values to sentences using a k-Nearest-Neighbor classifier with words as features. Although our results indicated that there was some room for improvement, we did show that text classification was amenable to automate the inference of human values. Subsequently we used an expanded test collection with 102 documents and better chosen human value categories consisting of six human values. We used this improved test collection to explore learning rates, showing that relatively modest quantities of training data can be suffice. That result has important practical implications for the adoption by social scientists of this approach to coding and for scalability for a large-scale document set.

In Chapter 4, we focused on cost-effective ways of generating the training data. To do this, we introduced an evaluation framework in which we counted coding decisions, rather than documents, thus better reflecting the true costs of generating training data when

multiple coders are used to improve data quality. We then used this framework to examine a quality vs. quantity tradeoff, seeking an approach that would maximize classifier effectiveness for a given level of human coding effort. Our task in this case was to extend our work on automatic classification of human values to assign human values to sentences in a collection of 450 editorials. In our earlier work on Net neutrality, each document had been manually selected as being on topic. In this case, however, we extended our task design to include automating the detection of on-topic documents. We worked with Japanese editorials, and sought to recognize those that meaningfully addressed the nuclear power debate. The automated coding process was therefore divided into three stages; (1) document selection (i.e., whether an editorial was on topic), (2) value sentence selection (i.e., whether a sentence in an on-topic editorial expressed or reflected a human value), and (3) automatically coding value sentences in on-topic documents for human values. For document selection, which was the principal focus of our experiments in Chapter 4, we subsampled our training data annotations to compare the utility of smaller amounts of training data of higher quality (based on multiple annotation) with the utility of equally costly larger amounts of lower quantity (single-annotation) training data. Our results show that when cost is tightly constrained, using larger quantities of lower quality training data is the better choice, whereas later in the learning curve more costly training data of higher quality, albeit in lower quantities, is the better choice.

Looking beyond these contributions, the research in this dissertation suggests a number of possible directions for future work.

- Bandit algorithms for adaptive management of training data quality

Although we have shown that larger quantities of lower quality training data are useful initially, and that at some point generating higher quality training data would be the better choice, that leaves open the question of how one could best recognize when to switch between the two strategies. It seems unlikely that we could find an analytical answer to that question, but if we cast this as a reinforcement learning problem we could use intermediate results to learn to choose between the two strategies. This is an application of what has been called a multi-armed bandit algorithm [1] in which the expected result from each alternative is used to optimally choose which alternative to choose (the name is an analogy to a gambler who always pulls the arm on the slot machine form which they expect the best payoff).

- Three-stage decomposition of annotation for topic-specific human values

In Chapter 4, a three stage decomposition of the process for coding human values in documents that address a specific topic was proposed. Our experiments in that chapter principally focused on the first (document selection) stage, however, with only preliminary results for the second (value sentence detection) stage. Although we have shown results for the third stage (value classification) in Chapter 3 with a different test collection, we have yet to study the interplay between the three stages on the same value labeling task.

To do this we will need a larger test collection, as every stage of our three-stage process decreases the size of the collection on which the next stage is tested.

- Extrinsic evaluation

This research has been on item-based intrinsic evaluation, using measures such as precision, recall, and F_1. These measures have proven to be useful for three reasons. First, systems which do well by these measures would clearly be useful because they could clearly be substituted for human effort with little distortion. Second, these types of measures are widely reported and well understood, which helps to make our work accessible to a broader audience. Third, these types of measures are easily calculated using, for example, cross-validation methods. However, intrinsic measures can't tell us what level (short of perfect) would be "good enough" for any particular use of our classifier's results. In Chapter 3, for example, it is necessary to use an evaluation measure indicating that human coding effort can be reduced by using classifiers, or the introduction of classifiers contributes scalability for large-scale document set. For that, we need an extrinsic evaluation measure that characterizes the degree to which the ultimate users of our classifiers (e.g., social scientists, in the case of our work on human values) are able to achieve their goals [2]. For example, social scientists often compare proportions, and offsetting errors (e.g., equal numbers of false positive and false negative classifier decisions) would have no adverse affect on such a use for our classification results. Extrinsic evaluation is application specific and it can be costly, so it makes sense to start with intrinsic evaluation, as we have done. But when building classifiers for use in specific application settings, it will be important to ultimately progress to conducting extrinsic evaluation as well.

- Supporting the design of coding frames

Our focus has been on the design of classifiers that learn from examples, but for those examples to be constructed the category set must already exist. At present, the process of conducting the category set (which social scientists often refer to as the "coding frame"), is entirely manual. If we truly wish to comprehensively address the category set, we must ultimately address not just its use, but also its creation. Supporting that human activity is not a classification task, but rather an abduction task. Humans are brilliantly good at recognizing patterns and at conducting goal-directed reasoning from examples, but the scale at which they can operate is fundamentally limited. What is needed, therefore, are tools to help people perform the abduction task involved in formalization of category at larger scale then they presently can accomplish. Perhaps some benefit might be obtained from unsupervised approaches to topic modeling [3] as a starting point. And perhaps visualization techniques might help to support human pattern recognition at scale. Regardless of the techniques employed, the key will be to design for synergy between human and machine.

In aggregate, our results clearly indicate the benefits that can accrue from the formalization of the structure of the three real application setting in text classification. By focusing on real problems that call for novel approaches, we have gained important insights into the considerations that arise in each of problems of a comprehensive approach to text classification.

References

[1] Berry, D. A. & Fristedt, B. (1985). *Bandit problems: Sequential allocation of experiments*. Monographs on Statistics and Applied Probability, Chapman and Hall, 275p.

[2] Jones, K. S., & Julia R. Galliers, J. R. (1996). *Evaluating natural language processing systems: An analysis and review*. Spinger-Verlag, 228p.

[3] Blei, D. M. (2012). Probabilistic topic models. *Communication of the ACM*, 55(4), 77-84.

Acknowledgments

Firstly, I would like to express my sincere gratitude to my advisor Professor Yoichi Tomiura for the continuous support of my research work, for his patience, motivation, and immense knowledge. His guidance helped me throughout my research and the writing of this dissertation. This dissertation would not have been possible without his supervision.

I would like to thank the members of my dissertation committee: Professor Masayuki Takeda, Professor Eiji Takimoto, and Professor Daisuke Ikeda. I am also grateful to the members of my advisory committee: Professor Shuji Kijima, and Professor Tsunenori Mine.

For the work described in Chapter 2, I would like express my gratitude to Emeritus Professor Shuichi Ueda (Keio University), Professor Teru Agata (Asia University), Professor Atsushi Ikeuchi (University of Tsukuba), and Professor Yosuke Miyata (Teikyo University). I worked with this excellent team over ten years. It was fun of working with them.

For the work described in Chapter 3 and Chapter 4, I would like express my gratitude to Professor Douglas W. Oard (University of Maryland), Professor Kenneth R. Fleischmann (The University of Texas at Austin), Professor An-Shou Cheng (National Sun Yat-sen University), Professor Yasuhiro Takayama (National Institute of Technology, Tokuyama College), Professor Toru Oga, and Dr. Satoshi Fukuda. They always supported me and shared great insights. Professors Oard and Fleischmann introduced me to computational social science research. This was a new research world for me with a lot of research questions to explore. I hope I can continue to work with them.

Emeritus Professor Shuichi Ueda and Professor Keiko Kurata (Keio University) were co-supervisors of my Ph.D program at Keio University. They guided me as I learned fundamental research skills. I could not have started my research career without their mentorship.

I would also like to express my appreciation to the supervisor of my master thesis, Emeritus Professor Takeo Yamamoto (National Institute of Informatics). He sparked my interest in research.

I would like to express my appreciation to colleagues at Kyushu University, research collaborators, former colleagues, and former research collaborators. All of these experiences have helped me to conduct my dissertation work.

This dissertation work was supported in part by JSPS KAKENHI Grant Numbers JP21300095, JP22500220, JP25280118, and JP18H03495.

Finally, I wish to thank my friends and my family who raised me.