

Demystifying Ambiguous Words in Request for Proposals of Information Systems in Japan

Toru Nakamichi

 <https://orcid.org/0009-0001-5913-0980>

Okayama University, Japan

Hiroki Inayoshi

 <https://orcid.org/0000-0003-3355-8804>

Okayama University, Japan

Kinari Nishiura

 <https://orcid.org/0009-0007-7168-1500>

Kyoto Institute of Technology, Japan

Mariko Sasakura

 <https://orcid.org/0000-0002-8909-9072>

Tottori University, Japan

Akito Monden

 <https://orcid.org/0000-0003-4295-207X>

Okayama University, Japan

ABSTRACT

Disagreements in interpreting words in software requirements specifications (SRSs) can lead to project failure. Various approaches to identifying and preventing ambiguous words in SRSs have been proposed. Yet, it is unclear which ambiguous words are used in the actual SRSs and to what extent they need to be modified. This paper quantitatively analyzes existing SRSs to clarify (1) how many ambiguous words are included in SRSs and (2) how many of these words require correction. This paper targets the Request for Proposals (RFPs), which describe the initial requirements of 40 systems of local governments, libraries, universities, and hospitals in Japan. Ten ambiguous Japanese words were analyzed. The result shows that “juubun” (sufficient) appeared most frequently, and 42% required correction when this word was used. The result also indicates that the number of ambiguous words varied greatly among the RFPs and that larger RFPs did not necessarily contain more ambiguous words.

KEYWORDS

Information Systems, Request for Proposal, Software Quality, Specification Document

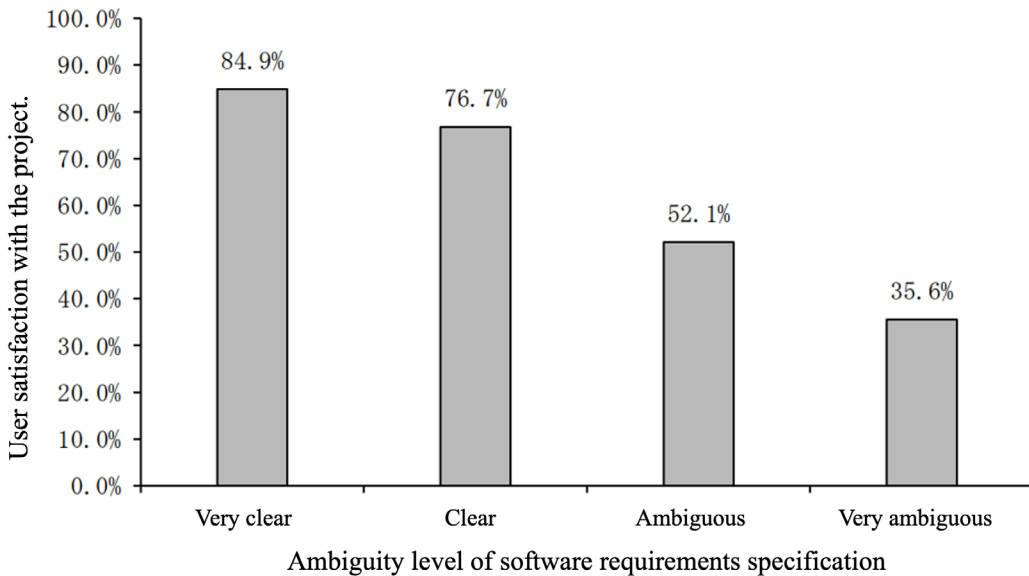
INTRODUCTION

It is extremely important in software development that specifications are unambiguously explained in the software requirement specification (SRS). This paper focuses on the ambiguous words contained in the request for proposal (RFP), one of the SRSs that is the gateway to software development and the key to successful software development (Saito et al., 2012). The RFP is a document prepared by a software purchaser company (i.e., the user) meant to clarify the objectives and fundamental

DOI: 10.4018/IJSI.358012

This article published as an Open Access article distributed under the terms of the Creative Commons Attribution License (<http://creativecommons.org/licenses/by/4.0/>) which permits unrestricted use, distribution, and production in any medium, provided the author of the original work and original publication source are properly credited.

Figure 1. Relationship between requirements ambiguity and user satisfaction (Japan users association of information systems, 2020)

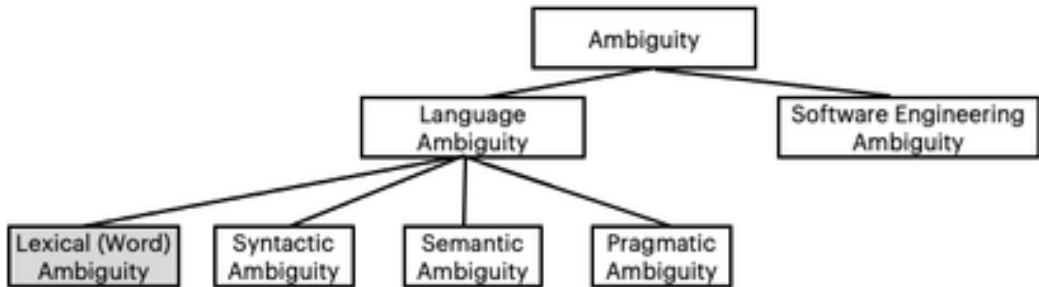


requirements of the software being developed. The intention is to communicate with the software developer company (i.e., the vendor) and to request proposals for technical specifications of the software.

Difficulties often occur due to differences in system understandings caused by ambiguous descriptions in the requirements (Berry, 2007). For example, after the contract is signed, there may be a comprehension gap between the requirements intended by the user and the requirements understood by the vendor. Another possibility is when users do not receive adequate maintenance services because their maintenance requirements are not adequately described in the RFP. The Japan Users Association of Information Systems collected and analyzed metrics data from 1,433 software projects in Japan and reported that the level of specification ambiguity affects user satisfaction with the software project (Japan Users Association of Information Systems, 2020). As shown in Figure 1, ambiguous requirement specifications reduce user satisfaction, which can be considered project failure for user companies.

To avoid ambiguities in requirement specifications, various requirement guidelines and templates have been proposed to describe requirements without causing misunderstandings (Berry et al., 2003; Mavin et al., 2019; Pohl & Rupp, 2015). Also, automated approaches to identifying ambiguous requirements have been proposed (Kiyavitskaya et al., 2008; Mishra & Sharma, 2019; Osman & Zaharin, 2018). As shown in Figure 2, according to the ambiguity handbook (Berry et al., 2003), there are several types of ambiguities in requirement specification documents. First, ambiguity is classified into “language ambiguity” and “software engineering ambiguity.” Language ambiguity means that the meaning of a statement is ambiguous, while software engineering ambiguity means that the statement itself is clear but some information is missing to describe system requirements. Software engineering ambiguities depend on the domain involved and can only be discovered by practitioners with sufficient domain knowledge. On the other hand, language ambiguity is classified into lexical ambiguity, syntactic ambiguity, semantic ambiguity, and pragmatic ambiguity. Lexical ambiguity occurs when a word itself has several meanings or interpretations. Syntactic ambiguity occurs when a given sequence of words can be given more than one grammatical structure and each has a different meaning. Semantic ambiguity occurs when it is possible to read a sentence in

Figure 2. Classification of ambiguity in requirements specifications



multiple ways within its context. Pragmatic ambiguity occurs when a sentence has several meanings in the context in which it is uttered. In this paper, we focus on the lexical (i.e., word) ambiguity. To date, various examples of ambiguities, including word ambiguity, have been presented (Berry et al., 2003); however, there are no empirical studies on which words are the most harmful in real world specifications. Therefore, in this paper we aim to identify harmful words that are very often used and are very often ambiguous so that practitioners can use the list of harmful words to check and improve the ambiguities of their requirements documents.

This paper targets the RFPs of 40 systems of local governments, libraries, universities, and hospitals in Japan and quantitatively analyzes these RFPs to clarify the following research questions:

1. How many ambiguous words are included?
2. How many of these words require correction?

Since RFPs are written by software user organizations, this study is meant to assist the user by clarifying to which words the user should pay particular attention.

Below, the second section presents the ambiguous words analyzed in this paper. The next section describes our method of analyzing ambiguous words in RFPs. The fourth section describes results of our analysis. Finally, the last section summarizes the paper.

POTENTIALLY AMBIGUOUS WORDS

Ambiguity of requirements often comes from modifier—or qualifier—words such as adjectives and adverbs. For example, the modifier “sufficient” is very often ambiguous because what is sufficient in a given situation varies from person to person. In this paper, we focus on the following modifiers that represent the degree of the modified words. In Japanese, these words are originally nouns; typically, with the “na” or “ni” suffix, they are used as adjectives or adverbs. We selected these words for the following three reasons:

1. They are very commonly used in daily life in Japan and thus are commonly used in requirement documents.
2. They contain inherent ambiguity—that is, sentences using them are always “locally” ambiguous unless additional explanation to resolve the ambiguity is provided.
3. These words have been often pointed out in the Japanese software industry as ambiguous words in requirement and specification documents (Nakayama, 2019).

These are the words we analyzed:

1. *Ippanteki* (i.e., common, commonly used, widely used): This word is often ambiguous because the criteria for judging “common” vary for each person. In particular, the user side and the vendor side have different levels of knowledge about information technology, and thus the perception of “common” technology may also differ.
2. *Juudai* (i.e., serious, seriously, critical, severe): This word is often used to describe the degree of system trouble (e.g., security incidents). However, there may be differences in the degree of severity between users and vendors.
3. *Juubun* (i.e., sufficient, sufficiently): This word is often used to qualify the capacity, quality, or performance of a system's components (memory, network, hardware, and software). For example, it is often used to mean “ensuring sufficient memory.” However, if the specifications do not specify the criteria for what constitutes “sufficient,” the word is subject to varying interpretations.
4. *Tekido* (i.e., adequate, adequately, appropriate, reasonable): This word is used when some value is neither too much nor too little. Generally, users do not want a poor system, but they also do not want a system that is too rich, so they may use this word. Since what is adequate is different for different people, this word is ambiguous.
5. *Kouritsuteki* (i.e., efficient, efficiently): This word is typically used to qualify the performance of a system's components. For example, it can be used as “this system must operate efficiently.” The word is ambiguous unless the specification indicates a criterion for efficient.
6. *Juuyou* (i.e., important): This word is commonly used in daily life; it can also be used in requirement documents. However, this word is highly subjective and should not be used to qualify factors that affect the procurement, design, or construction of system components.
7. *Shunji* (i.e., immediate, immediately), *sugu* (i.e., immediate, immediately), *sokuji* (i.e., immediate, immediately), and *tadachi* (i.e., immediate, immediately): These four words are often used to describe system behaviors—for example, a statement that describes input/output behavior such as “returns output immediately after user input.” However, “immediately” here may mean within 100 milli second or within one milli second . If no specific value is given, the word is ambiguous.

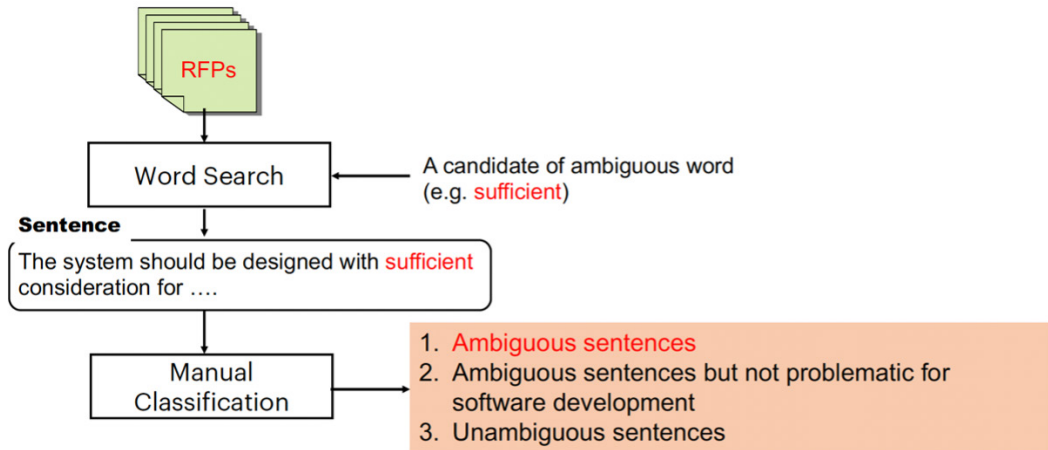
These modifiers are often used to indicate project deadlines, quality standards, and/or the speed and efficiency of expected results but can be ambiguous because the degree expressed by the modifiers can be interpreted in different ways. This type of ambiguity can be a significant barrier to communication between the vendor and the user and can have a direct impact on the quality of the final system and the success of the project. Even if the development team is highly skilled, ambiguous language in an RFP can lead to different interpretations, resulting in unnecessary rework, excessive cost burdens, and even overall project failure.

ANALYSIS METHOD

In this paper, we target the RFPs of backbone information systems for local governments, libraries, universities, and hospitals. These RFPs are publicly available on the websites of public organizations and reflect the unique requirements and expectations of each project. There were 10 RFPs from local governments, 10 from public libraries, 10 from universities, and 10 from hospitals.

Figure 3 shows the overview of our analysis method. In the analysis, for each RFP, sentences containing the potentially ambiguous words introduced before were extracted. This step was completed with keyword searching. Next, the sentences before and after the extracted sentences were manually read and classified into the following three groups: ambiguous sentences, ambiguous sentences but not problematic sentences for development, and unambiguous sentences. The following are examples of sentences that fall into each group.

Figure 3. Overview of the analysis method



Ambiguous Sentences

An example of an ambiguous sentence is “A function that enables efficient registration work to be performed even in cases where a large amount of user information is handled (personnel transfers, mergers, etc.), such as batch transfer processing and pre-registration functions.” In this example, the word “efficient” qualifies “registration process,” but because there are no details about the registration process either before or after the sentence, there is a risk that the project will proceed with varying assumptions if the sentence is left as is. There is no additional explanation about the “efficiency” of the registration process either before or after the sentence. We categorized such sentences that may pose a risk to the project progression as “ambiguous sentences.”

Ambiguous but Unproblematic Sentences

An example of a sentence that is ambiguous but not problematic for development is “When connecting to the campus LAN, the necessary settings shall be made after sufficient consultation with the university.” Although the potentially ambiguous word “sufficient” is used here, the essence of the sentence can be understood without the word “sufficient” in this sentence. Sentences containing such modifiers that do not significantly affect the meaning of the sentence are categorized as “ambiguous but not problematic.”

Unambiguous Sentences

An example of an unambiguous sentence is “For data centers, all of the following requirements shall be met; and, security measures and safety shall be sufficiently ensured.” The sentence itself is ambiguous, but the succeeding sentence explains what and how to ensure; thus, there is no ambiguity in the RFP. In such a case, we decide that the sentence is unambiguous because we want to clarify the reality of the practical ambiguities in RFPs. Therefore, even though the initial sentence is ambiguous, the subsequent explanation removes any ambiguity.

RESULT OF ANALYSIS

Before going into the quantitative results, Table 1 shows examples of modified words for each modifier in the ambiguous sentences of our analysis. Some of these are commonly found in RFPs in various domains. For example, *juudai* (i.e., serious) is very often used to qualify “issues” and

Table 1. Examples of modified words for each modifier

Modifier	Modified words
Ippanteki	Municipalities, security measures
Juudai	Issues, failure, security
Juubun	Skill and experience, discussion, capacity, performance, understanding, reliability, processing, testing, training
Kouritsuteki	Cost, operation and migration, system, function, task
Juuyou	Information, comment
Shunnji, sugu, sokuji, tadachi	Restoration, response, display, reflection, submit, search

Table 2. Classification of sentences containing a potentially ambiguous word in RFPs of local government systems

Modifier	Ambiguous	Ambiguous but not problematic	Unambiguous	Total
Ippanteki	2	1	1	5
Juudai	2	0	0	2
Juubun	21	15	12	48
Tekido	0	0	0	0
Kouritsuteki	14	11	11	36
Juuyou	3	1	3	7
Shunnji	0	0	0	0
Sugu	0	0	0	0
Sokuji	4	0	5	9
Tadachi	0	1	4	5

“failure.” Examples of whole, ambiguous sentences are shown in the Appendix. The following sections describe quantitative results for local government systems, library systems, university systems, and hospital systems.

Local Government Systems

In the local government RFPs, 112 sentences containing potentially ambiguous words were extracted. The classification results of those sentences are shown in Table 2.

In local government RFPs, sentences that contain tekido, shunji, and sugu were not detected. On the other hand, a relatively large number of ippanteki and kouritsuteki uses were detected. Although tadachi, shunnji, sugu, and sokuji have very similar meanings, their results differ from each other. The reason for this is probably that each person has a different preference for which words to use in an RFP, and the level of ambiguity varies from RFP to RFP, which is why we believe this difference in the results has emerged. Below is a detailed discussion of the findings and their implications:

1. Ippanteki: The word ippanteki was found in only two RFPs. In one of the RFPs, all sentences containing the word were classified as “ambiguous,” and in the other RFP, they were all classified as “ambiguous but not problematic.” This suggests that some document authors tend to use this word, and in the latter case, even this word was preferred, the sentences that followed would have described additional explanations to avoid ambiguity.

Table 3. Classification of sentences containing a potentially ambiguous word in RFPs of library systems

Modifier	Ambiguous	Ambiguous but not problematic	Unambiguous	Total
Ippanteki	0	0	0	0
Juudai	0	0	0	0
Juubun	16	13	6	35
Tekido	0	0	0	0
Kouritsuteki	0	1	0	1
Juuyou	2	1	2	5
Shunnji	0	0	0	0
Sugu	0	0	0	0
Sokuji	0	1	1	2
Tadachi	0	1	3	4

- Juudai: The word juudai was detected twice, both times in the context of security issues and classified as “ambiguous.” Although the word is used infrequently, when it is used, one needs to be careful because of its potential security relevance.
- Juubun: The word juubun was the most frequently detected word in the municipal RFPs. It is also the word most likely to be ambiguous in municipal RFPs. This word was found to be the most common word qualifying skills and capacities. The word accounted for five of the 18 ambiguous sentences. In addition, four ambiguous sentences about capacity were detected. However, the criteria for juubun are not clearly defined, making it difficult to read specific requirements and expectations. This can lead to misunderstandings about system performance and capacity.
- Kouritsuteki: The word kouritsuteki is often used to qualify the function of a system. The word is preferred in contexts where the goal is to improve processes or reduce costs, but the specific efficiency indicators or goals are often unclear. This makes it difficult to understand exactly what improvements are expected.
- Juuyou: The word juuyou is found primarily in sentences dealing with messages and data. Sentences containing this modifier often direct personal information handling and security precautions. However, the scope of “important” and specific measures are sometimes poorly explained, and thus, the requirements tend to be unclear.
- Sokuji: The word sokuji is often found in sentences about failures and operations. We also found that the words sokuji and “respond” are often used together. This expression is used in situations where prompt action is required, especially immediately after a system goes live or during the emergency response process. However, if a specific timeframe for sokuji is not defined, misunderstandings about the actual speed of response may arise.
- Tadachi: The word tadachi is often found in sentences related to reporting and communication. Sentences containing this word are usually found in situations that call for the quick sharing of information or immediate communication.

Library Systems

Forty-seven sentences containing potentially ambiguous words were extracted within the library's RFP. The classification results of those sentences are shown in Table 3.

Compared to the local government RFPs, the number of ambiguous sentences detected was relatively small. Of the 10 potentially ambiguous words, only two—juubun and juuyou—were detected in more than five sentences. Notably, while 36 sentences containing kouritsuteki were detected in

Table 4. Classification of sentences containing a potentially ambiguous word in RFPs of university systems

Modifier	Ambiguous	Ambiguous but not problematic	Unambiguous	Total
Ippannteki	1	1	0	1
Juudai	2	1	2	5
Juubun	14	6	1	21
Tekido	0	0	0	0
Kouritsuteki	2	1	2	5
Juuyou	7	1	0	8
Shunnji	0	0	0	0
Sugu	1	1	0	2
Sokuji	0	0	1	1
Tadachi	2	1	1	4

local government RFPs, only one sentence was detected in library RFPs. Below we describe the details of our findings for juubun and juuyou:

1. **Juubun:** The word juubun was the most detected word within the library RFPs. The word is often used within library RFPs in sentences related to stability and smoothness of systems and system operations. The frequent use of juubun within the library RFPs suggests a lack of clarity in project requirements. Particularly when the word is used in the context of assuring the stability and smooth operation of the system, it increases the likelihood that there is a lack of common understanding between the vendor and client regarding the level of performance expected.
2. **Juuyou:** The word juuyou was detected mainly in sentences dealing with messages and data. This characteristic was also found in the local government RFPs, which often included ambiguous sentences in descriptions related to personal information and other information. The use of this word gives the impression that the item has a high priority, but often no specific criteria are given as to what is important.

University Systems

Forty-six sentences containing potentially ambiguous words were extracted within the university's RFP. The classification results of those sentences are shown in Table 4.

This university RFP is like the RFP for local governments. The number of ambiguous sentences detected was relatively small. Of the 10 potentially ambiguous words, only two, juubun and juuyou, were detected in more than five sentences. This RFP has a low occurrence of words other than juuyou:

1. **Juubun:** The word juubun was the most detected word within the university RFPs. In addition, 14 out of 21 were ambiguous. This frequent usage indicates a lack of precision in defining project requirements. Specifically, the use of juubun in the context of system performance and capabilities suggests that there is often no clear agreement on what level of performance is considered “sufficient.” This ambiguity can lead to varying interpretations by vendors, which may result in proposals that do not meet the university's actual needs.
2. **Juuyou:** The word juuyou was detected mainly in sentences dealing with messages and data.

Table 5. Classification of sentences containing a potentially ambiguous word in RFPs of hospital systems

Modifier	Ambiguous	Ambiguous but not problematic	Unambiguous	Total
Ippanteki	2	1	0	3
Juudai	5	0	0	5
Juubun	47	71	12	130
Tekido	0	0	0	0
Kouritsuteki	26	3	17	46
Juuyou	8	8	4	20
Shunnji	0	1	0	1
Sugu	2	1	1	5
Sokuji	17	1	18	36
Tadachi	1	3	4	4

This characteristic was also found in the local government RFPs, which often included ambiguous sentences in descriptions related to personal information and other information. The use of this word gives the impression that the item has a high priority, but often no specific criteria are given as to what is “important.”

Hospital Systems

Forty-six sentences containing potentially ambiguous words were extracted within the university's RFP. The classification results of those sentences are shown in Table 5.

Hospital RFPs feature more pages and more occurrences of words than other RFPs. Among them, the words *kouritsuteki*, *juubun*, *juuyou*, and *sokuji* occur frequently. This higher frequency of occurrence suggests a greater emphasis on these aspects within hospital projects, reflecting the critical nature of healthcare services where efficiency, sufficiency, and immediacy are paramount:

1. *Kouritsuteki*: The term *kouritsuteki* was commonly found within university RFPs, with 26 out of 46 instances being ambiguous. This frequent use suggests a significant emphasis on efficiency in the project requirements. However, the ambiguity in defining *kouritsuteki* in the context of operational processes and system performance can lead to differing vendor interpretations. Consequently, this could result in proposals that fail to meet the university's efficiency standards and operational expectations.
2. *Juubun*: The word *juubun* was the most detected word within the university RFPs. In addition, 47 out of 130 instances were found in ambiguous sentences. This frequent usage indicates a lack of precision in defining project requirements. Specifically, the use of *juubun* in the context of system performance and capabilities suggests that there is often no clear agreement on what level of performance is considered “sufficient.” This ambiguity can lead to varying interpretations by vendors, which may result in proposals that do not meet the university's actual needs.
3. *Juuyou*: The word *juuyou* was frequently detected in the university RFPs, especially in sections concerning data security and project milestones. In addition, seven out of nine instances were found in ambiguous sentences. This frequent usage highlights a lack of precision in defining what is considered “important.” Specifically, the use of *juuyou* without clear criteria can lead to varied interpretations of priority by vendors, potentially resulting in proposals that do not align with the university's critical needs.
4. *Sokuji*: The word *sokuji* appeared frequently in the university RFPs, particularly in sections related to response times and service delivery. Of the 36 instances detected, 17 were considered

Table 6. Frequency analysis result of RFPs for local government systems

RFP No.	Number of pages	Frequency
G1	21	0.14
G2	31	0.19
G3	25	0.04
G4	45	0.09
G5	14	0.35
G6	59	0.14
G7	12	0.33
G8	23	0.17
G9	33	0
G10	55	0.2

Table 7. Frequency analysis result of RFPs for library systems

RFP No.	Number of pages	Frequency
L1	12	0
L2	34	0.03
L3	34	0.06
L4	14	0.21
L5	9	0.22
L6	9	0.22
L7	9	0.33
L8	23	0.17
L9	33	0
L10	55	0.2

ambiguous. This high frequency indicates a strong focus on prompt action and immediate responses. However, the lack of specific definitions can lead to various interpretations by vendors, which might result in proposals that do not adequately address the university's urgency requirements, potentially compromising timely project execution.

Frequency Analysis

Next, we show the detailed results on the number of pages and the frequency of ambiguous sentences in the RFPs. Table 6, Table 7, Table 8, and Table 9 show the results for local government, library, university, and hospital RFPs. The “frequency” column indicates the number of ambiguous sentences divided by the number of pages in each RFP. Also, Figure 4, Figure 5, Figure 6, and Figure 7 visually present the results of frequency analysis. The total number of pages was 313 for the local government RFPs, 156 for the library RFPs, 429 for the university RFPs, and 808 for the hospital RFPs. Table 6 shows the frequency analysis results of the RFPs for local government systems. Table 7 shows the frequency analysis results of the RFPs for library systems. Table 8 shows the frequency analysis results of the RFPs for university systems. Table 9 shows the frequency analysis results of the RFPs for hospital systems.

Table 8. Frequency analysis result of RFPs for university systems

RFP No.	Number of pages	Frequency
U1	41	0.12
U2	16	0.06
U3	47	0.04
U4	14	0.21
U5	39	0.1
U6	35	0.22
U7	108	0
U8	17	0.05
U9	37	0
U10	75	0.08

Table 9. Frequency analysis result of RFPs for hospital systems

RFP No.	Number of pages	Frequency
H1	27	0.18
H2	19	0.21
H3	26	0.61
H4	36	0.38
H5	158	0.11
H6	478	0.05
H7	15	0.4
H8	19	0.47
H9	14	0.21
H10	16	0.19

Interestingly, in Table 6 and Figure 4, documents with more pages tended to have a relatively low frequency—the correlation coefficient was -0.40. For example, No. G5 and G7 were the two smallest page RFPs (14 and 12 pages, respectively), and they had the two largest frequencies (0.35 and 0.33). In contrast, large RFPs like G4, G6, and G10 had relatively smaller frequencies. This may be because the larger the size of the RFP, the more likely it was to be a significant and more expensive system and the more likely it was to have its ambiguities checked by a larger number of people.

The same characteristics were also observed in the library RFPs, with documents with fewer pages tending to have a higher ambiguous frequency—the correlation coefficient was -0.62.

For university RFPs, Table 8 and Figure 6 indicate that the trend was similar but less pronounced. Shorter documents such as U2 and U4 showed higher frequencies of ambiguous sentences, while longer documents like U7 and U9 had lower frequencies, indicating a correlation coefficient of -0.50.

Hospital RFPs also exhibited this pattern, with a correlation coefficient of -0.58.

RFPs such as H3 and H4, which had fewer pages, displayed higher frequencies of ambiguous sentences (0.61 and 0.38, respectively), compared to longer documents like H6 and H5, which had lower frequencies (0.05 and 0.11, respectively).

For further analysis, Figure 8 shows boxplots of the frequency of ambiguous sentences in short RFPs (fewer than 30 pages) and long RFPs (30 or more pages). The average frequencies were 0.232

Figure 4. Frequency analysis result of RFPs for government systems

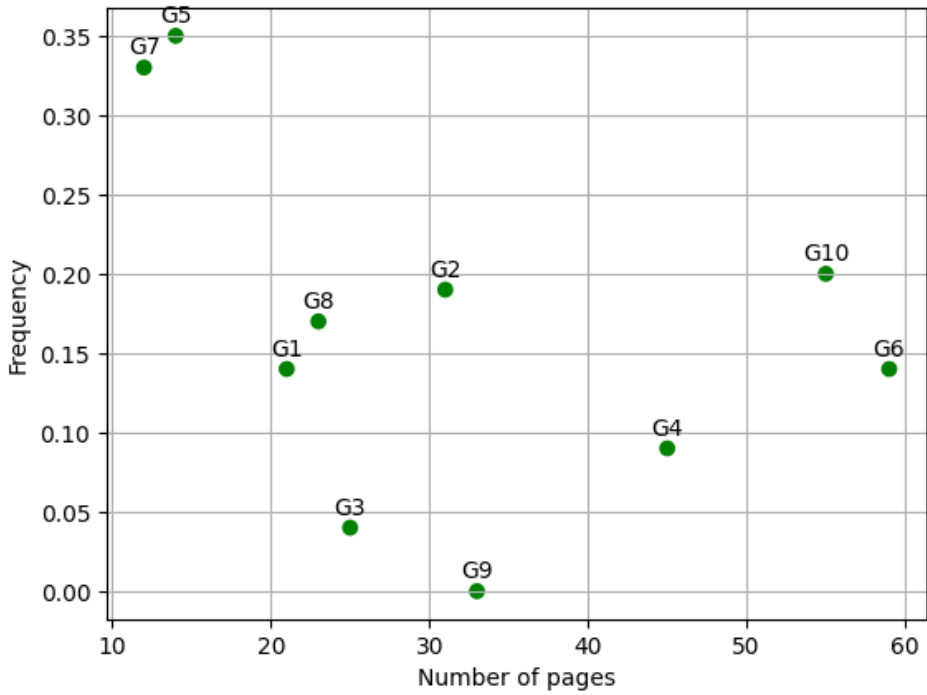


Figure 5. Frequency analysis result of library systems

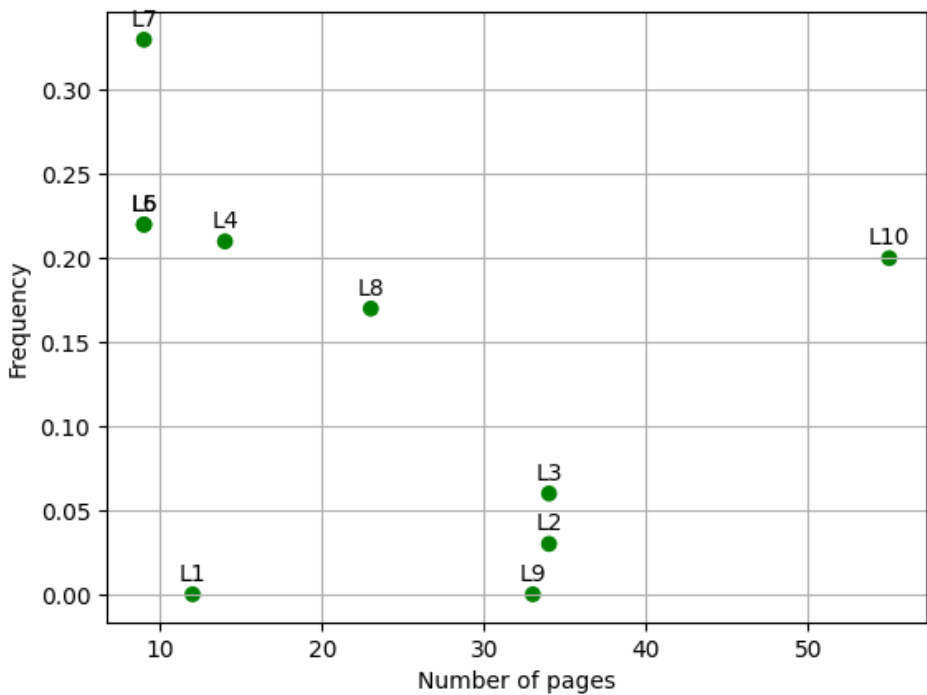


Figure 6. Frequency analysis result of RFPs for university systems

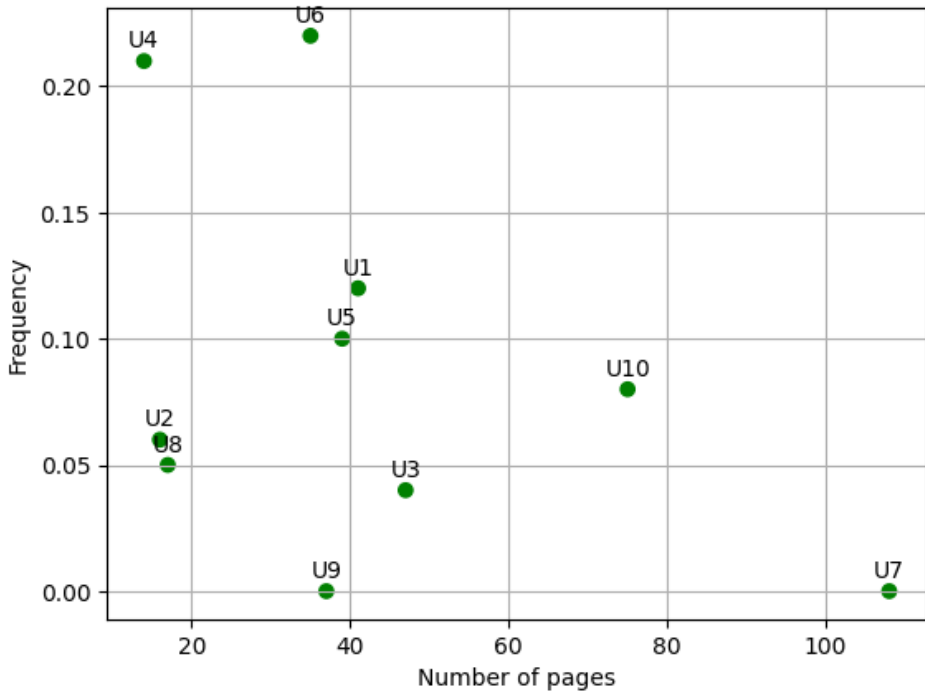


Figure 7. Frequency analysis result of RFPs for hospital systems

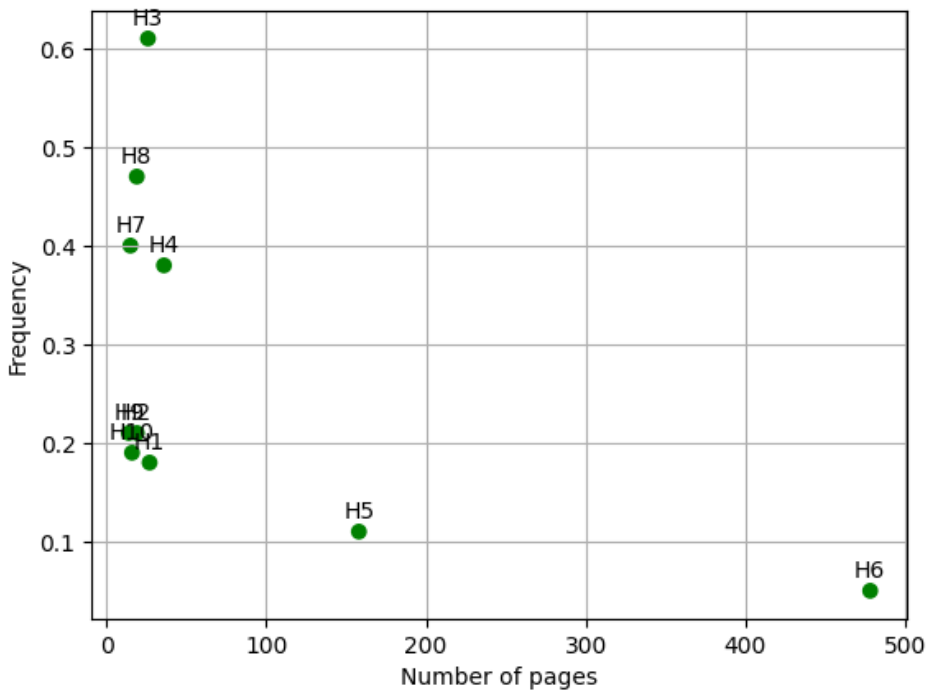
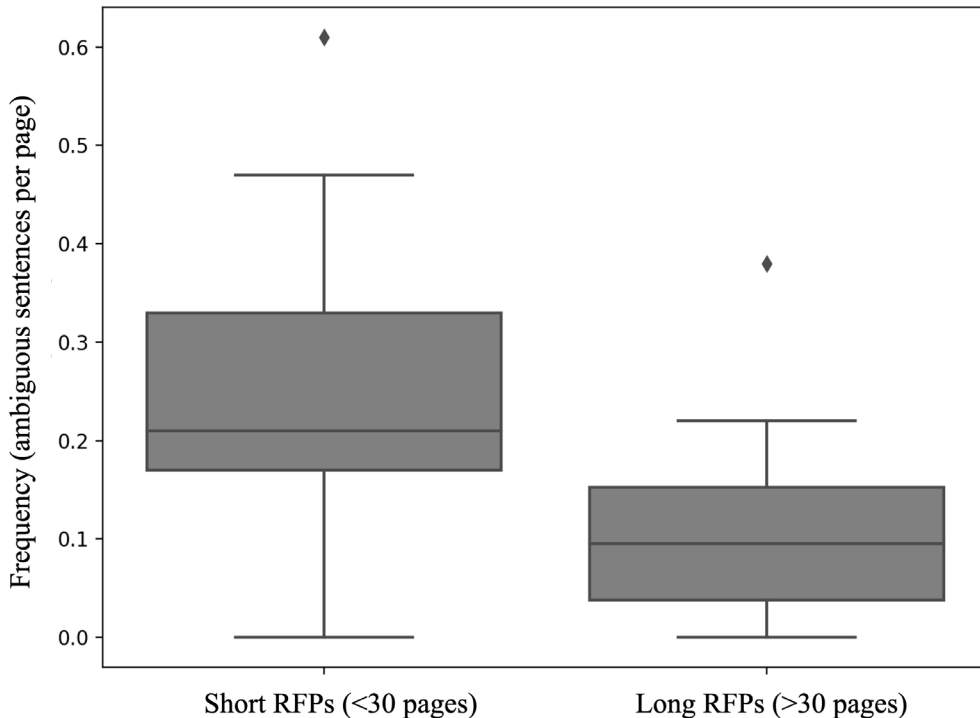


Figure 8. Frequency of ambiguous sentences in short RFPs and long RFPs



in short RFPs and 0.108 in long RFPs. We confirmed that short RFPs contained more ambiguous sentences per page than long RFPs. Therefore, practitioners should not take short RFPs lightly and should check them carefully for ambiguities.

SUMMARY

This paper targets RFPs of 40 systems of local governments, libraries, universities, and hospitals in Japan and quantitatively analyzes the ambiguity of sentences. The paper aimed to provide a collection of examples of ambiguous expressions based on the analysis. In our analysis, we manually classified sentences containing potentially ambiguous words into three groups: “ambiguous sentences,” “ambiguous but unproblematic for development,” and “unambiguous sentences.” We believe that the result of categorization helps vendors and users to understand the risk in RFP documents more clearly and resolve ambiguities in the early stages of a project. In particular, we found that *juubun* is the most frequently used ambiguous word in RFPs. Since word searching itself is easy and scalable, practitioners are encouraged to search for the word *juubun* in their RFP documents to see if there is any ambiguity. This process can be completed before releasing RFP documents to software engineers. Interestingly, RFPs with many pages tend to have relatively low percentages of ambiguous sentences. The analysis in this paper was limited by the authors’ subjective selection of ambiguous words and the classifications of ambiguity. In the future, we could introduce multiple examiners into the classification process to increase the reliability of the classification. It is also desirable to analyze more diverse types of RFP documents to increase the generality of the results.

COMPETING INTERESTS STATEMENT

The authors of this publication declare there are no competing interests.

FUNDING STATEMENT

This research was supported by the JSPS KAKENHI [grant number JP20H05706].

PROCESS DATES

September 30, 2024

Received: August 6, 2024, Revision: September 4, 2024, Accepted: September 18, 2024

CORRESPONDING AUTHOR

Correspondence should be addressed to Akito Monden (Japan, monden@okayama-u.ac.jp)

REFERENCES

- Berry, D. M. (2007). Ambiguity in natural language requirements documents. In *Innovations for requirement analysis. From stakeholders' needs to formal designs, Monterey workshop 2007, lecture notes in computer science*.
- Berry, D. M., Kamsties, E., & Krieger, M. M. (2003). From contract drafting to software specification: Linguistic sources of ambiguity, a handbook version 1.0. *University of Waterloo Technical Report*. <https://cs.uwaterloo.ca/~dberry/handbook/ambiguityHandbook.pdf>
- Japan Users Association of Information Systems. (2020). Software metrics analysis report 2020. *Japan Users Association of Information Systems*. <https://juas.or.jp/cms/media/2020/05/20swm.pdf>
- Kiyavitskaya, N., Zeni, N., Mich, L., & Berry, D. M. (2008). Requirements for tools for ambiguity identification and measurement in natural language requirements specifications. *Requirements Engineering, 13*(3), 207–239. DOI: 10.1007/s00766-008-0063-7
- Mavin, A., Wilkinson, P., Harwood, A., & Novak, M. (2019). Easy approach to requirements syntax (ears). In *Proceedings 17th international requirements engineering conference (RE'09)* (pp. 317–322).
- Mishra, S., & Sharma, A. (2019). On the use of word embeddings for identifying domain specific ambiguities in requirements. In *Proceedings 27th international requirements engineering conference workshops, 6th international workshop on artificial intelligence for requirements (AIRE)* (pp. 234–240). DOI: 10.1109/REW.2019.00048
- Nakayama, H. (2019). Ambiguous Japanese expressions, four NG patterns. *Nikkei XTech/Nikkei cloud first*. <https://xtech.nikkei.com/atcl/nxt/column/18/00611/022200001/?P=3>
- Osman, M. H., & Zaharin, M. F. (2018). Ambiguous software requirement specification detection: An automated approach. In *Proceedings 5th international workshop on requirements, engineering and testing (RET'18)* (pp. 33–40). DOI: 10.1145/3195538.3195545
- Pohl, K., & Rupp, C. (2015). *Requirements engineering fundamentals: A study guide for the certified professional for requirements engineering exam*. Rocky Nook.
- Saito, Y., Monden, A., & Matsumoto, K. (2012). Evaluation of non functional requirements in a request for proposal (RFP). In *Proceedings 2012 joint conference of the 22nd international workshop on software measurement and the 2012 seventh international conference on software process and product measurement* (pp. 106–111). DOI: 10.1109/IWSM-MENSURA.2012.23

APPENDIX

Examples of Ambiguous Sentences Discovered in the Analysis (Translated From Japanese)

Examples of ambiguous sentences containing *ippanteki* are the following:

1. “The functions for the common municipal mission-critical operations.”
2. “Describe the common status of efforts and achievements in core municipal operations.”
3. “If new servers are required, deliver equipment and other equipment in a configuration that enables common server operation and management.”
4. “The server should be configured to take into account commonly assumed security, such as stopping unnecessary services.”

Examples of ambiguous sentences containing *juudai* are the following:

1. “If a serious security problem occurs in the OS or other operating software, the corresponding security patch must be verified and applied immediately.”
2. “The ability to narrow down a potentially serious adverse drug reaction from the initial symptoms of a serious adverse drug reaction complained of by the patient, based on the drug history.”

Examples of ambiguous sentences containing *juubun* are the following:

1. “The management of a person with sufficient relevant skills and experience, such as a person who holds an information system-related qualification, etc.”
2. “Provide sufficient education and training by employees familiar with the operation of the system to explain the system and its operation.”
3. “Forecast the hardware capacity, required quality, and performance during the operation period and ensure sufficient capacity, quality, and performance in advance.”
4. “Sufficient air conditioning must be provided.”
5. “Sufficient consideration should be given to the screen structure so that users can operate the system without stress.”
6. “The system should be designed with sufficient consideration for the security of personal information and ease-of-data extraction.”
7. “The standard system should be in line with laws and national trends but should be able to reflect necessary modifications and additions while taking local characteristics and our requests into sufficient consideration and be based on the implementation of a package system.”
8. “Accuracy control of the transferred data shall be performed by the contractor, and sufficient accuracy shall be maintained in accordance with the instructions of the city.”
9. “Sufficient noise reduction measures must be taken for the equipment during operation.”
10. “In making the delivery, the supplier shall be able to demonstrate that the requirements specified in this procurement are sufficiently satisfied and that the quality of the product is satisfactory.”
11. “The system shall be designed with due consideration for operability and shall be sufficiently easy for beginners to understand through the help function.”

Examples of ambiguous sentences containing *kouritsuteki* are the following:

1. “Participants shall propose a safe, efficient, and least expensive method of handling the main output forms, from printing to sealing and shipping.”
2. “The system must have automatic operation functions, efficient input/output support, ease-of-batch processing, and efficiency.”
3. “Provide a function that enables the information management supervisor to efficiently manage user IDs and passwords.”
4. “For operations that are expected to undergo major system changes soon, measures should be devised to ensure that such changes can be implemented efficiently and at low cost.”
5. “To make suggestions on efficient system migration.”

6. “Each business system should have business support functions in addition to the basic functions so that staff can perform their tasks efficiently.”

7. “By introducing a system, student information will be managed centrally and administrative work will be standardized to improve student services and the administrative work of teaching staff through efficient administrative management.”

8. “To proactively make useful suggestions based on examples from other hospitals to ensure that the work of the hospital is efficient.”

9. “Efficient transmission of data and procedures used in medical information systems and medical imaging systems should be fully considered.”

10. “Provide advice on how to make the most efficient use of the new system at the request of the hospital.”

11. “Each business system should have business support functions in addition to the basic functions so that staff can perform their tasks efficiently.”

Examples of ambiguous sentences containing juuyou are the following:

1. “To enable users to use services such as reservations with peace of mind, a system that encrypts and communicates important information such as personal information using SSL or other data encryption is introduced to ensure secure communication.”

2. “Failure notifications by e-mail should also be implemented depending on the urgency and importance of the detection.”

3. “The ability to display important comments at the forefront of all screens.”

Examples of ambiguous sentences containing sokuji are the following:

1. “After the start of operation, until the normal operation of each function, etc. in normal operations can be confirmed, the installing business operator shall establish a support system that can respond immediately to requests from the city and shall witness the operation.”

2. “If immediate repair is not possible, the company shall provide spare equipment (e.g., terminals, printers, etc.) to deal with the fault as soon as possible and take other emergency measures. In the event that immediate repair is not possible, the company shall respond to the damage by providing spare equipment (e.g., terminals, printers, etc.) and other emergency measures.”

3. “If it is difficult to determine the cause of the problem and to take immediate restoration action, alternative options need to be offered.”

Toru Nakamichi is a master course student in the Graduate School of Natural Science and Technology, Okayama University, Japan. He received the BE degree in Computer Science and Systems Engineering from University of Miyazaki in 2023. His research interests include software measurement and analysis.

Hiroki Inayoshi received the B.E., M.E., and Ph.D. degrees from the Nagoya Institute of Technology, Japan, in 2018, 2021, and 2024, respectively. He has been an Assistant Professor with the Department of Computer Science, Okayama University, Japan, since 2024.

Kinari Nishiura received M.E. and Ph.D. degrees from Kyoto Institute of Technology in 2018 and 2021, respectively. Currently, he is an Assistant Professor at the Faculty of Information and Human Sciences, Kyoto Institute of Technology, Japan. He was an assistant professor at Okayama University from 2021 to 2024. His current research interests include software testing, software repository mining, and fault-prone module prediction.

Mariko Sasakura is an associate professor at the International Platform for Dryand Research and Education, Tottori University. Their specialty is human-computer interaction.

Akito Monden is a professor in the Graduate School of Natural Science and Technology at Okayama University, Japan. He received the BE degree (1994) in electrical engineering from Nagoya University, and the ME and DE degrees in information science from Nara Institute of Science and Technology (NAIST) in 1996 and 1998, respectively. His research interests include software measurement and analytics, and software security and protection. He is a fellow of JSSST, and a member of the IEEE, IEICE and IPSJ.