An Assistant Tool for Concealing Personal Information in Text

Tomoya Iwakura and Seishi Okamoto

Fujitsu Laboratories Ltd
Address: 1-1, Kamikodanaka 4-chome, Nakahara-ku, Kawasaki 211-8588, Japan
{iwakura.tomoya,seishi}@jp.fujitsu.com

Abstract. This paper presents an assistant tool for concealing personal information in text. Concealing personal information is one of the important roles for protecting privacy in disclosure of public documents, protection of accidental personal information leakages, and so on. However, concealing personal information is very time-consuming, because it is strongly depending on manpower. In order to alleviate tasks of concealing personal information, we have developed a graphical user interface (GUI) tool that has the following three characteristics: 1) Extracting candidates of personal information in text. 2) Presenting the candidates with colors indicating types of personal information. 3) Creating extraction rules for personal information from text including annotations of personal information. The experimental results on tasks of concealing person names in Japanese text showed that processing times of concealing personal names with candidates of person names were about 1.5 to 3.9 times faster than without candidates of person names.

1 Introduction

Protection of personal information is now one of the problems for disclosing documents, making ready personal information leakages, and so on. For example, Act on Access to Information Held by Administrative Organs [1] was effective fully in 1999 for providing for the right to request the disclosure of administrative documents. A problem of disclosing administrative documents is that documents have personal information. In order to disclose administrative documents, the Japanese low says that the protection of personal information in documents is obligation. Furthermore, the law related to protection of personal data applied to private company was fully executed in April of 2005 in Japan [2]. As the results, Japanese companies have to pay more attention than before.

One of the methods for protecting personal information is concealing personal information in documents, and it is necessary to disclose documents. However, concealing personal information is very time-consuming works, because the concealing process strongly depends on manpower. One of the crucial problems is finding personal information in text. Workers concealing personal information have to find personal information by reading all text in documents very carefully, because text written by natural language has no structures without processing texts.

M.J. Smith, G. Salvendy (Eds.): Human Interface, Part II, HCII 2007, LNCS 4558, pp. 38–46, 2007.
© Springer-Verlag Berlin Heidelberg 2007

In this paper, we present an assistant GUI tool for concealing personal information in text. Our GUI tool has the following characteristics

- Extracting personal information: Our GUI tool extracts several types of proper nouns and numeric expression with their classes, such as person names, locations, organizations, and so on, as personal information candidates in text.
- Presenting candidates of personal information: Our GUI tool presents candidates of personal information with colors indicating their classes, and users can conceal the candidates by just clicking them.
- Creating extraction rules for personal information: Our GUI tool creates extraction rules from text including annotations of personal information to be concealed.

This paper is organized as follows. In section 2, we present examples of concealing personal information in texts. In section 3, we present our assistant GUI tool for concealing personal information. In section 4, we report the experimental results on tasks of concealing personal information in Japanese text. In section 5, we discuss what our approach for extracting personal information differs from the other approaches. Finally, we conclude in section 6.

2 Concealing Personal Information in Text

Text data may include information identifying person, such as person names, dates of birth, addresses, and so on. To protect privacy, we must recognize person identifiable information and transform them to different forms not identifying him or her.

For example, a court decision in Japan is not disclosed without concealing personal information [2]. In order to disclose such documents, we have to conceal personal information in documents. The following example shows concealing personal information. If underlined person names are anonymized or concealed, we can disclose documents.

- Original text: An accused man, Yamada, was released after agreeing to testify against the others. Mr. Yamada testified that …
- Text including concealed person names: An accused man, **X**, was released after agreeing to testify against the others. Mr. **X** said that …

From the above example, we see that even if person names are concealed, we can understand most part of information except for the person names. However, concealing personal information are very time-consuming, because the concealing process strongly depends on man power. Thus, we think that assistant tools for concealing personal information are required.

3 Our Assistant Tool for Concealing Personal Information

We present our assistant tool for concealing personal information in text. Our tool consists of a Named Entity (NE) extractor for extracting personal information, a GUI for concealing personal information, and a rule learner for personal information extraction.

3.1 Extracting Personal Information by Using Context Information

One of the problems for concealing personal information in text is that we have to correctly discriminate the meaning of words. For example, most Japanese location names are used as Japanese family names, and we have to use context information to discriminate meanings of words.

In order to correctly find personal information in text, we apply Named Entity (NE) extraction technologies. NE extraction aims to identify word chunks , such as proper nouns, numerical expressions, and classify the word chunks into their classes, such as persons, locations, organizations, dates, and so on, in text . Table 1 shows NE examples. In this paper, we extract NEs as candidates of personal information. We are now focusing on Japanese. We briefly describe our personal information extraction methods for Japanese.

Table 1. NE examples

Numeric Expression

NE class	DATE	MONEY	PERCENT	TIME
Example	May 5th	200 JPY	100 %	10 PM

Proper Noun

NE class	ARTIFACT	LOCATION	ORGANIZATION	PERSON
Example	Novel Prize in Chemistry	Japan	Fujitsu	Jorge White

NE representation in text: We use a method that classifies words into NE chunk labels by using context information for extracting NEs, because NEs consist of one word or more than one word. We used Start/End (SE) representation. SE representation uses the five tags, S, B, I, E and O, for representing word chunks becoming NEs [3]. S means that the current word is a chunk consisting of only one word. B means in the start of a chunk consisting of more than one word. E is the end of a chunk consisting of more than one word. I is the inside of a chunk consisting of more than two words. O is the outside of any chunk.

We used the five tags with NE class labels as NE chunk labels for representing NEs in text. For example, the person names and the outside of person names in

…, Mr. Michael W. White said to Mr. Brow…

are represented by using NE chunk labels as follows.
…, Mr./O Michael/B-PERSON W./I-PERSON White/E-PERSON said/O to/O Mr./O Brown/S-PERSON…
B-PERSON, I-PERSON and E-PERSON indicates the beginning, inside, and end of a person name, respectively. S-PERSON indicates a person name consisting of only one word.

Classifying word into NE classes: To classifying words into NE labels, we use information of surrounding two words as context in addition to current word information as context information. We use the following word information as features for extracting NEs.

- Words: Asian languages, such as Japanese and Chinese, have no word boundary. In order to segment words from Japanese sentences, we use a morphological analyzer.
- Part of speech (POS) tags: We use POS tags of words tagged by a morphological analyzer, which is used for segmenting words.
- Character types: We use several types of character types, such as ‘Hiragana’, ‘Katakana’, ‘Chinese letter’, ‘capitalized alphabet’, ‘digit’, ‘sign’, and so on, and these combination.
- Dictionaries: We use dictionaries for augmenting features, if available. The dictionaries include person title lists, address dictionaries. In this experiment, we used an NE dictionary created from news articles and web pages by using several NE extractors [4].

For example, to classify the word of “W.” in the above sentence, word information of “Mr.”, “Michael”, “White” and “said” are used as context information to discriminate the NE class label of “W.”.

Classifying character into NE classes: Japanese NEs include a part of a word becoming begin or end of an NE sometimes, because Japanese words have no explicit word boundary. For example, the ”訪米(visit U.S.A)” in

田中使節団は訪米(Tanaka mission party visit U.S.A.)

dose not match with LOCATION ”*米*(U.S.A)” because this sentence is tokenized by a morphological analyzer as

”*田中*(Tanaka) / *使節*(mission) / *団*(party) / *は*(particle) / *訪米*(visited U.S.A)”,
”/” indicates a word boundary.

To solve this problem, we apply a character-unit-chunking-based NE extraction algorithm [5]. We use character-unit-chunking based NE extraction after classifying words into NE classes. To classifying characters into NE labels, we use information of surrounding two characters as context in addition to current character information.

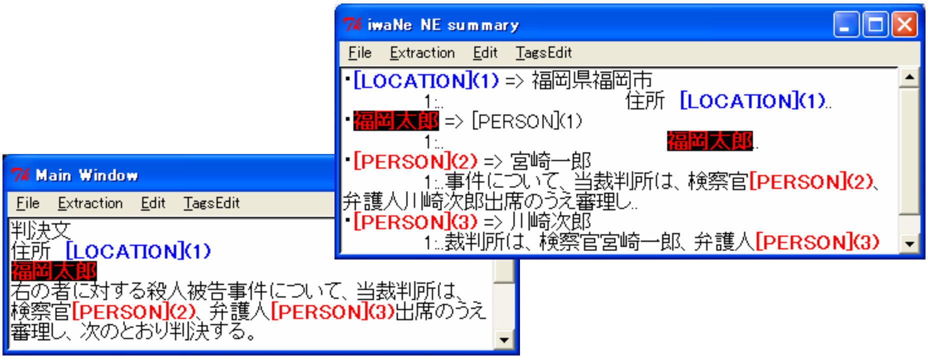
- Characters and words: We use words including characters within the current and surrounding two characters. Words are expressed with position identifiers to indicate where the character appears in words. We use B, I, E and S, which is the same as the SE representation
- POS tags: POS tags are annotated into words by a morphological analyzer. POS tags are expressed with position identifiers by SE representation, the same as characters’ ones.
- Character types: We use Kanji, Hiragana, Katakana, digit, lowercase alphabet, uppercase alphabet, and the others.
- NE labels of words: We use NE labels of words tagged by a word-unit-chunking-based NE extractor with stacking [6]. NE labels of words are also expressed with position identifiers by SE representation.

- NE labels of preceding extraction results: We used NE labels of previous character. We classify characters into NE labels in the direction of the end to begin of a sentence.

Each character is classified into NE labels represented by IOB2 [5] representation. IOB2 representation uses the three tags, B, I, and O. B means in the start of a chunk. I is the inside of a chunk consisting of more than two words. O is the outside of any chunk. For example, the above example is represented with character-based IOB 2 representation as the following.

”田/ORGANIZATION-B 中/ ORGANIZATION-I 使/ORGANIZATION-I 節 団 ORGANIZATION-I は/O 訪/O 米/LOCATION-B

ORGANIZATION-B and ORGANIZTION-I indicates a beginning and inside of an organization. LOCATION-B indicates a beginning of a location, and O indicates outside of organizations and locations.



・[LOCATION](1) => 福岡県福岡市
1:.. 住所 [LOCATION](1)..
・福岡太郎 => [PERSON](1)
1:.. 福岡太郎.
・[PERSON](2) => 宮崎一郎
1:..事件について、当裁判所は、検察官[PERSON](2)、弁護人川崎次郎出席のうえ審理し..
・[PERSON](3) => 川崎次郎
1:..裁判所は、検察官宮崎一郎、弁護人[PERSON](3)

判決文
住所 [LOCATION](1)
福岡太郎
右の者に対する殺人被告事件について、当裁判所は、
検察官[PERSON](2)、弁護人[PERSON](3)出席のうえ
審理し、次のとおり判決する。

Fig. 1. A snap shot of our tool: The left window is presenting whole text. The right window is presenting a summary. Concealed person names and a location name are represented by their classes and numbers. The person name focused by a mouse pointer is with black back ground color.

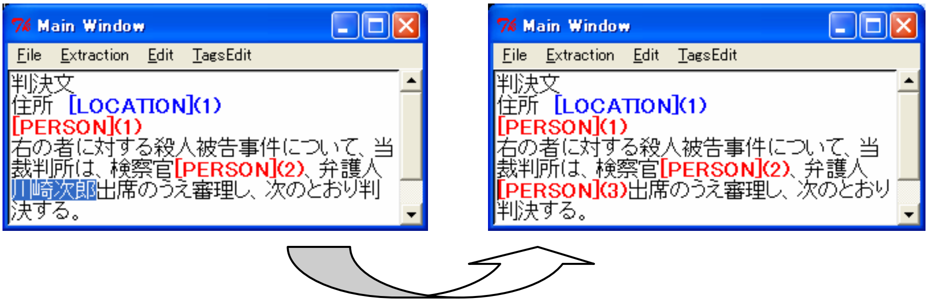
3.2 Our GUI for Concealing Personal Information

One of the annoying procedures for concealing personal information is finding personal information in text. Since text has no structure differing from structured data like Relational Database, workers concealing personal information have to read whole text for finding personal information. In order to alleviate such finding procedures, our GUI tool presents candidates of personal information as follows.

- Presenting candidates: Our tool presents candidates of personal information with colors corresponding to types of personal information. By presenting the candidates of personal information with the colors of their corresponding classes, workers can find personal information by their colors. The left window in Fig 1 shows a snapshot of our tool. The characters in bold font indicate personal information. The black background place indicates that a mouse pointer is focusing on a candidate of personal information. Users of our tool can conceal the candidates by just clicking them. [LOCATION](1), [PERSON](2) and [PERSON](3) in Fig. 1 indicate concealed person names and a location name.

- Creating summary: Our tool creates summaries for extraction results of personal information by collecting candidates of personal information with their surrounding words. By presenting summaries of extracted candidates of personal information, workers concealing personal information can conceal personal information without reading whole text as far as personal information extracted correctly. The right window in Fig. 1 shows a snapshot of a summary created by our tool. User can conceal personal information in text by clicking candidates of personal information presented in the summary.

Our tool also provides users shortcut keys for modifying wrong extractions. Fig. 2 shows a snapshot for a modification of a wrong extraction. For example, if a user defines 'key p with Alt-key' as annotating person tags, the user can annotate person tags by selecting the places to be annotated with a mouse and pressing 'key p with Alt-key' Our tool also provides users short cut for deleting wrong extractions. Since our tool provides users shortcut keys for modifying and deleting wrong extractions, users can edit wrong extractions only few steps.



判決文
住所 [LOCATION](1)
[PERSON](1)
右の者に対する殺人被告事件について、当
裁判所は、検察官[PERSON](2)、弁護人
川崎次郎出席のうえ審理し、次のとおり判
決する。

判決文
住所 [LOCATION](1)
[PERSON](1)
右の者に対する殺人被告事件について、当
裁判所は、検察官[PERSON](2)、弁護人
[PERSON](3)出席のうえ審理し、次のとおり
判決する。

Fig. 2. A snapshot of modifying missed extraction. Users can annotate tags of personal information by selecting the parts of personal information and pressing shortcut key corresponding to personal information classes.

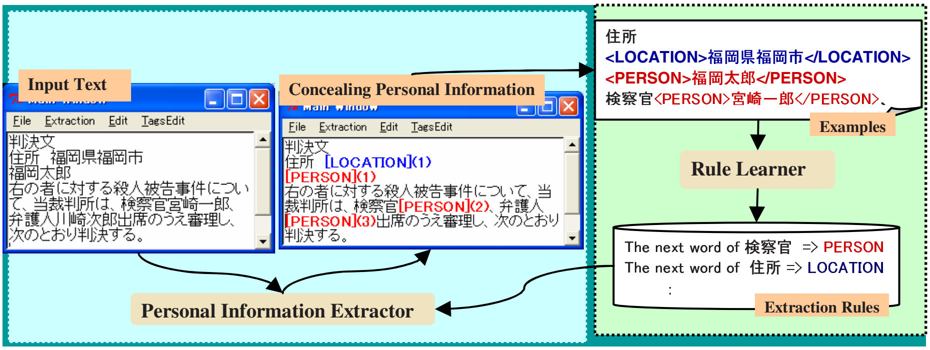
3.3 Creating Rules for Extracting Personal Information

There are two approaches for creating rules extracting personal information. First one is hand-crafted rule based approaches. The other is machine learning based approaches.

If we use hand-crafted rule based approaches for extracting personal information, we can revise extraction behaviors by adding new rules or modifying current rules. However, creation and modification of rules are very time consuming and it is necessary for rule developers to learn how to create rules. If we use machine learning based rule creation approaches, we can obtain extraction rules from text that personal information are concealed. However, it is difficult to control extraction behaviors.

The two approaches have different benefits and drawbacks. However, we apply a machine learning based approach, because machine learning based approaches can crate rules from new text including annotations of personal information created in routine work.

We use a boosting algorithm as the machine learning algorithm. Please refer to the paper [7] for the detailed explanation of boosting algorithms. Our rule learner creates rules extracting personal information from training data including annotations of personal information tagged with their class names. Fig 3 shows an example of a rule generation from data created in a concealing process.


Input Text
File Extraction Edit TagsEdit
Concealing Personal Information
File Extraction Edit TagsEdit
Examples
Rule Learner
Extraction Rules
Personal Information Extractor

判決文
住所　福岡県福岡市
福岡太郎
右の者に対する殺人被告事件について、当裁判所は、検察官宮崎一郎、弁護人川崎次郎出席のうえ審理し、次のとおり判決する。

判決文
住所　[LOCATION](1)
[PERSON](1)
右の者に対する殺人被告事件について、当裁判所は、検察官[PERSON](2)、弁護人[PERSON](3)出席のうえ審理し、次のとおり判決する。

住所
<LOCATION>福岡県福岡市</LOCATION>
<PERSON>福岡太郎</PERSON>
検察官<PERSON>宮崎一郎</PERSON>、

The next word of 検察官 => PERSON
The next word of 住所 => LOCATION
:

Fig. 3. A snapshot of a rule generation from data created in a concealing process: The left hand side is a concealing process. The right hand side is a sample of rule generation.

4 Experimental Results

We evaluated effectiveness of our tool by measuring times of concealing person names in Japanese text. We used the following data for our evaluation.

- A pseudo court decision: The pseudo court decision has 29 types of person names. The total number is 45. The size of the data is about 12KB. The person names appear in text all the here and there.
- A minute in a company: The minute has 25 types of person names. The total number is 76. The size of the data is about 12 KB. This minute is well organized compared to the pseudo court decision. For example, all the persons attending the meeting are presented at the beginning of the minute.

We conducted the evaluations as follows.

- Concealing person names: When we had candidates of person names presented by our tool, we used them at first for concealing person names. After that, we read whole text to find person names not concealed. If we did not have any candidates of person names, we concealed person names by reading whole text and using shortcut keys of our tool.
- Checking missed person names: We have done this phase for checking person name not concealed. We read whole text again.

We conducted all the tests using candidates before the tests without candidates for evaluating performance of our tool on fully unseen data. We used the rules extracting person names generated from training data including 7818 person names. The training data includes no text of data set for evaluation. We used MeCab, a morphological analyzer, for segmenting words from text[1]. We measured extraction accuracy of our GUI tool by using Recall, Precision, and F-measure, defined as follows.

Recall = NUM / (the number of correct person names)
Precision = NUM / (the number of person names extracted by our GUI tool)
F-measure = 2 * Recall * Precision / (Recall + Precision),

where NUM is the number of person names correctly extracted.

Table 2 shows the extraction accuracy of our tool. Our tool extracts person names more than 90 % F-measures.

Table 2. Our GUI tool performance on our evaluation data

Data Set	# of person names (# of types)	Precision	Recall	F-measure
A pseudo court decision	45 (29)	97.44	84.44	90.48
A minute in a company	76 (25)	98.65	96.05	97.33

Table 3 shows that the experimental results of concealing person names with two subjects for each task. The results showed that our tool enabled the subjects to conceal person names about 3.3, 3.9, 1.9 and 1.5 times faster than without using candidate of person names.

We think that the reasons are as follows. The first is that times for finding person names are greatly reduced by using candidates. The other is that times for concealing person names are also reduced because we can conceal them by just clicking them.

Table 3. Experimental results of concealing personal names

A pseudo court decision

Subjects	Manual operation (a)	With Our tool (b)	Improvement((a)/(b))
A	938 seconds	243 seconds	3.9 times
B	706 seconds	218 seconds	3.3 times

A minute in a company

C	486 seconds	245 seconds	1.9 times
D	578 seconds	384 seconds	1.5 times

5 Related Works

There are three approaches widely used for extracting personal information. The first approach is dictionaries based personal information extraction [8]. This approach has the advantage that users can control extraction behaviors by just adding new entries of

[1] http://mecab.sourceforge.net/

personal information into dictionaries. However, the approach can not discriminate meanings of words having meanings more than one. Furthermore, the approach tends to output many noises.

The second is hand-crafted rule based extraction [9]. This approach has the advantage that users can control extraction behaviors by creating and modifying rules. However, the creation of rules and the managements of rules require skills.

The third is machine-learning based extraction adopted by our tool. This approach crates extraction rules by machine learning algorithms. Although the approach has difficulties to control extraction behaviors, this approach has the advantage that users can crate new rules by creating data including annotations of personal information. Furthermore, this approach provides users to enhance extraction performance by incorporating data including concealed personal information created in daily work.

6 Conclusion

This paper has presented our GUI tool to help concealing personal information in text. Our GUI tool finds personal information in text by using a Japanese Named Entity extractor discriminating meanings of words. We have experimentally demonstrated that our GUI tool contributes to improve processing times of concealing personal information in text about 1.5 to 3.9 times.

We think that further improvements of concealing personal information will be realized by enhancing performances of personal information extractors. Future work should consider methods to combine approaches based on dictionaries, handcrafted rules, and machine learning algorithms, for further improvement.

References

[1] Act on Access to Information Held by Administrative Organs (Act No. 42 of 1999) web page. http://www.soumu.go.jp/english/gyoukan/060516_03.html

[2] Web page for disclosed court decision (in Japanese). http://www.courts.go.jp/saisinhanrei.html

[3] Uchimoto, K., Ma, Q., Murata, M., Ozaku, H., Utiyama, M., Isahara, H.: Named entity extraction based on a maximum entropy model and transformation rules. In: Proc. of the ACL 2000, pp. 326–335 (2000)

[4] Iwakura, T., Okamoto, S.: Improving Named entity extraction accuracy using unlabeled data and several extractors. CICLing (to appear 2007)

[5] Asahara, M., Matsumoto, Y.: Japanese named entity extraction with redundant morphological analysis. In: Proc. of HLT-NAACL 2003 (2003)

[6] Wolpert, D.H.: Stacked generalization. Neural Networks 5, 241–259 (1992)

[7] Schapire, R.E., Singer, Y.: BoosTexter: A boosting-based system for text categorization. Machine Learning 39(2/3), 135–168 (2000)

[8] Dehenken Web page (in Japanese): http://www.dehenken.co.jp/products/products-03/products-kansalib01.html

[9] Takemoto, Y., Fukushima, T., Yamada, H.: A Japanese Named Entity Extraction System Based on Building a Large-scale and High quality Dictionary and Pattern-matching Rules (In Japanese). IPSJ Journal 42(6), 158–159 (2001)