JAMIA PrePrint: Accepted Article. Published June 27, 2007 as doi:10.1197/jamia.M2444

Evaluating the State-of-the-Art in Automatic De-identification

Özlem Uzuner, Ph.D., University at Albany, SUNY

Yuan Luo, University at Albany, SUNY

Peter Szolovits, Ph.D., MIT CSAIL

For reprints, contact:
Özlem Uzuner
University at Albany, SUNY
Draper 114A
135 Western Ave.,
Albany, NY 12222
(518) 442-4687
ouzuner@albany.edu

Abstract

As a part of the i2b2 (Informatics for Integrating Biology to the Bedside) project, authors organized a Natural Language Processing (NLP) challenge on automatically removing private health information (PHI) from medical discharge records. This manuscript provides an overview of this "de-identification challenge", describes the data and the annotation process, explains the evaluation metrics, discusses the nature of the systems that addressed the challenge, and analyzes the results of received system runs.

The challenge data consisted of discharge summaries drawn from the Partners Healthcare system. Authors prepared this data for the challenge by replacing authentic PHI with synthesized surrogates. To focus the challenge on non-dictionary-based de-identification methods, the data was enriched with out-of-vocabulary PHI surrogates, i.e., made up names. The data also included some PHI surrogates that were ambiguous with medical non-PHI terms.

A total of seven teams participated in the challenge. Each team submitted up to three system runs, for a total of sixteen submissions. The authors used precision, recall, and F-measure to evaluate the submitted system runs based on their token-level and instance-level performance on the ground truth.

The systems with the best performance scored above 98% in F-measure for all categories of identifiers. Most out-of-vocabulary PHI could be identified accurately. However, identifying ambiguous PHI proved challenging.

The performance of systems on the test data set is encouraging. Future evaluations of these systems will involve larger data sets from more heterogeneous sources.

Keywords: Natural language processing, automatic de-identification, clinical data, medical discharge summaries.

1 Introduction

Clinical records can be an important source of information for clinical and laboratory researchers alike [[1], [2], [3]]. However, most of the information in these records is in the form of free text and extracting useful information from them requires automatic processing (e.g., index, semantically interpret, and search) [[4], [5], [6], [7], [8], [9]]. A prerequisite to the distribution of clinical records outside of hospitals, be it for Natural Language Processing (NLP) or medical research, is de-identification.

De-identification ensures the removal of all personally identifying private health information (PHI) from the records. Paragraph 164.514 of the Administrative Simplification Regulations promulgated under the Health Insurance Portability and Accountability Act (HIPAA) states that for data to be treated as de-identified, it must clear one of two hurdles.

1. An expert must determine and document "that the risk is very small that the information could be used, alone or in combination with other reasonably available information, by an anticipated recipient to identify an individual who is a subject of the information."
2. Or, the data must be purged of a specified list of 17 categories of possible identifiers relating to the patient or relatives, household members and employers, and any other information that may make it possible to identify the individual [10]. Many institutions consider the clinicians caring for a patient and the names of hospitals, clinics, and wards to fall into this final category because of the heightened risk of identifying patients from such information [[11], [12]].

Although technologies for automatic de-identification exist [[11], [13], [14], [15], [16], [17]], the development of such systems is limited by a chicken and egg problem: systems cannot be effectively developed without access to clinical records, but clinical records cannot be readily made available for research (even for de-identification) without being de-identified. To surpass this limitation, within the Informatics for Integrating Biology and the Bedside (i2b2) project [18], the authors created a data set for evaluating two NLP challenge questions:

Task 1: automatic de-identification of clinical records, i.e., de-identification challenge, and

Task 2: identification of the smoking status of patients, i.e., smoking challenge.

The two challenges were run as shared-tasks. In other words, all participating teams were asked to study the two challenge questions on the same data and their results were compared to the same gold standard. The challenges were accompanied by a workshop, co-sponsored by i2b2 and the American Medical Informatics Association (AMIA) [19], which provided a forum for the presentation and discussion of resulting research findings.

A total of 18 teams participated in the two challenges: 7 in de-identification and 11 in smoking. This manuscript describes the data and evaluation metrics of the de-identification challenge, provides an overview of the systems that participated in this challenge, and tries to draw conclusions about the state-of-the-art and the future of de-identification. The details of the smoking challenge can be found in Uzuner, et al. [20] .

2 Problem Definition and Related Work

De-identification resembles traditional Named Entity Recognition (NER). Outside of the biomedical domain, the Message Understanding Conference (MUC) [21] has served as a venue for the evaluation of NER systems. The Named Entity tasks of MUC tested systems for their ability to recognize organizations, persons, locations, dates and times (relative and absolute), currency/percentage values, artifacts (e.g., television), anaphoric references (e.g., the plane), etc. Following MUC, the National Institute of Standards and Technology (NIST) organized the Information Extraction—Named Entity task [22] and the Automatic Content Extraction (ACE) tasks [23]. Both of NIST's tasks and MUC's NER evaluations were run on newswire text.

In the biomedical domain, Critical Assessment of Information Extraction Systems in Biology (BioCreAtIvE) [24] enabled studies on automatic identification of biomedical entities, e.g., genes and proteins, in text drawn from various databases including PubMed [25], FlyBase [26], Mouse Genome Informatics [27], Saccharomyces Genome Database [28], and Swiss-Prot [29]. In addition to BioCreAtIvE, the TREC Genomics Track [30] provided a forum for the evaluation of extraction and retrieval of biomedical texts.

De-identification differs from NER in its focus on clinical records. The goal of de-identification, as defined in this challenge, is to find and remove PHI from medical records while protecting the integrity of the data as much as possible. De-identification algorithms need to remove such PHI from medical records in the presence of:

- Ambiguities: PHI and non-PHI can lexically overlap, e.g., "Huntington" can be the name of a disease (non-PHI) as well as the name of a person (PHI).

- Out-of-vocabulary PHI: PHI can include misspelled and/or foreign words that cannot be found in dictionaries.

Many approaches to traditional NER use dictionaries and gazetteers of person, organization, and location names. Ambiguous and out-of-vocabulary PHI reduce the contribution of dictionaries and gazetteers to de-identification and emphasize the value of studying context and language.

3 Data and Annotation

The data for the de-identification challenge came from Partners Healthcare and included solely medical discharge summaries. We prepared the data for the challenge by annotating and by replacing all authentic PHI with realistic surrogates.

3.1 PHI Categories, Ambiguous and Out-of-Vocabulary PHI

We found that, out of the 17 textual PHI categories listed by HIPAA, only six appeared in our data. As described in the Introduction, we added two additional categories, doctor and hospital, resulting in eight PHI categories in the data set. We defined the resulting PHI categories as follows (square brackets enclose PHI):

- Patients: includes the first and last names of patients, their health proxies, and family members. It excludes titles, such as Mrs., e.g., "Mrs. [Mary Joe] was admitted…".
- Doctors: refers to medical doctors and other practitioners mentioned in the records. For transcribed records, it includes the transcribers' names and initials. It excludes titles, such as Dr. and MD, e.g., "He met with Dr. [John Bland], MD".
- Hospitals: marks the names of medical organizations and of nursing homes where patients are treated and may also reside. It includes room numbers of patients, and buildings and floors related to doctors' affiliations, e.g., "The patient was transferred to [Gates 4]".
- IDs: refers to any combination of numbers, letters, and special characters identifying medical records, patients, doctors, or hospitals, e.g., "Provider Number: [12344]".
- Dates: includes all elements of a date except for the year. HIPAA specifies that years are not considered PHI. Therefore, we exclude them from this category.
- Locations: includes geographic locations such as cities, states, street names, zip codes, building names, and numbers, e.g., "He lives in [Newton]".
- Phone numbers: includes telephone, pager, and fax numbers.

- Ages: includes ages above 90. HIPAA dictates that ages above 90 should be collected under one category, 90+, and should be marked as PHI. Ages below 90 can be left as is.

Given the above definitions, we marked the authentic PHI in the records in two stages. In the first stage, we used an automatic system [31]. In the second stage, we validated the output of the automatic system manually. Three annotators, including undergraduate and graduate students and a professor, serially made three manual passes over each record. They marked and discussed the tokens they disagreed on and finalized these tags after discussion.

Next, we replaced the authentic PHI with realistic surrogates. For dates, IDs, phone numbers, and ages, we generated surrogates by replacing each digit with a random digit and each letter by a random letter; we respected the exact format of the authentic PHI and assured that synthesized dates remained valid dates. For patients, doctors, locations, and hospitals, we created surrogates by permuting the syllables of existing names from dictionaries such as the U.S. Census Bureau names database. We observed the orthography of the PHI in the authentic corpus and ensured that the surrogate PHI resembled the authentic PHI in their use of abbreviations, middle initials, full names, capitalization, etc. We made an effort to replace all proper noun references to a given entity with the same surrogate or orthographic variants of the same surrogate (as guided by the authentic data) so as to preserve co-reference information, e.g., replaced “John Smith” with “Jane Doe” and “J. Smith” with “J. Doe” consistently and throughout. We preserved relative time information by offsetting all dates in a record by the same number of days.

We allowed the generation of surrogates that could be found in dictionaries and did not make any effort to eliminate such surrogates from the data. Still, most of the generated surrogates could not be found in dictionaries, e.g., “Valtawnprinceel Community Memorial Hospital” and “Girresnet, Diedreo A”. To test systems on ambiguous PHI, we replaced some of the randomly-generated surrogate patient and doctor names with medical terms, such as diseases, treatments, and medical test names. We thus created ambiguities among the PHI and the non-PHI within specific records and within the complete corpus.

The Institutional Review Boards of Partners Healthcare, Massachusetts Institute of Technology, and the State University of New York approved the challenge and the data preparation process. In all, we annotated 889 records. 669 of these records were used for training. The remaining 220 were used for testing. The distributions of PHI categories in the complete corpus and in the training and test sets are shown in Table 1. Instances mark the number of PHI phrases in each category while tokens mark the number of words in each PHI category.

Table 1: Distribution of Instances and Tokens in the Complete Challenge Corpus, and in the Training and Test Corpora.

PHI Category	Complete Corpus		Training Data		Test Data	
	Instances	Tokens	Instances	Tokens	Instances	Tokens
Non-PHI	-	444127	-	310504	-	133623
Patients	929	1737	684	1335	245	402
Doctors	3751	7697	2681	5600	1070	2097
Locations	263	518	144	302	119	216
Hospitals	2400	5204	1724	3602	676	1602
Dates	7098	7651	5167	5490	1931	2161
IDs	4809	5110	3666	3912	1143	1198
Phone Numbers	232	271	174	201	58	70
Ages	16	16	13	13	3	3

Table 2 shows the distribution of out-of-vocabulary PHI in the complete challenge corpus and its subsets used for training and testing, after the authentic PHI has been replaced with surrogates and after ambiguities have been introduced. Note that the concept of being out-of-vocabulary does not apply to IDs, dates, ages, and phone numbers. Also note that out-of-vocabulary PHI tokens constitute 20 to 35% of each of patients, doctors, locations, and hospitals in the authentic corpus; the percentages of PHI instances that include one or more out-of-vocabulary tokens in these PHI categories range from 30 to 65%. The exaggerated percentages of out-of-vocabulary PHI in the challenge corpus allow us to emphasize the importance of successfully de-identifying such PHI. The random split of the complete challenge corpus into training and test sub-corpora does not guarantee similar distributions of out-of-vocabulary PHI in the two sub-corpora.

Table 2: Out-of-Vocabulary PHI in the Complete Authentic and Challenge Corpora, as well as the Challenge Training and Test Sub-Corpora.

PHI Category	Authentic Corpus		Complete Challenge Corpus		Challenge Training Sub-Corpus		Challenge Test Sub-Corpus	
	Instances (%)	Tokens (%)	Instances (%)	Tokens (%)	Instances (%)	Tokens (%)	Instances (%)	Tokens (%)
Patients	34	20	80	73	85	76	66	64
Doctors	43	26	85	67	86	67	82	67
Locations	30	22	69	56	72	60	64	50
Hospitals	65	35	91	49	91	49	91	50

As a part of the de-identification challenge, we aimed to evaluate the systems' abilities to resolve ambiguities of PHI with non-PHI. Many de-identification methods aim to remove all PHI from the records without paying much attention to non-PHI that may also get removed in the process. We believe that retaining the key medical concepts, such as diseases, is important for protecting the integrity of the data. Retaining such information in the records enables the use of de-identified records for studies on drug interactions, quality of service studies, etc.

Table 3: Ambiguity between PHI and non-PHI in Complete Authentic and Challenge Corpora, and in the Challenge Training and Test Sub-Corpora.

PHI	Ambiguity Scope	Complete Authentic Corpus		Complete Challenge Corpus		Challenge Training Sub-Corpus		Challenge Test Sub-Corpus	
		Instances (%)	Tokens (%)	Instances (%)	Tokens (%)	Instances (%)	Tokens (%)	Instances (%)	Tokens (%)
Patients	In record	2.04	1.09	8.36	4.48	6.14	3.15	14.7	8.96
	In corpus	22.7	3.73	17.9	9.71	14.8	7.72	22.5	13.7
Doctors	In record	0.64	0.31	12.9	6.30	11.5	5.50	16.5	8.44
	In corpus	26.9	4.67	29.9	15.1	27.5	13.3	29.2	15.0
Locations	In record	4.55	2.18	0.76	0.39	0.69	0.33	0.84	0.46
	In corpus	43.9	18.4	20.1	10.6	18.1	8.61	14.3	8.33
Hospitals	In record	29.8	18.9	32.0	15.5	28.2	14.2	41.9	18.2
	In corpus	46.6	42.2	50.9	37.2	47.9	38.2	54.7	33.3
Dates	In record	0.01	0.01	0.42	0.42	0.37	0.36	0.57	0.56
	In corpus	1.18	0.09	0.75	1.10	0.74	1.13	0.78	0.88
IDs	In record	0.00	0.00	0.00	0.00	0.00	0.00	0.57	0.00
	In corpus	0.25	0.10	0.06	0.06	0.08	0.08	0.78	0.08
Phone numbers	In record	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
	In corpus	1.29	0.37	0.43	0.37	0.57	0.50	0.09	0.00
Ages	In record	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
	In corpus	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00

Table 3 shows the percentage of ambiguity of each PHI category with non-PHI in the authentic and in the challenge corpora. The column marked "ambiguity scope" indicates whether the PHI is ambiguous with non-PHI within the same record or within the corpus. Note that out-of-vocabulary surrogate generation eliminated some of the ambiguities naturally present in locations and hospitals in the authentic corpus while our efforts to introduce ambiguity into the challenge data increased the ambiguity in patients and doctors. Also note that due to random split of the complete challenge corpus into training and test corpora, these two corpora exhibit different levels of ambiguity from each other. The columns marked tokens indicate the percentage of tokens that are ambiguous; the columns marked instances indicate the percentage of instances that include one or more ambiguous tokens.

Of the ambiguous PHI instances in our challenge corpus, 54% included ambiguities with medical terms; correspondingly, 41% of the ambiguous tokens were ambiguous with medical terms. In comparison, ambiguity with medical terms was observed in 22% of the ambiguous instances in the authentic corpus; 17% of the ambiguous tokens in this corpus were ambiguous with medical terms.

Table 4: Token-Level Inter-PHI Ambiguity in the Authentic and Complete Challenge Corpora.

X Y	Corpus	Non-PHI	Patients	Doctors	Locations	Hospitals	Dates	IDs	Phone Numbers	Ages
Non-PHI (%)	Authentic	-	0.56	0.96	0.86	3.12	0.03	0.01	0.00	0.00
	Challenge	-	3.17	8.27	0.50	1.18	0.41	0.00	0.00	0.00
Patients (%)	Authentic	3.73	-	35.5	0.86	5.28	0.00	0.00	0.00	0.00
	Challenge	9.71	-	18.0	1.61	0.92	0.00	0.00	0.00	0.00
Doctors (%)	Authentic	4.67	24.8	-	0.61	4.67	0.01	1.72	0.00	0.00
	Challenge	15.1	10.31	-	2.04	3.26	0.13	0.13	0.00	0.00
Locations (%)	Authentic	18.4	6.00	11.3	-	28.9	0.73	0.00	0.00	0.00
	Challenge	10.6	3.66	8.29	-	5.39	0.00	0.00	0.00	0.00
Hospitals (%)	Authentic	42.2	1.61	5.48	26.1	-	0.00	0.04	0.00	0.00
	Challenge	37.2	1.23	15.0	3.07	-	2.38	0.08	0.00	0.00
Dates (%)	Authentic	0.09	0.00	0.94	0.04	0.00	-	0.00	0.00	0.00
	Challenge	1.10	0.00	0.65	0.00	0.07	-	0.00	0.00	0.00
IDs (%)	Authentic	0.10	0.00	0.06	0.00	0.02	0.00	-	0.00	0.00
	Challenge	0.06	0.00	0.06	0.00	0.02	0.00	-	0.00	0.00
Phone Numbers (%)	Authentic	0.37	0.00	0.00	0.00	0.00	0.00	0.00	-	0.00
	Challenge	0.37	0.00	0.00	0.00	0.00	0.00	0.00	-	0.00
Ages (%)	Authentic	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	-
	Challenge	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	-

Note that ambiguity between different PHI categories, i.e., inter-PHI ambiguity, also exists. Table 4 shows the level of inter-PHI ambiguity in the authentic corpus and in the complete challenge corpus respectively. The Y dimension indicates the actual category of the PHI; the X dimension indicates the categories that they are ambiguous with, e.g., in the authentic corpus, 0.56% of actual non-PHI are ambiguous with patients. As can be seen, patients, doctors, locations, and hospitals in the challenge corpus are highly ambiguous with each other. However, Table 4 reveals that, for most pairs of PHI categories, the level of ambiguity is lower in the challenge corpus than in the authentic corpus. This is an artifact of the surrogate generation process. Given our focus on separating PHI from non-PHI, we did not inject extra inter-PHI ambiguities into the challenge corpus. Yet, differentiating between categories of PHI can be important for some applications that need to build on de-identified records, e.g., did the patient or the doctor report a particular fact? Therefore, despite underemphasizing this task, the de-identification challenge asks to retain the distinction between PHI categories as much as possible. Table 5 shows the distribution of inter-PHI ambiguity in the challenge training and test corpora respectively. As before, the random split of the complete challenge corpus into training and test does not guarantee similar distributions of inter-PHI ambiguities in these two sub-corpora.

Table 5: Token-Level Inter-PHI Ambiguity in the Challenge Training and Test Corpora.

X Y	Corpus	Non-PHI	Patients	Doctors	Locations	Hospitals	Dates	IDs	Phone Numbers	Ages
Non-PHI (%)	Training	-	1.50	5.60	0.32	0.86	0.41	0.00	0.00	0.00
	Test	-	2.07	5.83	0.16	0.79	0.41	0.00	0.00	0.00
Patients (%)	Training	7.72	-	14.9	0.37	0.45	0.00	0.00	0.00	0.00
	Test	13.7	-	15.7	2.24	1.00	0.00	0.00	0.00	0.00
Doctors (%)	Training	13.7	7.63	-	0.66	2.13	0.13	0.09	0.00	0.00
	Test	15.0	5.63	-	1.43	1.34	0.14	0.00	0.00	0.00
Locations (%)	Training	8.61	1.32	5.30	-	4.64	0.00	0.00	0.00	0.00
	Test	8.33	2.78	5.09	-	3.24	0.00	0.00	0.00	0.00
Hospitals (%)	Training	38.2	1.47	14.3	1.83	-	2.67	0.03	0.00	0.00
	Test	33.3	0.12	2.25	0.37	-	0.00	0.00	0.00	0.00
Dates (%)	Training	1.13	0.00	0.67	0.00	0.09	-	0.00	0.00	0.00
	Test	0.88	0.00	0.60	0.00	0.00	-	0.00	0.00	0.00
IDs (%)	Training	0.08	0.00	0.05	0.00	0.03	0.00	-	0.00	0.00
	Test	0.08	0.00	0.00	0.00	0.00	0.00	-	0.00	0.00
Phone Numbers (%)	Training	0.50	0.00	0.00	0.00	0.00	0.00	0.00	-	0.00
	Test	0.00	0.00	0.00	0.00	0.00	0.00	0.00	-	0.00
Ages (%)	Training	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	-
	Test	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	-

Dictionaries are often used as a source of information in many de-identification systems. Because of the randomly generated and/or ambiguous surrogate PHI, we expected dictionary-based approaches to be less successful in separating PHI from non-PHI in the challenge corpus than with the authentic data. We hypothesize that if the systems shed light on information that can help identify ambiguous and/or out-of-vocabulary PHI, they can easily be complemented with dictionaries in order to identify the more mainstream in-vocabulary and unambiguous PHI.

3.2 Annotation Format

For the challenge, the data were tokenized, broken into sentences, and converted into XML [32]. Each record was enclosed in <RECORD> tags and was identified by a random, unique ID number. The text of each record was enclosed within <TEXT> tags. Each PHI instance was enclosed in <PHI> tags. The TYPE attribute of the opening <PHI> tag marked the category of each PHI. An excerpt from a sample record is below:

```
<RECORD ID=“1”>
<TEXT>
<PHI TYPE=“ID”>101126659</PHI>
<PHI TYPE=“HOSPITAL”>MGH</PHI>
<PHI TYPE=“DATE”>10/29</PHI>/1997 12:00:00 AM
CARCINOMA OF THE COLON .
Unsigned
DIS
Report Status :
Unsigned
Please do not go above this box important format codes are contained .
DISCHARGE SUMMARY
<PHI TYPE=“ID”>FMT51 DS</PHI>
DISCHARGE SUMMARY NAME :
<PHI TYPE=“PATIENT”>SLOAN , CHARLES E</PHI>
UNIT NUMBER :
<PHI TYPE=“ID”>358-51-76</PHI>
ADMISSION DATE :
<PHI TYPE=“DATE”>10/29</PHI>/1997
DISCHARGE DATE :
<PHI TYPE=“DATE”>11/02</PHI>/1997
PRINCIPAL DIAGNOSIS :
Carcinoma of the colon .
ASSOCIATED DIAGNOSIS :
Urinary tract infection , and cirrhosis of the liver .
HISTORY OF PRESENT ILLNESS :
The patient is an 80-year-old male , who had a history of colon cancer in the past , resected approximately ten years prior to admission , history of heavy alcohol use , who presented with a two week history of poor PO intake , weight loss , and was noted to have acute on chronic Hepatitis by chemistries and question of pyelonephritis .
</TEXT>
</RECORD>
```

Figure 1: Sample Discharge Summary Excerpt.

4 Methods

We evaluated de-identification systems both at token- and instance-level. We used precision, recall, and F-measure as evaluation metrics.

4.1 Common Metrics

Precision, also known as positive predictive value (PPV), is the percentage of the correctly identified tokens (or entities) in a category in relation to the total number of tokens (or entities) marked as belonging to that category. Recall, also known as sensitivity, is the percentage of the correctly identified tokens (or entities) in a category in relation to the total number of tokens (or entities) in that category. In a binary decision problem, e.g., does the entity belong to category A or not?, the output of a classifier can be represented in a confusion matrix which shows true positives (TP), true negatives (TN), false positives (FP), and false negatives (FN). Precision (Equation 1) and recall (Equation 2) can be computed from such a matrix. F-measure is the weighted mean of precision and recall; it can favor either precision or recall (Equation 3). In de-identification, recall is generally considered more important than precision. However, in the absence of a well-established numeric value associated with the relative importance of recall over precision, we weigh them equally, i.e., β=1. We also report precision and recall of each system separately.

Precision:

$$P = \frac{TP}{TP + FP}$$

Equation 1

Recall:

$$R = \frac{TP}{TP + FN}$$

Equation 2

F-measure:

$$F_\alpha = \frac{(1+\beta)P \times R}{\beta P + R}$$

Equation 3

4.2 Token-Level Evaluation

Precision, recall, and F-measure are standard evaluation metrics in NLP [33]. They are often applied at the token level and measure the performance of systems on individual tokens.

4.3 Instance-Level Evaluation

Instance-level evaluation is used by MUC and NIST in their named entity recognition shared-tasks. When applied to de-identification, instance-level evaluation checks individual PHI instances and marks the presence of a correct instance or one of three types of errors: substitution, insertion, or deletion. This evaluation models a PHI instance as a combination of three slots: type, content, and extent [22]. Each slot has four possible values: correct, incorrect, missing, and spurious. Type marks a PHI instance's category, e.g., patient or doctor. Content marks the words in the labeled PHI instance. Extent specifies the starting and the ending points (in terms of character indices) of the PHI instance. A PHI instance has to consist of the correct type, content, and extent in order to be correct (Equation 4).[1]

Total Number of Correct Entities:

$$C = \sum_{e=1}^{\#\text{ hypothesized entities}} c_e \text{ , where } c_e = \begin{cases} 1, \text{if type, content, and extent are all correct} \\ 0, \text{otherwise} \end{cases}$$

Equation 4

An error in any one of type, content, or extent results in a substitution error (Equation 5). For example, an actual date marked as an ID gives an incorrect type and results in a substitution error. Marking "08/97" as a date when the ground truth marks only "08" gives incorrect extent and also results in a substitution error. In general, despite being counter-intuitive, MUC and NIST consider all partial matches substitution errors.

Total Number of Substitution Errors:

$$S = \sum_{e=1}^{\#\text{ hypothesized entities}} s_e \text{ , where } s_e = \begin{cases} 1, \text{if any one of type, content, or extent is incorrect} \\ 0, \text{otherwise} \end{cases}$$

Equation 5

Spurious type, spurious content, and spurious extent characterize an insertion error (Equation 6), e.g., a non-PHI being marked as PHI. Missing type, missing content, and missing extent characterize a deletion error (Equation 7), e.g., a PHI being marked as non-PHI.

Total Number of Insertion Errors:

$$I = \sum_{e=1}^{\#\text{ hypothesized entities}} i_e \text{ , where } i_e = \begin{cases} 1, \text{if type, content, and extent are all spurious} \\ 0, \text{otherwise} \end{cases}$$

Equation 6

[1] We used software designed by NIST [22].

Total Number of Deletion Errors:

$$D = \sum_{e=1}^{\#\text{ hypothesized entities}} d_e \text{ , where } d_e = \begin{cases} 1\text{, if type, content, and extent are all missing} \\ 0\text{, otherwise} \end{cases}$$

Equation 7

Instance-level precision (Equation 8), recall (Equation 9), and F-measure (Equation 10) are computed from correct instances and from substitution, insertion, and deletion errors. Note that instance-level F-measure also weighs precision and recall equally by setting β=1.

Instance-level Precision: Instance-level Recall: Instance-level F-measure:

$$P = \frac{C}{C+S+I}$$

Equation 8

$$R = \frac{C}{C+S+D}$$

Equation 9

$$F = \frac{2PR}{(P+R)}$$

Equation 10

4.4 Significance Testing

We tested the significance of the differences of system performances using a randomization technique that is frequently used in NLP [[34], [35]]. The null hypothesis is that the absolute value of the difference in performances, e.g., F-measures, of two systems is approximately equal to zero. The randomization technique does not assume a particular distribution of the system differences. Instead, it empirically generates the distribution. Given two actual systems, it randomly shuffles their responses to "units"[2] in the test set *N* (e.g., 9999) times and thus creates *N* pairs of pseudo-systems. It counts the number of times, *n*, when the difference between the performances of the pseudo-system pairs is greater than the difference between the performances of the two actual systems. It computes $s = \frac{n+1}{N+1}$. If *s* is greater than a pre-determined cutoff α, then the difference of the performances of the two actual systems can be explained by chance; otherwise, the difference is significant at level α. Following MUC's example, we set α to 0.1 and we used complete messages, i.e., records, as units. Using records as units requires that we treat the predictions on all of the tokens/instances in a record as a unit; we shuffle system responses to a record by exchanging all corresponding token/instance-level predictions of two systems in the record. Note that PHI instances and sentences could also be used as units. However, this would implicitly give more weight to records with more PHI instances or sentences.

[2] For each unit, flip a coin to decide whether to exchange system responses for it.

5 Submissions

We allowed each team to submit up to three system runs for the de-identification challenge. In the end, we received sixteen de-identification system runs from seven teams. Some of these teams viewed the de-identification task as a problem of classification of tokens. Others viewed it as a sequence tracking problem (using Hidden Markov Models (HMM) [36] or Conditional Random Fields (CRF) [37]). We review six of these systems below. Wrenn, et al., the engineers of the seventh system, participated in the challenge but provided us with no system description; therefore, we are unable to review their system characteristics in this manuscript.

Aramaki, et al. [38] use CRFs to learn the features that are important to identify PHI. They take a text chunking and sequence tracking approach to de-identification and mark all tokens as either beginning a chunk (B) or as being inside(I)/outside(O) a chunk using BIO tagging [39]. Their features include local, global, and external features. Global features encode sentential information and label consistency. Sentential features mark the position of a sentence in the record, the length of the sentence, and the last several tokens in the previous sentence. External features come from dictionaries for people, locations, and dates. Aramaki, et al.'s choice of sentential features are motivated by the observation that most PHI instances in discharge summaries appear in the beginning and end of records and that the sentences containing them are relatively shorter. The authors assume that the most likely label for a target, i.e., the token being tagged, is the valid label for all occurrences of that target in a record. They achieve consistency among the labels of a target by a two-step learning algorithm. In the first step, they train their system with the local features of the target (the word to be classified). Inspired by Sibanda and Uzuner [31], they use the target itself, token length, part of speech (POS) tag, orthography (e.g., capitalization), special characters (e.g., "-"), and format patterns (e.g., "\d{3}-\d{3}-\d{4}"[3] for phone numbers). In the second step, they retrain their system with all the features from the first step plus the most frequent label (predicted by the first step) for the target. The authors report that the sentence features are the most informative features for dates, IDs, and patients because of the fact that these PHI instances appear mostly in the beginning of the records. Dictionary information is found to be particularly useful for dates, doctors, locations, and patients. Given the emphasized levels of ambiguity between PHI and non-PHI within individual records, label consis-

[3] Throughout this paper, we describe regular expressions using Perl syntax.

tency does not help differentiate the PHI from non-PHI; however, given the underplayed inter-PHI ambiguity in records, it is useful in differentiating among PHI.

Guillen [40] implements a rule-based system utilizing global features (sentence position), local features (lexical cues, special characters, and format patterns), and syntactic features to identify PHI. For her, sentence positions indicate whether a sentence is in the header, body, or the footer of the record. Guillen observes that headers contain mainly IDs, hospitals, and dates; footers include doctors, IDs, and dates. Guillen employs regular expressions to identify dates and IDs; she processes the body of the records using lexical cues and format patterns (e.g., "Mr." and "Discharge summary name:" mark patient names, "live.*(in|with)" indicates locations, etc.). As syntactic features, Guillen explores the tense and modality of verbs. She shows that "local and syntactic features play a significant role in identifying PHI when no […] gazetteers and dictionaries are available".

Guo, et al. [41] use Support Vector Machines (SVM) [42] and the "General Architecture for Text Engineering" (GATE) system [43] to produce two system runs: "Guo 1" and "Guo 2"[4]. In "Guo 1", they train their system only on the local features including the target (root form) and its length, POS tags, orthography, affixes, and special characters. In "Guo 2", they enrich their feature set with lexical cues for doctors, patients, and hospitals; with contextual features for patients (6 tokens to the left and to the right of the target); with named entity, e.g., person, location, organization, dictionaries; with named entity types predicted by an information extraction system from the newswire domain; and with rules[5] for mapping the named entities recognized by the information extraction system to PHI categories (e.g., rules to map a person name to either a doctor or a patient). By comparing the results from their two runs, the authors show that in recognizing patients, context is more powerful than the named entity types provided by the information extraction system.

Hara [44] adopts a hybrid system of rules with SVMs. (Note: a description of the work done by Hara et al for the i2b2 de-identification challenge appears as a JAMIA on-line supplement to this overview, and can be viewed at www.jamia.org). His rules capture the patterns for phone numbers, dates, and IDs. SVMs trained on global and local features recognize hospitals, locations, patients, doctors, and ages. The features used for SVMs include headings of sections (the

[4] In this paper, we refer to systems with the last name of the first author and a submission id.

[5] Rules are captured by regular expression templates. In this manuscript, we use the terms "rules" and "regular expression templates", and "format templates" interchangeably.

heading closest to the target), the category of the sentence as determined by a sentence classifier, the root and the surface form of the target, POS tag, and orthography. The classification of the sentences in a record is based on the PHI categories they contain. For this, Hara uses two types of ordered trees, "n-gram trees" and dependency trees, with the Boosting Algorithm for Classification Trees (BACT) [45]. Given this information, he applies chunking techniques to determine the position and span of locations, hospitals, patients, and doctors in a sentence. Hara submitted 3 runs which differ mainly in the way they classify sentences. "Hara 1" uses n-gram trees, "Hara 2" uses dependency trees, and "Hara 3" skips sentence classification altogether. The results show that sentence classification hurts performance.

Szarvas, et al. [46] apply an iterative named-entity recognizer to de-identification. They treat de-identification as a classification task and use decision trees with local features and dictionaries. Their local features include lexical cues, phrasal information, orthography, token length, special characters, format templates (for ages, phone numbers, and IDs), and frequency information. Their phrasal information includes labels of the phrase preceding the target and whether the target is inside quotation marks or brackets. Frequency information includes the lower/uppercase ratio and term frequency. Szarvas, et al.'s system uses context defined by lexical triggers that are sorted based on the strength of their association with each PHI category as determined by use frequencies. They collect from training data the lexical context of PHI tokens (window of ±3) and instances (window of ±1). The authors hypothesize that the PHI found in the structured headers of the records can be used as trusted information which, if associated unambiguously with only one PHI category, can be more reliable than context for recognizing PHI in the unstructured narratives. To ensure consistency among the PHI labels within a record, Szarvas, et al. post-process the labels of PHI and mark all occurrences of a phrase with the label of the longest identified matching phrase. They submit three runs: "Szarvas 1", "Szarvas 2", and "Szarvas 3". In "Szarvas 1", they use a boosted decision tree consisting of AdaBoost [47] and C4.5 classifiers [48] trained on dictionaries, all local features, and some lexical triggers. In "Szarvas 2", they combine the votes of three boosted classifiers trained on dictionaries, all local features, and all lexical triggers. In "Szarvas 3", they add trusted information to "Szarvas 1".

Wellner, et al. [49] adapt two NER systems, LingPipe and Carafe, to the de-identification task. They submit three runs. "Wellner 1" and "Wellner 3" use Carafe [50]. "Wellner 2" uses LingPipe [51], an implementation of HMMs. Carafe is a CRF implementation that uses n-grams and location dictionaries; it is complemented by regular expressions that can capture the more standardized PHI, e.g., dates. In order to lend information from structured headers/footers to the narrative, the CRF systems treat the entire record as a single sequence. They model features in two ways: a transition model (using the target's local features as well as the current and previous PHI labels) and a label model (using the target's local features as well as the current PHI label). In general, "Wellner 1" and "Wellner 2" use traditional MUC like features, including lexical context (with window size of ±2), orthography, affixes (with length of two), special characters, and format templates. "Wellner 3" adds to this feature set lexical cues (e.g., "Medical center" indicates hospitals) and dictionaries for people, locations, and dates. It also increases the affix length to four characters, increases the lexical context window to three, and adds templates for phone numbers, hospitals, and locations. Wellner, et al. show that such task-specific feature engineering for Carafe results in a performance improvement of "Wellner 3" over "Wellner 1"; also, the two CRF systems generally outperform the HMM system.

A summary of the characteristics of all of the above described systems is in Table 6 where Pt means patient, Dr means doctor, Dt means date, Hp means hospital, Ag means age, Ph means phone number, Lo means location, Id means ID, and No means number. Systems are referred to using the last name of the first author and their run IDs. "*" marks entries corresponding to the model, feature, or approach employed by each system. Items in the brackets indicate the PHI categories these features are applied to.

Table 6: Summary of System Characteristics.

Systems		Aramaki 1	Guillen 1	Guo 1	Guo 2	Hara 1	Hara 2	Hara 3	Szarvas 1	Szarvas 2	Szarvas 3	Wellner 1	Wellner 2	Wellner 3
BIO Model		*		*		*						*		
Global features	sentence position	*	*						*	*	*			
	sentence length	*												
	words in previous sentence	*												
	Majority Label	*												
	trusted PHI										*			
	heading info					*	*	*						
Local features	Lexical cues[6]	*	* (Dr, Pt, Hp, Ph)		* (Dr, Pt, Hp)				n-gram trigger	all triggers	n-gram trigger	* W2		* W3
	tokens	*		*	*	*	*	*				*	*	*
	n-grams													
	phrasal info								*	*	*			
	pos-tag	*		*	*	*	*	*						
	ortho-graphic	*		*	*	*	*	*	*	*	*	*	*	*
	token length	*		*	*				*	*	*			
	affix			*	*							*L2		*L4
	special char	*	* (Dt)	*	*				*	*	*	*	*	*
	template	* (Dt, Ph)	* (Dt, Lo, Ag)		* (except Id)	*	*	*	* (Ag, Ph, Id)			* (No)		*(No, Ph, Hp, Lo)
	sentence classification					n-gram tree	dependency tree							
	frequency								*	*	*			
External features	dictionary	*			*				*	*	*			*
Machine Learning method or Rules		CRF	Rules	SVM		Rules, SVM			boosted c4.5	voting of 3 boosted c4.5's	iterated boosted c4.5	CRF	Hierarchical HMM	CRF

[6] W indicates window size for lexical cues (e.g., W2 means lexical cue contains the preceding and succeeding 2 tokens with respect to the target).

6 Evaluation, Results, and Discussion

We evaluated de-identification performance using precision, recall, and F-measure at token and instance level. All evaluations were performed against the same test corpus. See Table 1 through Table 5 for the details on the distribution of PHI in the challenge test corpus. Below, we present results on all seven team's submissions; however, we are unable to make sense out of Wrenn, et al.'s results due to lack of a system description.

Some of the tables discussed in this manuscript are available as JAMIA on-line data supplements at www.jamia.org. The online supplements are marked as such in the manuscript.

6.1 Token-level Evaluation

6.1.1 Overall Performance

We evaluated systems on their ability to differentiate PHI from non-PHI. For this, we lumped all eight categories of PHI into one overall PHI category and computed the precision, recall, and F-measure for differentiating PHI from non-PHI. For details, please see Table 7, available in the on-line data supplements.

Figure 2 demonstrates that recall shows more variance than precision among the 13 system runs. This is not surprising. Almost all of the systems used patterns, e.g., regular expressions or rules, for recognizing the format of the PHI (see Table 6). This led to high precision. However, our data consist of large amounts of ungrammatical text with possible misspellings, arbitrary abbreviations, etc. In addition, even for very structured PHI such as phone numbers, standard format assumptions do not always hold, e.g., "337-4296, 936", "678-233-5033, x 549", and "160-6305, x 2644. This makes it difficult to capture some PHI with patterns. The machine learning approaches and the features used amend the format patterns (orthography, tokens, POS tags etc.) but also fail to characterize all PHI. The relative strengths of the employed learning methods and features account for most of the variance in recall.

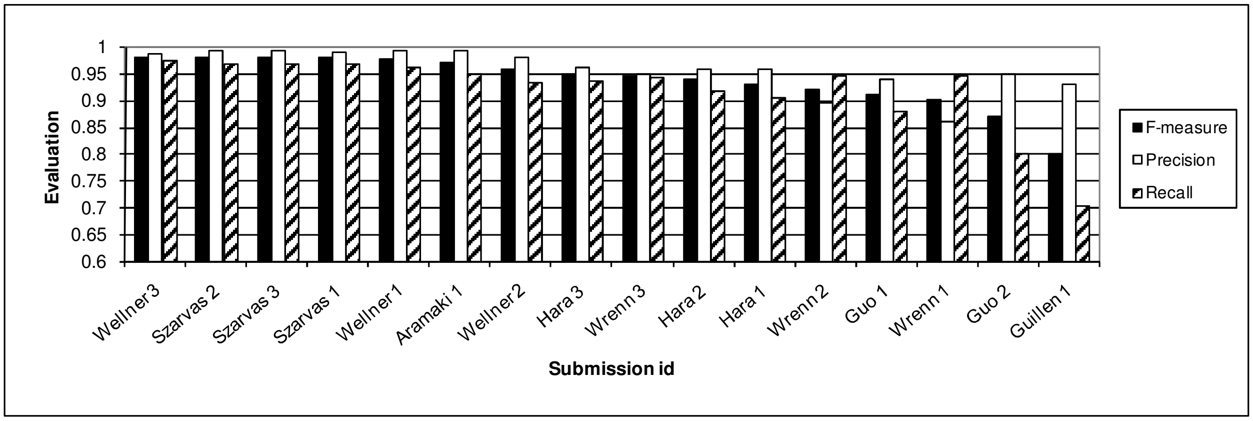

Figure 2: System Comparison Based on Overall Performance (rank ordered in F-measure).

6.1.2 Analysis

Table 8 shows that the performances of most pairs of systems are significantly different from each other. In this table, "F", "P", and "R" mark the pairs of systems that are *not* significantly different at $\alpha = 0.1$ in F-measure, precision, and recall, respectively. Note that we only mark the upper diagonal due to symmetry. Note also that the top four systems are not significantly different from each other in F-measure.

Among the submissions, "Guillen 1" is the only solely rule-based system; therefore, we use it as a baseline for comparison. All other systems augment rules with machine learning techniques to some extent. The systems that take advantage of machine learning generally perform significantly better than the baseline. Among the systems that use machine learning, "Guo 1" is the only one that does not employ any regular expression templates as features. With the exception of "Guo 2", all of the machine learning systems that employ regular expression templates as features perform significantly better than "Guo 1". Hybrid systems such as Hara's that employ rules for certain PHI categories and machine learning with regular expression template features for others generally fall behind the systems that use regular expression template features for all PHI categories.

Table 8: Significance Tests on F-measure, Precision, and Recall. Systems are sorted in F-measure.

	Szarvas 2	Szarvas 3	Szarvas 1	Wellner 1	Aramaki 1	Wellner 2	Hara 3	Wrenn 3	Hara 2	Hara 1	Wrenn 2	Guo 1	Wrenn 1	Guo 2	Guillen 1
Wellner 3	F	F	F, R												
Szarvas 2	-	F, P, R	F, R	F, P, R	P										
Szarvas 3	-	-	F, P, R	F, P, R	P										
Szarvas 1	-	-	-	F	P										
Wellner 1	-	-	-	-	P										
Aramaki 1	-	-	-	-	-						R		R		
Wellner 2	-	-	-	-	-	-	R								
Hara 3	-	-	-	-	-	-	-	F, R							
Wrenn 3	-	-	-	-	-	-	-	-				P		P	
Hara 2	-	-	-	-	-	-	-	-	-	P				P	
Hara 1	-	-	-	-	-	-	-	-	-	-	F			P	
Wrenn 2	-	-	-	-	-	-	-	-	-	-	-				
Guo 1	-	-	-	-	-	-	-	-	-	-	-	-	F		
Wrenn 1	-	-	-	-	-	-	-	-	-	-	-	-	-		
Guo-2	-	-	-	-	-	-	-	-	-	-	-	-	-	-	

Overall, statistical learning systems with regular expression template features for all PHI categories perform best. They are followed by hybrid systems combining rules for some PHI categories with learning for others, pure learning systems without regular expression template features or supplementary rules, and pure rule-based systems, in that order.

6.1.2.1 Best Systems

Among learning systems with template features, “Wellner 3”, “Szarvas 2”, “Szarvas 3”, and “Szarvas 1” are the best performers in terms of F-measure. “Wellner 1”, “Szarvas 2”, “Szarvas 3”, “Aramaki 1”, and “Szarvas 1” give the best precision. “Wellner 3”, “Szarvas 1”, “Szarvas 2”, and “Szarvas 3” give the best recall. In general, Wellner’s and Aramaki’s systems take a sequence tracking approach to de-identification and use CRFs to map features to PHI. Szarvas’ systems classify individual tokens with boosted decision trees.

Detailed review of the CRF systems, “Wellner 1”, “Wellner 3”, and “Aramaki 1”, shows that the reliance on majority labels does not significantly improve the precision of “Aramaki 1” but lowers its recall significantly. This is due to the ambiguity between PHI and non-PHI. For example, one of our records contains references to “DRS. RIGHT AND SIGNS” where a majority of the occurrences of both “RIGHT” and “SIGNS” are marked as non-PHI. When compared to “Wellner 1”, the task specific feature engineering in “Wellner 3” improves recall; more specifically, the lexical cues of “Wellner 3” trade precision for recall. For example, this system marks all acronyms with the lexical cue “H” as a hospital when only some such acronyms in fact are, e.g., “BPH” in “prostatectomy for BPH” stands for “Benign Prostatic Hyperplasia”.

Szarvas, et al. show that decision trees give competitive performance to CRFs in token-level classification of PHI. According to Szarvas’ own analysis [46], frequency information and trusted PHI help improve system F-measures. Their frequency information consists of term frequency (tf), lowercase/uppercase ratio, etc. Low tf tends to indicate some categories of PHI, e.g., IDs. Similarly, lowercase/uppercase ratio is a good indicator of names. For example, typical names either consist of all uppercase letters, e.g., “DR. RIGHT”, or are capitalized, e.g., “Dr. Kaystkote”. Hence, if lowercase to uppercase ratio is close to 0 or around $1-1/l$ (where l is averaged word length), it is highly likely that the target is in a name. Frequency information, when combined with lexical triggers for doctors, patients, locations, and hospitals, can be effective in predicting these PHI categories. The best lexical cues come from the trusted PHI obtained from structured headers and footers of the records. For example, “TR”, “DD”, “TD”, and “CC” lines of the footer in 3 clearly mark doctors, dates, dates, and doctors respectively.

```
TR:
<PHI TYPE=“DOCTOR”>jn</PHI>
DD:
<PHI TYPE=“DATE”>08/04</PHI>/1999
TD:
<PHI TYPE=“DATE”>08/09</PHI>/1999 1:46 P
CC:
<PHI TYPE=“DOCTOR”>LENNI E CAN</PHI>, M.D.
```

Figure 3: Sample Footer.

6.1.2.2 Performance on PHI Categories

In addition to overall system evaluation on distinguishing PHI from non-PHI, we evaluated systems for their ability to recognize the exact category of PHI. Given the differences in the percentages of out-of-vocabulary and ambiguous tokens included in each PHI category, we present results on each category separately (please see Tables 9 through 11 in the on-line data supplement). The results summarized in Figure 4, Figure 5, and Figure 6 show that all systems have the hardest time when identifying locations (F-measures below 80%) and phone numbers (F-measures generally below 90%). For all other PHI categories, including the categories with the highest ambiguities and most out-of-vocabulary tokens (i.e., hospitals and doctors), the system performances are generally comparable to, if not better than, their overall performance. Of the systems that are not in the top five in overall performance, "Hara 3" is among the top three systems in recognizing patients; this system gives its worst performance on locations.

For locations, poor performance is partly caused by the presence of few training examples. In the training set, only 144 instances (302 tokens) of locations exist (compare to other PHI categories in Table 1). The ability to learn locations is further limited by the fact that, unlike hospitals that are indicated by many lexical cues such as "admitted to", "transferred to", "Medical Center", and "Hospital", locations have few lexical cues. Even the system with the best performance on locations, i.e., "Wellner 3", completely misses 38, partially matches 7, and mislabels as hospitals 5 of the 119 location instances present in the test corpus. Its partial matches tend to miss the abbreviations associated with states, e.g., MA.

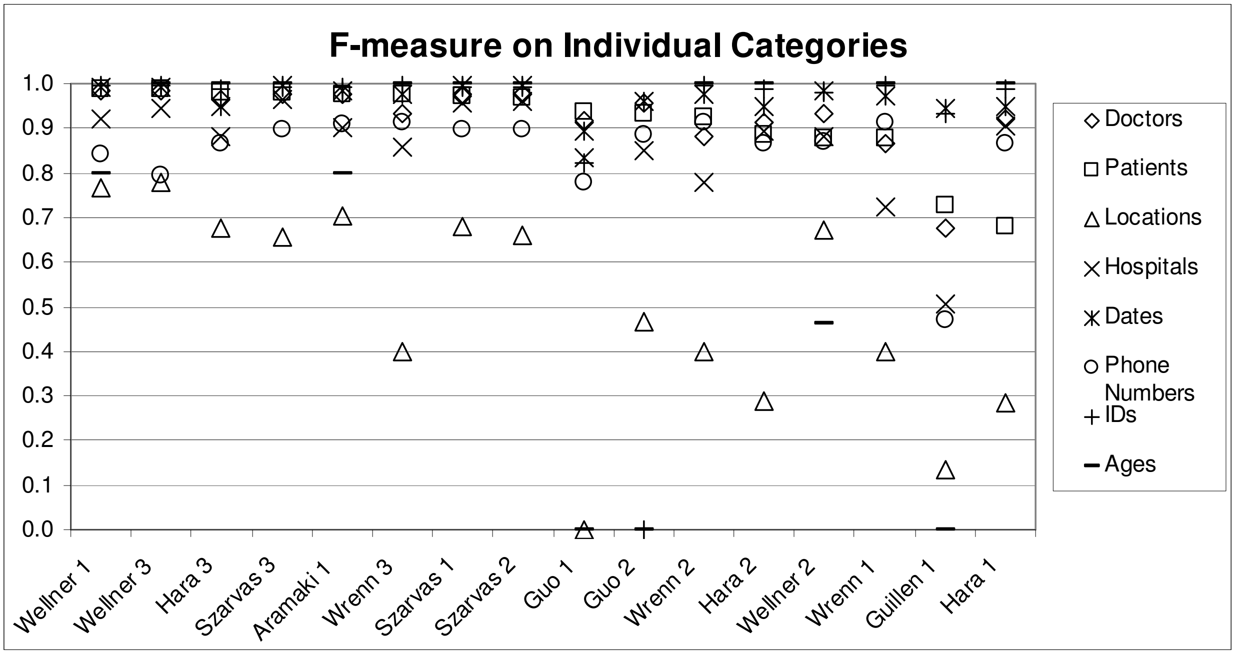

Figure 4: F-measure on Individual PHI Categories. Sorted by Performance on Patients.

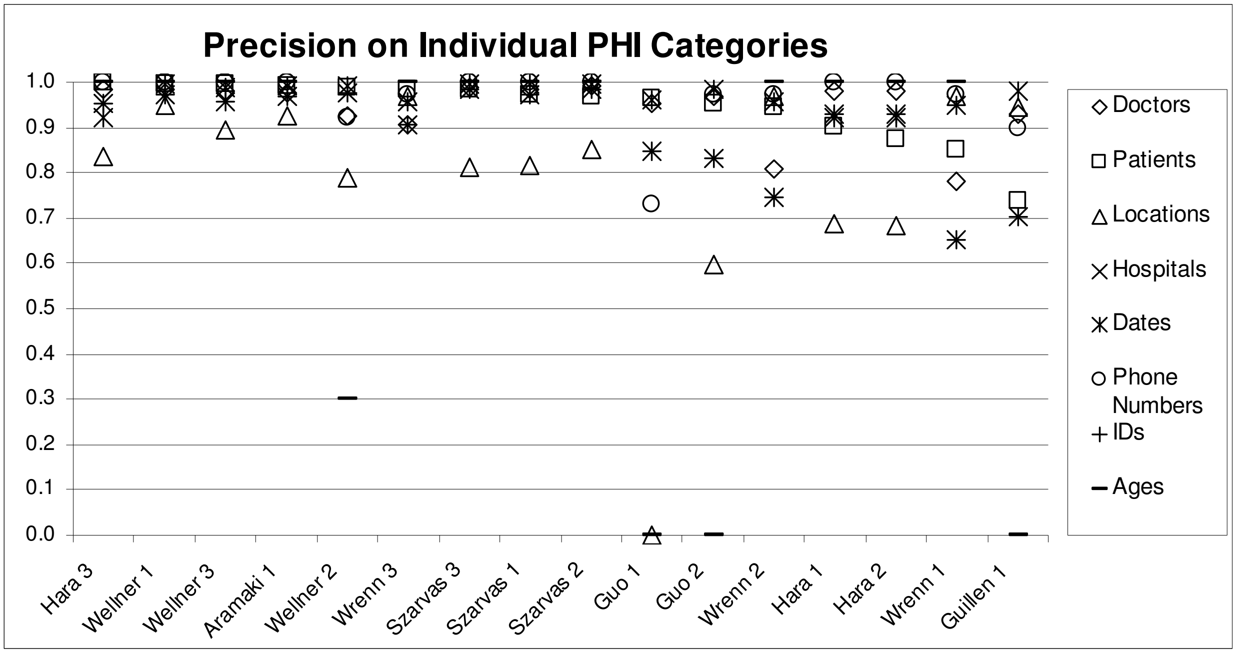

Figure 5: Precision on Individual PHI Categories. Sorted by Performance on Patients.

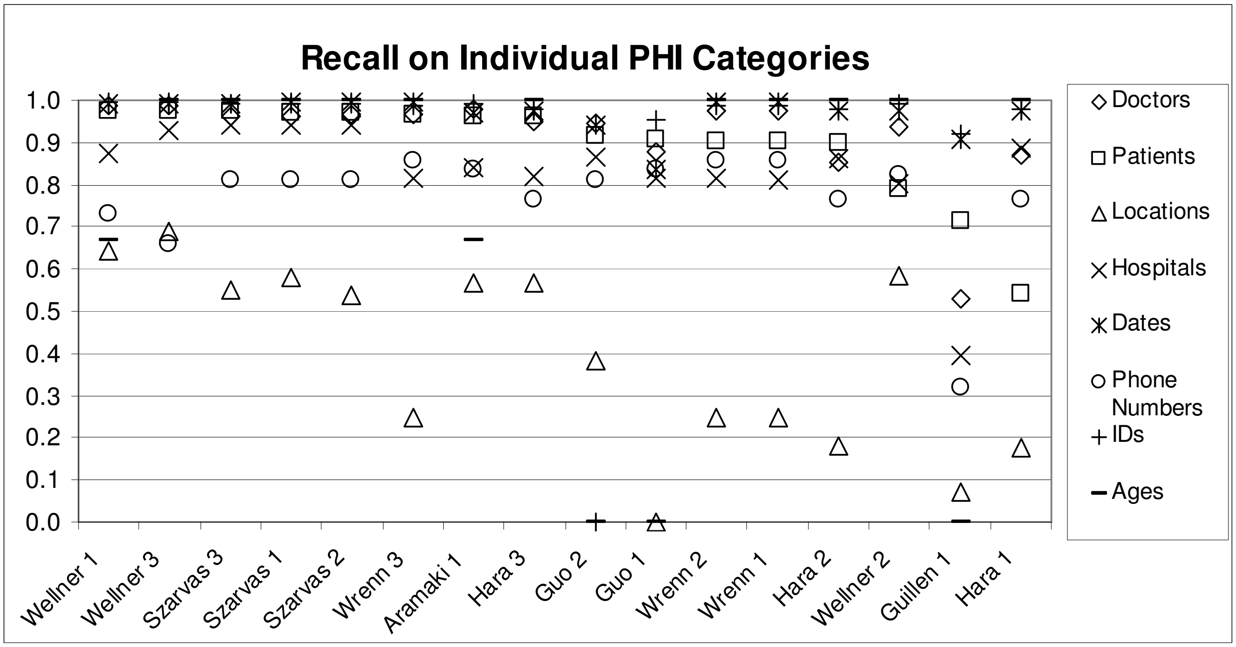

Figure 6: Recall on Individual PHI Categories. Sorted by Performance on Patients.

For phone numbers, examination of system responses show that many systems suffered from lack of comprehensive rules. Even the system with the best performance on phone numbers, i.e., “Aramaki 1”, missed the phone number “(692) 673 3025”, the pager number “917070-7689”, the extensions in “678-233-5033, x549” and in “160-6305, x2644”, and the area code in “337-4296, 936”. All of these PHI are considered phone numbers but deviate from the rules dedicated to recognizing phone numbers by “Aramaki 1”. In general, deviation from the expected templates complicate the recognition of phone numbers, which already suffer from a lack of training examples (174 instances consisting of 201 tokens in training data).

6.2 Results on Ambiguous and Out-of-Vocabulary PHI Tokens

To gain insight into the strengths of systems on out-of-vocabulary and ambiguous PHI tokens, we analyzed their performance on such tokens in the challenge test corpus separately. We found that the top six systems in Figure 2 are in the top six in terms of performance on out-of-vocabulary PHI also. Given the dominance of out-of-vocabulary PHI in the test corpus, this observation is not surprising.

In addition to performance on out-of-vocabulary PHI, Figure 7 shows the performance of systems in classifying ambiguous tokens into PHI and non-PHI categories, taking into consideration all ambiguities (including inter-PHI ambiguities) and taking into consideration only the PHI tokens that are ambiguous with non-PHI (please see Tables 12 through 14 in the on-line supplements). We see that “Guillen 1” gives the worst performance on ambiguous PHI tokens. This implies that Guillen’s rules capture the features present in unambiguous PHI tokens but don’t generalize well to ambiguous data. For the systems that employ machine learning with local features, performances on ambiguous PHI tokens are comparable to performances on all PHI. Note that systems using dictionaries also use abundant local features, thus their performances on ambiguous PHI tokens show no significant difference from overall performances, e.g., ”Wellner 3”, Szarvas, et al.’s systems, and Aramaki, et al.’s systems. Local features by themselves can recognize ambiguous PHI tokens and produce comparable results.

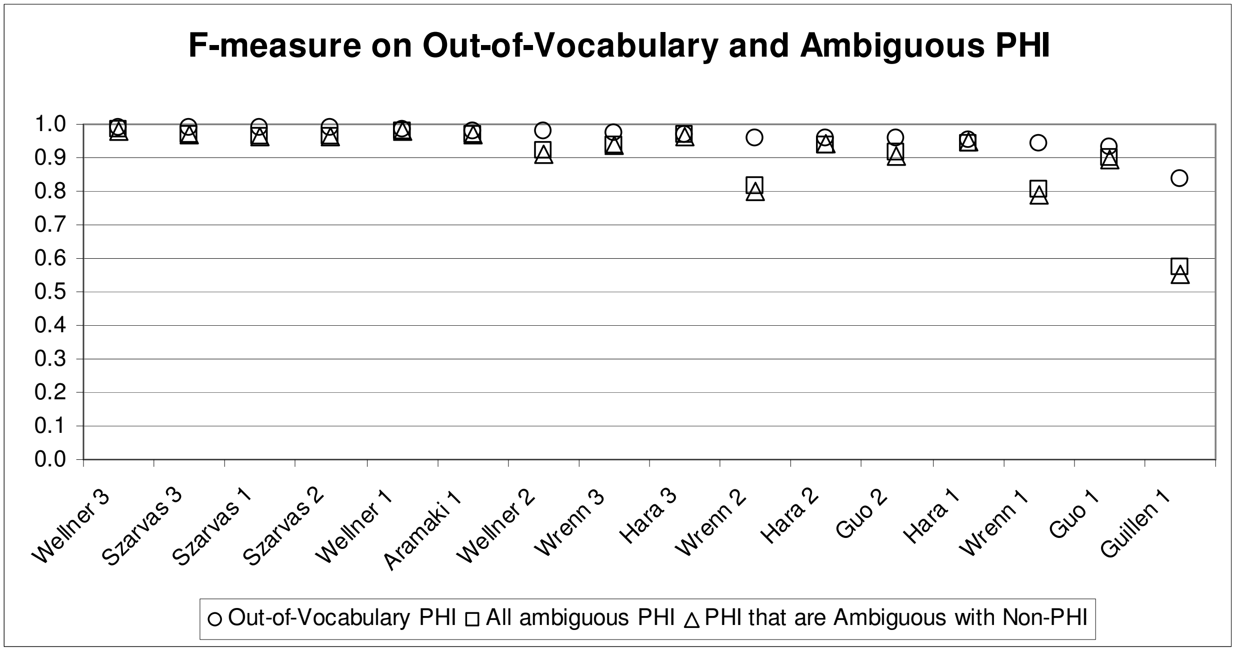

Figure 7: F-measure on Out-of-Vocabulary and Ambiguous PHI. Sorted by Performance on Out-of-Vocabulary PHI.

6.3 Instance-level Evaluation

In addition to token-level evaluation, we performed instance-level evaluation. We measured the performance of systems on differentiating instances of PHI from non-PHI, without any regard to the exact categories of PHI. Figure 8 shows that the top performing systems in token-level evaluation also gave the best performance in instance-level evaluation. Please refer to Table 15 in the on-line data supplements for details.

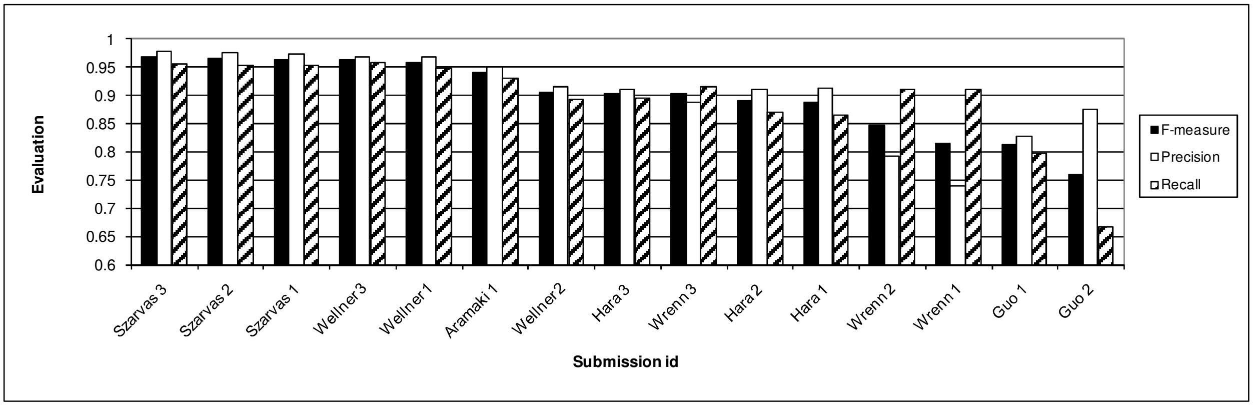

Figure 8: System Comparison Based on Instance-level Overall Evaluation. Sorted by F-measure.

Figure 8 confirms the findings in Figure 2 and shows that the 13 systems differ more in recall than in precision. However, comparing Figure 8 with Figure 2 also shows that all system performances are lower at the instance-level than they are at token-level. This is expected because instance-level evaluation gives systems no partial credit for marking part of a PHI instance correctly. It only gives credit for instances that are marked exactly correctly.

6.3.1 Significance Testing

Significance tests gave similar results on instance-level evaluation as they did on token-level. Resulting matrices are shown in Table 16. Again, "F", "P", and "R" mark the pairs of systems that are NOT significantly different at $\alpha = 0.1$ in F-measure, precision, and recall respectively.

Findings on instance-level evaluation confirm the results of token-level evaluation. In particular, despite the change in their absolute rankings relative to each other, "Szarvas 3", "Szarvas 2", "Szarvas 1", "Wellner 3", and "Wellner 1" are still the top five systems.

Table 16: Significance Test on Instance-Level F-measure, Precision, and Recall.

	Szarvas 2	Szarvas 1	Wellner 3	Wellner 1	Aramaki 1	Wellner 2	Hara 3	Wrenn 3	Hara 2	Hara 1	Wrenn 2	Wrenn 1	Guo 1	Guo 2
Szarvas 3	F, P, R	R	F, R											
Szarvas 2	-	F, R	F, R											
Szarvas 1	-	-	F, R	P										
Wellner 3	-	-	-	P										
Wellner 1	-	-	-	-										
Aramaki 1	-	-	-	-	-									
Wellner 2	-	-	-	-	-	-	F, P, R	F	P	P				
Hara 3	-	-	-	-	-	-	-	F	P	P				
Wrenn 3	-	-	-	-	-	-	-	-						P
Hara 2	-	-	-	-	-	-	-	-	-	P				
Hara 1	-	-	-	-	-	-	-	-	-	-				
Wrenn 2	-	-	-	-	-	-	-	-	-	-	-	R		
Wrenn 1	-	-	-	-	-	-	-	-	-	-	-	-	F	
Guo 1	-	-	-	-	-	-	-	-	-	-	-	-	-	

6.4 Common Errors

Analysis of the errors of the top six systems by Aramaki, et al., Szarvas, et al., and Wellner, et al. show that ambiguous and out-of-vocabulary PHI play a role in the overall performance of the systems. Some ambiguous and out-of-vocabulary PHI cause missed or partially recognized PHI. Overall:

- All of the top six systems perform generally well on patients; they only miss very few ambiguous and out-of-vocabulary patient names such as "Randdor So".

- All of the top six systems are challenged by ambiguous and out-of-vocabulary entries in doctors. In particular, they tend to miss the ambiguous token “Can” in “Ettrent Can” and fail to recognize the out-of-vocabulary “Freierm , Le”. In addition, the lack of correct punctuation in the records affects the performance of these systems in recognizing the doctor names correctly. In particular, many doctor names are followed immediately by the name of the hospital that the doctor is affiliated with; however, the records lack the correct punctuation that would mark the end of the doctor name and the beginning of the hospital name, e.g., “Dr. Fa Knottippsfyfe Fairm of Ijordcompmac Hospital”, i.e., “Dr. Fa Knottippsfyfe Fairm” who works at the “Ijordcompmac Hospital”. The systems are able to only partially recognize many such PHI.
- All of the top six systems make mistakes on marking locations such as “South Dakota Rangers” and “Port Authorities”. They also miss ambiguous and out-of-vocabulary locations such as “Ph”, “Goo”, and “Apem”.
- Among the top six, “Aramaki 1” has an especially hard time on and only partially recognizes many hospitals. In particular, this system misses “Ingree and Ot of” in the phrase “Ingree and Ot of Weamanshy Medical Center”; it misses the tokens “Fairm of” in the phrase “Fairm of Ijordcompmac Hospital”; and it over-marks the tokens “Cardiac Surgical” in the phrase “Nilame Hospital Cardiac Surgical”.
- Surprisingly, effective removal of dates proves to be non-trivial. We observe that dates are highly ambiguous with medical measurements and the systems tend to miss some dates with even standard formats, e.g., 9/10.

6.5 Generalization and Practical Use

The results of evaluations summarized in this paper are quite encouraging and suggest that the best techniques are able to find nearly all instances of PHI in the test data. We are left with two important unanswered questions:

1. Does success on this challenge problem extrapolate to similar performance on other, untested data sets?
2. Can health policy makers rely on this level of performance to permit automated or semi-automated disclosure of health data for research purposes without undue risk to patients?

Unfortunately, we have strong reasons to suspect that extrapolation will be difficult. We have noted that a number of the systems took advantage of the specific organization of discharge summaries that are characteristic of the institution from which these were drawn. We have also observed, anecdotally, that a Web-based demonstration program of our own construction, but based on techniques very similar to those evaluated here, did well on test data but suffers occasional serious lapses when challenged with volunteered examples not drawn from our own data [31]. For example, in almost all of the challenge data set (both training and test sets), patients are referred to as "Mr. Smith" or "Ms. Jane Doe," and never as "Bill" or "Sam Smith." Therefore, machine learning methods overfit the data's style and learn to rely heavily on lexical clues such as "Ms.". These and other similar observations argue that systems should be trained on much larger and more heterogeneous data sets in order to allow their developers to judge more accurately how well they really perform. Studies on confusion set disambiguation, i.e., "choosing the correct use of a word given a set of words with which it is commonly used", demonstrate that three orders of magnitude larger data sets lead to significantly improved performance using unchanged methods [52]; we suspect that the same would be true of de-identification. However, because creation of gold standard data sets is very time consuming and because this process requires access to sensitive patient data, it may be very difficult to increase the data set a thousand fold. Instead, we anticipate that unsupervised techniques, perhaps bootstrapped from data de-identified by programs such as those reported here, will need to be developed.

The second question, "how good is good enough?", combines many diverse issues of ethics, liability, law, and regulations with the performance questions we report on here. We are not aware of strict criteria to be enforced by Institutional Review Boards when they agree that release of (nearly completely) de-identified data is safe. We know from past experience that human performance on de-identification tasks is imperfect and some studies show that computer algorithms perform at least as well [11]. A currently popular approach is to approve release of data for research only under a limited data use agreement, where the recipient agrees contractually not to try to re-identify patients. In this case, institutions rely on automated de-identification methods only to reduce the risk of inadvertent disclosure. The need for such agreements to use data does, however, inject delays and may discourage widespread legitimate data exploitation.

7 Conclusion

In this paper, we described the i2b2 shared-task on de-identification, including the details of data preparation, overview of participating systems, details of evaluation metrics, and our findings. Our analysis shows that statistical learning systems utilizing rule templates as features give the best de-identification performance on our corpus, followed by hybrid systems of rules and machine learning, pure machine learning systems without rule features or supplementary rules, and pure rule-based systems, in that order. Results indicate that although ambiguity of tokens can deteriorate performance, out-of-vocabulary PHI can be effectively identified in this corpus.

Overall, the systems reviewed in this paper show that much can be accomplished to de-identify data with the best techniques. Future challenges remain, including the need to make systems robust to greater variation in challenge data, e.g., data from different sources and of different formats, and policy issues to delineate the circumstances under which automated de-identification methods can be safely used.

Acknowledgements

This work was supported in part by the National Institutes of Health through research grants 1 RO1 EB001659 from the National Institute of Biomedical Imaging and Bioengineering and through the NIH Roadmap for Medical Research, Grant U54LM008748. IRB approval has been granted for the studies presented in this manuscript. We thank all participating teams for their contribution to the challenge. We also thank AMIA for their support in the organization of the workshop that accompanied this challenge.

Bibliography

[1] Berner E., Detmer D., Simborg D. Will the Wave Finally Break? A Brief View of the Adoption of Electronic Medical Records in the United States. Journal of the American Medical Informatics Association, 2005; 12:3-7.

[2] Miller R. Medical Diagnostic Decision Support Systems -- Past, Present, and Future: a Threaded Bibliography and Brief Commentary. Journal of the American Medical Informatics Association, 1994; 1:8-27.

[3] Rollman B., Hanusa B., Gilbert T., Lowe H., Kapoor W., Schulberg H. The Electronic Medical Record. Archives of Internal Medicine, 2001; 161:189.

[4] Cao H., Stetson P., Hripcsak G. Assessing Explicit Error Reporting in the Narrative Electronic Medical Record Using Keyword Searching. Journal of Biomedical Informatics, 2004; 36:99-105.

[5] Chapman W., Bridewell W., Hanbury P., Cooper G., Buchanan B. A Simple Algorithm for Identifying Negated Findings and Diseases in Discharge Summaries. Journal of Biomedical Informatics, 2001; 34:301-310.

[6] Chapman W., Christensen L., Wagner M., Haug P., Ivanov O., Dowling J., Olszewski R. Classifying Free-text Triage Chief Complaints into Syndromic Categories with Natural Language Processing. Artificial Intelligence in Medicine, 2005; 33:31-40.

[7] Friedman C., Shagina L., Lussier Y., Hripcsak G. Automated Encoding of Clinical Documents Based on Natural Language Processing. Journal of the American Medical Informatics Association, 2004; 11:392-402.

[8] Huang Y., Lowe H., Klein D., Cucina R. Improved Identification of Noun Phrases in Clinical Radiology Reports Using a High-Performance Statistical Natural Language Parser Augmented with the UMLS Specialist Lexicon. Journal of the American Medical Informatics Association, 2005; 12:275-285.

[9] Friedman C., Hripcsak G. Natural Language Processing and its Future in Medicine. Academic Medicine: Journal of The Association of American Medical Colleges, 1999; 74:890-895.

[10] HIPAA Administrative Simplification Regulation Text. § 164.514 [Online] http://www.hhs.gov/ocr/AdminSimpRegText.pdf.

[11] Sweeney L. Replacing Personally-identifying Information in Medical Records, the Scrub System. Journal of the American Medical Informatics Association, 1996.

[12] Malin B., Airoldi E. The Effects of Location Access Behavior on Re-identification Risk in a Distributed Environment. Privacy Enhancing Technologies 2006: 413-429

[13] Zhou G., Zhang J., Su J., Shen D., Tan C. Recognizing Names in Biomedical Texts: a Machine Learning Approach. Bioinformatics, 2005; 20(7).

[14] Gupta D., Saul M., Gilbertson J. Evaluation of a Deidentification (De-Id) Software Engine to Share Pathology Reports and Clinical Documents for Research. American Journal of Clinical Pathology, 2004; 121(6).

[15] Douglass M., Clifford G., Reisner A. Moody G., Mark R. Computer-Assisted De-Identification of Free Text in the MIMIC II Database. Computers in Cardiology. 2005; 32:331-334.

[16] Ruch P., Baud R., Rassinoux A., Bouillon P., Robert G. Medical Document Anonymization with a Semantic Lexicon. AMIA. 2000.

[17] Beckwith B. A., Mahaadevan R., Balis U. J., Kuo F. Development and Evaluation of an Open Source Software Tool for Deidentification of Pathology Reports. BMC Medical Informatics and Decision Making 2006, 6:12.

[18] Informatics for Integrating Biology & the Bedside. [Online] Partners Healthcare, 2005. https://www.i2b2.org/.

[19] American Medical Informatics Association. [Online] http://www.amia.org.

[20] Uzuner Ö., Goldstein I., Kohane I. Identifying Patient Smoking Status from Medical Discharge Records. Forthcoming, Journal of the American Medical Informatics Association, 2007; 14(6).

[21] Grishman R., Sundheim B. Message Understanding Conference - 6: A Brief History. 16th International Conference on Computational Linguistics (COLING). 1996. pp. 466–471.

[22] IE-ER, NIST. [Online] 1999. http://www.nist.gov/speech/tests/ie-er/er_99/er_99.htm.

[23] ACE. [Online] http://www.nist.gov/speech/tests/ace/index.htm.

[24] Hirschman L., Yeh A., Blaschke C., Valencia A. Overview of BioCreAtIvE: critical Assessment of Information Extraction for Biology. BMC Bioinformatics, 2005; p. 6(S1).

[25] PubMed Central Homepage. [Online] National Institutes of Health (NIH). http://www.pubmedcentral.nih.gov.

[26] FlyBase Homepage. [Online] http://www.flybase.org.

[27] Mouse Genome Informatics. [Online] http://www.informatics.jax.org.

[28] Saccharomyces Genome Database. [Online] Department of Genetics at the School of Medicine, Stanford University. http://www.yeastgenome.org.

[29] UniProtKB/Swiss-Prot. [Online] European Molecular Biology Laboratory. http://www.ebi.ac.uk/swissprot/.

[30] TREC Genomics Track. [Online] Oregon Health & Science University. http://ir.ohsu.edu/genomics.

[31] Sibanda T., Uzuner Ö. Role of Local Context in De-identification of Ungrammatical, Fragmented Text. North American Chapter of Association for Computational Linguistics/Human Language Technology (NAACL-HLT). 2006.

[32] Friedman C., Hripcsak G., Shagina L., Liu H. Representing Information in Patient Reports Using Natural Language Processing and the Extensible Markup Language. Journal of the American Medical Informatics Association, 1999; 6:76-87.

[33] Hripcsak G., Rothschild A. S. Agreement, the F-Measure, and Reliability in Information Retrieval. Journal of the American Medical Informatics Association, 2005; 12:296-298.

[34] Chinchor N. The Statistical Significance of the MUC-4 Results. McLean, Virginia : Association for Computational Linguistics, 4th Conference on Message Understanding. 1992. pp. 30-50.

[35] Noreen E. W. Computer Intensive Methods for Testing Hypothesis: An Introduction. New York: John Wiley & Sons, 1989.

[36] Manning C. D., Schutze H. Foundations of Statistical Natural Language Processing. Cambridge Massachusetts: MIT Press, 1999.

[37] Lafferty J., McCallum A., Pereira F. Conditional Random Fields: Probabilistic Models for Segmenting and Labeling Sequence Data. ICML. 2001. pp. 282-289.

[38] Aramaki E., Miyo K. Automatic Deidentification by Using Sentence Features and Label Consistency. i2b2 Workshop on Challenges in Natural Language Processing for Clinical Data. 2006.

[39] Erik F., Sang T. K., Veenstra J. Representing Text Chunks. EACL. 1999. pp. 173–179.

[40] Guillen R. Automated De-Identification and Categorization of Medical Records. i2b2 Workshop on Challenges in Natural Language Processing for Clinical Data. 2006.

[41] Guo Y., Gaizauskas R., Roberts I., Demetriou G., Hepple R. Identifying Personal Health Information Using Support Vector Machines. i2b2 Workshop on Challenges in Natural Language Processing for Clinical Data. 2006.

[42] Cortes C., Vapnik V. Support-Vector Networks. Machine Learning. 1995. pp. 273-297.

[43] GATE, A General Architecture for Text Engineering. [Online] Natural Language Processing Group, Sheffield University. http://gate.ac.uk.

[44] Hara K. Applying a SVM Based Chunker and a Text Classifier to the Deid Challenge. i2b2 Workshop on Challenges in Natural Language Processing for Clinical Data. Available as a JAMIA on-line data supplement to the current overview article, at www.jamia.org.

[45] Kudo T., Matsumoto Y. A Boosting Algorithm for Classification of Semi-Structured Text. Empirical Methods in Natural Language Processing. 2004. pp. 301-308.

[46] Szarvas G., Farkas R., Ivan S., Kocsor A., Busa-Fekete R. An Iterative Method for the De-identification of Structured Medical Text. Journal of the American Medical Informatics Association, 2007; 14:[typesetter please place page numbers here, article in current issue].

[47] Freund Y., Schapire R. A Decision-Theoretic Generalization of On-line Learning and an Application to Boosting. Journal of Computer and System Sciences, 1997; 55:119-139.

[48] Quinlan J. R. C4.5: Programs for Machine Learning. Morgan Kaufmann Publishers, 1993.

[49] Wellner B., Huyck M., Mardis S., Aberdeen J., Morgan M., Peshkin L., Yeh A., Hitzeman J., Hirschman L.: Rapidly Retargetable Approaches to De-identification in Medical Records. Journal of the American Medical Informatics Association, 2007; 14:[typesetter please place page numbers here, article in current issue].

[50] Carafe. [Online] MITRE. http://sourceforge.net/projects/carafe.

[51] LingPipe. [Online] Alias-i corporation. http://www.alias-i.com/lingpipe/.

[52] Banko M., Brill E. Mitigating the Paucity-of-Data Problem: Exploring the Effect of Training Corpus Size on Classifier Performance for Natural Language Processing. First International Conference on Human language technology research, 2001: 1-5.