Investigator Name Recognition from Medical Journal Articles: A Comparative Study of SVM and Structural SVM

Xiaoli Zhang, Jie Zou, Daniel X. Le, George R. Thoma
National Library of Medicine, Lister Hill National Center for Biomedical Communications,
8600 Rockville Pike, Bethesda, 20894
1-301-4353245
{zhangxiaol, jzou, daniel, gthoma}@mail.nih.gov

ABSTRACT
Automated extraction of bibliographic information from journal articles is key to the affordable creation and maintenance of citation databases, such as MEDLINE®. A newly required bibliographic field in this database is "Investigator Names": names of people who have contributed to the research addressed in the article, but who are not listed as authors. Since the number of such names is often large, several score or more, their manual entry is prohibitive. The automated extraction of these names is a problem in *Named Entity Recognition (NER)*, but differs from typical NER due to the absence of normal English grammar in the text containing the names. In addition, since MEDLINE conventions require names to be expressed in a particular format, it is necessary to identify both first and last names of each investigator, an additional challenge. We seek to automate this task through two machine learning approaches: Support Vector Machine and *structural* SVM, both of which show good performance at the word and chunk levels. In contrast to traditional SVM, structural SVM attempts to learn a sequence by using contextual label features in addition to observational features. It outperforms SVM at the initial learning stage without using contextual observation features. However, with the addition of these contextual features from neighboring tokens, SVM performance improves to match or slightly exceed that of the structural SVM.

Categories and Subject Descriptors
H.3.3 [**Information Storage and Retrieval**]: Information Search and Retrieval – *Retrieval models*; I.2.7 [**Artificial Intelligence**]: Natural Language Processing - *Text analysis*; I.7.5 [**Document and Text Processing**]: Document Capture – *Document Analysis*.

General Terms
Algorithm, Design, Experimentation, Performance.

Keywords
Investigator Name, Named Entity Recognition, Support Vector Machine (SVM), Structural SVM, Document analysis, MEDLINE

DAS'10, *June 9-11, 2010, Boston, MA, USA*.

1. INTRODUCTION
MEDLINE®, the flagship database of the U.S. National Library of Medicine, contains over 17 million citations to the medical journal literature and is a critical source of information for biomedical research and clinical medicine. With the rapid increase of journal literature indexed by MEDLINE every year, it is essential to have automatic methods to retrieve bibliographic data, including article titles, author names, affiliations, abstracts and so on.

Beginning with journals published in 2008, personal names of those who are not entered as authors but belong to members of corporate organizations are required to be included in a new "Investigator Names" field in MEDLINE citation. The addition of these investigator names to MEDLINE allows retrieving information on the collaborative research one has taken part in. The investigator names are usually listed in one or several paragraphs in those articles containing such names. The investigator name paragraphs can appear at the beginning of the article, right below the author section or at the end of the article, in the appendix or footnote. It is common for an investigator name paragraph to contain over a hundred names, and sometimes well over a thousand. Manual extraction of these names is time-consuming, costly, tedious and error-prone.

Automatic investigator name recognition is a two-step process: (1) locate investigator name paragraphs; and (2) parse the paragraphs to extract investigator names. In this article, we assume investigator name paragraphs have already been identified by a preceding automated method or by a human operator. In this paper we discuss the second step, parsing the paragraph to recognize the names.

Figure 1 shows three examples of investigator name paragraphs. The investigator names are usually mixed with institute names, addresses, degrees and many other entities, which usually are not arranged into sentences complying with English grammar. In most cases, they freely co-occur with only some separators, e.g., commas, parentheses, or even spaces, in between. For most investigator names, first names precede the last names, but they may be in the reverse order, as in the example shown in Figure 1(c). The first name can be a complete word or just initials. MEDLINE conventions oblige us to identify not only names, but also their particles. In other words, the first and last names of each investigator need to be identified. For some long names, such as Vicente Rodrìguez Pappalard or Francisco J. García De La Corte shown in Figure 1(a), this is not a trivial task.

Extracting investigator names is a named entity recognition (NER) problem, but the variations and special requirements discussed above pose new challenges. Existing NER algorithms

ISTAPS Project Investigators (IAPP Network node)

- **Andalucia-Al Andalus**: Isabel Fernandez Fernandez (Network node responsable) Transito Cebrian and Inmaculada Romero (Locally research responsible), Beatriz López, César J. Costa, Ma Isabel Villafuerte, Vicente Rodríguez Pappalard, Antonia Aguilera, José Manuel Santos, Filomena Ballester, Francisco J. García De La Corte, Emilio Márquez

(a)

The following investigators participated in the Thrombolysis in Cardiac Arrest (TROICA) trial (the number of patients enrolled is given in parentheses). **Germany (199):** Charité, Campus Benjamin Franklin, Berlin — H.R. Arntz, J. Breckwoldt, D. Müller; Charité, Campus Rudolf Virchow, Berlin — U. Frei, L. Nibbe; Charité, Campus Humboldt, Berlin — S. Behrens, B. Lehmke; Deutsches Rotes Kreuz Kliniken Westend, Berlin — M. Toursarkissian; Städtische Kliniken, Bielefeld — H.P. Milz, A. Roeper; Knappschafts-Krankenhaus, Dortmund — U. Schniedermeier; Albert-Ludwig-University, Freiburg — C. Bode, T. Schwab; Georg-August University, Göttingen — M. Roessler; Martin-Luther University, Halle/Saale — S. Grond, O. Meyer; Ruprecht-Karls University, Heidelberg — B.W. Böttiger, A. Gries, J. Motsch, F. Spöhr; Friedrich-Schiller-University, Jena — K. Pahlke, J. Reichel, K. Reinhart; University Hospital Schleswig Holstein, Kiel — A. Seidenstücker, R. Simon; Berufsfeuerwehr, Kiel — M. Corzillius; Ruprecht-Karls University, Mannheim — H. Genzwürker, T. Viergutz; Technical University, Munich — M. Blobner, M. Heim; Klinikum Saarbrücken, Saarbrücken — K.H. Altemeyer, K. Flick, H. Krieter, T. Schlechtriemen; Ulm University, Ulm — B. Dirks, F. Weißer. **Belgium (152):** Algemeen Ziekenhuis (AZ) Sint-Jan Autonomische Verzorginginstelling, Brugge — P. Martens; Academisch Ziekenhuis, Vrije Universiteit, Brussels — L. Corne, S. Hachimi-Idrissi; Université Catholique de Louvain, St.-Luc, Brussels — P. Meert; Centre Hospitalier Universitaire (CHU) Saint-Pierre, Brussels — B. Claessens, D. de Longueville, P. Mols; Centre Hospitalier (CH) de Jolimont-Lobbes Hospital, Haine St. Paul-La Louvière — J.M. Jacques; Universitaire Ziekenhuizen, Leuven — D. Desruelles, M. Sabbe; Centre Hospitalier Régional (CHR) de la Citadelle, Liège — M. Vergnion; AZ Groeninge-Campus Maria's Voorzienigheid, Kortrijk — V. van Belleghem; CHU de Tivoli, La Louvière — P. Lefebvre, L. Stamatakis; CHR de Namur, Services Mobiles d'Urgence et de Réanimation (SMUR), Namur — G. Mazairac. **The Netherlands (133):** Regionale Ambulance Voorziening Gelderland Zuid, Nijmegen — P. van Grunsven; Canisius-Wilhelmina Hospital, Nijmegen — B.T.J. Meursing; University Medical Center St. Radboud, Nijmegen — W. Keuper, F. Verheugt; Medisch Centrum Rijnmond Zuid-Locatie Zuider, Rotterdam — T. Bruning; Ambulance Zorg Rotterdam Rijnmond, Rotterdam — P. van Loenen; Erasmus Medical Center, Rotterdam — M. Ent, J.M. Mekel; Sint Franciscus Gasthuis, Rotterdam — P.R. Nierop; Ruwaard van Putten Hospital, Spijkenisse — G.J. van Beek. **Austria (115):** Innsbruck Medical University, Innsbruck — M. Baubin, W. Lederer, M. Moritz, M. Luger, V. Wenzel; Graz Medical University, Graz — G. Brunner, G. Prause, H. Walch, A. Wasler, W. Weihs; Landeskrankenhaus St. Johanns-Spital, Salzburg — F. Chmelizek, S. Edtinger, A. Franz, E. Frauenschuh, E. Miller, T. Michalski; Wilhelminenspital, Vienna — K. Huber; Emergency Medical Services, Vienna — A. Kaff, R. Malzer; Vienna Medical University, Vienna — A. Geppert, N. Riechling, E. Riedmüller, W. Schreiber, H. Herkner, F. Seibert; Krankenhaus der Rudolfstiftung, Vienna — B. Enzelsberger, J. Slany, A. Valentin. **France (110):** Hôpital Jean Minjoz, Besançon — G. Capellier; Hôpital Avicenne, Service d'Aide Médicale d'Urgence (SAMU) 93, Bobigny — F. Lapostolle; Hôpital Henri Mondor, SAMU 94, Créteil — C. Brun-Buisson, A. Margenet-Baudry; Hôpital André Mignot, SAMU 78, Le Chesnay — J.P. Bedos, Y. Lambert; Centre Hospitalier Régional Universitaire, SAMU 59, Lille — P. Goldstein; CH Site de St. Luise, La Rochelle — M. Barboteau; CHU Marc Jacquet, Melun — K. Tazarourte; SAMU de Paris, Hôpital Necker, Paris — A. Cariou, P. Carli; Hôpital Charles Nicolle, SAMU 76, Rouen — H. Eltchaninoff, B. Jardel; Centre Hospitalier Purpan SAMU, Toulouse — S. Charpentier; Centre Hospitalier Général, Voiron — C. Escallier. **Norway (95):** Haukeland Universitetssykehus, Bergen — B. Vikenes; Sykehuset Ostfold, Moss — M. Ostensvig; Ullevål University Hospital, Oslo — L. Wik; Sykehuset Telemark Helseforetak, Skien — N.A. Waagsboe; Sentralsykehuset Rogaland, Stavanger — T.W. Lindner; Sykehuset Vestfold, Tonsberg — J.E. Steen-Hansen. **Italy (86):** Azienda Ospedalieria Ospedali Riuniti, Bergamo

(b)

The following Investigators and Institutions participated in NIMISCAD (Non Invasive Multicenter Italian Study for Coronary Artery Disease):

Radiologists:

Marano Riccardo, MD (A. Gemelli Hospital, Catholic University, Rome, Italy); Liguori Carlo, MD (A. Gemelli Hospital, Catholic University, Rome, Italy); Bonomo Lorenzo, MD (A. Gemelli Hospital, Catholic University, Rome, Italy); De Cobelli Francesco MD (S. Raffaele Scientific Institute and Vita-Salute University, Milan, Italy); Esposito Antonio, MD (S. Raffaele Scientific Institute and Vita-Salute University, Milan, Italy); Del Maschio Alessandro, MD (S. Raffaele Scientific Institute and Vita-Salute University, Milan, Italy); Becker Christoph, MD (Ludwig-Maximilians University, Munich, Germany); Herzog Christopher, MD (J.W. Goethe University, Frankfurt, Germany); Centonze Maurizio, MD (S.Chiara Hospital, Trento, Italy); Coser Daniela, MD (S.Chiara Hospital, Trento, Italy); Morana Giovanni, MD (Cà Foncello Hospital, Treviso, Italy); Salviato Elisabetta, MD (Cà Foncello Hospital, Treviso, Italy); Gualdi Gian Franco, MD (DEA Umberto I Hospital, La Sapienza University, Rome, Italy); Casciani Emanuele, MD (DEA Umberto I Hospital, La Sapienza University, Rome, Italy); Ligabue Guido, MD (University of Modena and Reggio Emilia, Italy); Fiocchi Federica, MD (University of Modena and Reggio Emilia, Italy); Pontone Gianluca, MD (Centro Cardiologico Monzino, Milan, Italy); Andreini Daniele, MD (Centro Cardiologico Monzino, Milan, Italy); Catalano Carlo, MD (Umberto I Hospital, La Sapienza University, Rome, Italy); Carbone Iacopo, MD (Umberto I Hospital, La Sapienza University, Rome, Italy); Chiappino Dante, MD (G. Pasquinucci Hospital, Massa, Italy); Midiri Massimo, MD (DIBIMEL, University of Palermo, Italy); Simonetti Giovanni, MD (Tor Vergata University, Rome, Italy); Marchisio Filippo, MD (University of Turin, Italy); Olivetti Lucio, MD (Istituti Ospitalieri of Cremona, Italy); Fattori Rossella, MD (S. Orsola University Hospital, Bologna, Italy); Scardapane Arnaldo, MD (Policlinico of Bari, Italy); Principi Massimo, MD (S. Maria Hospital, Terni, Italy); Romano Luigia, MD (A. Cardarelli Hospital, Naples, Italy); Arcadi Nicola, MD (Ospedali Riuniti of Reggio Calabria, Italy); Profili Manuel, MD (Istituto Clinico Humanitas, Rozzano, Milan, Italy); Volterrani Luca, MD (S. Maria alle Scotte Hospital, University of Siena, Italy).

(c)

Figure 1: Three examples of investigator name paragraphs.

usually expect sentences to follow natural language grammars, and do not identify name particles (first and last names), and therefore cannot be directly used for our recognition problem. We designed and compared two algorithms based on state-of-the-art machine learning tools, SVM and structural SVM. Both approaches achieve good recognition accuracies, and comparing them also reveals some interesting issues.

The rest of the paper is organized as follows: In Section 2, we review the related work in named entity recognition and also briefly describe SVM and structural SVM. In Section 3, we describe our method, including preprocessing, feature extraction, SVM and structural SVM classification and post-processing. Both SVM methods are evaluated and compared in Section 4. Finally, Section 5 provides the summary.

2. RELATED WORK

Investigator name recognition falls in the general category of named entity recognition (NER), which typically involves the

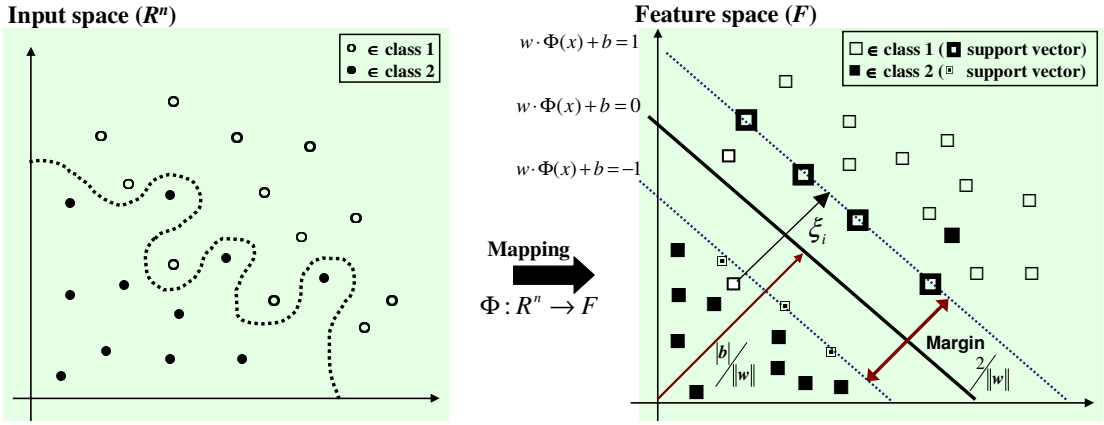

Figure 2: SVM learning algorithm for a non-linearly-separable case.

identification of named entities such as persons, organizations, locations, dates, times, etc., and has been well researched [16]. Currently, the main technique to address NER is supervised learning: Support Vector Machines [2], Naïve Bayesian classifier [21], and Decision Trees [18]. Due to the context existing among named entities, sequential learning is popular as observed from the successful application of HMM on name recognition in news corpora [3], MEMM on FAQs segmentation [14], and CRF on NER in news articles [15]. Compared to NER in the newswire domain, biomedical NER, used to identify technical terms in the biomedical domain (e.g. gene, protein, DNA, etc.), is more challenging and of increasing interest [1, 12]. Several machine learning approaches have been proposed for this domain, including Support Vector Machine [13] and Conditional Random Field [19], as well as combinations of several methods to further improve performance [20].

NER is very application dependent even though the same learning algorithm applies [5, 17]. SVM has been successfully used in many NLP research areas particularly for NER tasks. We implement two approaches – SVM and structural SVM in a new domain, namely, identifying the first and last names of investigators from medical journal articles.

Given a training set $\{(X_1, y_1), \cdots (X_L, y_L)\}$, where X_i is a feature vector and $y_i \in \{-1, +1\}$ is the corresponding label, SVM constructs a linear separable hyperplane with maximum margin between classes in a high-dimensional feature space by a nonlinear transformation of the input space $\phi(X)$ [7]. The hyperplane with the normal vector W is determined by maximizing the margin $2/\|W\|$, which is a primal optimization problem:

$$\min_{w,b} \quad \frac{1}{2}\|W\|^2 + C\sum_{i=1}^{L}\xi_i$$
$$s.t. \quad y_i(W^T\phi(X_i) + b) \geq 1 - \xi_i \quad \forall i = 1, \cdots, L$$
$$\xi_i \geq 0, \quad i = 1, \cdots, L$$

where ξ_i is a slack variable that allows a large soft margin with small errors. The introduction of slack variables helps solve more general classification problems where two classes are not strictly separable even in high feature space. Figure 2 shows the SVM converting a non-linearly-separable problem to a linear classification task by mapping the original input space to a higher dimensional feature space using a nonlinear transformation function $\phi(X)$. Instead of directly computing the mapping of input features in the primal optimization problem, we define a kernel function which is the inner product between a pair of input data mappings to solve the equivalent dual reformulation. The four most frequently used kernel functions include linear, polynomial, radial basic function (RBF), and sigmoid [4, 23].

Though SVM was originally introduced as a supervised learning algorithm for binary-class categorization, it has been extended to solve multi-class problems [8, 24, 25]. Owing to its generalizability especially in the presence of a large number of features, SVM has been used in a wide variety of applications such as text categorization [10], computer vision, speech recognition, gene classification, etc. [9].

Structural Support Vector Machines (Structural SVMs), first proposed in [22], is designed for predicting structured outputs, such as sequences, trees and graphs. Given a set of pairs of inputs $x \in X$ and discrete outputs $y \in Y$, $(x_1, y_1), \cdots (x_n, y_n) \in X \times Y$, structural SVMs exploit the structure and dependencies within *Y*, and perform supervised learning to approximate a mapping $f : X \to Y$. Structural SVM learns a discriminant function, which is a linear combination of some combined feature representations of inputs and outputs: $F(x, y, w) = w^T\Psi(x, y)$, where w is a parameter vector, and feature representation Ψ depends on the nature of the problem. For any given input *x*, structural SVM derives the prediction by maximizing *F* over *Y*: $f(x; w) = \arg\max_{y \in Y} w^T\Psi(x, y)$, which can usually be solved by efficient algorithms, such as Viterbi and CKY algorithms.

Structural SVM takes discriminative training to estimate the parameter vector *w*. Its training generalizes the maximum-margin principle employed in the traditional SVM. The training of structural SVMs can be very computation intensive. Recently, Joachims et al. proposed an efficient training algorithm, named "l-slack cutting-plane", which makes training on large databases feasible [11].

Structural SVMs can build highly complex, but still accurate discriminative models and show promising results in several

Table 1. 62 features extracted from each token for investigator name recognition

Dictionary features (1-6)	
First Name	Is the word in First Name dictionary?
Last Name	Is the word in the Last Name dictionary?
Affiliation	Is the word in the Affiliation dictionary?
Country	Is the word in the Country name dictionary?
US & Canada state	Is the word in the US state or Canadian province dictionary?
Degree	Is the word in the Degree List?
Text features (7-15)	
Name initial pattern	Is the word a pattern of initials, e.g. J., J.Z., J.-Z.?
First character upper case	Is the first character of the word an upper case letter?
All character upper case	Are all characters of the word upper case letters?
Diacritics	Does the word include diacritics?
All letters	Are all characters of the word letters?
All digits	Are all characters of the word digits?
All digits or letters	Does the word contain only digits and letters?
Ended with "'s"	Does the word end with "'s"?
Started with "Mc"	Does the word start with "Mc"?
Punctuation features (16-35)	Is the word preceded or followed by the following 10 punctuation marks: ':', ',', '-', '–', '—', ';', '(', ')', '\n', '*'?
Special word features (36-62)	Is the word one of the following: and, at, center/centre/centro, chair, chairman, co, college, coordinator, director, disease, for, group, hospital/hôpital, in, institute, investigator/investigadore, medical, member, of, PI, PhD/Ph.D., research, school, study, the, university/universitario/universitaire, van?
Contextual features	The features from neighboring words.

areas, such as classification with taxonomies, named entity recognition, sequence alignment and natural language context-free grammar parsing. We implemented a structural SVM algorithm for our investigator name recognition problem and compared it to our parsing algorithm using traditional SVM.

3. METHODS

In our task, each entity which we call a *token* in the subsequent discussion is a single word in the investigator name paragraph. As shown in Figure 1, words in an investigator name paragraph are separated by spaces and punctuations. Before investigator name recognition, preprocessing is conducted to segment the paragraph into tokens based on the spaces and punctuations.

3.1 Feature Extraction

Five types of features - dictionary features, text features, punctuation features, special word features, and contextual features are used in investigator name recognition. All our features are binary features, and they are described in Table 1.

Dictionary features

Dictionary features are collected by looking up First Name List, Last Name List, Affiliation Key Word List, Country Name List, US Canada State List and Degree List. We built these lists from MEDLINE data for about 8 million medical articles. If a candidate word is found in one of these lists, we set the corresponding dictionary feature to 1.

Text features

The text features examine character cases and special characters in a word. A word with all upper case characters can be an abbreviation of degree, state or special words. A name initial pattern appears as a capital (A-Z) usually followed by a period. Words containing digits can be excluded as names. These text features provide important information to distinguish named entities of different types.

Punctuation features

Due to the regularity of the appearance of groups of named entities in an investigator name paragraph, punctuations like spaces, commas, periods, hyphens, dashes, semi-colons, brackets, etc. before and after a word are important features and can signify that adjacent words are in the same group of named entities or have the same entity type. For example, a semi-colon or a comma before and after a word often indicates the start of a new group of named entities. Hyphens are generally used to connect words of the same entity type. A name or affiliation very likely consists of words separated by spaces. For each punctuation mark listed in "Punctuation features" in Table 1, we add two features of value 1: if the character before a word is the specific punctuation and if the character after a word is the punctuation.

Special word feature

Special words can eliminate the class confusion. For example, university, institute and hospital are often associated with affiliations. Investigator, coordinator or manager indicates a person's position usually followed by a name.

Contextual features

The features from neighboring tokens can be very informative. For example, in Figure 1(c), the word "Hospital" in "A. Gemelli Hospital" clearly indicates that "A. Gemelli" is not an investigator name. Therefore, to take advantage of the contextual dependencies between tokens, contextual features from neighboring words are also extracted for each token.

3.2 SVM and Structural SVM Classification

An SVM is a supervised learning method which involves training and test stages. The goal is to produce a model using a training set

and to predict unknown test data given the model. Our investigator name recognition is a three-class (First Name, Last Name, and Other) classification problem. A total of 62 observation features including dictionary, text, punctuation, and special word features are used to represent each word token. In addition, due to the context existing among tokens, the observation features from neighboring tokens are also used for SVM classification.

An essential step in designing a structural SVM is to define its feature presentation function $\Psi(x, y)$. Our investigator name recognition is a sequence labeling problem. Therefore, $\Psi(x, y)$ includes two kinds of features: state transition features and observation features extracted from individual tokens. State transition features model only the adjacent label dependencies. We use first order transition dependencies, i.e., only the dependencies between adjacent token labels are modeled. For observation features, we use the same 62 features defined in Section 3.1. We also experimented with adding contextual observation features from neighboring tokens. Details are described in Section 4.

Structural SVM applied specifically for sequence labeling is sometimes called SVMHMM, possibly because it uses similar types of feature representations as Hidden Markov Models. We used the SVM^{HMM} library, available at [26], to implement our structural SVM algorithm for investigator name recognition.

3.3 Postprocessing

After SVM or structural SVM classification, every token is assigned a label. However, a post-processing step is still required to analyze the labeled paragraph and then derive individual complete names by finding corresponding name particles. We take a heuristic approach based on the following rules:

- Consecutive first name tokens and last name tokens form first name chunks and last name chunks, respectively. Within a first or last name chunk, no punctuations are allowed.
- A name can consist of a first name chunk and a last name chunk pair, or a last name chunk only. A name with a first name chunk only is prohibited. Therefore, isolated first names would be removed.
- If a name consists of both a first name chunk and a last name chunk, either one can be in the front, and no punctuations are allowed between them.

Following these rules, an algorithm can be implemented to remove isolated single first name labels and organize the remaining first name and last name tokens into complete names.

4. EVALUATION

By searching MEDLINE citations after 2008, we found 370 articles which have investigator name paragraphs. After obtaining the full text of these articles, we manually identified investigator name paragraphs in the articles and saved them as plain text files. The ground truth labeling of these investigator name paragraphs is then created semi-manually. We randomly selected 100 from those 370 articles for training and reserved the remaining 270 articles for testing. Some statistics of this data collection are listed in Table 2.

We evaluate algorithm performance at two levels. One is at the token level, i.e., the labeling accuracy of individual tokens. The other is at the name chunk level, i.e., the precision and recall of retrieving individual full names. At the name chunk level, a name is considered correctly retrieved only when both first name and last name tokens are correctly labeled. For example, for the name, "Francisco J. García De La Corte", shown in Figure 1(a), we accept as a correct chunk labeling only when all three tokens "Francisco J. García" are labeled as the first name chunk, and all three tokens "De La Corte" are labeled as the last name chunk. A false-negative and a false-positive are counted as such, even if only a single token is mislabeled. Therefore, chunk-level evaluation is much more rigorous than token-level evaluation.

Table 2. Dataset statistics

	Training	Test
Articles	100	270
Total Tokens	22,077	74,864
First Name Tokens	5,393	19,013
Last Name Tokens	5,308	17,560
Other Tokens	11,376	38,291
Total Names	4,607	16,570

4.1 Evaluation of SVM Method

We use LibSVM [6], an SVM library developed at National Taiwan University, to implement our token classification. Radial Basis Function (RBF) is adopted as the kernel function. The two parameters in RBF, C (penalty parameter of the errors) and γ (RBF parameter), are optimized by an exhaustive grid-search using cross-validation on the training samples.

To observe the effects from neighboring tokens, 62 basic observation features together with different orders of contextual observation features are used in our SVM token classification. The "k^{th} order contextual observation features" means the observation features from k neighboring tokens on either side. For a token, each order of contextual features added implies that one token from either its left or right side contributes 62 observation features. The feature dimensionality of the current token is thereby extended by 124 (2 x 62). Considering the complexity, we compare the evaluation results only up to the second order contextual features. Tables 3 and 4 show the SVM evaluation at token and name chunk levels with 62 initial observation features, with 186 (62 + 2 x 62) features by adding the first order contextual observation features, and with 310 (62 + 4 x 62) features by adding the second order contextual observation features, respectively. Note that the accuracy increases significantly as the observation features extracted from neighboring tokens increase. The contextual information is very helpful to SVM modeling as there are potential dependencies among a sequence of tokens.

4.2 Evaluation of Structural SVM Method

We used SVM^{HMM}, an implementation of structural SVMs for sequence labeling by Thorsten Joachims [26], to conduct our experiments. In structural SVM method, the same 62 observation features are extracted from each individual token. We used linear-kernel due to the fact that other kernels, e.g., RBF, can be extremely computation intensive. Meta-parameters are determined with cross-validation on training samples.

Tables 5 and 6 show the evaluation at token and name chunk levels. Even though the structural SVM algorithm has already considered the contextual label information through state transition features, it is still of interest to know whether the features extracted from neighboring tokens would further increase the accuracy. Therefore, besides using 62 observation features

Table 3. Accuracy of token classification using SVM method

	First Name	Last Name	Other	Overall
Before post-processing				
Features from the token itself	90.52%	79.78%	86.27%	85.82%
Features from the token and its two neighbors	89.77%	93.53%	96.10%	93.89%
Features from the token and its four neighbors	93.32%	93.71%	96.63%	95.10%
After post-processing				
Features from the token itself	81.18%	84.61%	92.33%	87.69%
Features from the token and its two neighbors	90.35%	94.55%	96.89%	94.68%
Features from the token and its four neighbors	92.24%	94.79%	97.65%	95.60%

Table 4. Precision and recall of full name extraction using SVM method

	Precision	Recall	F-Measure
Features from the token itself	77.38%	78.82%	78.09%
Features from the token and its two neighbors	88.01%	88.90%	88.45%
Features from the token and its four neighbors	91.72%	91.03%	91.37%

Table 5. Accuracy of token classification using structural SVM method

	First Name	Last Name	Other	Overall
Before post-processing				
Features from the token itself	87.11%	91.49%	94.78%	92.06%
Features from the token and its two neighbors	91.90%	94.41%	94.89%	94.02%
Features from the token and its four neighbors	91.87%	92.95%	95.79%	94.13%
After post-processing				
Features from the token itself	86.83%	91.39%	96.30%	92.75%
Features from the token and its two neighbors	91.69%	94.43%	96.25%	94.67%
Features from the token and its four neighbors	91.61%	93.06%	96.81%	94.61%

Table 6. Precision and recall of full name extraction using structural SVM method

	Precision	Recall	F-Measure
Features from the token itself	87.48%	85.05%	86.24%
Features from the token and its two neighbors	89.45%	89.35%	89.40%
Features from the token and its four neighbors	91.35%	88.82%	90.07%

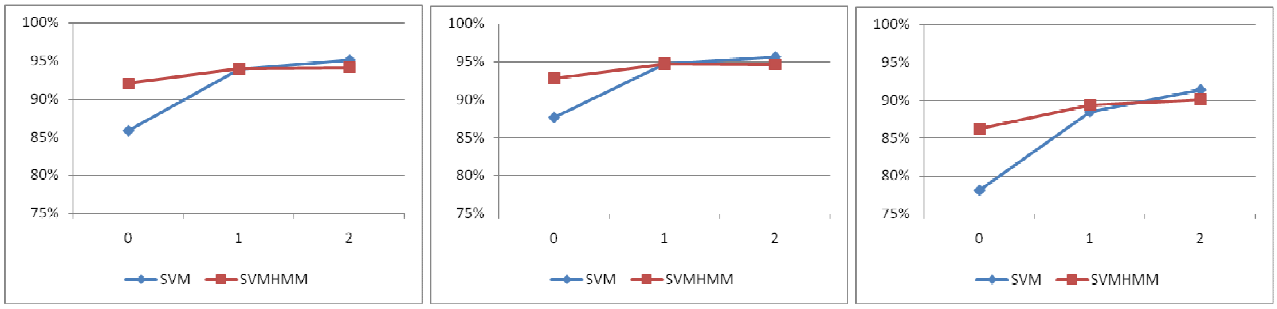

Figure 3: Performance values from Tables 3-6 plotted against *k*-order contextual observation features. Left: token classification accuracy before post-processing; Middle: token classification accuracy after post-processing; Right: name retrieval F-Measure.

extracted from the token itself, we also experimented with adding observation features extracted from neighboring tokens. Using the same settings as for SVM method evaluation, in Tables 5 and 6, we compare the results from using the features extracted from the token itself (62 features), the features extracted from the token itself and two adjacent (immediate one left and one right) neighbors (186 features), and the features extracted from the token itself and four neighboring (two left and two right) tokens (310 features).

4.3 Discussion

We summarize the following observations from our experiments. First of all, the information from neighboring tokens is very helpful and must be utilized. There are two kinds of contextual features: the labels assigned to the neighboring tokens and the observation features extracted from the neighboring tokens. We call the first one *contextual label features* and the second one *contextual observation features* in the following discussion.

In our opinion, the most important difference between our implementations of the two methods is that the SVM method uses only the contextual observation features, while structural SVM method may use both types of contextual features.

For SVM method, when we use only the observation features from the token itself, no contextual features are used, thereby providing a baseline performance. As expected, the performance

is relatively low: the overall token classification accuracies before and after post-processing are 85.82% and 87.69%, respectively (Table 3), and the F-Measure of the name chunk retrieval is 78.09% (Table 4). After combining the observation features from immediate left and right neighbors, the corresponding accuracies and F-Measure significantly increase to 93.89%, 94.68% and 88.45%. This clearly indicates the importance of the first order contextual observation features. After combining observation features from one further left and one further right neighbors, the corresponding accuracies and F-Measure increase to 95.10%, 95.60% and 91.37%. This indicates the second order contextual observation features are still helpful, but less significantly than the first order contextual observation features.

For structural SVM method, when we use only the observation features from the token itself, it does not use any contextual observation features, but it does use the contextual label features (see Section 3.2 for the discussion on state transition features of structural SVM). The token classification accuracies before and after post-processing are 92.06% and 92.75%, respectively (Table 5), and the F-Measure of name chunk retrieval is 86.24% (Table 6), much better than that of the SVM method in the same setting. This clearly indicates that contextual label features can be very helpful. After we add contextual observation features, the performance increases are much less significant compared to the SVM method at the same settings. This may indicate that the discriminative information provided by contextual observation and contextual label features are redundant. After adding the second order contextual observation features, there is no performance gain for structural SVM method even though it uses extra contextual label features.

For better visualization, we have plotted the performance data extracted from Tables 3, 4, 5 and 6 in Figure 3.

We also observe that after post-processing, the token classification accuracies for First Name all decrease, but the token classification accuracies of Other all increase. This is due to the second rule we used in post-processing and listed in Section 3.3. This rule re-assigns Other label to all tokens which are labeled as isolated First Name. This rule would make errors for some tokens, which are indeed First Name (though their corresponding Last Name tokens are mislabeled). On the other hand, this rule also helps correct many Other tokens, which are mislabeled as First Name. Overall, this rule increases performance.

4.4 Error Analysis

Partial screen-dumps of our GUI (Graphic User Interface) program for visually examining investigator name recognition results are shown in Figure 4. In this GUI program, the first name chunks are marked in red and the last name chunks are marked in blue. Most of the investigator names in these two samples are recognized correctly. Notice that in most cases, the algorithm recognizes those organizations named after people, e.g., Lozano Blesa Hospital in Figure 4(b).

Figure 4 also illustrates three kinds of possible recognition errors. For example, Figure 4(a) shows an under-labeling error (marked in blue), which is an uncommon name. Figure 4(b) illustrates an over-labeling error and a mis-chunking error (pointed by two arrows). San Sebastian is a city name, but the algorithm mislabels it as an investigator name. The first name chunk of "J. López del Val" should be "J. López" and the last name chunk should be "del Val". The algorithm mislabeled the word "López", and this error adds a false positive (because an extra false name is labeled) and a false negative (because the true name is mislabeled). This kind of mis-chunking error is the most common type for both SVM and structural SVM methods, and causes the large drop from the accuracy of overall token classification to the F-measure of full name recognition. It is not an easy task to eliminate this kind of error, and further research is required.

5. SUMMARY

We have implemented and evaluated two investigator name recognition methods. SVM method uses the observation features from the token itself and contextual observation features from neighboring tokens. Structural SVM method further utilizes contextual label features. Both contextual (observation and label) features provide important information and are very helpful for improving the recognition performance. After combining the

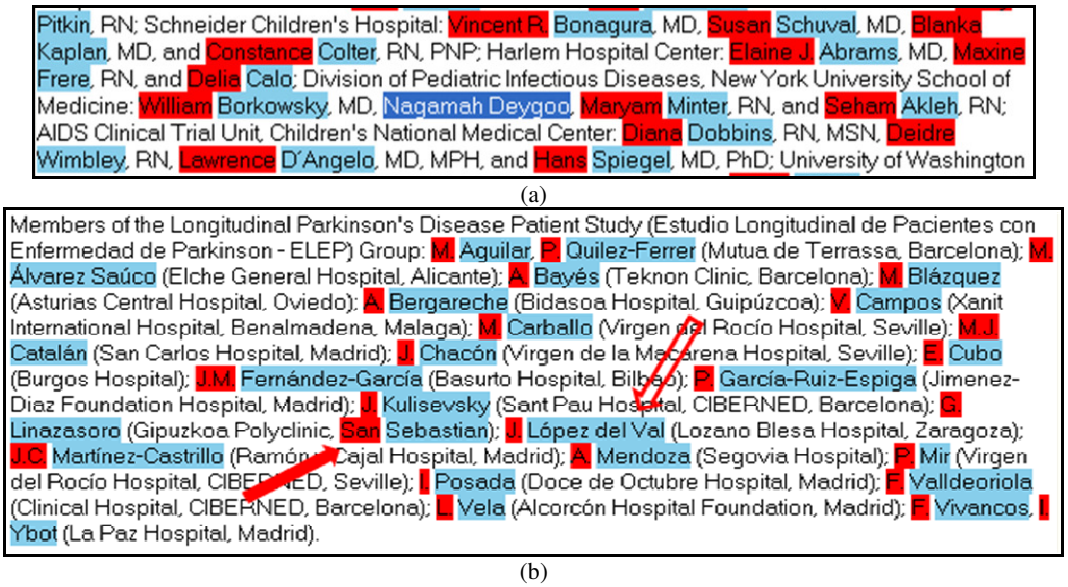

Figure 4: Two examples of visually examining investigator name recognition. Recognized first name chunks are marked in red, and recognized last name chunks are marked in blue. Three errors are discussed in text.

second order contextual features, both methods achieve above 94% overall token classification accuracy and above 90% full name recognition F-measure score. We are in the process of applying the proposed scheme to author name recognition since similar features are shared by author zones and investigator name paragraphs in an article. We also note that investigator name recognition is a structural learning problem due to the regular structures of investigator names organized in a paragraph. For example, in Figure 1(c), each name is followed by a degree, and then an affiliation, including institute, city and country, in a parenthesis. Recognizing and utilizing this kind of internal structures may provide a more general and more accurate solution to the investigator name recognition problem.

6. ACKNOWLEDGMENTS

We thank Dr. In-Cheol Kim for help in preparing Figure 2. This research was supported by the Intramural Research Program of the National Institutes of Health (NIH), National Library of Medicine, and Lister Hill National Center for Biomedical Communications.

7. REFERENCES

[1] Ananiadou, S., Friedman, Carol, and Tsujii, Jun'ichi. 2004. Introduction: named entity recognition in biomedicine. Journal of Biomedical Informatics, 37, 6, 393-395.

[2] Masayuki, Asahara, Matsumoto Y. 2003. Japanese named entity extraction with redundant morphological analysis. In Proc. Human Language Technology conference - North American chapter of the Association for Computational Linguistics. 8-15.

[3] Bikel, D.M., Schwartz, R.L., and Weischedel, R.M. 1999. An algorithm that learns what's in a name. Machine Learning, 34, 1-3, 211-231.

[4] Burges, C. J. C. 1998. A tutorial on support vector machines for pattern recognition. Data Mining and Knowledge Discovery, 2, 2, 1-43.

[5] Carvalho, V. R. and Cohen, W. W. 2004. Learning to extract signature and reply lines from email. Proc. of the Conference on Email and Anti-Spam 2004, Mountain View, California.

[6] Chang, C.-C. and Lin, C.-J. 2001. LIBSVM: a library for support vector machines. Software available at http://www.csie.ntu.edu.tw/~cjlin/libsvm.

[7] Corters, C. and Vapnik, V. 1995. Support vector network. Machine Learning, 20, 273-297.

[8] Crammer, K. and Singer, Y. 2001. On the algorithmic implementation of multi-class kernel-based vector machines. Machine Learning Research, 2, 265-292.

[9] Cristianini, N. and Shawe-Taylor, J. 2000. An introduction to support vector machines and other kernel-based learning methods. Cambrige University Press, Cambridge, UK.

[10] Joachims, T. 1998. Text categorization with support vector machine. Proc. Euro. Con. Machine Learning, 137-142.

[11] Joachims, T., Finley, T., Yu, Chun-Nam. 2009. Cutting-plane training of structural SVMs. Machine Learning Journal, 27-59.

[12] Kim, J.D., Ohta, T., Tateishi, Y. and Tsujii, J. 2004. Introduction to the bio-entity recognition task at JNLPBA. Proc. Joint Workshop on Natural Language Processing in Biomedicine and its Applications (JNLPBA).

[13] Lee, C., Hou, W. J. and Chen, H.-H. 2004. Annotating multiple types of biomedical entities: a single word classification approach. Proc. Joint Workshop on Natural Language Processing in Biomedicine and its Applications.

[14] McCallum, A., Freitag, D., and Pereira, F. 2000. Maximum entropy models for information extraction and segmentation. Proc. of the 17th International Conference on Machine Learning, 591-598.

[15] McCallum, A. and Li, W. 2003. Early results for named entity recognition with conditional random fields, feature induction and web-enhanced lexicons. Proc. of the 7th Conference on Natural Language Learning (CoNLL-2003), 4, 188-191.

[16] Nadeau, D. and Satoshi, S. 2007. A survey of named entity recognition and classification. Linguisticae Investigations, 30, 1, 3-26.

[17] Nguyen, N. and Guo, Y. 2007. Comparisons of sequence labeling algorithms and extensions. Proc. of the 24th International Conference on Machine Learning, 681-688.

[18] Satoshi, S., Nobata, C. 2004. Definition, dictionaries and tagger for extended named entity hierarchy. In Proc. Conference on Language Resources and Evaluation.

[19] Settles, B., 2004. Biomedical named entity recognition using conditional random Fields and novel feature sets. Proc. Joint Workshop on Natural Language Processing in Biomedicine and its Applications (JNLPBA).

[20] Si, L., Kanungo, T., Huang, X. 2005. Boosting performance of bio-entity recognition by combining results from multiple systems. Proc. Workshop on Data Mining in Bioinformatics (BioKDD).

[21] De Sitter, A. and Daelemans, W. 2003. Information extraction via double classification. In Proceedings of International Workshop on Adaptive Text Extraction and Mining, Dubrovnik.

[22] Tsochantaridis, I., Hofmann, T., Joachims, T., Altun Y. 2004. Support vector machine learning for interdependent and structured output spaces. Int'l Conf. on Machine Learning.

[23] Vapnik, V. 1995. The nature of statistical learning theory, New York: Springer-Verlag.

[24] Vapnik, V. 1998. Statistical learning theory. Wiley.

[25] Weston, J. and Watkins, C. 1999. Support vector machines for multi-class pattern recognition. In Proc. of the 7th European Symposium on Artificial Neural Networks.

[26] http://www.cs.cornell.edu/People/tj/svm_light/svm_hmm.html.