LAWRENCE
LIVERMORE
NATIONAL
LABORATORY

UCRL-CONF-213354

Information Extraction from Unstructured Text for the Biodefense Knowledge Center

N. F. Samatova, B.-H. Park, R. Krishnamurthy, R. Munavalli, C. Symons, D. J. Buttler, T. Cottom, T. J. Critchlow, T. Slezak

June 30, 2005

Working Together: Research & Development Partnerships in Homeland Security
Boston, MA, United States
April 25, 2005 through April 26, 2005

Information Extraction from Unstructured Text for the Biodefense Knowledge Center

Nagiza F. Samatova[1], Byung-Hoon Park, Ramya Krishnamurthy, Rajesh Munavalli, and Chris Symons, Oak Ridge National Laboratory

David Buttler, Teresa Cottom, Terence Critchlow, and Tom Slezak,
Lawrence Livermore National Laboratory

Abstract

The Bio-Encyclopedia at the Biodefense Knowledge Center (BKC) is being constructed to allow an early detection of emerging biological threats to homeland security. It requires highly structured information extracted from variety of data sources. However, the quantity of new and vital information available from every day sources cannot be assimilated by hand, and therefore reliable high-throughput information extraction techniques are much anticipated. In support of the BKC, Lawrence Livermore National Laboratory and Oak Ridge National Laboratory, together with the University of Utah, are developing an information extraction system built around the bioterrorism domain. This paper reports two important pieces of our effort integrated in the system: key phrase extraction and semantic tagging. Whereas two key phrase extraction technologies developed during the course of project help identify relevant texts, our state-of-the-art semantic tagging system can pinpoint phrases related to emerging biological threats. Also we are enhancing and tailoring the Bio-Encyclopedia by augmenting semantic dictionaries and extracting details of important events, such as suspected disease outbreaks. Some of these technologies have already been applied to large corpora of free text sources vital to the BKC mission, including ProMED-mail, PubMed abstracts, and the DHS's Information Analysis and Infrastructure Protection (IAIP) news clippings. In order to address the challenges involved in incorporating such large amounts of unstructured text, the overall system is focused on precise extraction of the most relevant information for inclusion in the BKC.

1. Introduction

The Bio-Encyclopedia (Bio-E) at the Biodefense Knowledge Center (BKC) [4, 5] is aimed to detect and prevent potential biological threats in the earliest time. By collecting and compiling a huge stack of related information, it will serve various government agencies to query potential bioterrorism and all related resources. In the end, it will be constructed as an elaborate knowledge base that allows effective and timely response to various biological terrorisms. In order to effectively perform its role, the center must incorporate as much relevant information as possible. Also much attention should be paid to ensure the quality of information stored therein.

The underlying data model in the BKC is a semantic graph where concepts are represented as nodes and the relationships between them are denoted as edges. A semantic graph is a powerful data model that permits advanced techniques for uncovering hidden relationships in the data. The information incorporated in the Bio-E is thus very structured. However, much of the information sources are in unstructured text format. For this reason, it is imperative to perform systematic analysis on source documents to extract valuable information in well structured format. More specifically, relevant information from unstructured text should be accurately identified so that it can be perpetuated up to the level of the graph.

The quantities of relevant texts are growing so fast, thus in the near future the amount of data processed daily for the BKC will be beyond the capacity of human curators. In such a context, assimilating free-text data at the level required by the semantic-graph model presents many challenges, and available technologies are inadequate for many requirements within the

[1] Contact Author: samatovan@ornl.gov

bioterrorism domain. Our effort in this direction embraces identification of relevant text entity in two levels: document and phrase. To identify the most relevant documents from a vast volume of corpus, we developed two keywords (or key phrases) extraction methods. To sift information in more detailed form, we developed semantic tagging methods that pinpoint phrases that are most pertinent to the concept under consideration.

This paper provides an overview of our initial effort performed in support of BKC's long-reaching goal. It describes our motivation, methodologies, and the results obtained from our empirical studies. In the first half of the paper, we describe two keyphrase-extraction methodologies, both corpus-based and single-document, which were designed to handle the free-text sources originally identified for inclusion in the BKC. In the latter half of the paper, we describe initial methodologies and preliminary results for applying semantic tags to terms and phrases belonging to semantic categories pertinent to the BKC mission.

2. Keyword Extraction from Both Domain Dependent and Independent Documents

Keywords extraction is a process of selecting a set of words (or phrases) that deliver concise and high-level description of a document. The importance of keyword extraction coincides with the increased need for instant understanding and handling of huge volume of documents for various purposes. The quality of the extracted keywords affects information retrieval processes like text clustering, classification, automatic text summarization, etc. Algorithms for keyword extraction can be classified into two broad categories: corpus dependent and independent approaches. While the former requires a large stack of documents and predetermined keywords to build a prediction model, the latter directly sifts keywords from a document without any previous or background information. Generally it is accepted that corpus dependent approaches yield better performance. However, a prediction model is practically restricted to a single domain, thus the quality of extracted keywords from a new document of unknown domain is not always guaranteed. In this regard, corpus independent (or domain independent) approaches may find many practical applications.

Document preprocessing is an important step in most information retrieval processes. It transforms documents in such a way the most essential information is readily available. In keyword extraction, it involves (1) selection of candidate phrases and (2) merge of candidate phrases for the final analysis. It has been reported that a successful preprocessing can improve the performance by 15-20%. In particular, we empirically verified that the most appropriate preprocessing steps can have a significant impact on the final results. In this section, we introduce two keyword extraction algorithms, domain dependent and domain independent, respectively that we developed for the BKC project. Also we describe various document preprocessing efforts applied during the course of the project.

2.1. Document Preprocessing

Our document preprocessing process includes three techniques to produce a suitable selection of candidate key phrases: stemming, stop word removal, and candidate phrase selection. These techniques help obtain unbiased frequency counting of each word in documents.

- Stemming: Morphological variants of the words could be considered semantically equivalent for information retrieval purposes. Studies have shown that there is a strong correlation between the performance of stemming and Information Retrieval. Hence the

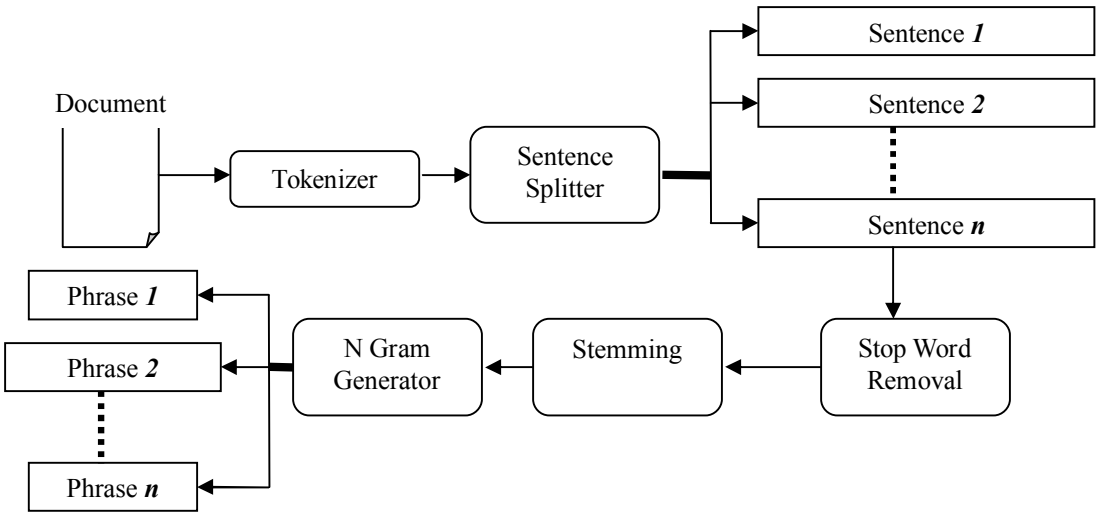

Figure 1. Document preprocessing steps

choice of stemmer can have far reaching effects on the quality of extracted key phrases. In practice, it is a process of collapsing words of the same root into a representative form by truncating suffixes and training numbers. We particularly used Iterative Lovins stemmer [7].

- Stop word removal: A word that does not carry any unique information about a document needs to be removed from candidate keywords. We filtered out such a word using a list of 650 commonly occurring stop words compiled from several text processing literature and SMART system [8].
- Candidate phrase selection: There are several methods that can be utilized for choosing the phrases that can be considered as coherent units, and hence potential key phrases. Utilizing *n*-grams alone can lead to incoherent phrases being output. In addition, inserting these incoherent phrases into the candidate set can negatively impact the scoring of other terms. This problem can be eased by requiring that an *n*-gram occurs a sufficient number of times together. To ensure that other coherent phrases are neglected, *n*-grams are formed by grouping *n* consecutive words in a given sentence. These *n*-grams are then filtered if they either start or end in a stop word. We selected up to 4 grams as candidate key phrases as input to the key phrase extraction algorithms.

2.2. Keyword Extraction Methods

Our implementations of keyword extraction are both based on Term Frequency (TF) of word in a document. For the corpus dependent approach, we further include Inverse Document Frequency (IDF) and Bayesian framework to build a prediction model. For the corpus independent (or domain independent) approach, we exploit statistical unusualness of certain words in terms of their co-occurring frequency patterns with frequent words in a document.

Our corpus dependent method basically implements naïve Bayes classifier, i.e.

$$\Pr[key \mid T, D] = \frac{\Pr[T \mid key] \times \Pr[D \mid key] \times \Pr[key]}{\Pr[T, D]}, \quad (1)$$

where T and D denote the TF-IDF score of candidate key phrase *key* and the location of *key* in the

document under consideration, respectively. A TF-IDF of candidate i is defined as, $tf_{i,j} \times \log(\frac{N}{df_i})$, where $tf_{i,j}$, df_i, and N denote term frequency of i in document j, the number of documents containing i, and the total number of documents in the corpus, respectively. We also consider the relevancy of a candidate key phrase with respect to a predefined domain specific dictionary, which we represent as a probability. Our final corpus dependent model is then constructed by adding the relevancy term to (1).

Our corpus independent (or domain independent) method utilizes a measure of co-occurrence with frequently occurring words as found in [9, 10]. More specifically, a distribution of co-occurrence frequencies is converted to a chi-square value which essentially denotes the degree of importance. Formally, the chi-square value of a candidate term w is computed as,

$$\chi^2(w) = \sum_{g \in G} \left\{ \frac{(freq(w,g) - n_w p_g)^2}{n_w p_g} \right\}, \quad (2)$$

where G denotes a set of frequent terms, n_w is the total number of co-occurrences of w with all the terms in G, and p_g represents the expected occurrence probability of a frequent term g in G.

We note that Chi-square estimation using (2) can be highly unreliable if G includes many terms that are semantically equivalent. To make the estimation more robust, we merged frequent terms by performing hierarchical clustering. Determination of the cut-off value in the dendogram is crucial, yet difficult task. Indeed our preliminary analysis revealed that the quality of the extracted keywords heavily depends on the carefully chosen cut-off values. To eliminate any spurious errors caused by improper choices, we averaged chi-square values measured from different clustering results.

Chi-square values can be compared only when they are computed from the same documents. There are, however, cases when comparisons between chi-square values from different documents are desirable. For example, we often need to choose the documents that are more relevant to the underlying topic from a large stack of corpus. For this reason, our corpus independent keyword extraction method transforms a chi-square value to z-scale following the method developed by Wilson & Hilferty [11] . More technically, the transformation is defined as:

$$W(\chi^2) = \frac{\left(\frac{\chi^2}{d}\right)^{1/3} - \left(1 - \left(\frac{1}{9}\right)\left(\frac{2}{d}\right)\right)}{\sqrt{\left(\frac{1}{9}\right)\left(\frac{2}{d}\right)}},$$

where d denotes the degrees of freedom.

3. Semantic Tagging

Over the last few years there has been a rapid growth in the availability of biomedical literature. Significant amount of valuable scientific information is hidden in these resources. Automatic information extraction is the first step towards extraction of relevant knowledge from these sources. Along the line of our effort to the BKC, we applied semantic tagging techniques to identify relevant terms in categories of our interest.

Semantic tagging, which is often referred to as named-entity recognition (NER), is the process of extracting proper names and classifying them into a set of predefined categories. Semantic tagging provides an added value to the process of information extraction by facilitating the discovery of structure in unstructured data. It is the foundation for building more complex information extraction processes which help build ontological information, event tracking and scenario. Many semantic categories contain polysemous terms, causing dictionary-based semantic tagging to be error-prone. Also some categories contain a wide variety of phrases, which makes construction of a comprehensive dictionary nearly impossible. In addition, new terms of a category can always arise, and we need to classify them as well. For example, we would like to be able to tag a new emerging disease as such, despite not having the term or phrase in a gazetteer.

Our current research is focused on NER of categories like *disease, bacterium, protein, etc.* which are of interest to the BKC. Due to the lapse in standard nomenclature, many terms in these categories have multiple names, abbreviations are common, and often exact phrase boundaries are not clearly defined. We found that there exists no NER tool for *disease* and *bacterium,* whereas there exist several tools for *protein*. Thus, we developed Conditional Random Field (CRF)-based tagger for *disease* and *bacterium*. For *protein*, we combine the outputs of existing tools into a single framework.

3.1 Named Entity Recognition for *disease* and *bacterium* using Linear Chain CRF Model

With the absence of existing tools for tagging *disease* and *bacterium*, we found Conditional Random Field (CRF) to be an appropriate choice. CRFs are discriminative/conditional probabilistic models that maximize conditional probability $\boldsymbol{P(s|o)}$ of label sequence $\boldsymbol{s}$ for a given particular observation sequence $\boldsymbol{o}$. When the real data distribution has higher order dependencies, CRFs have been shown to outperform other labeling and segmenting methods, such as Hidden Markov Models (HMMs) and Maximum Entropy Markov Models (MEMMs) [12, 13]. CRFs have several advantages over other labeling techniques as they relax the strong independence assumptions required by HMMs and avoid the label biasing problem in MEMMs. Moreover, convex nature of CRF loss function (log-likelihood) guarantees convergence of global optimum when estimating CRF parameters. They generalize easily to analogues of stochastic context free grammars which are useful in Natural Language Processing.

Our NER for *bacterium* and *disease* is modeled after the linear chain CRF model (See Figure 2). We particularly follow the structure described in [12], and utilize feature set described in [14]. Here a feature stands for a constraint over a set of states in a CRF. We consider the following features to properly model the NER for *bacterium* and *disease.*

- *Semantic Features:* Semantic domain knowledge in the form of lexicons is used for features. We manually prepared 4 lexicons for this purpose (Known Genus Names for Bacteria, Words resembling Species names, Known Diseases, Disease Indicating Words). These were compiled from different online resources based on the recommendations from domain experts.

- *Orthographic Features:* Several orthographic features are also considered. They are based on regular expressions that describe words ending with a particular substring, alphanumeric, punctuations etc. as well as suffixes and prefixes indicating a particular

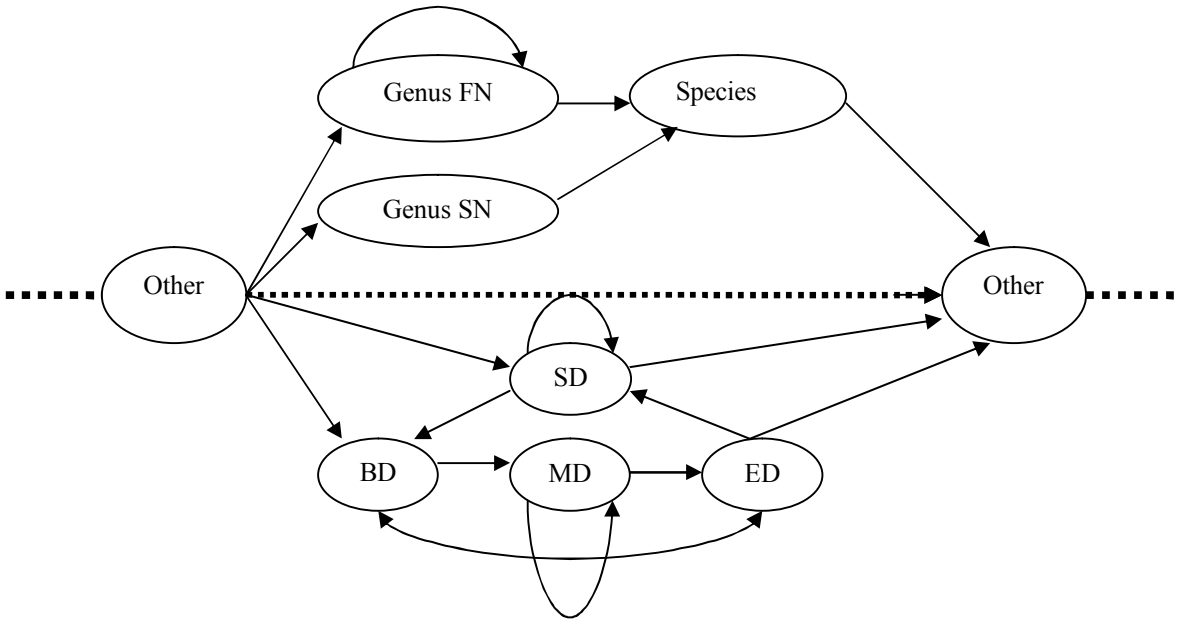

Figure 2: Intuitive integrated NER model for Bacteria and Disease

feature. Apart from these features, words observed in training data are also used as a feature set.

- *Other Features:* Semantic structure of the entity could also serve as a feature set. For example, in *Bacterium, Species* should always be preceded by *Genus* label in the model. These features put additional constraints and help improve the performance of the overall recognition system.

Each state in our CRF model corresponds to a sub-entity. A series of one or more sub-entities forms a named entity. More specifically, we define three entities as follows.

Bacterium = {Genus Full Name (FN), Genus Short Name (SN), Species}
Disease = {Begin Disease (BD), Middle Disease (MD), End Disease (ED), Single Disease (SD)}
Other = {Set of all other words}

Our linear-chain CRF model produces semantic tagging for *bacterium* and *disease* by identifying the most probable state sequences given a series of observed words (i.e. a sentence). Let ***S*** be a set of states (e.g. Single Disease, Genus, Species etc.). Let $s = <s_1, s_2, \ldots, s_n>$ be the sequence of states in ***S***. Let $\boldsymbol{o} = <o_1, o_2, \ldots, o_n>$ be a sequence of observed words of length ***n***. Our linear-chain CRF model computes the conditional probability of a state sequence **s**, given the observed input sequence ***o***. Technically, this is defined as:

$$P(s \mid o) = \frac{1}{Z_o} \exp\left(\sum_{i=1}^{n} \sum_{k} \lambda_k f_k(s_{i-1}, s_i, o, i)\right),$$

where $f_k(s_{i-1}, s_i, o, i)$ is a feature function, o is the observation sequence, i is the current position

in the sequence, s_{i-1} is the originating state, and s_i is the destination state. Z_o is a normalization factor, and λ_k is the weight associated with $f_k(s_{i-1}, s_i, o, i)$. Weights are determined to maximize the log-likelihood of state sequences in the training data. Training is performed via a quasi-Newton non-linear optimization routine, known as L-BFGS (Limited-memory BFGS).

3.2 **Named Entity Recognition for *proteins* using Predictive Data Mining Techniques**

Protein names are one of the most complicated entities to be extracted from unstructured text. It is primarily due to the fact that protein names contain Roman letters, digits, abbreviations, other symbols and unknown words. These names resemble species names, chemical names, mineral, and general English words. Furthermore these names are subject to orthographic variations originating from the differences in capitalization and hyphenation, multi-word formation etc. Hence it is not possible to extract all protein names from a document with high precision rate using a single methodology. Moreover from a performance evaluation of the individual predictors on un-seen data, it is observed that the overlap between the regions of predictions is considerably small. To achieve higher precision rate by utilizing overlapped and non-overlapped regions in predictions, we choose to combine individual predictions using predictive data mining techniques.

More specifically, we exploit the existing state of the art technologies in protein Named Entity Recognition, combine them using ensemble techniques, and produce a significantly improved precision rate.

We used two different predictive ensemble techniques: voting and meta-learning[15, 16]. Both techniques are based on merging predictions of base classifier (or predictors). Whereas voting simply takes majority predictions, meta-learning produces another classifier, which is called meta-classifier, based on predictions of base classifiers. The final classification is obtained from the meta-classifier which takes in base predictions as an input. In the current work, we considered 5 different base classifiers that are based on Conditional Random Field (CRF), Support Vector Machine (SVM), statistical model, and rule-based system.

4. Evaluation of Keyword Extraction and Semantic Tagging Methods

This section reports the result of our performance evaluation for both keyword extraction and semantic tagging methods.

4.1. Empirical Evaluation of Keyword Extraction Methods

For the evaluation of our two keyword extraction methods, we selected 20 documents; 6 from Aliweb, 6 journal papers and 8 documents from CSTR collection. The Aliweb corpus is a collection of HTML web pages gathered by Turney through Aliweb search engine for his study [17, 18]. CSTR is a collection of Computer Science research papers which were included as part of the New Zealand Digital Library (http://www.nzdl.org).

The performances of two keyword extraction methods (represented in bold italic font in Table 1) are compared with other four existing methods (See Table 1 for details). For each document, six sets of key phrases are retrieved, one from author assigned list that came with the data set, and the other five from each method. We limit the number of key phrases from each document to 15. In case a method produces $n << 15$ key phrases and it is the minimum of all methods, we only select n key phrases from all other methods. Each document and its six key phrase sets were

	Individual Keyphrase Quality			Topic Coverage		
	Average	Std	Rank	Average	Std	Rank
Author Assigned	5.8	1.7	10	5.9	1.2	7.4
Corpus Dependent (Domain Specific)	4.9	1.2	9	6.6	0.6	9.4
Corpus Dependent (Domain Unspecific)	4.7	1.3	7.8	6.4	0.7	8.4
TF-IDF	4.6	1.3	6.9	5.9	1.2	7.4
TF	4.1	1.5	5.4	5.2	1.1	5.2
Co-Occurrence	4.5	1.4	6.8	5.8	1.3	7.4

Table 1. Results based on human evaluation of key phrases extracted from 20 documents. Std denotes standard deviation. All the scores are within a scale of 0 to 10.

presented to human evaluators. An evaluator was asked to assign relevancy score to each key phrase set. More specifically, within a scale of 0 to 10 (the higher the better), the evaluators are asked to:

- Evaluate how an individual key phrase is relevant to the given document.
- Evaluate how the key phrase set as a whole covers the topics in the document.

Then, five methods and author assigned key phrases are ranked based on the scores given by the evaluator. Then the ranks are averaged over all the evaluators. As demonstrated in the Table 1, our corpus dependent method show the best (next to author assigned list) performance, and the other corpus independent also demonstrates a competitive result.

4.2 Empirical Evaluation of Named Entity Recognition

We used two separate document sets to evaluate the performances of NER model for *disease* and *bacterium*, and *protein*. For *disease*, we used 250 ProMED mails as training corpus, and 100 separate ProMED mails as validation corpus. For *bacterium*, we used 100 and 47 ProMED mails for training and testing corpus, respectively. The results are illustrated in Table 2. Although it is from a preliminary evaluation, the accuracy range is reasonable high.

For the evaluation of our integration approach to *protein* name recognition method, we considered the Protein Active Site Template Acquisition (PASTA)[19] data. PASTA contains 61 abstracts from the Journal of Molecular Biology. A three-fold cross validation was performed for the evaluation. The instances tagged as "PROTEIN" were considered as positive data set and the

Entity		Precision	Recall	F Measure
Disease		77.73	73.04	75.31
Bacteria	Genus	92.06	87.63	89.79
	Species	94.35	92.59	93.46

Table 2: The performance evaluation of Named Entity Recognition Model for *disease* and *bacterium*.

Tool	Description
ABNER and YAGI	ABNER and YAGI are biomedical Named Entity Recognizers [1]based on CRF. Their implementation mainly considers the case of CRF that uses the first order Markov independence assumption with Orthographic feature functions and Semantic feature functions.
KEX	KEX is a protein name annotation tool based on PROtein Proper-noun Extraction Rules (PROPER)[2]. Protein name extraction is done using surface clue on character string. This system uses the characteristics of proper noun description to extract protein names.
NLProt	NLProt is a protein Named Entity Recognition system [3, 4]based on SVM. It utilizes the word name, position of the word and the local context of the word in scientific literature to extract protein names.
LingPipe	LingPipe [6]is a collection of tools that could be used for performing linguistic analysis. Named Entity Recognition is a module in LingPipe that employees a generative statistical model based tag bigrams and word trigrams to tag protein names.

Table 3: Five existing Named Entity Recognition Methods for Protein names.

instances tagged as "SPECIES", "RESIDUE", "REGION", "NON_PROTEIN" were considered as negative data set. Initially, the untagged data was passed through the collection of predictors. The tagged outputs of the individual predictors were then extracted, post-processed and compared with the actual predictions. The performance was measured over predictions that were obtained by combining voting and the meta-learning. More specifically, given five predictions from methods listed in Table 3, we consider cases when three or more methods agree on their predictions. When all five agree, we take the prediction. For other cases, we get the prediction from meta-classifier. The meta-classifier implements Naïve Bayes with kernel density estimation. As illustrated in Table 4, our NER model shows significantly improved precision level at the expense of small decrease in recall level. This result empirically confirms the efficacy of our model. In future we intend to add additional filtering criteria using lexicons and rules thereby further increasing the precision and recall of our system.

5. Conclusion

The amount of text documents available for BKC is enormous. However, without powerful text processing technologies suitable for our interest, invaluable information hidden in the unstructured texts will be easily wasted. This paper describes our initial efforts toward the construction of systematic and highly precise system for identifying relevant text and phrases from free-texts. We particularly introduced the keyword extraction and the named entity recognition systems developed for the Bio-E of the BKC.

We demonstrated an impressive performance of our corpus dependent (and domain specific) keyword extraction method through meticulously designed manual evaluation processes. Also, we showed that our corpus independent (co occurrence-based) method yields highly competitive results compared to other corpus dependent methods. Considering the difficulty of compiling a

	ABNER	YAGI	KEX	LingPipe	NLProt	Meta-learning
Precision	25.656%	32.935%	15.255%	29.211%	40.003%	82.174%
Recall	58.872%	61.709%	63.963%	71.817%	55.699%	47.002%

Table 4: Performance evaluation of the combined protein Named Entity Recognition Model (Meta-learning & Voting)

training corpus for a certain domain in practice, this result highlights its paramount practical value.

With the absence of any available named entity recognition tool for *disease* and *bacterium*, we developed one such system based on the state-of-the-art probabilistic technique, and showed very promising results from our preliminary assessment. With the intention of improving precision level, our hybrid system for *protein* named entity recognition embraces ensemble classification framework in two layers: voting and meta-learning. Our cross validation result illustrates the significantly boosted precision level. Considering the volume of documents that are, or will be stored under the Bio-E system, high precision level in identifying relevant terms in domains of interest is a very precious asset.

Our efforts introduced in this paper are still in their preliminary stage. There are still rooms for our models to be improved in many directions. For example, we need more systematic way of incorporating domain knowledge into the corpus dependent keyword extraction system, and our CRF model for named entity recognition needs to be further refined to be more sensitive to rare but important cases.

Acknowledgements: This work was performed under the auspices of the U.S. Department of Energy by University of California Lawrence Livermore National Laboratory under contract No. W-7405-ENG-48.

6. References

1. Settles, B. *Biomedical Named Entity Recognition Using Conditional Random Fields and Rich Feature Sets*. in *In Proceedings of the COLING 2004 International Joint Workshop on Natural Language Processing in Biomedicine and its Applications (NLPBA)*. 2004. Geneva, Switzerland.
2. Fukuda, K., et al. *Toward information extraction: Identifying protein names from biological papers*. in *In Proceedings of the Pacific Symposium on Biocomputing '98 (PSB'98)*. 1998.
3. Mika, S. and B. Rost, *NLProt: extracting protein names and sequences from papers.* Nucl. Acids Res., 2004. **32**(suppl_2): p. W634-637.
4. Mika, S. and B. Rost, *Protein names precisely peeled off free text.* Bioinformatics, 2004. **20**(suppl_1): p. i241-247.
5. Kolda, T., et al., *Data Sciences for Homeland Security Information Management and Knowledge Discovery*. Jan. 2005, Sandia National Laboratories: Livermore, California.
6. Alias-i, I., *Alisa-i LingPipe*.
7. Lovins, J.B., *Development of a Stemming Algorithm.* Mechanical Translation and Computational Linguistics, 1968. **11**: p. 22-31.
8. Salton, G., *Automatic text processing: the transformation, analysis, and retrieval of information by computer*. 1989, Addison-Wesley Longman Publishing Co., Inc.
9. Dagan, I., L. Lee, and F.C.N. Pereira, *Similarity-Based Models of Word Cooccurrence Probabilities.* Machine Learning, 1999. **34**: p. 43-69.
10. Pereira, F., N. Tishby, and L. Lee. *Distributional clustering of English words*. in *Proceedings of the 30th Meeting of the Association for Computational Linguistics*. 1993.
11. Wilson, E.B. and M.M. Hilferty, *The distribution of chi-square.* Proceedings of the

National Academy of Sciences of the United States of America, 1931. **17**: p. 684-688.
12. Lafferty, J., A. McCallum, and F. Pereira. *Conditional Random Fields: Probabilistic Models for Segmenting and Labeling Sequence Data*. in *In Proceedings of the Eighteenth International Conference on Machine Learning (ICML-2001)*. 2001.
13. Gregory, M. and Y. Altun. *Using Conditional Random Fields to Predict Pitch Accents in Conversational Speech*. in *In Proceedings of the 42nd Annual Meeting of the Association for Computational Linguistics (ACL 2004)*. 2004.
14. Cohen, W.W. and S. Sarawagi. *Exploiting Dictionaries in Named Entity Extraction: Combining Semi-Markov Extraction Processes and Data Integration Methods*. in *Tenth ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*. 2004.
15. Saso, D., et al., *Is Combining Classifiers with Stacking Better than Selecting the Best One?* Mach. Learn., 2004. **54**(3): p. 255-273.
16. Philip, K.C. and J.S. Salvatore, *Experiments on multistrategy learning by meta-learning*, in *Proceedings of the second international conference on Information and knowledge management*. 1993, ACM Press: Washington, D.C., United States.
17. Turney, P.D., *Learning Algorithms for Keyphrase Extraction.* Information Retrieval, 2000. **2**: p. 303-336.
18. Turney, P.D., *Learning to Extract Keyphrases from Text*. 1999, Technical Report ERB-1057, National Research Council, Institute for Information Technology.
19. Gaizauskas, D.R., et al., *Protein Active Site Template Acquisition*.