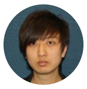
Semi-supervised feature learning from clinical text

Conference Paper · December 2010

DOI: 10.1109/BIBM.2010.5706610 · Source: DBLP



Zhuoran Wang
Tricorn (Beijing) Technology Co., Ltd. (trading a…
25 PUBLICATIONS **151** CITATIONS

John Shawe-Taylor
University College London
522 PUBLICATIONS **37,047** CITATIONS

Anoop Dinesh Shah
University College London
60 PUBLICATIONS **612** CITATIONS

2010 IEEE International Conference on Bioinformatics and Biomedicine

Semi-Supervised Feature Learning from Clinical Text

Zhuoran Wang
Department of Computer Science
University College London
London, United Kingdom
z.wang@cs.ucl.ac.uk

John Shawe-Taylor
Department of Computer Science
University College London
London, United Kingdom
j.shawe-taylor@cs.ucl.ac.uk

Anoop Shah
Clinical Toxicology
Guy's and St Thomas NHS Foundation Trust
London, United Kingdom
anoop.shah@gstt.nhs.uk

***Abstract*—This paper is focused on the automated identification of the clinical free-text records that contain useful information (e.g. symptoms, modifiers, diagnosis, etc) of a certain disease. We introduce a novel semi-supervised machine learning algorithm to address this problem, by training the set covering machine in a bootstrapping procedure. The advantage of the proposed technique is that not only can it find the documents of interest more accurately than searching based on diagnostic codes, the features it learned could also be directly used as a knowledge representation of the given topic and to assist either further machine learning algorithms or manual post-processing and analysis.**

***Keywords*-set covering machine; semi-supervised learning; sparse feature learning; medical text processing;**

I. Introduction

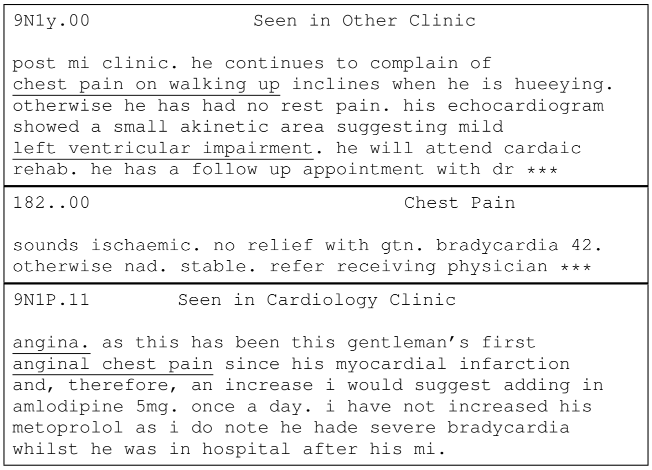
In recent years, the medical research community has shown increasing interest in automatic text analysis in electronic health records (EHR) using natural language processing (NLP) and machine learning techniques. Previous research demonstrated that the information extracted from those medical textual documents can be used for coding, decision support or to enrich the EHR itself [1]. Furthermore, [2] presented that using NLP-based method to identify patients with angina pectoris significantly outperformed searching based on International Classification of Diseases, Ninth Revision (ICD-9) codes, which suggested implications for epidemiological and clinical studies.

Clinical text sometimes is very ungrammatical, consisting of various shorthand expressions, e.g. abbreviations, acronyms, telegraphic phrases, local dialectal shorthand phrases, etc, (and thus is also called free text). To process such text, NLP systems usually highly rely on domain-specific knowledge base (and/or ontology) or manually created linguistic and morphological rules, which are expensive to build. Machine learning algorithms have the advantage that it can gain useful knowledge from some training examples without manually enumerating all possible language phenomena. Reference [3] applied naïve bayes and perceptron learners to identify patients diagnosed with congestive heart failure based on clinical notes, and showed that both the two models gave better precision and recall than ICD code based search. Reference [4] utilized a support vector machine (SVM) algorithm to classify patient smoking status in discharge summaries and achieved the top place in the i2b2 Shared-Task 2006 [5]. Reference [6] compared different types of supervised and unsupervised classification and regression methods for topic labeling and relevance ranking of clinical text, and presented satisfactory results.

Previous works on computerized clinical text analysis are mostly based on the patient level, e.g. to identify whether a patient has some disease by exploring all his medical history records. But sometimes there are also special needs for clinical researchers to process those free-text EHR in record level. For example, Tate *et al.* were interested in dating of diagnosis of ovarian cancer [7]. Their aim was to find, from a patient's free-text records, the first suspicion and/or previous ambiguous diagnosis of ovarian cancer prior to the earliest recorded date of a definite diagnostic code. Hence, the essential task is to identify those records that contain diagnosis information about ovarian cancer, where may or may not exist an explicit diagnostic code implying the fact. Another example is that, when training a supervised machine learning algorithm for information extraction in such free-text records, people must provide a set of manually annotated relevant examples (called training examples). In this case, an efficient methodology of finding those informative records will significantly reduce the amount of data for manual processing, and therefore reduce the human effort. As argued in [2] and [3], diagnostic code based search is not always reliable or satisfactory. Moreover, in the record level analysis the situation could be worse. Figure 1 shows an example of three medical history records of a same patient with angina, where the diagnostic codes are Read codes (that have been commonly used in UK primary care databases). It can be found that the record coded as '182..00 Chest Pain' actually gives little information besides the code itself. In contrast, the other two records containing more information, such as the chest pain modifier, impairment of the left ventricular function and angina pectoris, will not be recognized by their Read codes.

Although the diagnostic codes are usually insufficient for clinical research, one can easily obtain a collection of relevant records for a given disease based on them, which gives us the motivation to develop a machine learning algo-

978-1-4244-8305-1/10/$26.00 ©2010 IEEE



```
9N1y.00              Seen in Other Clinic

post mi clinic. he continues to complain of
chest pain on walking up inclines when he is hueeying.
otherwise he has had no rest pain. his echocardiogram
showed a small akinetic area suggesting mild
left ventricular impairment. he will attend cardaic
rehab. he has a follow up appointment with dr ***
```

```
182..00                                  Chest Pain

sounds ischaemic. no relief with gtn. bradycardia 42.
otherwise nad. stable. refer receiving physician ***
```

```
9N1P.11        Seen in Cardiology Clinic

angina. as this has been this gentleman's first
anginal chest pain since his myocardial infarction
and, therefore, an increase i would suggest adding in
amlodipine 5mg. once a day. i have not increased his
metoprolol as i do note he hade severe bradycardia
whilst he was in hospital after his mi.
```

Figure 1. Example clinical records with Read codes.

rithm that takes those relevant records as training examples and automatically learns information from them to identify more records of interest. This paper proposes a semi-supervised learning solution to the above problem, behind which the insight is that language expressions commonly shared among the labeled documents (i.e. the documents with definite disease codes) can be extracted as potential features, whilst more language phenomena could be gained via exploring to unlabeled documents along each feature direction, reflecting the core topics associated to the disease. Due to the sparseness problem of language corpora, we utilize the set covering machine (SCM) [8] here to learn features and classify unlabeled documents, which is combined with the bootstrapping technique to exhibit a semi-supervised manner. Its performance is evaluated on two manually labeled data set, where the semi-supervised SCM (S^{3}CM) is used to identify documents containing angiogram test results and diagnosis of ovarian cancer, respectively. The S^{3}CM successfully identified 75% and 130% more relevant records than diagnostic codes, where we obtained the F-score 0.73 (precision 0.66 and recall 0.81) and 0.78 (precision 0.69 and recall 0.88) respectively. In addition, we show that the features extracted by S^{3}CM yield a rather comprehensive knowledge representation of the given disease, suggesting that they could be used directly either to serve further machine learning algorithms or to assist manual post-processing or analysis.

The remainder of this paper is organized as follows. Section II explains the notations and terminology used in our context. Section III details the S^{3}CM algorithm. We report our experimental results in Section IV and finally conclude in Section V.

II. Notations and terminology

To avoid the language ambiguities between machine learning and medical literatures, here we clarify the following machine learning terms and notations adopted in this paper. We call each clinical record an example or a data point denoted by x, and assign it a label $y \in \{0,1\}$. The examples with their labels $y = 1$ are called positive examples, indicating that they are documents of interest. In contrast, those with $y = 0$ are called negative examples, meaning that they are not documents of interest. In the case of semi-supervised learning, we also have examples with unknown labels y namely unlabeled data, which are supposed to assist the learning algorithm during the training process and will be labeled after the training. We use $\mathcal{P}$, $\mathcal{U}$ and $\mathcal{N}$ to represent the set of positive, unlabeled and negative examples respectively. Each example x is expressed by a feature vector as $\phi : x \mapsto [h_1(x), h_2(x), \ldots, h_n(x)]$, where the elements $h_i(x)$ can have either binary or real values. In our case, the feature components h_i could be indexed by words, phrases (i.e. word chunks) or word combinations (i.e. phrases with word ordering ignored). Then given a training set $\mathcal{S} := \{\mathcal{P}, \mathcal{U}, \mathcal{N}\}$, the goal of the algorithm is to find a predictive function $f \in \mathcal{F}_\phi$ such that $f(x) = y$.

III. Semi-Supervised Set Covering Machine

In medical free text, the information distribution are very sparse. The symptoms or diagnosis of interest are usually indicated by a few key words. This gives us the intuition that a sparse algorithm learning determinative features would be a suitable solution for this task, where the set covering machine introduced in [8] could possibly be a good candidate.

The original SCM works in an iterative manner as follows. In each iteration, it greedily selects a feature h highest-scored by a score function and removes those examples covered by such feature before starting the next iteration, until all prospective (positive or negative) examples are covered or the size of the learned function f reaches a predefined value K. Note that the feature components $h(x)$ here are binary values, hence the predictive function f is in the form of logical conjunction of a set of features (indication rules). The score function is defined as the number of uncovered positive (or negative) examples it identifies penalized by the number of unexpected examples it covers. That is:

$$C(h) := |\mathcal{P}(h)| - \rho \cdot |\mathcal{N}(h)| \qquad (1)$$

where $\mathcal{P}(h)$ and $\mathcal{N}(h)$ represent the respective subsets of the positive and negative examples that have feature h, ρ is a weight coefficient, and $|\cdot|$ denotes the size of a set. The pseudo-code of SCM is given in Algorithm 1.

To adapt it to semi-supervised learning, we first add an additional penalty item to the score function, as:

$$\tilde{C}(h) := |\mathcal{P}(h)| - \rho_1 \cdot |\mathcal{U}(h)| - \rho_2 \cdot |\mathcal{N}(h)| \qquad (2)$$

where we use the ρ_1-weighted number of the unlabeled examples that h covers ($|\mathcal{U}(h)|$) to give it an extra penalty, since there will be a potential risk for covering an unlabeled point that could be negative. A natural feature definition in our task would be combinations of words, which means the

Algorithm 1: Set Covering Machine [8]

```
input: $\mathcal{P}$, $\mathcal{N}$, $\rho$, $K$
initialization: $f \leftarrow \emptyset$
repeat
    $\hat{h} = \arg\max_{\substack{1\le i\le n \\ h_i \notin f}} |\mathcal{P}(h_i)| - \rho \cdot |\mathcal{N}(h_i)|$
    $\mathcal{P} \leftarrow \mathcal{P} \backslash \mathcal{P}(\hat{h})$, $\mathcal{N} \leftarrow \mathcal{N} \backslash \mathcal{N}(\hat{h})$
    $f \leftarrow f \vee \hat{h}$
until $\mathcal{P} = \emptyset$ or $|f| \geq K$
return $f$
```

Algorithm 2: Modified SCM with Feature Generation

```
input: $\mathcal{P}$, $\mathcal{U}$, $\mathcal{N}$, $\rho_1$, $\rho_2$, $K$
       $f_{init}(\emptyset)$ // initial rule set ($\emptyset$ by default)
                 // used in bootstrapping
initialization: $f \leftarrow f_{init}$, $\mathcal{U}_+ \leftarrow \emptyset$
                $\mathcal{H} \leftarrow \{h : h(x) \wedge h(x'), \exists x, x' \in \mathcal{P}, x \neq x'\}$
                $\mathcal{P}' \leftarrow \mathcal{P}$, $\mathcal{U}' \leftarrow \mathcal{U}$, $\mathcal{N}' \leftarrow \mathcal{N}$
repeat
    $\hat{h} = \arg\max_{h \in \mathcal{H} \backslash f} |\mathcal{P}'(h)| - \rho_1 \cdot |\mathcal{U}'(h)| - \rho_2 \cdot |\mathcal{N}'(h)|$
    for each $x_+ \in \mathcal{P}'(\hat{h})$, $x \in \mathcal{U}'(\hat{h})$
        $\mathcal{H} \leftarrow \mathcal{H} \cup \{h : h(x_+) \wedge h(x)\}$
    end for
    $\mathcal{U}_+ \leftarrow \mathcal{U}'(\hat{h})$
    $\mathcal{P}' \leftarrow \mathcal{P}' \backslash \mathcal{P}'(\hat{h})$, $\mathcal{U}' \leftarrow \mathcal{U}' \backslash \mathcal{U}'(\hat{h})$, $\mathcal{N}' \leftarrow \mathcal{N}' \backslash \mathcal{N}'(\hat{h})$
    $f \leftarrow f \vee \hat{h}$
until $\mathcal{P} = \emptyset$ or $|f| \geq K$
return $f$, $\mathcal{U}_+$
```

Algorithm 3: Semi-Supervised SCM

```
input: $\mathcal{P}$, $\mathcal{U}$, $\mathcal{N}$, $\rho_1$, $\rho_2$, $K$, $M$
initialization: $f \leftarrow \emptyset$, $\mathcal{P}_u \leftarrow \emptyset$, $t = 1$
for $i = 1$ to $M$
    $[f, \mathcal{U}_+] \leftarrow \mathrm{mSCM}(\mathcal{P} \cup \mathcal{P}_u, \mathcal{U}, \mathcal{N}, \rho_1 t, \rho_2, K, f)$
    $\mathcal{P}_u \leftarrow \mathcal{P}_u \cup \mathcal{U}_+$, $\mathcal{U} \leftarrow \mathcal{U} \backslash \mathcal{U}_+$
    $t \leftarrow t + |\mathcal{U}_+|$
end for
return $f$, $\mathcal{P}_u$
```

explicit feature vector of an example x will be all possible word subsets that can be generated from the text.

To avoid dealing with such exponentially large explicit vectors, our algorithm is designed as follows. Firstly, an initial set of candidate features is created from the positive examples by extracting all common word subsets among them, with the expectation that it contains common information expressions of a specific disease. It means a word combination will be initially selected as a potential feature if and only if it is shared by at least two positive examples. The point here is that it not only significantly reduces the feature space, but also cancels the contributions of isolate features in the classification process, which otherwise might lead to overfitting under certain situation. After this, in each iteration we compute the common word combinations among the covered unlabeled examples and the positive examples and insert them into the candidate feature set as well, in order to recall those potential features that had been initially missed by considering the positive examples only. Algorithm 2 gives the pseudo-code of the modified SCM with the feature generation process. We call it mSCM for the convenience of future discussion.

After this, in order to yield the semi-supervised manner, i.e. to benefit from those unlabeled examples by gaining extra information beyond the labeled examples, we design a bootstrapping procedure as follows. In each bootstrap iteration, the unlabeled examples covered by the mSCM in the previous iteration are moved into the positive set, namely pseudo-positive examples. Then a new mSCM based on the updated data set partitions are re-trained. This procedure is repeated M times, where M is a pre-defined number. Algorithm 3 gives the complete pseudo-code for S^{3}CM.

The insight behind the bootstrapping procedure is that those unlabeled points covered by the mSCM in each iteration have the possibility to be positive examples. Hence, we give them the chance to positively contribute to the score function. Such positive contributions will eliminate the penalty for the remaining unlabeled examples that share common features with them, and consequently raise the possibility of the features shared among them to be selected. But note that, as the pseudo-positive set grows, it tend to increase the chance for the unlabeled points to be covered, which suggests that the risk of the classifier making errors is increasing at the same time. Therefore, we enlarge the penalty weight for unlabeled examples in each iteration by making it grow linearly to the size of the pseudo-positive set, as shown in Algorithm 3.

IV. Experimental Results

We applied the S^{3}CM introduced above on two different data sets to identify records containing angiogram test results and ovarian cancer diagnosis, respectively. In addition, we compared the S^{3}CM with a (non-sparse) semi-supervised support vector machine named TSVM[1] [9] and the original (fully supervised) SCM, to demonstrate the superiorities of both the sparsity and the semi-supervised settings. The following two subsections describe the data and the experimental results. In the experiments reported below, the clinical records are lowercased and stemmed before fed into the learners. The parameter settings of the models (S^{3}CM, TSVM and SCM) are tuned based on the leave-one-out cross-validation (LOO-CV) method. Note here, to enable LOO-CV for semi-supervised learning, each time we remove the label of one positive example to make it an unlabeled

[1] Code available at: http://vikas.sindhwani.org/svmlin.html

Table I
READ CODE LIST TO IDENTIFY ANGIOGRAM.

Read Code	Read Term	Frequency
55...11	Angiography - cvs	42
55...12	Angiogram	51
554..11	Coronary arteriography	8
554..00	Coronary arteriogr.-general	14
554Z.00	Coronary arteriog-general NOS	1
55...00	Cardiovascular sys.angiography	4
Total		120

Table II
EXPERIMENTAL RESULTS ON ANGIOGRAPHIC DATA.

Model	#TP	#FP	#FN	Precision	Recall	F-score
S^{3}CM	90	46	21	0.66	0.81	0.73
SCM	67	19	44	0.60	0.78	0.68
TSVM	2	64	109	0.03	0.02	0.02

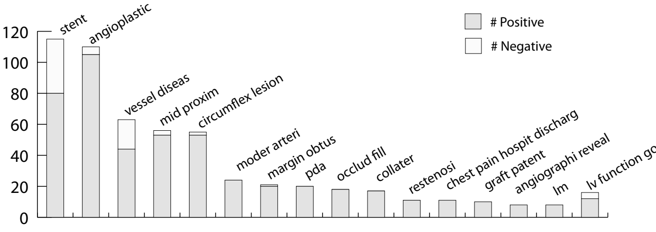


Figure 2. Features extracted from the angiogram records.

data point, train the algorithm based on this modified data set and test its classification result for that pre-selected example. This process is applied to every positive example to obtain an average LOO-CV error rate.

A. Angiogram

The corpora used in the first experiment are coded records from the General Practice Research Database (GPRD). There are two portions of the data for a case-control study. The case data consist of 2,090 free text records of 178 unique patients with angina, where there is at least one angiographic record for each patient. (But not every patient has a coded record indicating that.) Two controls for each case patient are provided, yielding 356 patients with 3539 records in the control data, where there is no angiographic record for any patient. Besides the records that have Read codes indicating they are angiographic results, we find there are significant amount of other records which contain information about angiographic findings. In addition, some coded angiogram and angiography records are non-informative (e.g. "55...12 Angiogram: Hospital admission."), with the relevant information "hidden" in other diagnosis records.

Our task here is to identify those records that contain information about angiographic findings but are not coded so. In the following experiments, the records in the case data whose Read codes can be found in a pre-define code list are taken as positive examples. The records in the control data are used as negative examples. Then the remaining records in the case data are treated as unlabeled examples. The list of the Read codes of concern can be found in Table I, with their frequencies in the case data.

By manually analyzing the case data, we find there are actually 231 records containing angiographic information, meaning that 111 records are missed by searching Read codes only. We evaluate the performance of our S^{3}CM algorithm on the unlabeled data to recall these 111 records. Table II shows the results, where #TP, #FP and #FN stand for the number of true positive, false positive and true negative examples output by the models respectively. When the S^{3}CM (with parameter settings: $\rho_1 = 0.01$, $\rho_2 = 10$, $K = 4$, $M = 5$) are compared to TSVM (with regularization coefficient $\lambda = 1$, unlabeled data influence parameter $\lambda' = 0.1$ and positive class fraction of unlabeled data $r = 0.1$) and supervised SCM (with $\rho = 10$), it significantly outperforms the other two models in both precision and recall. Note here, to train the SCM, we use the positive and negative examples only. Firstly, it proves that a semi-supervised classifier are more competitive in this task. Secondly, the weak performance of TSVM suggests the unfitness of non-sparse learners in this task.

Finally, we plot the features learned by the S^{3}CM in Figure 2, with their frequencies among the true positive and negative examples. It can be found that the features can sketch a rather complete knowledge representation of angiographic findings, suggesting that they could directly feed some further research.

B. Cancer of Ovary

In the second experiment, the data we use are also from GPRD, but are about ovarian cancer diagnosis. There are 7807 records for 340 unique patients, among which 585 unique records (representing 245 patients) are found by manual analysis to contain a reference to the patient's ovarian cancer in the text. But only 236 of them have a Read code indicating the diagnosis. The Read codes of concern and their frequencies are listed in Table III.

We apply the S^{3}CM (with $\rho_1 = 0.1$, $\rho_2 = 10$, $K = 4$ and $M = 5$) to this data set to identify in the remaining records whether there are referrals or suspicions of ovarian cancer in them, i.e. to recall the rest 349 records that have been manually identified. There is no control for this data set. Therefore, when training the S^{3}CM, we use the case data in the previous angiographic experiments as the negative examples, with the assumption that none of the patients in it would also have ovarian cancer at the same time. In this experiment, we include the read term of each record as a part of its text to make it more informative, since the text itself tends to be too brief for postive examples. The supervised SCM (with $\rho = 10$) and TSVM (with $\lambda = 0.01$, $\lambda' = 1$ and $r = 0.1$) are also tested on this corpus for comparison. The results are displayed in Table IV, which

Table III
READ CODE LIST TO IDENTIFY OVARIAN CANCER.

Read Code	Read Term	Frequency
B440.00	Malignant neoplasm of ovary	112
B440.11	Cancer of ovary	115
B44..00	Malignant neoplasm of ovary and other uterine adnexa	9
Total		236

Table IV
EXPERIMENTAL RESULTS ON THE OVARIAN CANCER DATA.

Model	#TP	#FP	#FN	Precision	Recall	F-score
S^3CM	308	139	41	0.69	0.88	0.78
SCM	95	53	254	0.64	0.27	0.38
TSVM	26	534	323	0.05	0.07	0.06

confirms similar findings to Table II. The features learned with their frequencies are shown in Figure 3.

V. CONCLUSIONS

In this paper, we developed a novel sparse semi-supervised learning algorithm to identify clinical text records of interest, whose effectiveness has been proved by the experiments on two different corpora. The main advantages of the proposed model can be summarized as follows. Firstly, as a semi-supervised learner, its training process does not require any human effort for corpus annotation, which can be easily fed by a 'cheap' diagnostic code search. Moreover, the features learned by it are representative and usually can sketch a good description of the given topic. Deep explorations of the specified topic can be achieved by including unlabeled data, even when the positive examples are not informative enough. In addition, it does not rely on any prebuilt knowledge base or linguistic rule set, which makes it easy to adapt to other domains and/or languages. However, there is a common drawback of purely machine learning based language analysis models, i.e. some rarely observed language expressions might never be favored by the model, which would correspond to the drop on recall rate. Integrating linguistic and domain knowledge to our machine learning method will be undoubtedly helpful. This issue is left open here, but would be addressed in our future research.

ACKNOWLEDGEMENTS

The research leading to these results has received funding from the Wellcome Trust and the National Institute of Health Research (NIHR) under the project CALIBER. This work was supported in part by the IST Programme of the European Community, under the PASCAL2 Network of Excellence. The authors acknowledge GPRD for providing the corpora used in the above experiments. The authors also would like to thank Rosemary Tate, Harry Hemingway, Julie Sanders and Spiros Denaxas for helpful discussions.

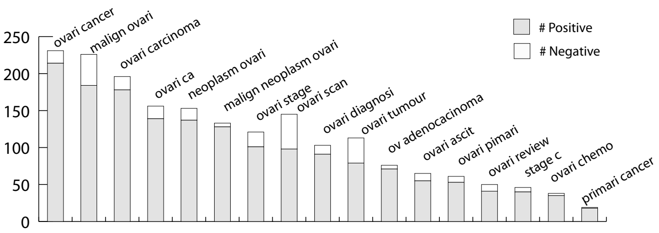


Figure 3. Features extracted from the ovarian cancer records.

REFERENCES

[1] S. Meystre, G. Savova, K. Kipper-Schuler, and J. Hurdle, "Extracting information from textual documents in the electronic health record: a review of recent research," *Yearb Med Inform*, pp. 128–144, 2008.

[2] S. Pakhomov, H. Hemingway, S. Weston, S. Jacobsen, R. Rodeheffer, and V. Roger, "Epidemiology of angina pectoris: Role of natural language processing of the medical record," *American Heart Journal*, vol. 153, no. 4, pp. 666–673, 2007.

[3] S. Pakhomov, J. Buntrock, and C. G. Chute, "Prospective recruitment of patients with congestive heart failure using an ad-hoc binary classifier," *Journal of Biomedical Informatics*, vol. 38, no. 2, pp. 145–153, 2005.

[4] C. Clark, K. Good, L. Jeziernyb, M. Macpherson, B. Wilsonb, and U. Chajewska, "Identifying smokers with a medical extraction system," *Journal of the American Medical Informatics Association*, vol. 15, no. 1, pp. 36–39, 2008.

[5] O. Uzuner, I. Goldstein, Y. Luo, and I. Kohane, "Identifying patient smoking status from medical discharge records," *Journal of the American Medical Informatics Association*, vol. 15, no. 1, pp. 14–24, 2008.

[6] H. Suominen, "Machine learning and clinical text. supporting health information flow," PhD Dissertation, University of Turku, Finland, 2009.

[7] A. R. Tate, A. G. R. Martin, T. Murray-Thomas, S. R. Anderson, and J. A. Cassell., "Determining the date of diagnosis - is it a simple matter? The impact of different approaches to dating diagnosis on estimates of delayed care for ovarian cancer in UK primary care," *BMC Medical Research Methodology*, vol. 9, 2009.

[8] M. Marchand and J. Shawe-Taylor, "The set covering machine," *Journal of Machine Learning Research*, vol. 3, pp. 723–746, 2003.

[9] V. Sindhwani and S. S. Keerthi, "Large scale semi-supervised linear SVMs," in *Proceedings of the 29th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval*, 2006, pp. 477–484.