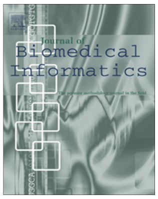
Journal of Biomedical Informatics 52 (2014) 55–64

Contents lists available at ScienceDirect

Journal of Biomedical Informatics

journal homepage: www.elsevier.com/locate/yjbin



Transfer learning based clinical concept extraction on data from multiple sources

Xinbo Lv, Yi Guan *, Benyang Deng

School of Computer Science and Technology, Harbin Institute of Technology, Harbin, Heilongjiang 150001, China

ARTICLE INFO

Article history:
Received 6 July 2013
Accepted 6 May 2014
Available online 21 May 2014

Keywords:
Clinical concept extraction
Transfer learning
TrAdaBoost
Bagging
Machine learning

ABSTRACT

Machine learning methods usually assume that training data and test data are drawn from the same distribution. However, this assumption often cannot be satisfied in the task of clinical concept extraction. The main aim of this paper was to use training data from one institution to build a concept extraction model for data from another institution with a different distribution. An instance-based transfer learning method, TrAdaBoost, was applied in this work. To prevent the occurrence of a negative transfer phenomenon with TrAdaBoost, we integrated it with Bagging, which provides a “softer” weights update mechanism with only a tiny amount of training data from the target domain. Two data sets named BETH and PARTNERS from the 2010 i2b2/VA challenge as well as BETHBIO, a data set we constructed ourselves, were employed to show the effectiveness of our work's transfer ability. Our method outperforms the baseline model by 2.3% and 4.4% when the baseline model is trained by training data that are combined from the source domain and the target domain in two experiments of BETH vs. PARTNERS and BETHBIO vs. PARTNERS, respectively. Additionally, confidence intervals for the performance metrics suggest that our method's results have statistical significance. Moreover, we explore the applicability of our method for further experiments. With our method, only a tiny amount of labeled data from the target domain is required to build a concept extraction model that produces better performance.

1. Introduction

Clinical documents are valuable resources in which abundant personalized health information, such as symptoms, medicines and tests, is recorded by physicians in natural language. As a subtask of automatic acquisition of knowledge from these unstructured clinical texts, concept extraction aims to identify words and phrases that stand for clinical concepts from the narrative texts in clinical documents. This is the key component of text processing systems for understanding the content of clinical documents. Only when clinical concepts are correctly identified can other more complex tasks, such as concept relation extraction, assertion classification, co-reference, health information retrieval and health information recommendation, be performed effectively.

In the biomedical literature domain, research similar to concept extraction has been conducted in named entity recognition tasks such as gene name recognition [1]. However, research on clinical concept extraction for clinical documents appears to be rather sparse. One important reason for the lag of clinical concept extraction is the lack of shared annotated clinical documents due to patient privacy and confidentiality requirements. Fortunately, efforts to construct de-identified clinical documents are finally allowing studies on clinical concept extraction. For example, the 2010 Informatics for Integrating Biology and the Bedside (i2b2)/Veteran's Affairs (VA) challenge [2] provided a total of 394 training documents, 477 test documents, and 877 un-annotated documents for all three tasks. However, annotated clinical documents are always scarce and are created by a number of different institutions; the 2010 i2b2/VA challenge's data consist of four sets from three institutions. Such small-scale data sets limit the performance of a statistical machine learning model. One solution to this problem is to increase the training data sets by gathering data from multiple sources. Nevertheless, different vocabularies and writing styles of multiple sources make the combined data sets heterogeneous, to which the statistical machine learning model is sensitive. Specifically, the marginal probability distributions of words in clinical texts from different institutions are not equal, which violates the traditional machine learning's basic assumption: the training and test data should be under the same distribution. Therefore, a learner trained by one institution's data may perform worse when it is applied to data from another institution. The normal way of tackling this problem is to annotate data from the new institution,

* Corresponding author. Address: Mailbox 321, West Da-zhi Street 92, Harbin, Heilongjiang 150001, China.

E-mail address: guanyi@hit.edu.cn (Y. Guan).

http://dx.doi.org/10.1016/j.jbi.2014.05.006
1532-0464/© 2014 Elsevier Inc. All rights reserved.

but this is always expensive and time-consuming. Abandoning old data would also be a waste.

The objective of this paper is to discuss approaches and strategies for clinical concept extraction from multiple sources. Using a training data set with a different distribution from one institution, we build a clinical concept extraction model for data from another institution. Transfer learning is a family of algorithms that can relax the traditional machine learning's same-distribution assumption. It leverages and transfers knowledge from the source domain to the target domain, and in this way, helps improve the model when the target domain's training data are insufficient. Specifically, we apply an instance-based transfer learning method – TrAdaBoost [3] – to the clinical concept extraction task. TrAdaBoost aims to re-weight the instances in the source domain in order to decrease the diversity between the data of the source domain and the target domain. It was originally created to solve binary classification problems, and we apply it to the sequence labeling problem with multiple labels. Additionally, to avoid the negative transfer problem caused by the over-discarded risk of TrAdaBoost, we integrate Bagging with TrAdaBoost to provide a "softer" weight update mechanism. Two data sets, BETH and PARTNERS, from the 2010 i2b2/VA challenge, as well as one data set we built by combining BETH and a biomedical literature data set (BIOLITERATURE), are used to verify the effectiveness of our method's transfer ability. Experiments show that with only a small amount of annotated training data from the target domain, our framework outperforms the baseline method, which simply combines data from the source domain and data from the target domain as training data.

2. Background

Methods for clinical concept extraction generally fall into three categories: dictionary-based methods, rule-based methods and statistical machine learning methods [4].

Dictionary-based methods search through dictionaries such as UMLS [5] and SNOMED-CT [6] to extract clinical concepts. MedLEE [7] is a typical system that uses a domain-specific vocabulary and semantic grammar to extract and encode clinical information in narrative reports. A structured representation is then constructed by these clinical terms. It is adapted to extract the concepts in clinical documents, and these concepts are mapped to semantic categories and semantic structures [8]. MetaMap [9] is also an early dictionary-based program developed at the National Library of Medicine (NLM) to recognize and categorize entities in texts from the biomedical domain and then to map them to UMLS Metathesaurus. It is applied to both IR and data mining applications; additionally, it is used to index the biomedical literature at the NLM. Systems described in [10–12] also adopt dictionary-based methods. The advantage of these methods is that they are easy to implement, while the disadvantage is that they suffer from low recall since many concepts may fail to be covered by the dictionary.

Rule-based methods require experts to define hand-coded rules or regular expressions for the extraction task. For example, in the sentence "systemic granulomatous diseases such as Crohn's disease or saroiaosis", the phrase "such as" implies that "Crohn's disease" and "saroiaosis" are disease names. Long [13] used regular expression to extract the diagnoses and procedures from the past medical history and discharge diagnoses in the discharge summary. Turchin et al. [14] designed a software tool to extract blood pressure values and anti-hypertensive treatment intensification from the texts of physician notes; regular expressions are also employed in their work. Rule-based methods are always difficult to achieve and time-consuming because rules have to be collected by hand.

In recent years, more and more researchers have resorted to statistical machine learning methods for clinical concept extraction. Several models, such as the Hidden Markov Model (HMM) [15], Support Vector Machine (SVM) [16], Maximum Entropy Model (MEM) [17] and Conditional Random Fields (CRF) [18], have been used to solve the information extraction problem. CRF has been proven to be the state-of-art model among these models. Taira and Soderland [19] first used MEM for the task of knowledge acquisition, parsing, semantic interpretation and evaluation of radiology reports; then, they moved to a vector space model to extract concepts about anatomy defined in the UMLS. A set of 2551 unique anatomical concepts was finally extracted from brain radiology reports, and an *F*-score of 87% was achieved [20]. Sibanda et al. [21] employed SVM trained with syntactic, contextual clues and ontological information from UMLS to recognize semantic categories in discharge summaries. They extracted eight types of semantic categories, and an *F*-score above 90% was achieved. There are also some methods of multiple classifier fusion for this task. For example, Wang and Patrick [22] combined MEM, SVM and CRF to recognize 10 types of clinical entities from 311 admission summaries, and an *F*-score of 83.3% which is 3.35% higher than the baseline stand-only CRF model, was obtained. Li et al. [23] compared CRF with SVM for disorder named entity recognition in clinical texts, and the experimental results showed that CRF obtained a higher score than did SVM.

All of the works described above are, however, not evaluated on the same data set, so it is difficult to compare them. The 2010 i2b2/VA challenge provides an opportunity for researchers to demonstrate their methods on a shared data set. Most of the submitted systems are based on machine learning methods. The best performance was achieved by a discriminative semi-Markov HMM that was trained by passive–aggressive (PA) online updates. The system obtained an *F*-score of 0.8523 [24]. Roberts and Harabagiu [25] proposed a flexible feature selection mechanism that makes it easy to find a near-optimal subset of features for a task in their system. For more details about this challenge, refer to [2]. There have also been some works based on the data set after this challenge. Xu et al. [26] developed a system that outperforms the best system in the challenge. Their main contribution to concept extraction was using two separate CRF models to handle medical concepts and non-medical concepts. Chen et al. [27] applied active learning to assertion classification, and their experiments showed that a comparable performance can be achieved with fewer annotated training instances. Abacha and Zweigenbaum [28] compared three methods of medical entity recognition: a semantic method that relies on domain knowledge, a method that first extracts noun phrases and then uses SVM to classify their entity types, and a method that uses CRF to identify entity boundaries and types simultaneously. Their work showed that the hybrid method that combined machine learning and domain knowledge yielded the best results.

Although statistical machine learning methods have obtained certain achievements, the lack of abundant annotated clinical documents and the diversity in clinical documents from multiple institutions present great challenges to researchers. One solution to this problem is to accomplish the task with fewer or even no training data. For example, Zhang and Elhadad [29] attempted an unsupervised method to extract named entities in both biological and clinical text without any rules or annotated data. The advantage of this method is that it is easy to use in different applications; however, it is not as competitive as supervised methods. Another solution is to achieve clinical concept extraction by increasing the training data. Torii et al. [30] found that the performance may be improved if more training data are available; however, they also found that the performance of a model trained on one institution's data degraded when data from another institution were tested. Their work inspires us to explore new machine learning methods to improve the performance of clinical concept extraction models with the help of training data from other sources with different distributions.

Although generalization, which is the ability to perform accurately on unseen examples after having experienced a learning data set, is a core objective of machine learning, it is still a challenge to most machine learning methods because they are not designed to work appropriately when a distribution changes. Transfer learning [31] presents a possible way to improve the generalization ability by relaxing the restriction of traditional machine learning's same-distribution assumption, allowing the previous data to be reused for a new task. Transfer learning has attracted more and more researchers since it was first presented in a NIPS-95 workshop, and it is referred to by different names, such as "learning to learn" [32], "life-long learning" [33], and "multi-task learning" [34]. It has been used in many applications such as sentiment classification [35] and image classification [36]. Currently, transfer learning is also applied in the bioinformatics and computational biology. Ganchev et al. [37] proposed a framework called Transfer Rule Learner (TRL) for the task of biomarker discovery, which aims to find measurable variables that can reflect and predict a disease state. TRL transferred knowledge in the form of rules from an old data set and used them to learn a new classifier on a new data set. The performance of their method exceeded that of using one data set alone and the union of the data sets. Lee et al. [38] applied transfer learning to flow cytometry, which is a technique for rapid cell analysis. An important process in flow cytometry is to label every cell as belonging or not belonging to the cell type of interest, which is called "gating". They leveraged existing datasets that were previously gated by experts to automatically gate a new flow cytometry dataset. The most relevant research to our work is to recognize protein names from biological journals coming from two sources using a maximum entropy based transfer learning algorithm [39]. To the author's best knowledge, no transfer learning algorithm has been attempted to perform the clinical concept extraction task on clinical documents.

3. Task definition

Clinical concept extraction aims to automatically identify the boundaries of concepts and assign the concept types to them. It can be seen as a sequence labeling problem that assigns each token a label indicating both the boundary and concept type. Given an unstructured text $X = x_1, \ldots, x_n$ and a label set Y, the statistical machine learning method's object is to obtain a probability $P(y|x_i)$ to x_i which is labeled as y. The 2010 i2b2/VA challenge defined three medical concept types: medical problems (e.g., diseases, viruses, abnormalities), treatments (e.g., drugs, procedures, medical devices) and tests (e.g., examinations and evaluations of the patient, physiologic measures and vital signs). We will follow the definitions proposed by this challenge in our work. In our clinical concept extraction task, $Y = \{B\_Pr, B\_Tr, B\_Te, I\_Pr, I\_Tr, I\_Te, O\}$. In this label denotation, *Pr*, *Tr*, *Te* are the category labels, indicating the label's type, that is, problem, treatment and test; *B* and *I* indicate a token is the beginning of a concept and the inside of a concept, respectively; *O* means that a token does not belong to a concept. An example is shown in Table 1, in which "chlorhexidine soap bath" is labeled as a concept, and its type is treatment.

Table 1
An example of clinical concept extraction.

the	patient	will	use	chlorhexidine	soap	bath	once	daily
O	O	O	O	B_Tr	I_Tr	I_Tr	O	O

To construct a prediction model, traditional machine learning methods must be trained by a training data set $(X_{train}, Y_{train}) = \{(x_1, y_1), \ldots, (x_N, y_N)\}$; then, the model will be designated to annotate test data set $X_{test} = (x_1, \ldots, x_M)$. The basic assumption of traditional machine learning is that X_{train} and X_{test} must be under the same distribution D, that is, the marginal probability distributions $P(X)$ between the training data set and test data set are the same. In the case of our clinical concept extraction task, that means distributions of words in clinical texts between the training data and the test data are identical (we consider distributions of words in clinical texts that are from domain vocabularies such as SNOMED). However, in practice, this assumption is always difficult to follow. For the clinical concept extraction task, in most cases, we have built a model using data from one institution but apply it directly to data from another institution, and this may lead to poor performance.

Transfer learning can, however, allow the model to be applied to data sets drawn from some distribution different from the one upon which it was trained. Fig. 1 shows the difference between traditional machine learning and transfer learning. The main contribution of transfer learning is transferring knowledge from the source domain to target domain. We now formally define our task: we have two clinical data sets with different distributions D^{source} and D^{target} that are constructed by two institutions. Our task is to assign labels Y_{test}^{target} to test data X_{test}^{target} drawn from D^{target}, given the training data $(X_{train}^{source}, Y_{train}^{source})$ drawn from D^{source}. In addition, we have a tiny amount of training data $(X_{train}^{target}, Y_{train}^{target})$ drawn from D^{target}, the quantity of these data is not sufficient to train a high-quality model alone but can help to initialize a rough model upon which transfer learning will further refine.

Among the different approaches to transfer learning, we prefer instance-based transfer, which assumes that some instances in the source domain can be reused. By re-weighting weights of instances in the source domain, effects of dissimilar instances will be

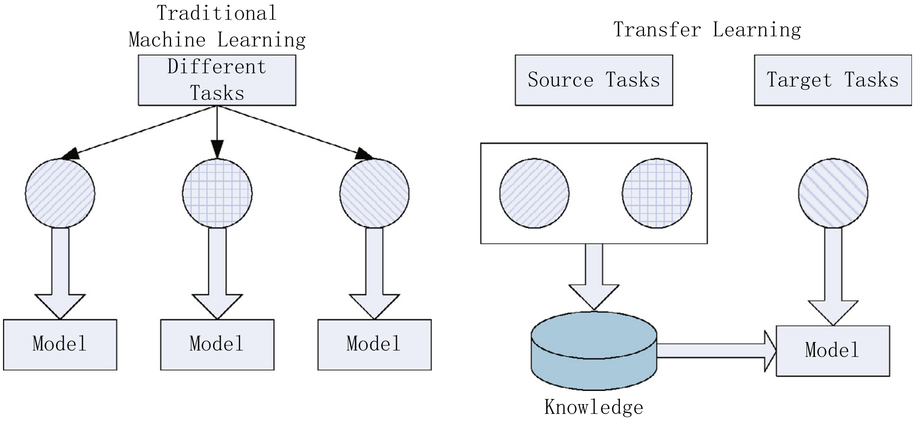


Fig. 1. Difference between traditional machine learning and transfer learning.

reduced, while similar instances will contribute more to the target domain and may thus lead to a more accurate model. For example, in our task, “medicine” can be a similar instance between the source domain and the target domain because it is a common word, while “Alzheimer” can be a dissimilar instance between the source domain and the target domain because it only appears in some specific clinical documents.

4. Methods

4.1. Framework

The framework of our work is shown in Fig. 2. We have training data in both the source domain and target domain, but their distributions are not the same. Additionally, the quantity of training data in the target domain is not sufficient, which makes it impossible to learn an accurate model with these data alone. Our goal is to learn a model that will extract clinical concepts on test data in the target domain with high accuracy based on training data from the source domain and a tiny amount of training data from the target domain. We employ an instance-based transfer learning method named TrAdaBoost to learn a model with transfer ability. Similar to other transfer learning methods, TrAdaBoost cannot always transfer the right knowledge to the target domain, which sometimes causes lower system performance, called negative transfer. To prevent this situation, we adopt Bagging (Bootstrap aggregating), which takes K TrAdaBoost’s results as base learners and aggregates them into the final learner. Under this schema, the risk of negative transfer will be averaged to K learners, resulting in a lower overall influence of negative transfer. In the end, the final learner will be used to extract concepts in the test data from the target domain.

4.2. TrAdaBoost

TrAdaBoost (Transfer AdaBoost) [3] is an instance-based transfer learning method extended from Adaboost [40]. It allows users to leverage the training data from an old domain to construct a high-quality model for the new test data. The key idea is to re-weight instances from the source domain based on a few of annotated training instances from the target domain. Although under different distributions, some instances from the source domain may be helpful to construct the training data set combined with the instances from target domain. The algorithm uses boosting to filter out “bad” source domain instances while encouraging the “good” ones to build a more accurate model in target domain.

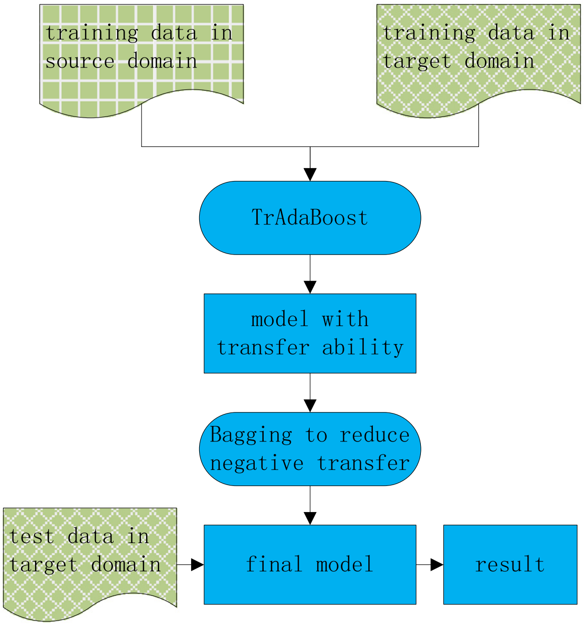

training data in source domain
training data in target domain
TrAdaBoost
model with transfer ability
Bagging to reduce negative transfer
test data in target domain
final model
result

Fig. 2. Framework of transfer learning-based clinical concept extraction.

There are many choices in the family of transfer learning algorithms; among them, we prefer TrAdaBoost for three major reasons: first, it assumes feature spaces and labels between the source domain and target domain are exactly identical but distributions are different, which is consistent with our task; second, it is suitable for the condition that the two domains’ dissimilarity is not too great, as is the case in our task; finally, TrAdaBoost is a flexible machine learning framework that can accommodate other machine learning models without modifying them, which increases the versatility of our method.

As an instance transfer learning method, the goal of TrAdaBoost is to reuse T^{source}_{train} as much as possible by discovering which part of T^{source}_{train} is specific for the source domain and which part may be common between source and target domains. It boosts to re-weight the instances from both T^{source}_{train} and T^{target}_{train}. The mechanism of TrAdaBoost is shown in Fig. 3. On the one hand, weights of instances from T^{source}_{train} that are wrongly predicted will be decreased in order to weaken their impacts; on the other hand, weights of instances from T^{target}_{train} that are wrongly predicted will be increased in order to emphasize them.

4.3. TrAdaBoost in clinical concept extraction

TrAdaBoost is an extension from AdaBoost, which aims to train N weak learners in N rounds to improve the overall performance of these weak learners by re-weighting training instances. In each round, AdaBoost re-weights training instances depending on whether they are correctly classified by this weak learner, and weights of the wrongly classified instances will be increased by multiplying a parameter β_t to strengthen their effects on the training of the next learner in the next round. After N rounds, N trained weak learners will be integrated to a final learner.

A formal description of the TrAdaBoost-based algorithm in clinical concept extraction is shown in Fig. 4. The inputs of this algorithm include clinical training data from two institutions (T^{source}_{train} and T^{target}_{train}), test data to be labeled (T^{target}_{test}, which is from the same institution with T^{target}_{train}), and a LEARNER, which can be any machine learning model, such as MEM or CRF, that acts as the base learner in AdaBoost. The algorithm maintains a weight vector for training instances whose value at round t is $w^t = \{w^t_1, \ldots, w^t_{m+n}\}$, which is arbitrarily assigned with weight $\{w^1_1, \ldots, w^1_{m+n}\}$ in the initializing step. $w^t_1, \ldots, w^t_m$ are weights for training instances from source domain while $w^t_{m+1}, \ldots, w^t_{m+n}$ are

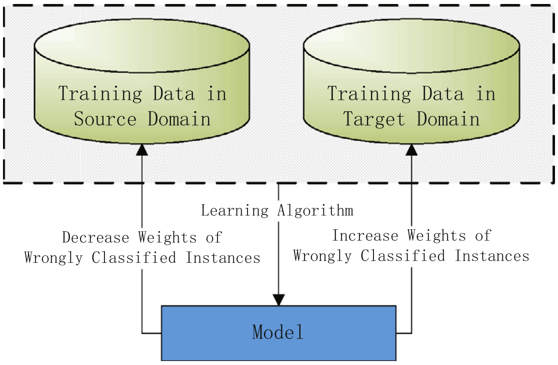

Training Data in Source Domain
Training Data in Target Domain
Learning Algorithm
Decrease Weights of Wrongly Classified Instances
Increase Weights of Wrongly Classified Instances
Model

Fig. 3. The mechanism of TrAdaBoost.

```
Input:
  Clinical training data $T_{train}^{source}$ from one institution,
  clinical training data $T_{train}^{target}$ from another institution, test
  data from $T_{test}^{target}$, base learning algorithm LEARNER,
  the times of iterations N, $n = |T_{train}^{source}|$, $m = |T_{train}^{target}|$,
  $m \ll n$
1 Initialize the weight vector $w^1 = \{w_1^1, ..., w_{n+m}^1\}$
2 for $t = 1, ..., N$ do
3     Set $p^t = w^t / \sum_{i=1}^{n+m} w_i^t$
4     Call LEARNER, providing the combined training set
      $T_{train}^{source} \bigcup T_{train}^{target}$ with distribution $p^t$ and the test
      data $T_{test}^{target}$. Get back a hypothesis $h_t : X \mapsto Y$
5     Calculate the error of $h_t$ on $T_{train}^{target}$
      $\epsilon_t = \sum_{i=n+1}^{n+m} \frac{w_i^t \cdot I[h_t(x_i) \neq c(x_i)]}{\sum_{i=n+1}^{n+m} w_i^t}$
6     Set $\beta_t = \epsilon_t/(1-\epsilon_t)$ and $\beta = 1/(1+\sqrt{2\ln n/N})$
7     Update the weights of instances which $h_t(x_i) \neq c(x_i)$
      $w_i^{t+1} = \begin{cases} w_i^t\beta, & 1 \leq i \leq n \\ w_i^t{\beta_t}^{-1}, & n+1 \leq i \leq n+m \end{cases}$
8 end
Output: the hypothesis
  $h_f(x) = \arg\max_{y \in Y} \sum_{n=\lceil N/2 \rceil}^{N} I[h_n(x) = y] \log(1/\beta_n)$
```

Fig. 4. Algorithm of concept extraction based on TrAdaBoost.

weights for training instances from target domain. The algorithm then iterates N times to update the weight vector. In each iteration round, a weak hypothesis h_t:$X \to Y$ is learned in line 4 based on the training data set $T_{train}^{source} \cup T_{train}^{target}$ with normalized weight vector p^t. Parameter ε_t is the error rate of the weak hypothesis h_t on training data from the target domain. It is used to evaluate h_t: the larger the ε_t is, the weaker is the h_t. TrAdaBoost enables AdaBoost with transfer learning by adding another parameter β, whose function is to re-weight training instances in the source domain; β_t, whose function is to re-weight training instances in the original AdaBoost algorithm, is diverted to re-weight training instances in target domain. β and β_t are set in line 6: β_t is a function of ε_t: β_t will have a larger value if h_t has a higher error rate. The weight vector is then updated by multiplying by the two parameters. If training instances from T_{train}^{source} are incorrectly predicted, weights of these instances will be decreased by multiplying β ($\beta \in (0, 1]$). Then, in the next round, the re-weighted "bad" training instances from T_{train}^{source} will have less impact than the current round. Meanwhile, weights of miss-predicted training instances from T_{train}^{target} will be increased by multiplying β_t^{-1} in order to emphasize them in the next round. After N iterations, common instances between T_{train}^{source} and T_{train}^{target} will have higher weights, and the combined training set $T_{train}^{source} \cup T_{train}^{target}$ will be more suitable for training an accurate clinical concept extraction model. At the end of the algorithm, the final hypothesis $h_f(x)$ is voted by the latter half of these weak hypotheses with higher confidence. The main difference between our algorithm and the original TrAdaBoost exists in that we extended it to multi-class problems in order to tackle our clinical concept extraction task. It is easy to understand as shown in line 5, line 7 and the output step.

Time complexity of line 3, line 5 and line 7 are $O(n + m)$, $O(m)$, and $O(n + m)$, respectively. The overall time complexity of this algorithm is $N \times (2O(n + m) + O(m))$.

4.4. TrAdaBoost with Bagging

Transfer learning is not guaranteed to improve the performance of the model. Sometimes, it even lowers the performance, which is called negative transfer; this phenomenon also occurs with TrAdaBoost. For the weights update mechanism of TrAdaBoost, weights of training instances from the source domain may decrease exponentially by multiplying β after several iterations, and they may be too small to be effective. However, TrAdaBoost cannot ensure such instances are noisy, and the performance of the model may be negatively impacted because many training instances are discarded.

To prevent the over-discarded risk of TrAdaBoost, a "softer" weights update mechanism is presented in this paper. Bootstrap aggregating (Bagging) [41] is a method that generates K base learners and aggregates them to a final learner to improve the final results. Suppose the size of the training data set is n. The K base learners are trained on K subsets of size $\rho n (0 < \rho < 1)$ by sampling with replacement from the original training data set, as shown in Fig. 5. In this paper, we integrate Bagging with TrAdaBoost in order to decrease the possibility of negative transfer. TrAdaBoost serves as the base learner in Bagging. For the two domain's training data set in TrAdaBoost, we handle the source domain only because only weights of instances in this domain are decreased. As shown in Fig. 6, K TrAdaBoost learners are generated on K subsets of training

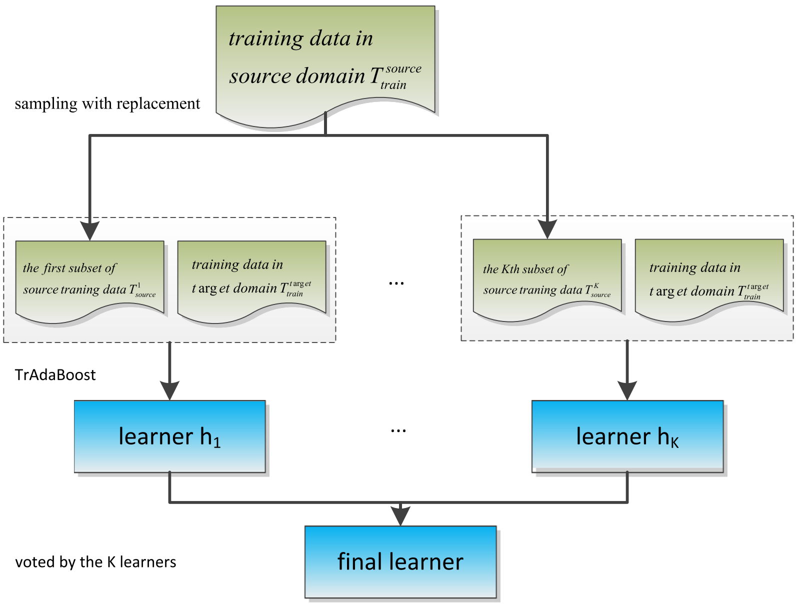

training data in
source domain T_{train}^{source}
sampling with replacement
the first subset of
source traning data T_{source}^{1}
training data in
target domain T_{train}^{target}
...
the Kth subset of
source traning data T_{source}^{K}
training data in
target domain T_{train}^{target}
TrAdaBoost
learner h_1
...
learner h_K
final learner
voted by the K learners

Fig. 5. The mechanism of TrAdaBoost with Bagging.

```
Input:
Source training data $T_{train}^{source}$, target training data $T_{train}^{target}$
TrAdaBoost parameters: base learning algorithm L, the number of iterations N
Bagging parameters: the number of learning rounds K, the percentage of the training size $\rho$
for $k = 1, ..., K$ do
    Generate a new training set $T_{source}^{k}$ from $T_{train}^{source}$ by sampling with replacement, its size is $\rho|T_{train}^{source}|$
    Get a TrAdaBoost learner $h_k = TrAdaBoost(T_{source}^{k}, T_{train}^{target}, L, N)$
end
Output: the hypothesis $h_f(x) = \arg\max_{y \in Y} \sum_{k=1}^{K} I[h_k(x) = y]$
```

Fig. 6. Algorithm of TrAdaBoost with Bagging.

data from source domain along with the entire target training data. The final result is then voted by the K learners. Through this algorithm, the wrongly discarded instances in the source domain by a single TrAdaBoost learner have the opportunity to be averaged to different subsets; therefore, the risk of abandoning them will be reduced. Consequently, negative transfer could be avoided to some extent.

4.5. The overall system

With the method described above, we constructed a clinical concept extraction system with transfer ability. A flow diagram of the overall system is shown in Fig. 7. In contrast to traditional methods that have a single training data set, this system has clinical training data from both institution A and institution B. TrAdaBoost with Bagging is applied to transfer knowledge from institution A to institution B. The base learner of TrAdaBoost is MEM. For MEM, we employ “A Maximum Entropy Modeling Toolkit for Python and C++” package [42]. The parameters of MEM are set to be default.

The major contribution of our work is to provide a framework with transfer ability. Feature engineering is not the key component we focus on. A fundamental set of features for MEM is listed in Table 2.

Among the features listed in Table 2, we provide further details for the following features:

Word shape features: Tokens with similar word shape may be labeled as the same concepts. We convert uppercase letters and lowercase letters to “A” and “a”, respectively, and digital letters are converted to “0”.

POS features: The POS of a token is always helpful; we use GENIA [43], which is trained on biological literature to do POS tagging.

Table 2
Features in our work.

Category	Features
Word features	the word itself the word shape the POS 4-character-prefix-and-suffix contain with digit begin with digit contain with uppercase letter begin with a uppercase letter
Context features	two previous tokens two next tokens two previous tokens’ 4-character-prefix-and-suffix two next tokens’ 4-character-prefix-and-suffix
Sentence features	end with colon sentence’s tense
Section features	heading subsection heading

Sentence features: This feature includes whether a sentence ends with a colon and whether a sentence is in the past or the future tense.

Section features: Tokens with all uppercase letters and ending with a colon are headings; tokens with mixed-case letters and ending with a colon are subsection headings. Headings are divided into eight classes in our work, namely, medication, diagnosis, illness, complication, review, allergy, regimen and procedure.

5. Experiments

5.1. Data sets

Our experiments involve two data sets: one from the 2010 i2b2/VA challenge’s data, and the other from the biomedical literature.

The i2b2’s data consist of four sets: three are discharge summaries from Beth Israel Deaconess Medical Center, Partners HealthCare, and the University of Pittsburgh Medical Center, which are called BETH, PARTNERS and UPMCD in this paper, respectively; and the last is progress notes from the University of Pittsburgh Medical Center, which is called UPMCP in this paper. UPMCD and UPMCP were not available by the end of the challenge; as a result, BETH and PARTNERS are used in our experiments. The two data sets come from two institutions, with different writing styles and vocabularies, and thus are under different distributions.

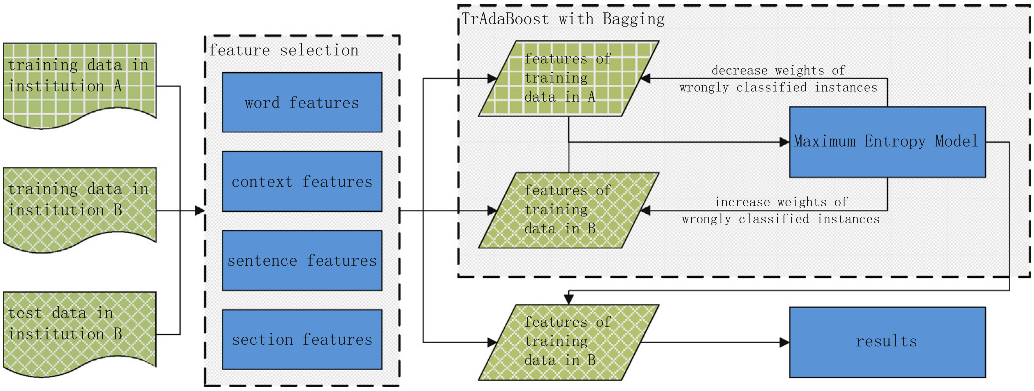

training data in institution A
training data in institution B
test data in institution B
feature selection
word features
context features
sentence features
section features
TrAdaBoost with Bagging
features of training data in A
features of training data in B
decrease weights of wrongly classified instances
increase weights of wrongly classified instances
Maximum Entropy Model
features of training data in B
results

Fig. 7. Flow diagram of the overall system.

To make the experimental results more convincing, we introduced an even more diverse data set in the biomedical domain, which was originally used for semantic relations classification, obtained from MEDLINE 2001 (we call it BIOLITERATURE) [44]. The data set consists of the first 100 titles and the first 40 abstracts from the 59 files named medline01n∗.xml in MEDLINE 2001 labeled by experts. We convert the labels in BIOLITERATURE to "treatment" and "problem", which are consistent with the labels in the i2b2's data (there are no correspondent "test" labels in BIOLITERATURE). We then combine the BETH data and the re-labeled BIOLITERATURE data into the third data set BETHBIO, which is more diverse than BETH and PARTNERS because of the importing of data from biomedical literature.

These three data sets are illustrated in Table 3.

For our transfer learning experiments, we take BETH and BETHBIO as source domain training data sets. Additionally, we split PARTNERS into two parts: the larger part, consisting of 80% of the data, serves as target domain training data set, and the smaller part, consisting of 20% of the data, serves as target domain test data set.

KL-divergence is introduced to quantify the diversity of data from different sets. Given two discrete distributions, their KL-divergence is defined as Eq. (1).

$$\mathrm{KL}(p(x), q(x)) = \sum_x q(x) \ln \frac{q(x)}{p(x)} \tag{1}$$

Table 4 shows the statistical information of the data sets in our transfer learning experiments. KL-divergences between different data sets are also presented in this table. For the same-distribution case, the KL-divergence is close to zero. In contrast, the KL-divergence between BETH and PARTNERS is 0.36. An even larger 0.79 KL-divergence between BETHBIO and PARTNERS demonstrates a larger diversity between the two data sets. Our framework aims to use a small amount of labeled corpus from target domain and a large amount of labeled corpus from source domain to build a more accurate concept extraction model.

5.2. Comparison methods

Our method is compared with three baseline methods. The distinctions among these methods depends on their training data, while the test data in all methods are the same, that is, T_{test}^{target}.

NoTr(S): Using training data from the source domain T_{train}^{source} alone to build a clinical concept extraction model.

NoTr(T): Using a small amount of training data from the target domain T_{train}^{target} alone to build a clinical concept extraction model.

Table 5
Description of the four methods.

Methods	Training data	Test data
NoTr(S)	T_{train}^{source}	T_{test}^{target}
NoTr(T)	T_{train}^{target}	T_{test}^{target}
NoTr(S∪T)	$T_{train}^{source} \cup T_{train}^{target}$	T_{test}^{target}
Our method	$T_{train}^{source} \cup T_{train}^{target}$	T_{test}^{target}

NoTr(S∪T): Combining T_{train}^{source} with T_{train}^{target} to build a clinical concept extraction model without transfer learning.

Our method: Similar to NoTr(S∪T), T_{train}^{source} and T_{train}^{target} are combined to build the training data, but we apply TrAdaBoost with Bagging to re-weight the instances in T_{train}^{source} in order to filter out "bad" instances while encouraging the "good" ones.

The description of the above methods is shown in Table 5; in each method, $|T_{train}^{target}| \ll |T_{train}^{source}|$. We introduce a parameter "r", which is the ratio between T_{train}^{target} and T_{train}^{source}, to observe how this value impacts the effect of transfer learning. Other parameters in our method are as follows: $N = 20$, $k = 5$, $rho = 0.8$. Performances of these methods are evaluated using the three standard performance metrics: precision (P), recall (R), and F measure (F), as shown in Eqs. (2)–(4). In the equations, TP stands for positives, FP stands for false positives, and FN stands for false negatives.

$$\text{Precision (P)} = \mathrm{TP}/(\mathrm{TP} + \mathrm{FP}) \tag{2}$$

$$\text{Recall (R)} = \mathrm{TP}/(\mathrm{TP} + \mathrm{FN}) \tag{3}$$

$$F = 2 \times \mathrm{P} \times \mathrm{R}/(\mathrm{P} + \mathrm{R}) \tag{4}$$

To indicate the reliability of P, R and F, we introduce confidence interval (CI) for them. CI gives an estimated range of values, in which the true value of a population parameter p is likely to be included. For example, a usually used 95% CI means that there is a 95% probability that the calculated confidence interval encompasses the true value of the population parameter. Given the Central Limit Theorem for Bernouilli trials [45], then we can calculate the 95% confidence interval according to Eq. (5), in which p will be P, R or F, and n will be the size of the test data set. We use Eq. (5) because n is large enough to meet the Central Limit Theorem. Additionally, CI is symmetric when using Eq. (5). To decide whether two systems' performances are significantly different on a task, one just has to observe whether their confidence intervals overlap.

$$\mathrm{CI} = \pm 1.96\sqrt{p(1-p)/n} \tag{5}$$

Table 3
The three data sets in our experiments.

Set name	Document type	Documents	Tokens	Problems	Tests	Treatments
BETH	Discharge summaries	73	88,722	4187	3036	3073
PARTNERS	Discharge summaries	97	60,819	2886	1572	1771
BETHBIO	Discharge summaries & Biomedical literatures	216	118,325	4631	3036	3296

BETHBIO is a combination of BETH and BIOLITERATURE. The number of the documents in BETHBIO is 73 (from BETH) plus 143 (from BIOLITERATURE); the rest can be performed in the same manner. Because there are no correspondent "test" labels in BIOLITERATURE, the number of "tests" is identical between BETH and BETHBIO.

Table 4
The data sets for transfer learning experiments.

T_{train}^{source}	T_{train}^{target}	T_{test}^{target}	$\#T_{train}^{source}$	$\#T_{train}^{target}$	$\#T_{test}^{target}$	KL
BETH	PARTNERS (80%)	PARTNERS (20%)	88,722	48659	12160	0.36
BETHBIO	PARTNERS (80%)	PARTNERS (20%)	118,325	48659	12160	0.79

5.3. Results and analysis

We design two groups of experiments to verify the effectiveness of our method. In the first group, we compare our method with the three baseline methods to demonstrate our method's transfer ability when the training data in the target domain is tiny; and in the second group, we reveal results of our method when the ratio between the training data in the target domain and the training data in the source domain changes by the means of parameter "*r*".

The results of the first group experiments are shown in Table 6. Here, the ratio "*r*" between $|T^{target}_{train}|$ and $|T^{source}_{train}|$ is set to be 0.02 by randomly selecting part of the data from T^{target}_{train}. In table 6a, T^{source}_{train} and T^{target}_{train} are training data from BETH and the 80% PARTNERS, respectively, and T^{target}_{test} is test data from the 20% PARTNERS. NoTr(T) yields the worst results due to the insufficient training data. NoTr(S) performs better because of the much larger training data set; however, due to the different distributions between the source domain and the target domain, the result has the potential to be improved. NoTr(S∪T) imports some training data from the target domain, but the improvement is not obvious. Our method yields the best results, which demonstrates the effectiveness of its transfer ability. Table 6b presents the results of BETHBIO vs. PARTNERS. Similar results appear in Table 6b compared with Table 6a, whereas the improvement is more obvious because the difference between the two domains is much larger, which can be reflected by the KL-divergence in Table 4. On the one hand, the larger difference between them plays a negative role in the three baseline methods due to the noisy data in BETHBIO. On the other hand, the original intention of our method was to solve the different distribution problem between the source domain and the target domain, and this data set is more suitable for the verification of our method. As seen in the above two mentioned experiments, the CIs for our method do not overlap with the CIs for the baseline models in almost all cases, which suggests that the better performance of our method has statistical significance. The situation that CIs overlap appears three times in Table 6a, while just one time in Table 6b, which also suggests that our method is more effective in the second experiment, as discussed above.

The results of the second group experiments are shown in Figs. 8 and 9. Here, we can see how the ratio between $|T^{target}_{train}|$ and $|T^{source}_{train}|$

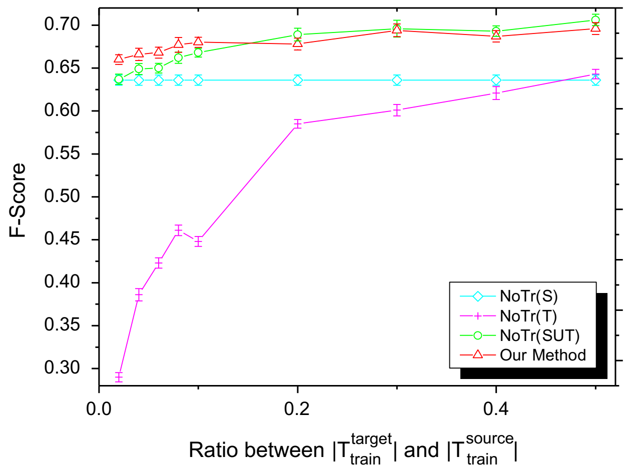


Fig. 8. The *F*-score curves on BETH vs. PARTNERS for four methods.

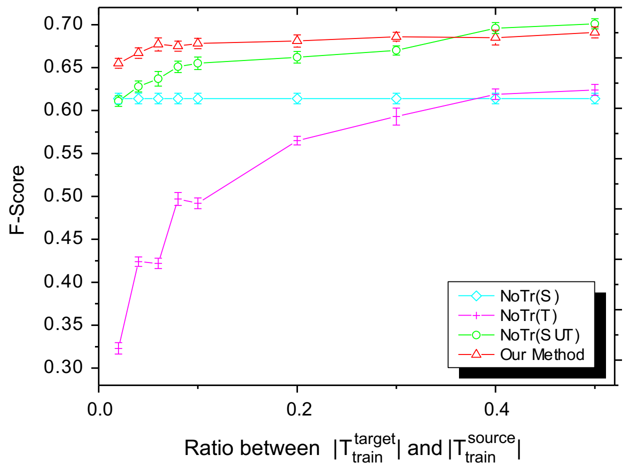


Fig. 9. The *F*-score curves on BETHBIO vs. PARTNERS for four methods.

Table 6
Performance of the four systems on BETH vs. PARTNERS and BETHBIO vs. PARTNERS when the ratio between $|T^{target}_{train}|$ and $|T^{source}_{train}|$ is set to be 0.02.

Method	Problem			Treatment			Test			Total		
	P CI(±)	R CI(±)	F CI(±)	P CI(±)	R CI(±)	F CI(±)	P CI(±)	R CI(±)	F CI(±)	P CI(±)	R CI(±)	F CI(±)
a. Results of BETH vs. PARTNERS (KL = 0.36)												
NoTr(T)	0.418	0.255	0.316	0.442	0.230	0.303	0.235	0.209	0.221	0.374	0.237	0.290
	0.012	0.011	0.008	0.015	0.012	0.009	0.014	0.015	0.010	0.008	0.007	0.005
NoTr(S)	0.625	0.642	0.633	0.686	0.652	0.669	0.566	0.715	0.631	0.630	0.643	0.636
	0.013	0.013	0.009	0.015	0.013	0.010	0.018	0.016	0.012	0.009	0.008	0.006
NoTr(S∪T)	0.611	0.664	0.636	0.671	0.622	0.646	0.574	0.669	0.618	0.621	0.653	0.637
	0.014	0.012	0.009	0.014	0.014	0.010	0.019	0.017	0.013	0.009	0.008	0.006
Our method	**0.648**	**0.676**	**0.662**	**0.701**	**0.652**	**0.676**	0.562	**0.723**	**0.632**	**0.644**	**0.676**	**0.660**
	0.014	0.012	0.009	0.014	0.014	0.010	0.019	0.017	0.013	0.009	0.008	0.006
b. Results of BETHBIO vs. PARTNERS (KL = 0.79)												
NoTr(T)	0.394	0.397	0.395	0.333	0.193	0.244	0.488	0.362	0.415	0.395	0.273	0.323
	0.013	0.013	0.009	0.014	0.011	0.009	0.017	0.017	0.012	0.008	0.008	0.006
NoTr(S)	0.600	0.668	0.632	0.629	0.607	0.618	0.542	0.690	0.607	0.589	0.641	0.614
	0.015	0.012	0.010	0.016	0.014	0.011	0.017	0.017	0.012	0.009	0.008	0.006
NoTr(S∪T)	0.598	0.605	0.601	0.638	0.605	0.621	0.584	0.658	0.619	0.604	0.618	0.611
	0.015	0.013	0.010	0.014	0.014	0.010	0.019	0.017	0.013	0.009	0.008	0.006
Our method	**0.642**	**0.678**	**0.659**	**0.673**	**0.642**	**0.657**	**0.603**	**0.714**	**0.653**	**0.658**	**0.652**	**0.655**
	0.013	0.012	0.009	0.013	0.014	0.010	0.016	0.016	0.012	0.008	0.008	0.006

P, R, F are the three performance metrics standing for precisions, recalls and *F*-score, which are defined in Section 5.2; CI(±) is confidence interval for P, R, and F.
Bold values mean that our system outperforms the baseline systems.

impacts the results of our method and the three baseline methods. In Fig. 8, when the ratio between the training data in PARTNERS and BETH increases from 0.02 to 0.5, the *F*-score of our method is always higher than NoTr(T) and NoTr(S). It exceeds NoTr(S∪T) when the ratio is less than 0.1 in either case. Additionally, CIs for the two methods do not overlap either. In such a situation, our method is effective. However, when the ratio is larger than 0.2, our method is a little worse than NoTr(S∪T) because the amount of data in T_{train}^{target} is more sufficient to train a satisfactory model. Therefore, there is no need to draw support from data in another domain. Training data with a different distribution may even be harmful to the model. Similar results appear in Fig. 9 for BETHBIO vs. PARTNERS. There are two differences compared with Fig. 8. First, the improvement is more obvious; second, our method outperforms the other baseline methods when the ratio is less than 0.4, which is a much larger value than the corresponding value in Fig. 8. The reason for these two differences is that BETHBIO has larger diversity that our method aims to reduce, and the latter data set is more suitable for revealing our method's effectiveness.

In summary, we draw the following conclusions: first, one of the major advantages is that our method is appropriate for the situation in which we have sufficient clinical training data from the source institution but only a tiny amount of clinical training data from the target institution. It can be seen in Table 6 that our method improves the total *F*-score by 2.3% and 4.4% compared with the best results of the three baseline methods. It also suggests that our method has a good generalization ability that can perform accurately on new, unseen instances. Second, our method is more suitable to situations where the ratio between training data in target domain and training data in the source domain is small, which is depicted as in Figs. 8 and 9. In fact, when the ratio is large, there is no need to import training data from the other domain because the training data in the target domain are sufficient. Lastly, we believe the degree of diversity between the source domain and target domain, which can be measured by KL-divergence, will impact our method's transfer ability. From the results displayed above, we can see that the improvement of our method is more obvious in the data set BETHBIO vs. PARTNERS than BETH vs. PARTNERS. Intuitively, we hypothesize that our method will work more effectively when the two domains have a larger diversity. However, it is very likely to be beyond our method's ability when the diversity in the two domains is too large. Restricted by the data sets, we cannot discuss this problem in a quantified manner with more experiments on more data sets in this paper, but we believe it is necessary for further study.

6. Conclusion and future work

This work presents a study using clinical documents from one institution to train a clinical concept extraction model for data from another institution. The two data sets from two institutions have different distributions, which violates traditional machine learning's basic assumption that training and test data must be under the same distribution. To address this problem, we applied TrAdaBoost, which is an instance-based transfer learning algorithm, to the clinical concept extraction task. Moreover, to prevent negative transfer, we combine Bagging with TrAdaBoost. Experiments show our framework is effective when using just a tiny amount of labeled training data from the target domain, and we can obtain a comparable clinical concept extraction system.

Future work will be carried out for the following items: 1. to develop other transfer learning methods such as feature-based transfer learning; 2. to evaluate data from multiple source domains simultaneously to extend the work; and 3. to transfer knowledge from biomedical literature to clinical documents, which is more challenging compared with the current work.

Acknowledgments

Deidentified clinical records used in this research were provided by the i2b2 National Center for Biomedical Computing funded by U54LM008748 and were originally prepared for the Shared Tasks for Challenges in NLP for Clinical Data organized by Dr. Ozlem Uzuner, i2b2 and SUNY.

We are also thankful for the reviewers' comments concerning our manuscript. These comments were valuable and very helpful for revising and improving our paper and have a profound guiding significance on our research.

References

[1] Simpson MS, Demner-Fushman D. Biomedical text mining: a survey of recent progress. In: Aggarwal CC, Zhai C, editors. Mining text data. US: Springer; 2012. p. 465–517.

[2] Uzuner Ö, South BR, Shen S, DuVall SL. 2010 i2b2/VA challenge on concepts, assertions, and relations in clinical text. J Am Med Inform Assoc 2011;18(5):552–6. http://dx.doi.org/10.1136/amiajnl-2011-000203.

[3] Dai W, Yang Q, Xue G-R, Yu Y. Boosting for transfer learning. In: Proceedings of the 24th international conference on machine learning. Corvalis, Oregon: ACM; 2007. p. 193–200.

[4] Demner-Fushman D, Chapman WW, McDonald CJ. What can natural language processing do for clinical decision support? J Biomed Inform 2009;42(5):760–72. http://dx.doi.org/10.1016/j.jbi.2009.08.007.

[5] Unified Medical Language System (UMLS). US National Library of Medicine, National Institutes of Health. <http://www.nlm.nih.gov/research/umls/>.

[6] Stearns MQ, Price C, Spackman KA, Wang AY. SNOMED clinical terms: overview of the development process and project status. In: Proceedings of AMIA annual symposium. AIMA; 2001.

[7] Friedman C, Alderson PO, Austin JHM, Cimino JJ, Johnson SB. A general natural-language text processor for clinical radiology. J Am Med Inform Assoc 1994;1(2):161–74. http://dx.doi.org/10.1136/jamia.1994.95236146.

[8] Friedman C, Shagina L, Lussier Y, Hripcsak G. Automated encoding of clinical documents based on natural language processing. J Am Med Inform Assoc 2004;11(5):392–402. http://dx.doi.org/10.1197/jamia.M1552.

[9] Aronson AR. Effective mapping of biomedical text to the UMLS Metathesaurus: the MetaMap program. In: Proceedings of AMIA annual symposium; 2001. p. 17–21.

[10] Hersh WR, Hickam D. Information retrieval in medicine: the SAPHIRE experience. J Am Soc Inf Sci 1995;46(10):743–7. http://dx.doi.org/10.1002/(SICI)1097-4571(199512)46:10<743::AID-ASI5>3.0.CO;2-C.

[11] Zhou X, Zhang X, Hu X. MaxMatcher: biological concept extraction using approximate dictionary lookup. In: Yang Q, Webb G, editors. PRICAI 2006: trends in artificial intelligence. Berlin, Heidelberg: Springer; 2006. p. 1145–9.

[12] Zou Q, Chu WW, Morioka C, Leazer GH, Kangarloo H. IndexFinder: a method of extracting key concepts from clinical texts for indexing. In: Proceedings of AMIA annual symposium. AMIA; 2003.

[13] Long W. Extracting diagnoses from discharge summaries. In: Proceedings of AMIA annual symposium. AMIA; 2005.

[14] Turchin A, Kolatkar NS, Grant RW, Makhni EC, Pendergrass ML, Einbinder JS. Using regular expressions to abstract blood pressure and treatment intensification information from the text of physician notes. J Am Med Inform Assoc 2006;13(6):691–5. http://dx.doi.org/10.1197/jamia.M2078.

[15] Bikel DM, Miller S, Schwartz R, Weischedel R. Nymble: a high-performance learning name-finder. In: Proceedings of the fifth conference on applied natural language processing. Washington, DC: Association for Computational Linguistics; 1997. p. 194–201.

[16] Joachims T. Making large-scale support vector machine learning practical. Advances in kernel methods. MIT Press; 1999.

[17] McCallum A, Freitag D, Pereira F. Maximum entropy Markov models for information extraction and segmentation. In: Proceedings of the seventeenth international conference on machine learning; 2000.

[18] Lafferty JD, McCallum A, Pereira FCN. Conditional random fields: probabilistic models for segmenting and labeling sequence data. In: Proceedings of the eighteenth international conference on machine learning. Morgan Kaufmann Publishers Inc.; 2001. p. 282–9.

[19] Taira RK, Soderland SG. A statistical natural language processor for medical reports. In: Proceedings of AMIA annual symposium. AMIA; 1999.

[20] Bashyam V, Taira RK. Indexing anatomical phrases in neuro-radiology reports to the UMLS. In: Proceedings of AMIA annual symposium. AMIA; 2005.

[21] Sibanda T, He T, Szolovits P, Uzuner O. Syntactically-informed semantic category recognizer for discharge summaries. In: Proceedings of AMIA annual symposium. AMIA; 2006.

[22] Wang Y, Patrick J. Cascading classifiers for named entity recognition in clinical notes. In: Proceedings of the workshop on biomedical information extraction. Borovets, Bulgaria: Association for Computational Linguistics; 2009. p. 42–9.

[23] Li D, Kipper-Schuler K, Savova G. Conditional random fields and support vector machines for disorder named entity recognition in clinical texts. In: Proceedings of the workshop on current trends in biomedical natural language processing. Columbus, Ohio: Association for Computational Linguistics; 2008. p. 94–5.

[24] De Bruijn B, Cherry C, Kiritchenko S, Martin J, Zhu X. Machine-learned solutions for three stages of clinical information extraction: the state of the art at i2b2 2010. J Am Med Inform Assoc 2011;18(5):557–62. http://dx.doi.org/10.1136/amiajnl-2011-000150.

[25] Roberts K, Harabagiu SM. A flexible framework for deriving assertions from electronic medical records. J Am Med Inform Assoc 2011;18(5):568–73. http://dx.doi.org/10.1136/amiajnl-2011-000152.

[26] Xu Y, Hong K, Tsujii J, Chang EI-C. Feature engineering combined with machine learning and rule-based methods for structured information extraction from narrative clinical discharge summaries. J Am Med Inform Assoc 2012;19(5):824–32. http://dx.doi.org/10.1136/amiajnl-2011-000776.

[27] Chen Y, Mani S, Xu H. Applying active learning to assertion classification of concepts in clinical text. J Biomed Inform 2012;45(2):265–72. http://dx.doi.org/10.1016/j.jbi.2011.11.003.

[28] Abacha AB, Zweigenbaum P. Medical entity recognition: a comparison of semantic and statistical methods. In: Proceedings of BioNLP 2011 workshop. Portland, Oregon: Association for Computational Linguistics; 2011. p. 56–64.

[29] Zhang S, Elhadad N. Unsupervised biomedical named entity recognition: experiments with clinical and biological texts. J Biomed Inform 2013;46(6):1088–98. http://dx.doi.org/10.1016/j.jbi.2013.08.004.

[30] Torii M, Wagholikar K, Liu H. Using machine learning for concept extraction on clinical documents from multiple data sources. J Am Med Inform Assoc 2011;18(5):580–7. http://dx.doi.org/10.1136/amiajnl-2011-000155.

[31] Sinno Jialin P, Qiang Y. A survey on transfer learning. IEEE Trans Knowledge Data Eng 2010;22(10):1345–59. http://dx.doi.org/10.1109/TKDE.2009.191.

[32] Schmidhuber J. On learning how to learn learning strategies; 1995.

[33] Thrun S. Is learning the nth thing any easier than learning the first? Advances in neural information processing systems; 1996. p. 640–6.

[34] Caruana R. Multitask learning. Mach Learn 1997;28(1):41–75. http://dx.doi.org/10.1023/a:1007379606734.

[35] Blitzer J, Dredze M, Pereira F. Biographies, bollywood, boom-boxes and blenders: domain adaptation for sentiment classification. Annual meeting-association for computational linguistics; 2007.

[36] Wu P, Dietterich TG. Improving SVM accuracy by training on auxiliary data sources. In: Proceedings of the twenty-first international conference on machine learning. Banff, Alberta, Canada: ACM; 2004. p. 110.

[37] Ganchev P, Malehorn D, Bigbee WL, Gopalakrishnan V. Transfer learning of classification rules for biomarker discovery and verification from molecular profiling studies. J Biomed Inform 2011;44:S17–23. http://dx.doi.org/10.1016/j.jbi.2011.04.009.

[38] Lee G, Stoolman L, Scott C. Transfer learning for auto-gating of flow cytometry data. J Mach Learn Res – Proc Track 2012;27:155–66.

[39] Arnold A, Nallapati R, Cohen WW. A comparative study of methods for transductive transfer learning. In: Seventh IEEE international conference on data mining workshops. ICDM workshops 2007, 28–31 October 2007.

[40] Freund Y, Schapire RE. A decision-theoretic generalization of on-line learning and an application to boosting. J Comput Syst Sci 1997;55(1):119–39. http://dx.doi.org/10.1006/jcss.1997.1504.

[41] Breiman L. Bagging predictors. Mach Learn 1996;24(2):123–40. http://dx.doi.org/10.1023/A:1018054314350.

[42] Zhang L. A maximum entropy modeling toolkit for python and C++. <https://github.com/lzhang10/maxent>.

[43] Tsuruoka Y, Tateishi Y, Kim J-D, et al. Developing a robust part-of-speech tagger for biomedical text. In: Bozanis P, Houstis E, editors. Advances in informatics. Berlin, Heidelberg: Springer; 2005. p. 382–92.

[44] Rosario B, Hearst MA. Classifying semantic relations in bioscience texts. In: Proceedings of the 42nd annual meeting on association for computational linguistics. Barcelona, Spain: Association for Computational Linguistics; 2004. p. 430.

[45] Grinstead CM, Snell JL. Introduction to probability. 2nd ed. Providence (RI): American Mathematical Society.