Supervised Identification and Linking of Concept Mentions to a Domain-Specific Ontology

ABSTRACT

We propose a supervised learning approach, SDOI, to the task of identifying concept mentions within a document and of linking these mentions to their corresponding concept node, if it exists, in a domain-specific ontology. Concept mention identification is performed with a trained sequential tagging model. Each identified mention is then associated with a set of candidate ontology concepts along with their feature vectors. We formalize feature spaces proposed in the literature and expand it into new data sources, such as from the training corpus itself. An iterative algorithm is defined for handling collective features which assume that some of the labels are known in advance. The approach is validated against the ability to identify the concept mentions within the 139 KDD-2009 conference paper abstracts, and to link these mentions to a domain-specific ontology for the field of data mining. We show a lift in over existing approaches applicable to the task. Additional experiments on a separate corpus from the same domain suggest that the trained models are portable both in terms of accuracy and in their ability to reduce annotation time.

Categories and Subject Descriptors

H.3.3 [**Information Search and Retrieval**]: Information Systems – *Information Storage and Retrieval*.

General Terms

Algorithms, Experimentation.

Keywords

Concept mentions; Semantic annotation; Supervised learning, Ontology; Collective inference.

1. INTRODUCTION

The value that can be gained from the growing availability in both electronic documents and ontologies will increase significantly once these two resource types are deeply interlinked. Imagine for example the time when the concepts mentioned within a research paper are linked to the corresponding concept within an ontology from the paper's research area; or when the concepts mentioned in the business rules within a corporation's policy documents are linked to the concepts in their corporate ontology. The documents would now behave more like hyperlinked webpages than static Adobe PDF or Microsoft Word files, and would thus enable more navigational strategic reading, when required. Further, searches on domain specific concepts such as "*supervised approaches to concept mention linking"* could also be more effective than the current ad hoc approach of ever more finely tuned keyword-based searchers. Similarly, the use and development of ontologies will benefit from links to a concept's usage in natural language. By seeing how a concept is described, used and constrained in linguistic expressions the meaning of a weakly developed concept can generally be more quickly understood and improved upon. Finally, deep interlinking could enable new forms of information retrieval, extraction and analysis.[1]

As described, the task addressed in this paper can be naturally decomposed into two subtasks: 1) the identification of relevant concept mentions in a text; and, 2) the linking of each of these mentions to some appropriate concept in an ontology, if such a concept exists. An obstacle however to the vision of deeply interlinked information is the significant amount of manual effort required by both subtasks. Some automation of these tasks is a precondition for deep interlinking, and some of the recent research described in the next section hints at the large-scale feasibility of this automation.

1.1 Related Work

The task of annotating text with semantic information, possibly as a pre-processing step, has been addressed by several research areas ranging from natural language processing, information retrieval, and information extraction. In the field of natural language processing two related tasks within lexical semantics are word sense disambiguation [1] and named entity recognition [5]. In word sense disambiguation however, the mentions sought are infrequently multi-word expression, and the lexical database is assumed to be complete both in alternate spellings and in word senses. Our task has the additional challenge of frequent multi-word mentions and the inventory (the ontology) can be incomplete and contain few of the alternate phrasings. In named entity recognition the mentions sought are often multi-token expressions, but the number of concepts to be linked to is very small (e.g. person, protein, organization, and location) and the mentions have additional structure (e.g. nouns with frequently capitalized first letters). In our task it is common to have no dominant conceptual category to link to and for the mentions to have less structure. Other related research arises from the extraction of technical terms to automatically create a book's subject index [12], and automated population of database and ontologies [10].

The field of biomedical text mining has actively investigated the ability to identify concepts in research papers and to link them to domain specific databases, such as the gene and protein database Swiss-Prot[2] or to Gene Ontology[3] [14]. The field's focus however

Conference'04, Month 1–2, 2004, City, State, Country.
Copyright 2004 ACM 1-58113-000-0/00/0004…$5.00.

[1] This paper's abstract illustrates the envisioned annotation. Its concept mentions are identified and linked to an ontology (in this version of the document the hyperlinks lead nowhere, and would be reinserted after the double-blind review process).

[2] http://www.ebi.ac.uk/swissprot/ http://expasy.org/sprot/

with respect to identification and linking of mentions (as opposed to detecting relations between mentions) continues to be on named entity mentions such as of proteins, genes, and organisms; with solutions generally proposing ways of cope with the multitude of spellings and abbreviations of their entities. Less attention has been given to the processing of more general concept mentions, such as biological or experimental processes.

Recently, some research has begun to investigate the more general task of identifying and linking of concept mentions to those concepts that are found in Wikipedia using supervised learning [4,8,9]. Further, while Milne & Witten in [9] and Kulkarni & al in [4] focus on identifying and linking concept mentions within Wikipedia pages, their research also begins to explore the application on non-Wikipedia documents such as news articles. Both propose a relatively small feature space with a significant focus on features that require some of document's links to be accurately pre-predicted in order to inform the linking decisions for the remaining mentions based on the document's context.

Milne & Witten in [9] propose the use of three features for linking mentions to Wikipedia. One feature is simply the proportion of inlinks associated with the concept (its commonness). The two other features are based on a proposed semantic relatedness measure between a candidate concept and the concepts mentioned in the document that are naturally disambiguable. As a first phase they detect the "context concepts" that are assumed not to require disambiguation, and then proceed to a supervised learning phase. They also use of the number of links into the candidate concept from within Wikipedia (the concept's commonness) as a feature. A challenge of applying their approach to the more general task of non-Wikipedia documents is their requirement that some of the mentions in the document can be naturally linked to the ontology without the need for disambiguation (in order to provide the context for the relatedness features). Their two phase approach could benefit from a more conservative approach that continually increases the mentions that will be deemed as disambiguated, and for these context concepts to be decided by the same trained classifier.

Kulkarni & al in [4] extend the link selection work of [9] in two main ways. They add additional features based on the similarity between the bag-of-words representations of the text-window surrounding the concept mention and of the concept's description in the ontology. They also propose a more sophisticated scheme to handle the collective features. Specifically they propose the use of an objective function that sums up the probability estimate produced by the trained classifier (based on bag-of-word features) and the relatedness measure proposed in [9] tested on all pairs of candidate concepts. They empirically show that optimizing on the proposed function closely tracks *F1*-measure performance (on several of their test documents, as the value of their objective function increases so did *F1* performance). They explore two optimization algorithms for finding an optimal link assignment to the objective function. One algorithm is based on integer linear programming while the other based on greedy hill-climbing. Foreseen challenges to the application of their proposed approach to our task include its use of the longest matching sequence heuristic for concept mention identification which will reject many candidate mentions. Also, updating the proposed objective function to include additional features, or new definitions of relatedness, could unwittingly degrade algorithm performance. Finally, they do not explore the use of features that directly compares the mention or document to the concept and its description.

1.2 Our Contribution

We propose supervised algorithm to the tasks of concept mention identification and linking to an ontology: `SDOI`.

`SDOI` first trains a sequential classifier to identify concept mentions that need not have been mentioned before in the corpus nor be present in the ontology. The use of sequential models has been successfully used in the NLP community to the related tasks of text chunking and named-entity recognition, thus any improvements to these other solutions, such as the handling of global features can be naturally imported into our solution.

Next, `SDOI` identifies a set of candidate concepts for each mention based on heuristic candidacy rules that are more general than those currently proposed in order to expand the recall rate. For each candidate concept, an expanded set of features is defined in order to improve precision. We address the procedural challenge of collective features with an iterative classification algorithm. This approach simplifies the algorithm's reimplementation, naturally allows for the addition of more features, and enables the use of an off-the-shelf supervised binary classification algorithm.

We validate `SDOI` on a novel corpus and domain specific ontology consisting of the 139 abstracts of the papers accepted to the KDD-2009 conference, and whose concept mentions have been manually identified and linked, where possible, to the concepts within a nascent data mining ontology. We further validate the portability of the trained model by assessing accuracy and time savings on papers from a separate conference: ICDM'09.

The remainder of the paper is structured as follows: The next section formally defines the task. Sections 3 through 6 describe the `SDOI` algorithm, starting with concept mention identification, candidate set generation, the feature space and finally the use of iterative classification. Section 7 describes the corpus, ontology and reports on the empirical evaluation, and Section 8 concludes with a discussion of possible research directions.

2. TASK DEFINITION

Assume that we are given a corpus of text documents $d_i \in D$ where each document is composed of sentences based on sequences of *tokens* (orthographic words or punctuation).

Assume also the existence of an ontology of interrelated *concepts*, $o_c \in O$, the represent and describe some concept within some domain. The concepts are interconnected by directed edges referred to as *internal links* (λ) that link one concept to another concept, $\lambda(o_{c'}, o_{c''})$. Each concept o_c can be associated with: a *preferred name*, p_c, a set of (also-known-as) synonyms A_c, and descriptive text t_c. As described, an ontology is a directed and labeled multigraph that could be used to represent such diverse structures as Wikipedia[4] (with its rich text and weak semantics) to the Gene Ontology (with its rich semantics and terse descriptions).

[3] http://www.geneontology.org/GO

[4] http://www.wikipedia.org

Assume next that each document d_i has a set of non-overlapping non-partitioning subsequences of tokens referred to as *concept mentions*, $m_m \in d_i$, that refers to a domain specific meaning not generally found in a dictionary. We assume that the domain of the corpus overlaps the domain of the domain-specific ontology.

Every concept mention m_m is connected via a directed edge to either the concept o_c that captures the mention's intended meaning, or to the symbol "**?**" that denotes the absence of the concept within the ontology. We refer to these edges as *external links* and denote them as φ (m_m, o_c). An *unlinked* concept mention, φ (m_m, ?),is one that cannot be linked to the ontology because the concept is not yet deemed to be present in the ontology. We can refer to a mention's token sequence as its *anchor text*, a_m, to distinguish the text from the concept it links to.

Figure 1 illustrates the concept mentions within a document. Next, Figure 2 illustrates the objects and relations available for analysis. Finally, Table 1 contains some additional terminology related to the task description.

Given a document from the same domain as the ontology that lacks the concept mention information, the task is to identify each of the concept mentions within the document: both their *anchor text* and their corresponding *external link*.

```
[[Collaborative    Filtering    Algorithm|
Collaborative   filtering]]   is   the  most
popular   [[Algorithm|approach]]   to   build
[[Recommender System|recommender systems]]
and has been successfully employed in many
[[Computer   Application   |applications]].
However,   as   [[?|(Schein   &   al,   2002)]]
explored,  it  cannot  make  recommendations
for so-called  [[?|cold start users]]  that
have  rated  only  a  very  small  number  of
[[Recommendable Item|items]].
```

Figure 1 – The example above uses wiki-style formatting to illustrate concept mention identification and linking. The doubled square-brackets identify *concept mentions*, with the internal vertical bar (|) separating the *anchor text* (on right) from *concept* (on left). The question mark character (?) signals that the corresponding *concept* is not present in the ontology.

Table 1 – Terminology associated with the task

m_i	The i^{th} concept mention in the corpus, $m_i \in D$.
o_i	The i^{th} concept node in the ontology, $o_i \in O$.
$I(o_i)$	The set of internal links into o_i from some o_k
$O(o_i)$	The set of internal links from o_i into some o_k.
$E(o_i, D)$	The set of external links into o_i from some $m_k \in D$.

3. IDENTIFYING CONCEPT MENTIONS

To identify concept mentions in a document we train a sequence tagging model in the same spirit as proposed in [13] for the task of text chunking, and in [4] for named entity recognition. We use BIO tagging, which requires the first token of a mention to be labeled with character B, any remaining mention tokens labeled with I, and all other tokens are labeled with O. Figure 3 illustrates the labels used to identify concept mentions.

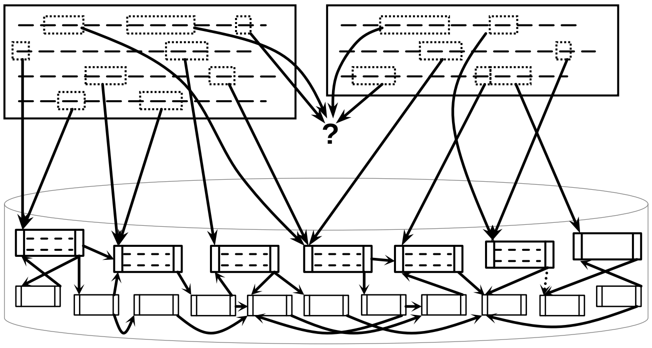


Figure 2 – An illustration of the task's training data. The two objects on top represent two text *documents*. The object below represents the *ontology* of *concept nodes* and *internal links*. Some non-overlapping subsequences in the documents are *concept mentions* mapped to either concept nodes or the unknown concept symbol (?), via *external links*.

These approaches generally make use of at least two feature sources: the token and its part of speech (POS) role. For the POS information the use of an automated part-of-speech tagger (rather than manual annotation) is accepted practice. We include features also proposed for the named entity recognition task of: whether the first letter is capitalized, whether a token contains a number or a special character and whether the token contains fewer than four characters. We use a five token window and test the unigram, bigram and trigrams that include the target token.

```
Collaborative/B filtering/I is/O the/O
most/O   popular/O   approach/B   to/O
build/O recommender/B systems/I and/O
has/O been/O successfully/O employed/O
in/O many/O applications/B ./O
```

Figure 3 – Sample of the first sentence in Figure 1 labeled for concept mention identification.

4. CANDIDATE CONCEPT SETS

A concept mention can be linked to any one of the many concept nodes in the ontology. However, knowledge of the mention's anchor text can be used to significantly reduce the number of concept nodes that should be realistically considered as candidates for the assignment, without discarding the correct node in the process. As an example, assume that a concept mention contains the anchor text composed of the single token of "*features*" then its candidate concept set might include the concepts for "Predictor Feature", "Application Feature", and "Data Table Attribute", one of which ideally is the correct concept.

This section explores a composite heuristic used to create the *candidate set* for a given anchor text; where a candidate set is composed of zero or more distinct concepts from the ontology: $a_m \rightarrow C_m = \{\varnothing, o_{c'}, o_{c''}, \ldots\}$. The heuristic is composed of a series of eight individual tests between a concept mention's anchor text and some information about the concept $t_i(a_m, o_c)$. Each test results in a set of accepted concepts and the overall heuristic accepts the union of all accepted concepts. Thus, a concept must pass at least one of the tests to become a member of a mention's candidate set. We describe a set of eight heuristics below. The actual subset of these tests that will be used for *SDOI* will be determined empirically (see section 7.3).

The first test to be considered, t_1, requires an exact match between the anchor text and the concept's preferred name. The second test, t_2, extends this pattern and requires that the anchor text exactly match any one of the concept's pre-identified synonyms (e.g. as materialized in the redirect pages in Wikipedia). These two tests can be used to replicate the proposals in [4, 8]. In more specialized domains however, with complex multi-token mentions and with nascent ontologies that have small and incomplete synonym sets, these two tests would result in a weak recall rate of the correct concept. The anchor text of "*supervised learning of a sequential tagging model*" for example would be missed because it is too specialized an expression to become an official synonym.

We define two additional candidacy tests for consideration. One of the tests, t_3, probes into the documents in the training corpus to determine whether the anchor text was also linked to this concept. In a sense, this test extends that of the synonym test in that at some future time some of these matching anchor texts will likely become official synonyms for the concept.

Given that a large proportion of concept mentions and concept synonyms are composed of more than one token, the final primary test, t_4, accepts a concept node where any of the component tokens match. Table 2 summarizes the four *primary* tests of candidacy.

Table 2 – the primary tests used to determine whether concept node (o_c) becomes a member of the candidate concept set (C_m) for anchor text (a_m).

t_1	The anchor text (a_m) matches the concept's preferred name (p_c)
t_2	The anchor text (a_m) matches a synonym of the concept (o_c)
t_3	The anchor text (a_m) matches a linked *anchor text* (in some other document) to the concept, $a_{k'} \in d_k$, $\varphi(a_{k'},o_c)$ *and* $k \neq m$
t_4	A token in the anchor text (a_m) matches a token within the preferred name (p_c), a synonym (s in S_c), or a linked anchor text (a_k) in some other document

Finally, each of the four primary tests is associated with an alternative test that is based on the use of the stemmed versions of the text being compared. We denote these tests as: t_{s1}, t_{s2}, t_{s3}, and t_{s4}. Note that, if a test succeeds on a primary test then it will also succeed on the stemmed version of the test.

5. CONCEPT LINKING FEATURES

Given a candidate concept set for each mention, and given training examples that identify the correct concept link, the task of identifying the concept links in unseen documents can be accomplished by training a supervised binary classification model. This section describes the feature vector associated with each paired mention/concept training case that will be used to train the classifier. Table 3 illustrates the structure of the training data produced and Table 4 presents a summary of the (non-collective) features about to be described.

Note that the features under the category of "Collective Features" are recursively defined (they are based on predictions for some mentions). The evaluation of these features is achieved with an iterative classification algorithm described in Section 6.1.

Table 3 – Illustration of the structure of the training data used for the linking task. A feature vector is created for each mention in the training corpus and concept candidate from the ontology. The vector is then labeled based on whether it corresponds to the link assigned by the human annotator.

Mention	Concept	Predictor Features													Label
		Anchor Txt.		Text Wind.	Document		Concept		Corpus		Cand. Set		Collective		
m_m	o_c	a_m,o_c		t_m,o_c	d_m,o_c		o_c		$D\text{-}d_m$		C_{mc}		o_c,S_m		T/F
1	1903	0	…	0.01	0.03	…	3	…	0	…	15	…	4	…	F
1	1021	0	…	0	0.01	…	1	…	2	…	30	…	1	…	F
1	829	1	…	0.02	0.02	…	12	…	1	…	15	…	7	…	T
2	4028	0	…	0.08	0.08	…	5	…	11	…	30	…	9	…	T
…	…	…	…	…	…	…	…	…	…	…	…	…	…	…	…

5.1 Anchor Text-based Features *f*(a_m,o_c)

Each of the eight candidacy tests defined in the previous section are included as binary features. The intuition for their inclusion is that these tests can signal how closely a mention's anchor text matches text associated with the concept node.

5.2 Text Window-based Features *f*(t_m,o_c)

Another source of features can be based on text near the concept mention (its *text window*) and from the text used to describe the concept in the ontology. The feature included in *SDOI* that is based on this information is the cosine distance between the normalized bag-of-word vector representations of the text window and of the ontology description. This feature is proposed in [4] (they also include dot product and Jaccard similarity).

5.3 Document-based Features *f*(d_m,o_c)

The entire document can be used to inform the classification decision. Two proposed features are: 1) the cosine distance between the normalized bag-of-word vector representations of the document and the ontology description (also proposed in [4]), and 2) the token position of the concept mention within the document (1st token, 2nd token …). The intuition of the later feature is that different types of concepts are expressed near the beginning of a document rather than later on.

5.4 Concept-based Features *f*(o_c)

The candidate concept node on its own along with its role within the ontology (without knowledge of the specific concept mention being considered) can also inform the classification decision. For example, [4] and [9] propose the use of the frequency that a concept is linked to (its inlink count) as a feature. We include this count as a feature, $|I(o_c)|$ and also include the count of internal links extending out of the concept, $|O(o_c)|$. The first feature signals the popularity of the concept as a reference. The second feature can signal whether the concept has received significant attention by the ontology designers in the form of additional links.

5.5 Corpus Based-based Features $f(o_c,D\text{-}d_m)$

Interestingly, the training corpus can also be an alternative source of information for predictor features. Indeed the t_3 candidacy test signals the presence in the corpus of an identical mention. An additional corpus-based feature is the count of documents that also have *external* links to the ontology concept: $|E(o_c)|$.

An implementational challenge with this feature source, particularly during cross-validation studies, is that when calculating a mention's corpus-based feature care must be taken to exclude all of the other mentions that occur within the document. If an anchor text for example is repeated elsewhere in the same document then in order to truthfully replicate the testing environment the other mentions in the document cannot influence the calculation of the feature. Thus, the value associated with this feature for a given mention can differ for every document in the training corpus.

5.6 Candidate Set-based Features $f(C_{mc})$

Awareness of the size and membership of entire set of candidate concepts associated with the concept mention can inform the classification decision. For example it is riskier to pick a concept from a large candidate set than from a candidate set composed of only two members.

5.7 Collective-based Features $f(o_c,S_m)$

We describe a set of recursively defined features whose calculation requires knowledge about the label for some of the links that we are trying to predict. With possession of some disambiguated links to the ontology, the ontology can be used to provide some background knowledge into the classification decision for the remaining links. For example, if we knew that a document mentioned the concept "*supervised learning algorithm*" then the decision of which candidate concept to predict for the mention of "*feature*" may be improved (i.e. *predictor feature*, not *computer program feature*, nor *database attribute*). How to attain such a partial set of labeled links will be addressed in the next section. Let S_m be the context set of disambiguated concepts in document d_m.

In order to replicate the work in [9] we include the relatedness measure they propose, which in turn is based on the Normalized Google Distance (*NGD*) metric [2] that assesses the dissimilarity between two sets.

$$NGD(A,B) = \frac{\log(\max(|A|,|B|)) - \log(|A \cap B|)}{\log(|O|) - \log(\min(|A|,|B|))}$$

Table 4 – *SDOI*'s Features for Concept Mention Linking

FEATURE	DEFINITION
$\cos(d_m,o_c)$	The bag-of-word cosine similarity between the document and the concept description, as proposed in [4].
$\text{tok}(a_m,d_m)$	Number of tokens between the start of the document and the first token in the mention.
$\|I(o_c)\|$	Cardinality of all internal links into o_c, as proposed in [4] and [9].
$\|O(o_c)\|$	Cardinality of all internal links from o_c.
$\|E(o_c)\|$	Cardinality of all external links into o_c from the corpus D
$\|C_i\|$	Cardinality of the set of candidate concepts. i.e.
$\Sigma\|I(C_i)\|$	Count of internal links into all candidate concept nodes. $\|I(o_{j'})\| + \|I(o_{j''})\| + \ldots$, for all $o_j \in C_i$
$R\|I(o_j,C_i)\|$	Relative proportion of the internal links into the candidate concept relative to overall size. $\|I(o_j)\| / \Sigma\|I(C_i)\|$
$\Sigma\|O(C_i)\|$	Count of internal links out from all candidate concept nodes. $\|O(o_{j'})\| + \|O(o_{j''})\| + \ldots$, for all $o_j \in C_i$
$R\|O(o_j,C_i)\|$	Relative proportion of the internal links out from the candidate concept relative to overall size. $\|I(o_j)\| / \Sigma\|O(C_i)\|$
$\Sigma\|E(C_i)\|$	Count of external links into all candidate concept nodes. $\|E(o_{j'})\| + \|E(o_{j''})\| + \ldots$, for all $o_j \in C_i$
$R\|E(o_j,C_i)\|$	Relative proportion of the external links into the candidate concept relative to overall size. $\|E(o_j)\| / \Sigma\|E(C_i)\|$

Given two ontology concept nodes (o_a, o_b) the relatedness function proposed in [9] tests the links into two concepts nodes(o_a, o_b), where $A=I(o_a)$, and $B=I(o_a)$. Also, the $[0,\infty]$ function range of *NDG* is converted to a similarity metric by truncating the output to [0,1] and then subtracting it by 1.

$$MW08rel(o_a,o_b) = 1 - NGD(o_a,o_b), \quad if\ NGD(o_a,o_b) < 1 \qquad else\ = 1$$

We extend this feature space by also including the components used in the calculation of *NDG*. We also include a Jaccard set similarity feature. Table 5 defines the collective features. In the case where the context set of mentions is empty (when no mentions have been linked) these features all calculate to zero (0).

Table 5 – Definition of the collective features.

FEATURE	DEFINITION
$\|S_i\|$	The cardinality of the set of context concepts.
$\Sigma IS(S_i)$	The count of internal links to the set of context concepts.
$AIS(S_i)$	The average number internal links into each of the context concepts.
$AM\cap S(o_j,S_i)$	The average cardinality of the intersection between the links into concept o_j and the internal links into the anchor concepts in S_i. $Ca\cap b(o_j,S_i)$, where $o_k \in S_i$ and $o_j \neq o_k$.
$AMW08rel(o_j,S_i)$	The average weighted relatedness between the concept node o_j and each of the concept nodes in S_i, as proposed in [9].
$\Sigma MW08\ (S_i)$	The sum of the relatedness between each concept in S_i. to the other concepts in S_i. This feature is proposed in [9] to inform the classifier about the entire context set.
$AJacc(o_j,S_i)$	The average Jaccard set similarity between the links into the o_j concept and each of the concept nodes in S_i.

6. ITERATIVE SET-BASED CLASSIFIER

Given a set of feature vectors with a binary label one could apply an off-the-shelf supervised classification algorithm to the task. However, two issues impede the casting of the problem in this manner: the recursive definition of the collective features and the need to classify at the level of the mention (not at the level of the mention/concept case). These two issues are addressed below:

6.1 Collective Feature Handling

The "Collective-based" features defined in Section 5.7 require that some portion of a document's concept mentions be already linked to the ontology. Milken and Witten in [9] accomplish this assignment by first identifying some mentions heuristically as "context" mentions that do not require disambiguation. Kulkarni et al in [4] accomplish this assignment by specifying a custom objective function that is then optimized by, for example, iteratively committing to the next highest scoring mention.

We also propose an incremental approach, but one that is directed by an iterative supervised classification algorithm inspired by the one proposed in [11]. Our algorithm first trains a model on an idealized context set, S_m, with all the correctly labeled mentions; then, during the testing phase, it iteratively grows the context set based on an increasing proportion of the most likely predictions.

Because our collective features all zeroed (0) initially, we enhance the approach of [11] by first training a model on all but the collective features to seed the first guesses with informed choices. Assume that we define a constant number of iterations μ and a set of N training instances. The algorithm used is as follows:

1. Train model ($M_{\cancel{col}}$) without the collective features
2. Train a model (M_{col}) with the collective features
3. For each iteration of ι from 1 to μ
    a. Calculate the value for the collective features
    b. Apply model $M_{\cancel{col}}$ to the test set, if ι is 1 otherwise, apply model M_{col} to the test set.
    c. Select the κ most probable links, where $\kappa = N(\iota/\mu)$.
4. Output the final set of predictions on all mentions.

6.2 Candidate Set-based Predictions

Recall that the goal of the task is to select at most one concept/link per candidate set. The classifier however may assign the label of "True" to more than one concept associated to a mention. When this is the case *SDOI* uses a tie-breaking rule. A possible tie-breaking rule is to make a random selection. However, if the supervised classifier used also reports a value that can rank the predictions according to their likelihood (e.g. support vector machines, decision tree, and logistic regression) then *SDOI* uses this number to select the more likely concept.

7. EMPIRICAL EVALUATION

The proposed *SDOI* algorithm is evaluated on a corpus whose concept mentions have been identified and linked to a domain-specific ontology. The corpus is composed of the paper abstracts of ACM's KDD 2009 conference, and the ontology is based on a semantic wiki created specifically for the field of data mining. To our knowledge this is the first ontology and annotated corpus for a computing discipline. The corpus and ontology are publicly available[5]. They are summarized below and additional details are forthcoming in a separate publication[6]. Further we test the portability of the model trained on the above data to identify and link concepts mentions from abstracts of a separate conference track: IEEE's ICDM 2009 conference.

7.1 Benchmark Datasets

7.1.1 The `kdd09cam1` Corpus

The annotated corpus used in the study, `kdd09cma1`, is composed of the 139 abstracts for the papers accepted to ACM's SIGKDD conference which took place in 2009 (KDD-2009)[7]. The competitive peer-reviewed conference on the topic of data mining and knowledge discovery from databases has acceptance rates in the range of 20% -25%. The annotation of the corpus (identification and linking of concept mentions) was performed by one of the authors and was performed in two separate phases. We first identified mentions of concepts that would be understood and/or are often used within the data mining community without consideration for what concepts existed in the ontology. Next an attempt was made to link the mentions to the concept in the ontology (described in the next section) that stood for the intended concept in the mention. On average the identification task took approximately 6 minutes per abstract, while the linking task took approximately 17 minutes per abstract. To evaluate the quality of the annotation, sixteen abstracts were randomly selected and the paper's author was asked to review the annotation. Fourteen authors responded and simply accepted the annotation as is.

[5] A URL is temporarily withheld due to the double-blind review.

[6] The reference is temporarily withheld due to the double-blind review.

[7] The KDD-2009 abstracts are freely accessible from ACM's Digital Library http://portal.acm.org/toc.cfm?id=1557019

The corpus bears similarities to corpora from the bio-medical domain such as the GENIA[8] and BioCreAtIvE[9] that are based on research paper abstracts found in MEDLINE abstracts and the terms are linked to concept in some ontology. Those corpora however focus on the annotation of basic named entities such as molecules, organisms, and locations. The `kdd09cma1` corpus on the other hand contains very few named entities. Being from a formal science, its concept mentions range from single token ones such as "*mining*" to multi-token ones such as "*minimal biclique set cover problem*". Also, in cases where named entities do appear they often are embedded within an abstract concept mention, as in "*Gibbs sampling method*". The text was tokenized and assigned a part-of-speech role by using Charniak's parser [3]. Table 6 summarizes some key statistics about the corpus.

Table 6 – Summary statistics of the `kdd09cma1` corpus, including the minimum, median, and maximum per abstract.

Documents		139	Per Document (min/med/max)
Sentences		1,186	3 | 8| 17
Tokens		29,139	105 |220| 367
Concept Mentions	(100%)	7,580	26 | 52| 96
Single Token	(~66%)	5,001	12 | 35| 65
Multi Token	(~33%)	2,579	4 | 18| 38

7.1.2 *The* `dmswo1` *Data Mining Ontology*

The ontology used in the study was based on a custom built semantic wiki[10] created specifically for the field of data mining and text mining by one of the authors. Each concept has its own distinct wiki page[11] and follows the structured English approach described in [5], where each concept contains: 1) A preferred name; 2) A one sentence definition in the form of "*an X is a type of Y that …*"; 3) A set of possible synonyms; 4) A set of relationships to other concepts stated in structured English; 5) A set of sample instances of the concept; 6) A set of counter-examples of the concept; 7) A set of related terms whose relationship has not been formally defined; and 8) a set of relevant external references for the concept. Table 7 summarizes some statistics of the ontology.

Table 7– Summary statistics of the `dmswo1` ontology

Concepts			5,067
Concept links			27,408
	Min	**Median**	**Max**
Links into a Concept	0	3	157
Links out of a Concept	2	3	444
Synonyms per Concept	0	1	8

Given the novelty of the corpus and ontology Table 8 summarizes some additional key statistics of the linking (external links) between the corpus and ontology.

Table 8 – Summary statistics of the external links from the `kdd09cma1` corpus to the `dmswo1` ontology.

Documents	139	Per Document (min/median/max)
Linked Mentions	51.7% 3,920	10 | 26 | 66
Unlinked Mentions	48.3% 3,660	3 | 25 | 49
Distinct Concepts linked to by corpus	820	9 | 19 | 50
Concepts uniquely linked to by a single document		0 | 2 | 17

7.2 Baseline Algorithms

We compare performance against baseline algorithms for each of the subtasks.

For the mention identification task our baseline is a dictionary based algorithm (*dict*) that selects the longest sequence of tokens that matches a concept's preferred name or synonym. This is the identification method used in [4] and the non-Wikipedia experiments in [9]. For our task this baseline will likely achieve poor recall rate because it cannot identify concept mentions that are not yet in the ontology.

The main baseline algorithm for the linking task is the supervised approach proposed in [9]. We reimplemented the three features defined in their proposal (see Section 5: *RI*, *AMW08rel*, and Σ*MW08*), and the algorithm's two phased approach to handle the two collective features used. The first phase selects a set of context concepts (S_{MW08}), and the second phase applies a binary classifier on the three features. We replicate the first phase by committing to all candidate concepts that pass tests t_1 and t_2, and that result in a single candidate concept.

For the joint task of identification and linking, both the baseline and the proposed *`SDOI`* algorithm simply direct the output of their identification algorithm (the predicted anchor text for the concept mentions) as input to their linking algorithm.

7.3 Mention Identification Evaluation

The performance of the sequential model-based algorithm on the mention identification task was evaluated on the `kdd09cma1` corpus[12]. We present both the results of a leave-one-document-out analysis on the entire data in Table 9 and a learning curve analysis in Figure 4 that illustrates performance trends for different training set sizes. We also present performance on exact and partial matches of anchor text; where a partial match is defined as starting on the correct token but ending on a different token. From Table 9 we see that *`SDOI`* outperform the dictionary-based baseline algorithm both in precision and recall, but particularly in recall. This lift is due to *`SDOI`*'s sequential model's ability to identify mentions not in the nascent ontology.

From the learning curve in Figure 4 we infer that SDOI overtakes baseline performance after 30 to 50 annotated mentions, and that performance continues to improve significantly as additional training data is provided. Interestingly, the performance lift between feature spaces decreases with additional data. This

[8] http://www-tsujii.is.s.u-tokyo.ac.jp/GENIA

[9] http://biocreative.sourceforge.net/

[10] A semantic wiki is a wiki that captures semantic information in a controlled natural language that enables the generation of a formal machine-processable ontology http://www.semwiki.org/

[11] A URL to sample ontology page is withheld due to the double-blind requirements.

[12] We use the conlleval.pl evaluation script from CONLL-2000.

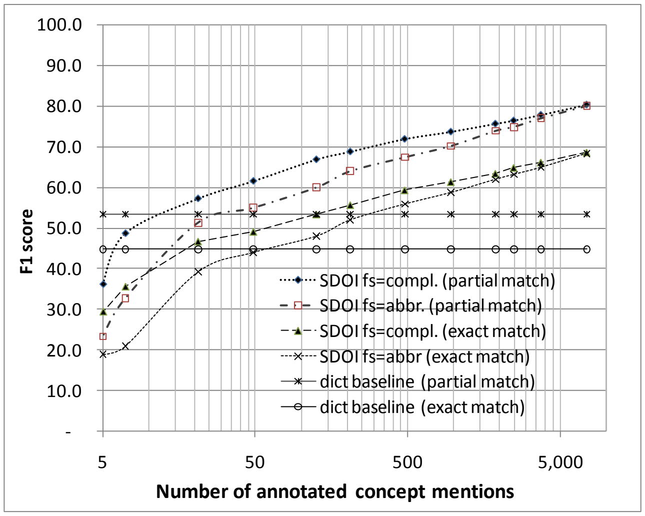
suggests that high, expert-level performance, will require additional data more than advanced feature engineering.

Finally, there is a significant effect when a partial match criteria is applied on multi-word mentions. A visual inspection of some of the cases where partial match succeeds suggests the issue is due to multi-word expressions that are divided differently by the annotator and the algorithm. When these mismatches occur the algorithm is doubly penalized for making two false predictions and for missing one true prediction. With partial matches then the algorithm receives one correct prediction and one false prediction.

Table 9 -Average concept mention identification performance (Precision, Recall, and F1) and lift ratio, of the dictionary-based baseline and `SDOI` algorithms on the `kdd09cma1`, under exact and partial definitions of mention matching.

	Exact Match			Partial Match		
	P	R	F1	P	R	F1
SDOI	70.8%	67.3%	69.0%	82.7%	78.6%	80.6%
dict	51.9%	40.2%	44.8%	61.5%	50.4%	54.1%
lift	36.4%	67.3%	54.0%	34.4%	56.0%	48.9%

Figure 4 Log-scale learning curve analysis of `SDOI`'s and the baseline's F1 performance on the kdd09cma1 dataset under exact and partial match criteria. For `SDOI` two features spaces: complete and abridged (POS and token)



7.4 Mention Linking Evaluation

7.4.1 Candidacy Definition

Before proceeding to assessing *SDOI*'s performance we first identify the subset of the eight candidacy tests defined in Section 4 by empirical means. The definition of the candidacy selection heuristic can impact performance. Too restrictive a policy will limit the maximal attainable recall performance. Too liberal a policy could swamp the classifier with a large proportion of negative-to-positive training cases. We empirically test the effect on accuracy of incrementally adding individual tests (primary and stemmed[13]) in the following order: t_1+t_{1s}+t_2+t_{2s}+t_3+t_{3s}+t_4+t_{4s}.

Table 10 summarizes the impact of sequentially adding each of the tests. As the tests become more inclusive more training cases are available and the maximum accuracy increases. Based on this empirical analysis the candidacy test selected for `SDOI` included tests t_1+t_{1s}+t_2+t_{2s}+t_3+t_{3s}. Adding more tests beyond this point drops accuracy significantly, likely because the average number of training cases per mention increases from approximately 2.5 to 47 cases per mention on average.

Table 10 – Effect of the four candidacy tests (plus their stemmed version) on linking accuracy. The selected combination of tests is highlighted.

Test	training cases	max. possible accuracy	*Accuracy*
+ t_1	536	9.0%	13.7%
+ t_{1s}	1,278	19.1%	26.4%
+ t_2	3,126	40.4%	46.1%
+ t_{2s}	5,206	53.0%	46.0%
+ t_3	9,390	74.8%	56.7%
+ t_{3s}	**11,598**	**77.3%**	**57.3%**
+ t_4	296,086	90.1%	50.3%
+ t_{4s}	386,537	91.5%	49.3%

7.4.2 Mention Linking Performance

To estimate algorithm performance we performed a leave-one-out cross-validation study. Specifically, we iterated through all 139 documents, leaving one document out of the training corpus and testing on all the mentions within the excluded document. The CRF++[14] package was used to generate the sequential tagging model used in the identification task. SVMlight[15] was used as the classification model training system used for the linking task. The number of iterations for the iterative classifier was set to five (μ=5).

Performance is reported on the separate tasks of: 1) predicted concept node versus actual concept node in the ontology on the manually annotated concept mentions, and 2) predicted anchor text and concept node vs. actual anchor text and concept. Table 11 summarizes the performance of the *SDOI* and baseline algorithms.

Table 11 – Accuracy of the `SDOI` and baseline algorithms on the linking subtask applied to the kdd09cma1 corpus when based on true or predicted anchor text spans.

Linking accuracy	Exact Match		Partial Match	
	SDOI	MW08	SDOI	MW08
On 'true' anchor text	57.30%	44.70%	63.21%	46.82%
On predicted anchor text	45.40%	17.70%	47.08%	19.14%

On true anchor texts *SDOI* performed much better at linking concept mentions to the ontology than the baseline. This is likely due to the additional features and the expanded definition of candidacy.

[13] http://search.cpan.org/perldoc?Lingua%3A%3AStem

[14] http://crfpp.sourceforge.net/

[15] http://svmlight.joachims.org/

On predicted anchor texts *SDOI* performed significantly better than the baseline because of the cumulative effects of performance on mention identification and linking. For many mentions the baseline algorithm could not make a link prediction because it had failed to identify them in the identification task.

7.4.3 Collective Features & Iterative Classification

Unexpectedly the collective features contributed negligibly to overall linking performance. As seen in Table 12, the first iteration of the algorithm (which does not yet benefit from collective features) is only marginally increased by the final iteration. This is a surprising result given the lift attributed to collective features in [4] and [9]. We explored whether *SDOI*'s marginal increase in performance is due to the significantly expanded feature space which leaves fewer ambiguities requiring deep insight into the roles of the concepts. As Table 12 shows, the collective features contribute more noticeably to the performance when only the anchor-text based features are retained.

Table 12 – Average accuracy of *SDOI* after each iteration on the full feature set, and on only anchor text-based features.

Iter.	All Features	Anchor Text + Collective
1	57.2%	47.9%
2	57.3%	49.7%
3	57.3%	49.9%
4	57.3%	50.0%
5	57.3%	50.2%

7.5 Evaluation on ICDM'09 Abstracts

To assess the portability of the models trained on `kdd09cma1` we tested the models on a different corpus that is also from the data mining domain. For this corpus we also performed two additional types of analysis: inter-annotator agreement and the time savings achieved when annotating a pre-annotated abstract.

The second corpus is composed of twenty two manually annotated abstracts from the papers accepted into IEEE's annual conference on data mining in 2009 (ICDM'09)[16]. Seven domain experts were involved in the annotation task. Each annotator was asked to select five abstracts of interest to them. To ensure that most abstracts would have more than one annotated version, the selected abstracts were ranked according to number of annotators that selected it, and the annotators were asked to annotate at least two abstracts in the ranked order. Some of the abstracts were pre-annotated by *SDOI* or by the baseline algorithm (based on [9]). We created an annotation environment based on the semantic wiki that houses the `kddo1` ontology. The annotation environment is unsophisticated but realistic since thousands of people use this annotation style and technology every day to edit pages on wikis such as Wikipedia. Annotators were asked to annotated using the following four-step procedure: 1) read the abstract on the official IEEE webpage for the paper, 2) identify and annotate concept mentions without referencing the ontology, 3) link mentions to their first best guess of the concepts preferred name in the ontology, and 4) revise their annotations based on active search of the ontology (either by hyperlink navigation or keyword search). In eleven cases, abstracts were annotated by more than one person; ranging from two abstracts by four annotators to two abstracts by two annotators. We then created a single ground truth version of each abstract after reviewing the annotations.

Two issues complicated the measurement of inter-annotator agreement and time-savings on this task. The first challenge was due to annotator's "learning curve" with the annotation process. The first abstract was annotated more slowly and resulted in lower agreement than each of the subsequent documents. In response, we sometimes report that the results exclude the annotator's first abstract. A second challenge with the timing data was the significant variance in the time required per mentions by each annotator. To accommodate for the variance, the time-savings analysis is restricted to the five annotators who annotated three or more abstracts.

7.5.1 Portability Analysis

The accuracy of *SDOI* and the baseline algorithm with respect to the ground truth was analyzed and the results reported in Table 13. Performance on the second corpus is only slightly lower than those reported in the second row of Table 11; suggesting that the trained models are portable to other corpora.

Table 13 - *SDOI* and baseline accuracy, trained on the *kdd09cma1*, and tested on the eleven ICDM'09 abstracts, where matches are exact or partial (correct start token).

Linking accuracy on predicted anchor text	Exact Match		Partial Match	
	SDOI	MW08	SDOI	MW08
ICDM-2009	42.2%	15.6%	46.5%	17.5%
KDD-2009 (from Table 11)	45.4%	17.7%	47.1%	19.1%

7.5.2 Inter-Annotator Agreement

This section analyzes the agreement between the annotators' output and the ground truth. This analysis empirically sets a maximum accuracy that we would expect an automated approach to achieve. The results presented in Table 14 indicate that, on average, 70% of each person's annotations are identical to the ground truth. This result suggests subjectivity in the annotation (though this performance would be improved on with additional training for the annotators). Still, it also shows that *SDOI*'s current accuracy of 57.3% has room for improvement.

Table 14 – Average accuracy of the annotator's abstract versus the consolidated "gold" annotation. The second row accounts for the annotator's "learning curve" by excluding the first abstract processed by each annotator.

Match type:	Exact	Partial
All abstracts	65.8%	66.9%
First abstract withheld	71.8%	72.8%

7.5.3 Time Savings Evaluation

A final performance measure of interest is the time savings in the annotation process achieved by making use of an automated system's predictions. This measure is particularly relevant to scenarios were annotation will be an ongoing process and where high-accuracy is required (such as for linking mentions in high-quality conferences, and within business environments linking policy documents to official term definitions). We evaluate

[16] http://ieeexplore.ieee.org/xpl/mostRecentIssue.jsp?reload=true&punumber=5360037

whether it is faster for an annotator to work from a pre-annotated document than from working from an unprocessed document. For this analysis we selected the five individuals who annotated three or more abstracts and excludes their first abstract. Table 15 presents the timing results.

Table 15 – Seconds required for annotators (who annotated three or more abstracts) to annotate each unique concept mention: per phase and three pre-annotation scenarios.

Phase	No Pre-Annot.	MW08	SDOI
B - Identification	8.0	9.3	3.5
C - Linking	15.6	12.8	6.9
D - revision	9.9	12.2	8.5
B, C, D	33.6	34.3	18.9

On average, annotators required significantly less time on all three phases when abstracts were pre-annotated using *SDOI*'s output. The annotators also benefited little or negatively from correcting the baseline predictions - they had to spend a significant amount of time to correct mistakes. This result suggests that *SDOI* has achieved sufficient accuracy to be of value on some tasks. To the best of our knowledge, this type of time-savings evaluation has not been performed to date on a related task.

8. CONCLUSION

In this paper we present a supervised learning based algorithm, *SDOI*, for the task of identifying and linking concept mentions to an ontology. The algorithm is validated against a novel corpus of abstracts from a data mining conference that have been annotated and linked to a data mining ontology; and further tested on an additional corpus to assess the portability of the produced models.

Our main contributions are the ability to identify mentions not yet present in the ontology, proposing a set of tests for selecting candidate concepts, and proposing a formalized and expanded feature set. We explore the use of iterative classification as a principled and purely supervised approach to handling the collective features; however, we also present evidence that their contribution is significantly reduced by the introduction of the expanded feature set, compared to [4] and [9]. Finally, we present a novel time savings analysis which suggests that *SDOI* achieves high enough accuracy to be of practical value in some tasks.

This paper suggests several directions for future research. It will be interesting to explore an integration of the identification and linking models in order to better inform the boundary decisions for mentions. In addition, alternative evaluation criteria are worth investigating that would award partial credit for predictions that are close to the correct answer in terms of overlap with the correct mention or selection of a parent or child concept. We plan to expand the corpus to include all past KDD and ICDM conference abstracts, and to expand the data mining ontology to include many of the main concepts and relationships discovered in the process. Ideally we would like to integrate *SDOI* into the submission process of future data mining conferences in order to have the authors themselves validate and correct the pre-annotated versions of their abstracts.

9. ACKNOWLEDGEMENTS

We thank Anoop Sarkar for valuable discussions of the mention identification problem, and related work from the Natural Language Processing community. We also thank the volunteers who participated in the annotation of ICDM'09 abstracts.

10. REFERENCES

[1] Satanjeev Banerjee, and Ted Pedersen. (2002). An Adapted Lesk Algorithm for Word Sense Disambiguation Using WordNet. In: Proceedings of CICLing (2002). Lecture Notes In Computer Science; Vol. 2276.

[2] Rudi L. Cilibrasi, and Paul M. Vitanyi. (2007). The Google Similarity Distance. In: IEEE Transactions on Knowledge and Data Engineering 19(3). [doi>10.1109/TKDE.2007.48]

[3] Eugene Charniak. (2000). A Maximum-Entropy-Inspired Parser. In: Proc. of NAACL Conference (NAACL 2000).

[4] Sayali Kulkarni, Amit Singh, Ganesh Ramakrishnan, and Soumen Chakrabarti. (2009). Collective Annotation of Wikipedia Entities in Web Text. In: Proc. of ACM SIGKDD Conference (KDD 2009). [doi>10.1145/1557019.1557073]

[5] Andrew McCallum, and Wei Li. (2003). Early Results for Named Entity Recognition with Conditional Random Fields, Feature Induction and Web-Enhanced Lexicons. In: Proc. of Conference on Natural Language Learning (CoNLL 2003).

[6] Withheld during double-blind review

[7] Withheld during double-blind review

[8] Rada Mihalcea, and Andras Csomai. (2007). Wikify!: Linking documents to encyclopedic knowledge. In: Proceedings of the Sixteenth ACM Conference on Information and Knowledge Management (CIKM 2007).

[9] David N. Milne, and Ian H. Witten. (2008). Learning to Link with Wikipedia. In: Proceeding of the 17th ACM Conference on Information and Knowledge Management, (CIKM 2008). [doi>10.1145/1458082.1458150]

[10] Roberto Navigli, Paola Velardi, and Aldo Gangemi. (2003). Ontology Learning and Its Application to Automated Terminology Translation. In: IEEE Int. Systems, 18(1).

[11] Jennifer Neville, and David Jensen. (2000). Iterative Classification in Relational Data. In: Proceedings of the Workshop on Statistical Relational Learning.

[12] Francesco Sclano, and Paola Velardi. (2007). TermExtractor: A web application to learn the common terminology of interest groups and research communities. In: Proc. of the 9th Conference on Terminology and AI (TIA 2007).

[13] Fei Sha, and Fernando Pereira. (2003). Shallow Parsing with Conditional Random Fields. In: Proceedings of Conference of the North American Chapter of the Association for Computational Linguistics on Human Language Technology (HLT-NAACL 2003). [doi>10.3115/1073445.1073473]

[14] Pierre Zweigenbaum, Dina Demner-Fushman, Hong Yu, and Kevin B. Cohen. (2007). Frontiers of Biomedical Text Mining: current progress. In: Briefings in Bioinformatics 2007, 8(5). Oxford Univ Press. [doi>10.1093/bib/bbm045]