SUPERVISED ONTOLOGY TO DOCUMENT INTERLINKING

by

Gabor Settimio Melli
M.Sc., SFU, 1998
B.Sc., UBC, 1989

A THESIS SUBMITTED IN PARTIAL FULFILLMENT OF
THE REQUIREMENTS FOR THE DEGREE OF
DOCTOR OF PHILOSOPHY

In the
School of Computing Science

SIMON FRASER UNIVERSITY

Fall 2010

APPROVAL

Name: **Gabor Melli**

Degree: **Doctor of Philosophy**

Title of Thesis: **Supervised Ontology to Document Interlinking**

Examining Committee:

Chair:

_____

Dr. Fred Popowich
Professor

_____

Dr. Martin Ester
Professor
Senior Supervisor

_____

Dr. Anoop Sarkar
Associate Professor
Supervisor

_____

Dr. Jian Pei
Associate Professor
Internal Examiner

_____

Dr. Daxin Jiang
Researcher, Microsoft
External Examiner

Date Defended/Approved: _____

ABSTRACT

The value from the growing availability of online documents and ontologies will increase significantly once these two resources become deeply interlinked at the semantic level. We focus our investigation on the automated identification and the linking of concepts and relations mentioned in a document that are (or should be) in a domain-specific ontology. Such semantic information can allow for improved navigation of the information space: users can more quickly retrieve documents that mention the relations sought; Ontology engineers can enhance concepts with relations extracted from the literature; and more advanced natural language-based applications such as text summarization, textual entailment, and machine reading become ever more possible.

In this thesis, we present the task of supervised semantic interlinking of documents to an ontology. We also propose a supervised algorithm that identifies and links concept mentions that are (or should be) in the ontology, and also identify mentions of binary relations that are (or should be) in the ontology. The resulting system, SDOI, is tested on a novel corpus and ontology from the data mining field on intrinsic measures such as accuracy, and extrinsic measures such time saved by the annotator in the annotation process.

One day many high-value documents and ontologies will be interlinked to each other. This thesis presents a principled step towards that outcome.

Keywords: Concept Mention; Relation Mention; Reference Resolution; Ontology; Supervised Classification

ACKNOWLEDGEMENTS

This thesis - my hand in it - has many to thank. My grade six school-principal, for example: the one who picked me out to “play” with his freshly self-assembled Tandy TRS-80; I thank him and the many other dedicated teachers who guided me to the time that I finally decided to attempt a Ph.D. that would help me to participate in the future creation of an artificial intelligence.

That decision precipitated when I stumbled onto Martin Ester at a computing science tradeshow where he immediately encouraged me to apply at SFU. Through the years, his positive attitude and living example of excellence animated me to “dig deeper”. Thank you, Martin.

As hoped, my first course was on the topic of statistical natural language processing. It was there that Anoop Sarkar challenged me to imagine new ways to extending and repurposing computing machinery. He was the one who pushed me and our team of the document understanding competition (DUC) to explore and write about about the impact of “semantic role labeling” performance on text summarization – which led to my most cited paper (Melli & al, 2005). Thank you, Anoop.

Other faculty members that helped me through these years include Fred Popowich, Jian Pei, Fiona Brinkman, and Oliver Schulte. My thanks also goes to Val Galat and the other staff members who shepherded me through several bureaucratic hurdles.

It was also during that first class in NLP where I met my academic companions: Irene Pekerskaya, Zhongmin Shi, Yudong Liu, and Baohua Gu. Soon after that, the crew grew to include Flavia Moser, Nancy Yu, and Yang Wang. Thank you for your help and friendship.

Finally, I thank friends and family who encouraged me through these recent years. Thank you especially, Marco Romero and Karen Tripson for your many a caring prod.

But in the end, it was my mother, Heidi Schmidt, who can take the most credit for this thesis. She is the one who dedicated herself to ensure that I receive a Canadian education rather than a Venezuelan one; the one who stayed with me during my erratic teenage years; and the one who pushed me to continue with higher education. Thank you, mom!

DEDICATION

To my mother, Heidi Schmidt.

For your love, encouragement, and example of perseverance.

TABLE OF CONTENTS

LIST OF FIGURES

LIST OF TABLES

1: INTRODUCTION

"*We need to have a way for computers to read books on chemistry and learn chemistry. ... Or biology. Or whatever. ... reading from text in structured domains I don't think is as hard. It is a critical problem that needs to be solved.*" –Edward Feigenbaum (Shustek, 2010)

1.1 Motivation

As more documents and ontologies become readily available in electronic format their combined value will increase significantly once they become deeply interlinked so that the mentions within a text are grounded to formalized concept descriptions, and the concepts within an ontology are grounded in the dominant form of knowledge exchange (Staab & Studer, 2009; Renear & Palmer, 2009; Buitelaar & al, 2009). Imagine, for example, the day when research papers, legal contracts, how-to manuals, or even the entirety of this thesis[1] had their concept and the relation mentions linked to their corresponding item within an ontology from their respective domain. With such interlinking in place, several new applications become possible.

Strategic reading, for example, will be further enabled when a person can 'click' or 'hover over' on an important but unfamiliar term in the text (for example "ontology" or "strategic reading"); be taken to the formal concept description (for

[1] Some of the concept mentions within the abstract and body of this thesis are indeed hyperlinked to an ontological reference. Refer to http://www.gabormelli.com/RKB/2010_PhDThesis if you are not reading an electronic version of this document.

example, the webpages wikipedia.org/wiki/Ontology_(information_science) or gabormelli.com/RKB/Ontology and gabormelli.com/RKB/Strategic_Reading); and, be able navigate to a seminal document that used the concept (such as to Gruber, 1993 or Marchionini, 1997). Currently however such information seeking activity would require several Web sessions using several different search services ranging from google.com to scholar.google.com, and likely the use of some specialized digital libraries, such as portal.acm.org, or knowledge bases such as wikipedia.org (Renear & Palmer, 2009).

Query-based information retrieval will also be better serviced when concepts with many lexical variations (such as "*Gaussian distribution*" or "*normal density function*", or "*supervised approaches to concept mention linking"*) are identified and linked both in the text and in the queries. Currently queries on lexically varied concepts force users to iteratively fine-tune their keyword searches to account for these variations, while also ensuring that unintended matches are excluded (Tran & al, 2007; Lei & al; 2006). Once concept mentions are identified and linked to a concept in an ontology then the performance of identifying relevant documents will improve both in terms of precision (only documents that contain the intended concept) and of recall (all documents that contain the intended concept) despite of the many alternate ways to express the concept in text.

Another area would benefit from document to ontology interlinking is the use and administration of ontologies themselves. An ontology's knowledge engineer could more quickly understand the meaning and scope of two slightly

differing concepts by seeing them “in action” within relevant natural language passages (Buitelaar & al, 2009; Schaffert, 2006). Ontologies are generally also required to evolve with the domain they represent as time passes. Ontology engineers would benefit from a listing of concept or relation mentions that appear to be relevant to the domain (because several documents mention them) but are absent in the ontology (Buitelaar & al, 2009).

Finally, other applications that would benefit from the existence of ontology-linked documents include: information retrieval (Manning & al; 2008), information extraction (Sarawagi, 2008), topic modelling (Blei & Lafferty, 2007; Griffiths & Steyvers, 2004), textual entailment (Burchardt & al, 2007; Dagan & al, 2006), question answering (Strzalkowski & Harabagiu, 2006), text summarization (Melli & al, 2006), and eventually machine reading (Etzioni & al, 2006).

An obstacle to this vision of semantically grounded documents however is the significant amount of effort required from domain experts both to semantically annotate the text within the documents, and to evolve an ontology in order to keep it aligned with its domain (Uren & al, 2006; Erdmann & al, 2000). The obstacle has been noted in the biomedical domain (Zweigenbaum & al, 2007), the business reengineering domain (Melli & Quinn, 2008), and the legal domain (Wyner, 2010).

Some automation of the annotation task is a precondition to the envisioned future of deeply interlinked information. Fortunately, some recent research suggests the feasibility of such automation by the application of data-

driven algorithms (Cucerzan, 2007; Mihalcea & Csomai, 2007; Milne & Witten, 2008; Kulkarni & al, 2009).

The research efforts to-date however have been focused on a domain-independent interpretation of the task that uses Wikipedia[2] or WordNet[3] as the knowledge base. Certainly, the use of a knowledge base with a broad but shallow coverage of concepts is a good starting point to show the feasibility of automation; but there is also a need to support scenarios where a domain-specific ontology is provided. Also, the empirical results reported in the literature are mostly for either Wikipedia-based or newswire-like articles where the annotation is focused on named entities (such as "*Michael I. Jordan*", "*iPhone 4G*" and "*E.coli O157:H7*") and simple categories (such as "*person*", "*product*", or "*organism*"), while excluding nuanced concepts (such as "*self-trained information extraction algorithm*" or "*contractual consideration*"). Finally, current proposals focus on the recognition of some specific relation, either domain independent ones, such as IS-A, or domain-specific ones such as SUBCELLULAR-LOCALIZATION (Banko & al, 2007; Buitellar & Cimano; 2008).

An organization or company that decides to link their high-value documents to their domain-specific ontology cannot make direct use of current research to reduce the workload (and cost) of their annotation team as their collection of annotated documents grows. We propose the application of supervised learning as a solution for this scenario.

[2] http://www.wikipedia.org
[3] http://wordnet.princeton.edu/

1.2 Topic and Approach

This thesis examines the composite task of ontology-based concept mention identification and linking, and relation mention identification. The tasks and approaches most closely related to our are those proposed in (Milne & Witten, 2008; Kulkarni & al, 2009). Our topic however is for linking domain-specific ontologies, rather than Wikipedia. Further, we extend our focus beyond concept mention linking to also include concept mention and relation mention identification.

We decompose the task into three separate subtasks: 1) the identification of mentions in a document of concepts relevant to the specific domain; 2) the linking (disambiguation) of each of these mentions to the appropriate concept in an ontology, if such a concept exists; and 3) the identification of mentions in a document of relations between the mentioned concepts. This decomposition can support scenarios were one person first identifies the text segments that appear to refer to specialized meaning, some other person with likely greater expertise links the mention to the correct concept in the ontology (if the concept exists), and a third person with further expertise identifies relevant relations between the concepts mentioned in the document that exists in the ontology. Our proposed supervised solution, named SDOI, follows this three-fold decomposition: SDOI_{CMI}, SDOI_{CML}, and SDOI_{RMI}.

SDOI_{CMI} first trains a supervised sequential classifier to identify token subsequences in a document as concept mentions. Motivations for the application of a sequential tagger include their successful use in the NLP

community to the related tasks of text-chunking (Sha & Pereira, 2003) and named entity recognition (McCallum & Li, 2003), and the possibility that any future improvements in the use and training of sequential taggers in other domains can be naturally imported into our framework. A further motivation of this sequential tagging approach is that it identifies lexically varied concept mention even when the token sequence is not present in the training corpus, nor recorded as a possible alternate spelling within the ontology.

Next, due to the large number of concepts in the ontology, we propose that the $\texttt{SDOI}_{\texttt{CMI}}$ module apply a binary supervised classifier to the concept-mention linking task rather than to directly train a multi-class classifier (Rifkin & Klatau, 2004). To accomplish this transformation, each mention is associated with a subset of candidate concepts by means of heuristic candidacy tests that can be used to remove cases that are very unlikely to be true (i.e. to undersample). Next, each candidate concept is associated with a rich feature vector, including recursively defined (collective) features that account for global context, and then labelled as true or false based on whether the concept is indeed the one that the mention must link to. In order to support the collective features we propose the use of an iterative supervised classifier (Neville & Jensen, 2000).

The final module of the pipeline, $\texttt{SDOI}_{\texttt{RMI}}$, is another binary classifier for solving the relation mention identification task: $\texttt{SDOI}_{\texttt{RMI}}$. For each permutation of two concept mentions, we build a feature vector and heuristically associate a label based on whether the relation is present in the ontology. A difference for

this subtask is that we do not require that a person manually label each of the multitude of concept mention combinations within each document. Instead, we propose the use of a self-supervised approach that makes use of a labelling heuristic (Banko & Etzioni, 2008). The proposed labelling heuristic is to assign a label if the candidate mention refers to a link that exists or does not exist in the ontology.

To estimate `SDOI's` empirical performance we created an annotated corpus composed of the abstracts from papers accepted to the KDD-2009 and ICDM-2009 data mining conferences. Each abstract is annotated such that its concept mentions are identified and, where possible, linked to the concepts in a data mining ontology, `kddo1`, which we have also created in support of the thesis. Our evaluation suggests that `SDOI` is able to save time in the annotation process.

1.3 Contributions

The main contributions of this thesis[4] are that:

1. We present a formal definition of the task of ontology-based concept mention identification and linking, and relation mention identification. This definition is general in that it subsumes a variety of related tasks such the linking to Wikipedia pages, or the task of named entity recognition and disambiguation where the ontology contains only

[4] Some of the contributions are previously presented in (Melli & al, 2007; Melli & McQuinn, 2008; Melli, 2010; Melli & Ester, 2010).

named entities. Also, the task definition can be naturally extended to include additional future requirements, such as the classification of the type of the identified relation mentions.

2. We propose a principled process composed of three pipelined learning algorithms ($\texttt{SDOI}_{\texttt{CMI}}$, $\texttt{SDOI}_{\texttt{CML}}$, and $\texttt{SDOI}_{\texttt{RMI}}$) that are feasible to re-implement and extend by others in the community, and so become the de facto baseline. The system's performance is intended to improve as the annotated corpus grows, and to not require expertise in natural language processing or statistics to, for example, develop lexico-syntactic detection patterns or to fine tune algorithm parameters[5].

3. Our proposed algorithms include extensive feature sets that subsume and extend the features of related proposals. The feature sets can include recursively defined collective features, and for these we propose an iterative classification algorithm that does not require (Milne & Witten, 2008)'s heuristic of which mentions to assume as correctly linked, nor requires a customized "objective" relatedness function as proposed in (Kulkarni & al, 2009).

4. We present a novel and publicly available dataset for the task that may become a benchmark used by the community. The dataset is composed of the `kddo1` ontology for the domain of data mining, and of two corpuses (`kdd09cma1`, `icdm09cma1`) derived from the 139

[5] Consequently, the algorithm is not tied to a particular domain or natural language, though it does require a natural language parser for the language of the corpus.

abstracts for the KDD 2009 and 22 of the abstracts of the ICDM 2009 data mining conference proceedings. We are not aware of another similar dataset for any field within computing science, and believe that the dataset may continue to grow and evolve with each future edition of the KDD and ICDM annual conference proceedings.

5. Finally, we evaluate the $\texttt{SDOI}_{\texttt{CML}}$ and $\texttt{SDOI}_{\texttt{RMI}}$ algorithms not only on standard intrinsic measures of accuracy and F1, but also on task-oriented extrinsic measures. For $\texttt{SDOI}_{\texttt{CML}}$ we report time saved in the annotation process and for $\texttt{SDOI}_{\texttt{CML}}$ the ability to suggest new direct relations. In this manner, we demonstrate that the algorithm is ready to participate in some real-world settings.

1.4 Thesis Outline

The remainder of the thesis is structured as follows: Sections 2 formally defines the task. Section 3 presents related work that will be used to ground, compare and contrast the proposed algorithms. Section 4 presents the ontology and corpus that we have created to the evaluated the task. Sections 5 through 7 present the proposed algorithm. Specifically, section 5 presents the solution to the concept mention identification task: $\texttt{SDOI}_{\texttt{CMI}}$, and its evaluation; section 6 presents the solution to the concept mention linking task: $\texttt{SDOI}_{\texttt{CML}}$, and its evaluation; and, section 7 presents the solution to the relation identification task: $\texttt{SDOI}_{\texttt{RMI}}$, and its evaluation. Finally, section 8 concludes the thesis with a discussion of recognized limitations and future research directions.

2: TASK DEFINITION

In this chapter, we define the task of supervised concept and relation mention[6] recognition with respect to an ontology in terms of its input requirements, output requirements, and performance measures.

2.1 Input Requirements

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 text tokens (orthographic words or punctuation).

Assume also the existence of an ontology of interrelated *concepts*, $o_c \in O$, that represent and describe some concepts 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, some descriptive text t_c. As described, an ontology is a directed and labelled multigraph that could be used to represent such diverse structures as Wikipedia[7] (with its rich text and

[6] We generally affix the word "*mention*" to a term to distinguish the concept of a referring expression in a text and its referent; be it a mental state in the reader or a thing in the real-word (Cruse, 1986): For example: *entity* ⇔ *entity mention*, *named entity*⇔*named entity mention*, *concept*⇔*concept mention*, and *relation*⇔*relation mention*.

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

weak semantics) and the Gene Ontology[8] (with its rich semantics and terse descriptions).

Assume next that each document d_i has a set of non-overlapping non-partitioning subsequences of tokens referred to as *concept mentions*[9], $m_m \in d_i$, that refers to a domain specific meaning not generally found in a dictionary. We assume that there is a significant overlap between the concepts intended for the ontology and the concepts mentioned in the corpus.

Every concept mention m_i is connected via a directed edge to either the concept o_j that captures the concept 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 $\varphi(m_m, o_c)$. An *unlinked* concept mention, $\varphi(m_m, ?)$,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.

Next, a *candidate relation mention,* r_i, is any pairing of two concept mentions within the document $<m_{i'}, m_{i''}>$ that are linked to different concepts[10]. A relation mention is labelled as *true* if the mention signifies an internal link in the ontology, and *false* otherwise.

[8] http://www.geneontology.org

[9] The term "concept mention" is previously used in (Beigman, Klebanov & Shamir, 2005).

[10] The type of the underlying relation does not play a role in our task definition of relation identification. No distinction is made between domain independent semantic relations such as: subsumption (IS-A-TYPE-OF), meronymy (IS-A-PART-OF), antonymy (IS-AN-OPPOSITE-TO), or sibling (SHARE-A-PARENT) relations; nor, between domain-specific relations such as ORGANIZATION-HEADQUARTERCITY, PROTEIN-LOCALIZATION-SUBCELLULARCOMPARTMENT, or IS-AN-ALGORITHM-FOR.

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.

Figure 1 – 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 token sequences in the documents are *concept mentions* mapped to either concept nodes or the unknown concept symbol (?), via *external links*. The concept nodes in the ontology have side bars to suggest that the data structures can record inlinks to the concept and outlinks from the concept.

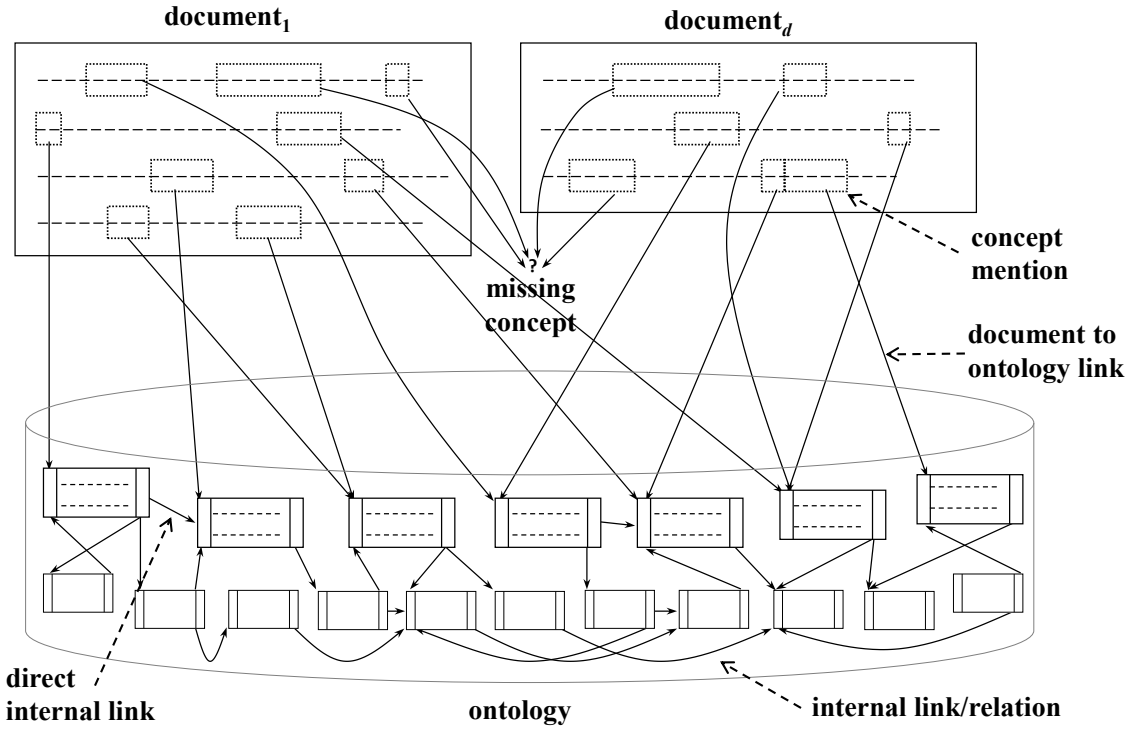


Figure 2 – The example of concept mention annotation using wiki-style formatting. Mentions are identified with doubled square brackets. The internal vertical bar (|) separates the *anchor text* from the concept reference. A question mark (?) refers to an unlinked concept.

```
 [[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]].
```

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 $m_k \in D$.

2.2 Output Requirements

Given a previously unseen document from the same domain as the ontology that lacks concept and relation mention annotations, the required outputs of the task under investigation are:

1) The complete set of concept mentions within the document (both their *anchor text* and their corresponding *external link*)

2) The set of relation mentions in the document that are in the ontology.

2.3 Evaluation

Several relevant evaluation criteria are available to measure performance. Concept mention identification can be naturally assessed in terms of F1-measure (and its precision and recall components). An F1-measure of 1.0 for example will occur when all concept mentions within a document are identified and no non-existing (false) mentions are predicted.

Concept mention linking is a multiclass classification problem that lends itself to measure of accuracy. An accuracy measure of 1.0 will occur when all true mentions within a document are linked to the correct concept node (one of which is the unlined concept symbol). However, at this phase performance must also be evaluated against the concept mentions predicted by the concept mention identification algorithm (not the manually annotated ones). This alternate evaluation will better correlate with the experience of an annotator who waits to this stage to begin to review (and fix) mistakes made by the two-step algorithm.

Finally, while relation mention identification is a binary classification task that lends itself to an F1-based assessment. An obstacle to such an assessment is that our task does not require manually annotated relation mention data. Thus the performance analysis of this task will be based on its ability to aid in some extrinsic task for some given dataset.

2.3.1 Partial Credit

Another evaluation metrics that is investigated is to grant partial credit for incorrect predictions that are relevant predictions. A motivating intuition for this

extension is that the task can have a subjective aspect to it where even two people may not agree on exact assignments because ambiguities in language.

For the concept mention identification task, partial credit can be assigned if there is an overlap between the true mentions and the predicted mention. For the mention linking task partial credit can be assigned if the concept node selected only has a one edge distance to the correct node. For the relation mention identification task partial credit can be assigned if the relation is between concept mentions that only received partial credit.

2.3.2 Concept Mention Annotation - Time Savings

Some settings will require that all annotations be reviewed by an expensive person with domain expertise. In these settings a relevant performance measure is whether it would take less time for the annotator to fix the mistakes made by the algorithm or to not use the algorithm whatsoever. The measure to compare algorithm performance is the proportion of time spent per concept mention under the two competing approaches. A proportion that is greater than 1.0 would indicate that some time was saved and would also suggest that the technology is ready for broad use. This human factors assessment assumes that the mechanisms implemented for annotating and for fixing annotations will not change significantly after more careful design of the data curation interfaces.

3: RELATED WORK

"*... nos* (we) *esse quasi nanos* (dwarfs)*, gigantium* (giants) *humeris* (shoulders) *insidentes* (on)*, ut possimus plura eis et remotiora* (far) *videre* (see)"
--John of Salisbury quoting Bernard of Chartres (1159 AD)

In this chapter, we summarize some the published research that has informed the solution that we propose in subsequent chapters to the task of supervised document-to-ontology interlinking. Subsequent chapters will also, on occasion, present more localized references to the literature; while in this chapter, we present the key ideas that we have extracted from past published research. We assume that the reader is familiar with state-of-the-art approaches to fully-supervised learning in vector-based feature spaces with noisy input data (Cortes & Vapnik, 1995) and the general availability of implementations such as `SVMlight` (Joachims, 1999).

3.1 Overview

Many diverse research areas have investigated topics that bear some similarity to that of document-to-ontology interlinking. These include: natural language processing research; research into applied tasks, such as information extraction; and research into general learning algorithm considerations, such as ways to address input data with a highly skewed label distribution. Generally, we focused the review on past research that was document centric (rather than

database centric) and assumed and accounted for the availability of background knowledge (ideally, of an ontology).

The field of natural language processing has a natural influence on our design because it addresses ways to convert natural language utterances into a more structured representation with only partial regard of the extrinsic benefit to some real-world application. While much of their work is on identifying syntactic structure, such as part-of-speech tags and parse trees, a large amount of work has also gone into identifying preliminary semantic structure from the words used in sentences. The two topics within lexical semantic processing that we review in more detail are word sense disambiguation and named entity recognition.

Several application areas are of interest to us because they have investigated techniques to (preprocess documents to) identify semantic structure. The main application-driven area that we review is that of corpus-based information extraction (Sunita, 2008) which requires that a database be populated with information drawn from text. We review some work that use or learn lexico-syntactic patterns, and also work into the populating of biomedical entity databases from biomedical literature. Another particularly relevant application-driven research that has received some recent attention is automated semantic annotation (Reeve & Han, 2005), particularly the addition of links from word mentions to the Wikipedia article that most closely addresses the meaning of the word. We closely review the proposals by (Milne & Witten, 2008) and (Kulkarni & al, 2009).

Some of the application-driven research areas that we investigated but do not review here include: information extraction techniques whose focus is to populate a database from scratch without the use of background knowledge (Etzioni & al, 2008; Hassell & al, 2006); semantic information retrieval (Manning & al, 2008; Shah & al, 2002), preprocessing for question answering (Strzalkowski & Harabagiu, 2006; Melli & al, 2006); database record deduplication which links a database record to some canonical one (Bhattacharya & Getoor, 2004; Bilenko & al, 2005); ontology learning from text (Navigli & Velardi, 2006); ontology matching which requires the linking of equivalent concepts in two ontologies (Euzenat & Shvaiko, 2007); the automatic identification of the keywords commonly required when submitting a paper for publication (Turney, 2000); and the generation of a document's subject index (Sclano & Velardi, 2007; Jacquemin & Bourigault, 2003).

Finally, we review three research areas within machine learning that apply to us. The first is graph edge prediction of whether two labeled nodes in a graph are linked. The second is the handling of classification tasks with many labels. And, thirdly, techniques to handle significantly skewed distributions of classification labels.

3.2 Word Sense Disambiguation

Within lexical semantic processing, one of the more closely tasks to that of concept mention identification and linking is that of *word sense disambiguation* (WSD) which requires that words in some given text be linked to the appropriate word sense in the provided dictionary (Stevenson & Wilks, 2003). Canonical

examples of challenging words that the task must cope with are "*bank*" and "*pike*" because of their multitude of wide ranging senses.

A dominant data-driven algorithm for the task is the unsupervised Lesk algorithm (Lesk, 1986; Banerjee & Pedersen, 2002). The algorithms uses a similarity measure that compares the overlap in the words in the definition of the word sense in the dictionary (its gloss) and the words before/after the target word in the text (its text window of predefined size), and selects the word sense with the highest score.

The task of WSD differs from ours in important ways that limit the portability of proposed solutions to our task. WSD solutions, for example, can assume that the provided lexical database (dictionary) will contain most word senses in the language. This assumption is largely valid because several electronic dictionaries, such as WordNet (Miller & al, 1993), exist with very broad lexical coverage. The Lesk algorithm will return a word sense for every word mention. For our task of concept mention linking on the other hand, one can assume that a large proportion of words will not be present in the ontology. Further, WSD's subtask of identifying which token sequences to deem as "words" is trivial, particularly in English, because dictionary words are typically composed of only one orthographic word or can be identified with a longest-matching-token-subsequence heuristic. In our task the mentions can often be multi-word expressions that do not necessarily abide by the matching heuristic and (again) our ontology-based sense inventory is possibly incomplete.

3.3 Named Entity Recognition

Another relevant task within lexical semantics is that of *named entity recognition*, NER, (Nadeau & Sekine, 2007) which requires the identification of proper nouns that refer to some set of basic categories such as person, protein, organization, and/or location. A seminal set of categories of was defined for the MUC-6 task (Grishman & Sundheim, 1996) and the CoNNL-2002 competition (Tjong Kim Sang, 2002) but has since been expanded a more granular categories (Jijkoun, 2008; Bunescu & Pasca, 2006). Examples of challenging names include words such as the location of “Java” (an island), the company “Amazon”, and the organization “the Ronald MacDonald charity”. A successful approach to corpus-based named entity recognition is the use of a trained sequential model based on conditional random fields, CRFs, (Lafferty & al, 2001). (McCallum & Li, 2003) for example successfully apply a linear chain CRF.

The task of named entity recognition resembles the task of concept mention identification in that named entity mentions are often multi-token sequences. As we will see in Section 5, we also propose the use of a CRF-based sequential model. A notable difference in that named entities have more natural surface features to draw on, such as the presence of capitalized first letters and non-letters, than exist for concept mentions. Further, the task of NER does not typically assume the presence of background knowledge – although sometimes it assumes the presence of simple listings of synonyms, gazetteers. Our proposal takes account of these differences by including features based on information drawn from the ontology.

The similarity between named entity recognition and the concept mention linking subtask is weaker that for identification because concept mentions, in general, are labeled from a very large set (of ontology concepts) not just a handful of categories. Further, concept mention linking has the special and possibly dominant label "?" to indicate an unknown category. Thus our proposed concept mention linking solution does not draw from this research area.

3.4 Information Extraction with Lexico-Syntactic Patterns

A similar task to ours that has received significant analysis is that of corpus-based information extraction which requires the population of a data structure based on the information available in a corpus (Sunita, 2008). Initially the research focused in the filling-in of predetermined "templates" such as a template for ORGANIZATION [name, headquarter location, ...], or PERSON [name, birthdate, ...] (Miller & al, 1998; Cardie, 1997). With the availability of the web as a corpus, less supervised solutions have been proposed that also extract some of the structure directly from the text (Banko & al, 2007).

As a pre-processing step, information extraction system must identify token sequences in the text that represent entities and relations between the entities. One of the more successful approaches to this pre-processing is the use of extraction patterns based on the tokens near the candidate (tokens) under consideration, or that can include some syntactic information such as the part-of-speech role: lexico-syntactic patterns. The use of such surface patterns for information extraction was suggested initially by (Hearst, 1992). A sample Hearst pattern is: "<Class> such as <Member$_1$>, …, <Member$_i$>". For the phrase

“Bacteria such as E.coli, P.aeruginosa and B.subtilis are commonly tested for…” the pattern can detect the semantic relations IsA(bacteria, E.coli), IsA(bacteria, P.aeruginosa), and IsA(bacteria, B.subtilis). Other patterns proposed by (Pantel & al, 2004) include:

- <Class> like <Member1> and <Member2>
- <Member> is a <Class>
- <Class>, including <Member>
- <Class>, especially <Member>
- <Member> (<Class>)

(Pantel & al, 2004) and (Etzioni & al, 2005) use surface patterns to identify homonyms and extract subsumption relations from very large corpora such as the Web. (Builtelaar & al, 2008) and (Magnini & al, 2006) use surface patterns to populate an ontology.

A challenge to the use of manually designed extraction patterns is the ad hoc nature of constructing them for new types of concept and relation mentions. A solution to this challenge is the use of learning algorithms to produce the patterns (Pantel & al, 2009; Suchanek & al, 2006; Agichtein & Gravano, 2000). These proposals generally use bootstrapped weak-supervision based on a small number of training examples of the sought entities and relations. With these examples they look for likely mentions of them in the unannotated corpus, and then induce patterns to match these text segments. The patterns are tested against the corpus to assess their precision and the patterns that introduce too

many errors are discarded. An iterative (bootstrap) process is begun with the retained patterns by using them to extract new items from the corpus, assume that the extracted items are all correct, and then re-execute the process.

A challenge for the direct use of these lexico-syntactic extraction patterns to our task is their conservativeness. These patterns exploit the redundancy in very large corpora where facts will eventually be stated (and re-stated) in just the way that the pattern is designed to catch. Our task however requires that every concept and every relation mention in a document be identified, but natural language allows for a multitude of irregular ways to express meaning.

3.5 Named Entity Normalization in Biomedicine

One of the early fields to actively investigate the application of information extraction techniques is that of Biomedicine (Zweigenbaum & al, 2007). The main reason for their interest is the significant value that bioinformaticians extract from databases on biomedical entities, such as proteins, and the fact that much of this information is 'locked-up' in published papers. The focus of the field however, as in the BioCreAtIvE II benchmark task, remains on identifying named entity mentions, such as of molecules, genes, cellular locations, and organisms, and to then link ("normalize") the mentions to entity-based databases, such as Swiss-Prot, or the entities contained in Gene Ontology (Morgan & al, 2008). State-of-the-art solutions to this task all make use large lists of synonyms for the sought entities (Fundel & al, 2007; Crim & al, 2005). Their challenge is to find ways to cope with the multitude of possible spellings and abbreviations.

3.6 Ontologized Information Extraction

A related direction to normalization in biomedical text mining research is the “ontologization” of extracted entities proposed in (Pantel, 2005) and the ontologization of extracted relations proposed (Pantel & Pennacchiotti, 2008; Pennacchiotti & Pantel, 2006). Given an information extraction system’s output of extracted terms and relations between the terms, the task is to link each of the terms to an ontology concept or relation. For example, given the output of PARTOF("*Holland*", "*Europe*") and PARTOF("*Netherlands*", "*Europe*") the proposed solutions must recognize that "*Netherlands*" and "*Holland*" refer to the same concept in the given ontology – their experiments use WordNet 2.0 as the ontology. Two unsupervised algorithms that use of a similarity function are proposed and evaluated on several relation types (PARTOF, SUCCESSION, and CHEMICALREACTION).

An initial challenge to the application of the proposed algorithms to our concept-mention linking task is the assumption that extracted facts will have been extracted several times in the corpus. Given this assumption, the algorithms can exploit Harris’ distributional hypothesis (Harris, 1954) that words are semantically similar to the extent to which they share syntactic contexts. Given several instances of PARTOF(Country, Continent) facilitates the detection of a syntactic similarity. In our task, a new document may result in fact types that have not been encountered before. A second challenge arises from the use of a similarity function and our likely scenario that the ontology is incomplete. For example, WordNet includes two word senses for “London”: the capital of England

and the author Jack London. It does not have the sense of London as a city in Ontario, Canada. The similarity function is forced to select one of the two available senses (likely the city sense). As we will show in Chapter 6:, our proposal is to train a classifier that can determine, in a sense, a situation specific threshold for when none of the candidate concepts is similar enough.

3.7 Linking to Wikipedia

The most similar applied task to ours that we have encountered is the recent thrust to identify and to link concept mentions that are described in Wikipedia (Cucerzan, 2007; Mihalcea & Csomai, 2007; Milne & Witten, 2008; Kulkarni & al, 2009). These proposals are document-centric rather than extraction-centric. The initial focus of the investigations was on reproducing the link mentions of named entities in a Wikipedia page; but the research has evolved to the identification of more general concepts and on text other than Wikipedia-style pages – typically news articles. While the proposed solutions are tailored to take advantage of Wikipedia's internal structures, such as category pages, disambiguation pages and list pages, they can naturally extended to work against a more general defined ontology, and thus be applicable to our task. In the following two sections, we review the proposals by (Milne & Witten, 2008) and (Kulkarni & al, 2009) in more detail. Both proposals apply some degree of supervised learning to the task, and both investigate features based on the context of the document. Neither solution addresses the task of relation identification.

3.7.1 Supervised Balancing of Commonness and Context

Given a mention that matches more than one Wikipedia page (Milne & Witten, 2008) propose the training of a supervised classifier that is based on three features. The first feature is based on how many Wikipedia pages link to the Wikipedia page under consideration (its commonness). The more popular the page, the more likely that it will be the correct page to link to. The two other features are based on the context of the document. First it assumes that some of the mentions in the document will require no disambiguation. Based on these mentions they propose the use of a semantic relatedness measure that averages the commonness over all unambiguous mentions[11]. When a mention has more than one prediction they apply the heuristic of selecting the one with the highest likelihood score.

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. Also, they use a very limited feature set because they conceived of the learning step mainly as a data-driven means of setting a threshold on their three variables.

[11] Because we incorporate (Milne & Witten, 2008)'s context-based features we withhold a detailed description of the features until Section 6.4.7

3.7.2 Document Features and Collective Features

(Kulkarni & al, 2009) extends the link selection work of (Milne & Witten, 2008) in two main ways. They add additional features based on the similarity between the bag-of-words representation of the text window surrounding the concept mention and the concept's description in the ontology. They also propose a more sophisticated scheme to handle the relatedness-based 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 (Milne & Witten, 2008) 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 this 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 and new definitions of relatedness could unwittingly degrade algorithm performance.

3.8 Very Large Label Set Classification

In the three remaining sections of this chapter we move away from application-based research to research into general algorithm topics within

supervised classification that relate to our task. The first area that we address is research into multi-class classification algorithms. Our main motivation for this topic is the linking phase of our supervised task which involves a very large label set, which can include every concept in the ontology.

Large label sets are a known challenge for supervised classification algorithms (Dietterich & Bakiri, 1995; Breiman & al, 1984). One of the extensively investigated and defended mechanisms to handle such scenarios is to convert the problem to a binary classification one (Allwein & al, 2001); with two common approaches being the training a model for each class, OVA (Rifkin & Klautau, 2004), or the training of a model for each pair of classes, AVA (Fürnkranz, 2002).

Challenges to the application of these two generalized approaches to the task of concept linking include:

1) Some classes in our test set will have few if any examples in the training set. The application of an OVA or AVA based approach would result in many concept mentions never being predicted.

2) The number of classes in our task is far larger than that tested in the literature. The dataset with the most classes tested by (Rifkin & Klautau, 2004) is the `spectrometer` dataset from the UCI repository[12] with 48 classes. Our task could require thousands of classifiers to be trained which would likely result in a loss of performance because a large number of classifiers increase the

[12] http://kdd.ics.uci.edu/

chances that one of the classifiers will falsely claim to be the correct answer, and because we do not expect to have large amounts of training data per class.

Because of these two challenges we do not propose the direct application of OVA or AVA to our solutions. Instead we propose an approach more similar to the graph-edge prediction solution proposed in (Al Hasan & al, 2006) which we review in the next section.

3.9 Graph-Edge Prediction

Another relevant research area to the task is that of graph mining (Getoor & Diehl, 2005), specifically the prediction of whether two nodes are linked based on positive and negative examples of links. We focus the review on the work by (Al Hasan & al, 2006) which casts the problem as a binary classification task by creating feature vectors for each node pair and labeling the one that represents an existing link as TRUE. An additional contribution of theirs is an exploration of several different sources of information for predictive features ranging from topological features (such as the number of local edges) to intrinsic ones such as (matches between attribute-values).

A notable difference between their task and ours is they allow for all graph nodes to have multiple links. The scenario they investigate is drawn from social network analysis where people can interact with many people. A citation network is a canonical example of this scenario, where authors often write papers with a variety other authors. In our case edges represent reference relations, and so a concept mention must only link to one referent concept node in the ontology.

Thus for our proposal we include in Section 6.6 an additional step of selecting a single link in situation where the classifier predicts more than concept for a mention.

Finally, while (Al Hasan & al, 2006) point out that there is an inherent and significant skew of negative examples to positive examples, they only point to the literature on algorithms designed to handle classification in the presence of skewed data. We review this research area next.

3.10 Classification Modeling with Skewed Data

The final research area that we survey focuses on handling skewed training data where a few of the labels (possibly just one of the labels) are overrepresented, while the other labels are associated with few training cases. This topic is particularly relevant to us when we cast a problem as a binary labelling task, with most cases labelled as FALSE, and where a large number of mentions are labelled as UNLINKED.

Two proposed approaches to handling scenarios with label skew are either to pre-process the data by undersampling the cases with a dominant label (Chawla & al, 2004); or, to apply an algorithmic remedy, such as the adjustments to kernel function definition (Wu & Chang, 2004; Morik & al, 1999).

We use both approaches in our proposed solution. In Section 6.2, we present a heuristic that removes (undersamples) candidates from the training set that are likely to be FALSE. In Section 7.5, we adjust the SVM kernel to outweigh errors on TRUE cases more than on FALSE cases.

3.11 Conclusion

In this chapter, we reviewed the research that has most significantly influenced our solution to the task of supervised document to ontology interlinking. We focused our attention on the proposals by (Milne & Witten, 2008) and (Kulkarni & al, 2009), but also noted the relevant of sequential tagging models in named entity recognition (McCallum & Li, 2003) to the concept mention identification task, and the general need to support skewed data distributions. In the next chapter, we describe a dataset that we have created which will make the task more concrete and provide working examples for subsequent chapters.

4: THE KDDO1, KDD09CMA1 AND ICDM09CMA1 DATASETS

"*.. In 1993, very few scientific, technical, and medical (STM) journals had an electronic version, and yet by 2003, virtually all of them did.*" (Renear & Palmer, 2009)

This section describes three novel real-world datasets created to evaluate the proposed task. We describe the resource prior to the description of the proposed algorithm in order to draw examples from the resource in describing the algorithm. The datasets are an ontology for the field of data mining, and two annotated corpuses of research paper abstracts, also from the field of data mining. The corpuses and ontology are publicly available[13] and summarized below. To our knowledge, this is the first ontology and annotated corpora for a computing discipline.

4.1 The kddo1 Data Mining Ontology

The kddo1 ontology is based on a custom built semantic wiki[14] created specifically for the field of data mining and text mining by the author. In the wiki, each concept has its own distinct page[15] and follows the structured English approach described in (Melli & McQuinn, 2008). Each concept contains: 1) A preferred name; 2) A one sentence definition in the form of "*an X is a type of Y*

[13] http://www.gabormelli.com/Projects/kdd/data/

[14] 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/

[15] For example, http://www.gabormelli.com/RKB/Information_Extraction_Task

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 2 summarizes some statistics of the ontology.

Table 2– Summary statistics of the `kddo1` ontology

CONCEPTS			5,067
INTERNAL 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

4.2 The `kdd09cma1` Annotated Corpus

The author has also created an annotated corpus, `kdd09cma1`, in order to evaluate the proposed algorithm. Additional motivations for the creation of the corpus include the lack of similar resources, and the possibility that this corpus could be the seed of a valuable and naturally expanding corpus.

The `kdd09cma1` corpus is composed of the 139 abstracts for the papers accepted to ACM's SIGKDD conference, which took place in 2009 (KDD-2009)[16]. 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 in two separate phases. We first identified mentions of

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

the most specific concepts understood or used within the data mining community. This phase was performed without consideration for what concepts existed in the `kddo1` ontology. Next, an attempt was made to link the mentions to the concept in the ontology 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.

The corpus bears similarities to corpora from the bio-medical domain such as the GENIA[17] and BioCreAtIvE[18] 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 3 summarizes some key statistics about the corpus.

[17] http://www-tsujii.is.s.u-tokyo.ac.jp/GENIA
[18] http://biocreative.sourceforge.net/

Table 3 – 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

Given the novelty of the corpus and ontology, Table 4 summarizes some additional key statistics of the linking (external links) between the corpus and ontology.

Table 4 – Summary statistics of the external links from the `kdd09cma1` corpus to the `kddo1` 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

4.3 The `icdm09cma1` Annotated Corpus

A second corpus we present is one composed of twenty two manually annotated abstracts from the papers accepted into IEEE's annual conference on data mining in 2009 (ICDM'09[19]).

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

For this corpus, seven data mining experts[20] participated in the annotation. To ensure that most abstracts would be annotated by more than one person and that each person would annotate an abstract from a topic that they were comfortable in, we followed the following procedure. First, sixteen abstracts were randomly selected (from the 139[21] accepted papers), and each annotator was requested to select five abstracts of interest to them from this list. Their selected abstracts were then ranked according to the number of annotators who selected it, and each person was asked to annotate at least two abstracts in this order. Some of the abstracts were pre-annotated either by the proposed *SDOI* algorithm or by the baseline algorithm based on (Milne & Witten, 2008). The first abstract was always pre-annotated in order to familiarize the person with the complete annotation process.

We created an annotation environment based on the semantic wiki that houses the `kddo1` ontology. Wiki-style annotation identifies text with double brackets, e.g. "...a log fn was..." ⇒ "... a [[log fn]] was...". Further, wiki-style allows the linking of bracketed (anchor) text to a webpage by inserting the page name and a "|" character to the left of the text, e.g. "... a [[log fn]] was..." ⇒ "... a [[Log Function|log fn]] was..." The annotation environment is crude but realistic given that thousands of people use this annotation style and technology daily to edit pages on wikis such as Wikipedia.

Annotators were asked to follow a four-step procedure:

[20] Includes four active computing science graduate students, two former graduate students, and a professor who specialize in data mining research

[21] By coincidence, both KDD'09 and ICDM'09 accepted an identical number of papers: 139.

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;*
4) *Revise their annotations based on active search of the ontology (either by hyperlink navigation or by keyword search).*

Eleven abstracts were annotated by more than one person: ranging from five abstracts by four annotators, to two abstracts by two annotators. The author reviewed the annotated abstracts and then created a single ground truth version of each abstract.

Finally, eleven additional abstracts were randomly selected an annotated by the author in order to increase the total number of available documents to a larger quantity of documents that would result in performance estimates with stronger statistical confidence.

4.3.1 Inter-Annotator Agreement

This section analyzes the general agreement between the annotated documents produced by the annotators, and the single version of each document that is based on a consolidation of their inputs. We assess agreement by measuring the accuracy of each manually annotated abstract with respect to its consolidated version[22].

[22] The few abstracts per annotator make it infeasible to characterize individual performance bias.

A complication to the measurement of inter-annotator agreement was the "learning curve" associated with the annotation process. We noticed that the annotators annotated their first abstract more slowly, and the annotation resulted in lower agreement, than each of the other documents that they annotated. To account for this effect, we generally report results both on the entire annotated set and on a set that excludes the first abstract to be annotated by an annotator. Table 5, for example, reports the inter-annotator agreement with both sets. On average, 65.8% or each person's annotations were identical to the consolidate version. When the first abstract is removed the agreement increases to an accuracy of 71.8%. From this analysis, we infer that 70% is a more realistic maximum accuracy to expect from an automated solution than a 100% ideal accuracy[23].

Table 5 – 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%

Table 5 also reports performance based on a loosened definition of accuracy that does not require an exact match between the annotated concept and the predicted one. We define a "partial (near) match" to occur when the predicted concept is directly linked to the concept that the annotator linked to,

[23] We note that the "learning curve" persists to a lesser degree on their future abstracts, therefore we believe that the estimates of human performance underestimate their true long-term performance

and further, that the annotated concept is also directly linked to the predicted concept. This definition is intended to captures some of the subjectivity that is encountered during annotation when there is ambiguity exactly which concept the writer meant to express in the text. On the `icdm09cma1` corpus, for example, a term in the text such as “collaborative filtering” can be difficult to disambiguate as to whether the mention refers to a COLLABORATIVE FILTERING ALGORITHM or a COLLABORATIVE FILTERING TASK. However, since a direct link exists between the two concepts in the ontology predicting either one would result in a “near match”. Certainly, this loosened definition will also however lead to crediting unambiguously incorrect predictions. As expected, the loosened definition of a match increases the ensuing calculation of ‘accuracy’. We note that there is an approximately 1% increase in accuracy when based on a “partial (near) match”.

4.4 Conclusion

In this chapter we presented an ontology (`kddo1`) and two annotated corpuses (`kdd09cma1`, `icdm09cma1`) from the data mining domain that we have created to evaluate the task for document to ontology interlinking. The two corpuses are based on abstracts drawn from the KDD-2009 and ICDM-2009 conferences. Each of the abstract’s concept mentions have been manually identified and subsequently linked to a concept in the data mining ontology. With this data we can evaluate the performance of the three component algorithms of our proposed solution to the task that we now turn our attention to in the next three chapters.

5: FULLY-SUPERVISED CONCEPT MENTION IDENTIFICATION: $SDOI_{CMI}$

This chapter begins the description of our proposed solution to the task of document to ontology interlinking. We start with our proposed algorithm for the task of supervised concept mention identification: $SDOI_{CMI}$ and its empirical evaluation on the `kdd09cma1` corpus. Our main contributions in this chapter are to initiate the principled supervised solution to the overall task; and to propose a solution that can identify spans of text as concept mentions that are not explicitly represented in the ontology, as currently required by baseline solutions.

5.1 Proposed Approach

To identify concept mentions in a document we propose the use of a supervised sequential tagging algorithm based on linear-chain conditional random fields (Lafferty & al, 2001). Motivations for this approach to the task include its successful application to the similar tasks of syntactic chunking (Sha & Pereira, 2003) and of named entity recognition (McCallum & Li, 2003). Also, unlike the longest-matching substring heuristic generally used in the literature that requires the text span to be predefined in the ontology, this approach enables the identification of novel text spans as the anchor text of candidate mentions. The approach enables $SDOI_{CMI}$ to be also applied in complex domains where ontologies are incomplete (and evolving), and where concept mentions can often be expressed with many different token sequences.

A trained linear-chain CRF is a undirected conditional probability network with jointly distributed random variables that provides the conditional probability $P(\mathbf{Y}|\mathbf{X})$ of a label (tag) sequence $\boldsymbol{y} = y_1,...,y_n$ given a corresponding input token sequence $\boldsymbol{x} = x_1,...,x_n$. The structure of a linear-chain CRF on $(\boldsymbol{X}, \boldsymbol{Y})$ is defined by a feature vector, $\boldsymbol{f}$, and a corresponding weight vector, $\boldsymbol{\lambda}$. Each dimension of the feature vector is associated to a set of binary-valued functions (features), $\boldsymbol{b}(\boldsymbol{x}, i)$, that express some property of the empirical distribution of the training data that is expected to remain valid in the model distribution.

In the next section we define how sequences are to be labeled, and then continue with the definition of the features proposed for our task

5.2 Label Assignment

We propose the use of IOB tagging to our task (Ramshaw & Marcus, 1995). In IOB tagging a “B” label denotes the token that begins a span; the “I” label denotes an intervening token in the span; and the “O” label denotes any token that is outside a span – where a “span” in our case refers to the “anchor text” or a concept mention[24]. Figure 3 illustrates the labels used to identify concept mentions in a sample passage.

[24] Empirical analysis by (Tjong Kim Sang & Veenstra, 1999) suggest that performance on text chunking is minimally affected by alternate label assignments, such as IOE where ti end ("E") of the segment is demarcated.

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

```
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
```

5.3 Feature Space

We propose eight features for the sequential tagging model: five based on information about the token themselves, and three that draw on information from the ontology.

The first two (of five) token-based features that we propose are: (`Token`), the token itself, and (`POS`), its part-of-speech role[25]. These two features are commonly used in supervised text segmentation tasks (Sha & Pereira, 2003). The three other token-based features that we proposed are: (`CAP`) whether the first letter is capitalized, (`sChar`) whether a token contains a number or a special character, and (`fourC`) whether the token contains fewer than four characters. These three additional features are commonly used for named entity recognition tasks (Nadeau & Sekine, 2007), and while our task is not focused on named entities we do not precluded their presence in the text and the ontology.

Finally, given the availability of a domain-specific ontology, we propose three features O_1, O_2, and $O_{>2}$ that indicate whether a text token appears in the ontology within any single-token, two-token, or greater than two-token preferred name. We borrow this approach from prior work in supervised named entity

[25] While the token's POS is not available from our text data directly, the use of an automated part-of-speech tagger is accepted practice.

recognition that addresses the availability for large lists names (Minkov & al, 2005; Tsai & al, 2006).

Table 6 presents a sample sequence, the corresponding feature (unbinarized) values, and label.

Table 6 – Sample text sequence along with the associated features (the Token and POS features would be binarized)

Token	POS	CAP	sChar	fourC	O_1	O_2	$O_{>2}$	Label
$f_1(x)$	$f_2(x)$	$f_3(x)$	$f_4(x)$	$f_5(x)$	$f_6(x)$	$f_7(x)$	$f_8(x)$	$f(y)$
Collaborative	JJ	1	0	0	0	1	1	**B**
filtering	VBG	0	0	0	1	1	1	**I**
is	AUX	0	0	1	0	0	1	O
the	DT	0	0	1	0	0	1	O
most	RBS	0	0	0	0	0	0	O
popular	JJ	0	0	0	0	0	0	O
approach	NN	0	0	0	1	1	1	**B**
to	TO	0	0	1	0	0	0	O
build	VB	0	0	0	0	0	0	O
recommender	NN	0	0	0	0	1	1	**B**
systems	NNS	0	0	1	1	1	1	**I**
and	CC	0	0	1	0	0	1	O
has	AUX	0	0	1	0	0	0	O
been	AUX	0	0	0	0	0	0	O
successfully	RB	0	0	0	0	0	0	O
employed	VBN	0	0	0	0	0	0	O
in	IN	0	0	1	0	0	0	O
many	JJ	0	0	0	0	0	0	O
applications	NNS	0	0	0	1	0	0	**B**
.	.	0	1	1	0	0	0	O

Another common practice in the application of CRFs to tagging text sequences is to apply the features on the nearby tokens. The use of five token wide text window is common (Nadeau & Sekine, 2007). Similarly, bigram and trigram combinations of the features are commonly used. We also use a five token window and test on unigram, bigram and trigrams combinations.

5.4 Empirical Evaluation of Concept Mention Identification

In this section, we empirically evaluate the performance of the proposed concept-mention identification algorithm, $\texttt{SDOI}_{\texttt{CMI}}$. For this evaluation, we used the `CRF++` package[26], with its default parameter settings, to train the models.

5.4.1 Evaluation Setup

We assessed performance on the `kdd09cma1` corpus by using of a leave-one-document-out approach. For each document we train a model on the other (138) documents, apply the model to the held-out document, and evaluate the predicted concept mentions with the evaluation script from the CONLL-2000 text chunking shared task[27] which reports the achieved recall, precision, and F1 rates. Performance was further evaluated on predicted mentions that started on the correct token but ended on a different token – which we refer to as "partial matches". Overall performance is based on the average of the corresponding metric on the 139 holdout tests. The process was similarly followed for a dictionary-based baseline algorithm.

5.4.2 Baseline Definition

The baseline algorithm that we use to compare against on the concept identification task is an unsupervised dictionary-based approach (*dict*) that heuristically selects the longest token subsequence in the text that matches a concept's preferred name or synonym (raw or stemmed[28]). This is the identification method used in (Kulkarni & al, 2009) and in the experiments by

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

[27] http://www.cnts.ua.ac.be/conll2000/chunking/conlleval.txt.

[28] We used the Porter stemmer in http://tartarus.org/martin/PorterStemmer/

(Milne & Witten, 2008) on non-Wikipedia documents. For our task, which involves a nascent ontology for a complex domain, we expect this baseline to achieve poor recall rate because it cannot identify concept that are not yet in the ontology, nor ones with alternative spellings.

5.4.3 Full Data Analysis

Table 7 reports the performance results on the leave-one-document out analysis. SDOI outperforms the baseline on all metrics, with relative differences (lift) in performance measures ranging from 34% to 67%. In general, the lift is higher for precision than for recall[29].

We attribute SDOI's superior performance to its sequential model's ability to identify mentions not in the nascent ontology. Certainly, as more concepts and synonyms are added to the ontology, the longest subsequence heuristic's recall would improve. However, at the same time, the baseline's precision may also drop when able to guess at more predictions.

Table 7 – Average and relative (lift) concept mention identification performance (Precision, Recall, and F1-Score) of the baseline and SDOI algorithms on kdd09cma1, and under exact and partial matching criteria.

	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%

[29] Note that SDOI's recall performance can be traded-off for additional recall performance, by accepting predictions with a lower expected likelihood probability.

5.4.4 Learning Curve Analysis

In addition to the performance evaluation on the entire training corpus, we evaluated F1 performance on subsets of the training data in order to provide insight into the real-world effect of having access to additional annotated data. We again perform hold-one-document-out, but continually reduce the proportion of the dataset used: one-half, one-third, one-quarter, and so on[30]. For each proportion set, we average the performance on each of the training sets and average the number of concept mentions in the training documents. Because the heuristic baseline's performance is unaffected by additional training we simply retain the performance results reported in the previous section. In this analysis, we also investigate the effect of a reduced feature set. Figure 4 presents the results.

5.4.5 Observations

As expected from a supervised learning solution, additional training data results in increased F1 performance, with the rate of F1 improvement rapidly slowing. A learning curve will eventually 'plateau' given sufficient training data, however as Figure 4 shows, the `kdd09cma1` training corpus does not contains sufficient annotated data to reach $\texttt{SDOI}_{\text{CMI}}$'s performance plateau. A simple extrapolation however suggests that it would require one order of magnitude of additional training data to reach this plateau for partial match performance – two orders of magnitude of additional data for exact match.

[30] We randomly assign documents to the training subsets and test on each of the remaining documents. Starting with the proportion of one-eighth only five subsets are created to speed up the process while still achieving a stable estimate.

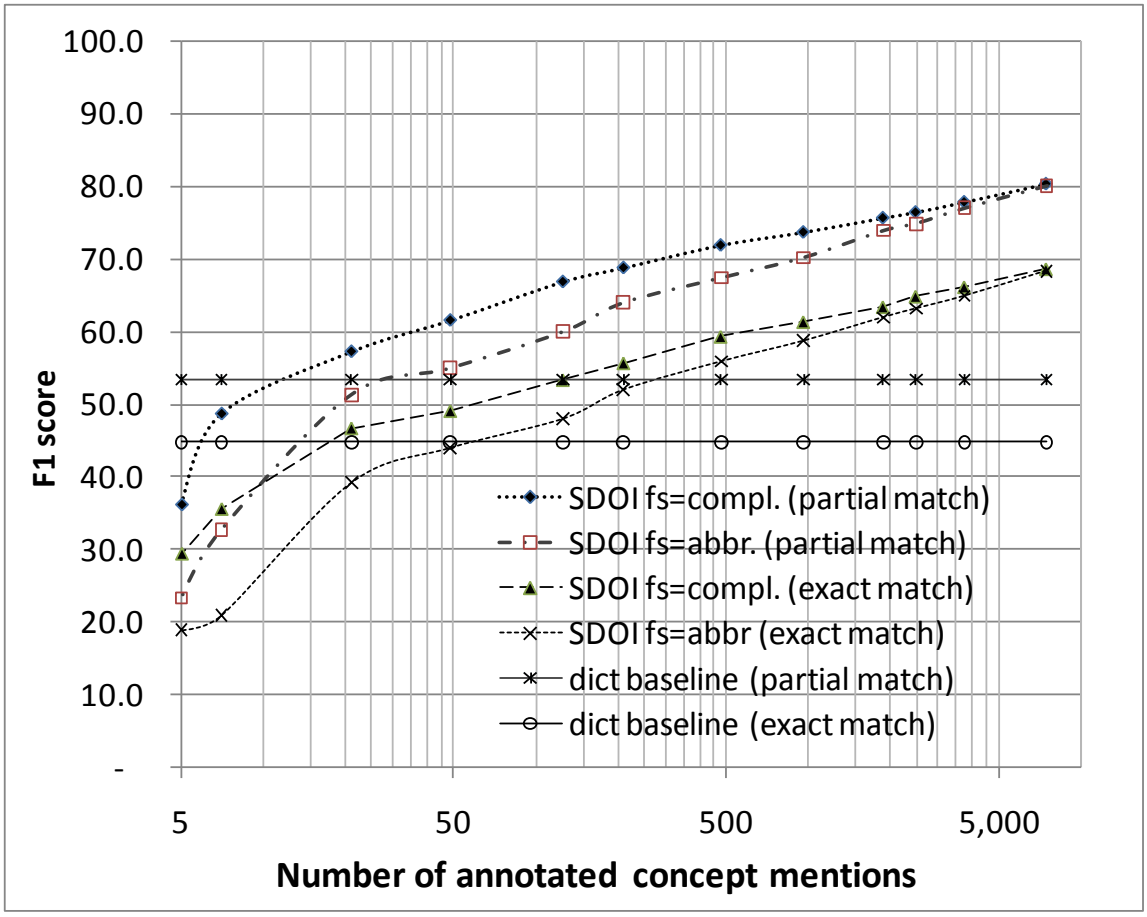


Figure 4 – Log-scale learning curve analysis of SDOI's and the baseline's F1 performance on the kdd09cma1 dataset under exact and partial match criteria. For SDOI two features spaces: complete and abridged (only POS and token)

The trend supports the recent dictum within the statistical NLP community that "there is no data like more data". While few domains or organizations can justify the resources required for this amount of annotation, we will see in the next section that the generated models both may be portable across corpuses and may be used to reduce annotation time. These two outcomes suggest that organizations could pool their annotation efforts to improve overall performance and speed-up the annotation[31].

[31] For example the organizations that organized the KDD, ICDM, PKDD, PAKDD, and SIAM conferences could pool their annotation work and achieve better performance.

We observe that SDOI's F1 performance overtakes that of the baseline after approximately 30 to 50 annotated concept mentions have been trained on. This information can be used to determine the minimal amount of effort required to invest in the approach when applying it to other domains (assuming that the data in other domains behaves in a similar manner).

The use of all of the proposed features results in higher performance than when only the two most commonly used features (the token and part-of-speech tag) are used. Interestingly however, as Figure 4 illustrates, the relative benefit in performance shrinks as more data is made available, and becomes marginal once the entire kdd09cma1 corpus is available for training. This shrinking benefit from feature engineering, along with the distant performance plateau suggests that future research could focus more on ways to reduce the effort required to annotate mentions rather than on improvements in the feature space.

Finally, there is a significant and persistent lift in performance when partially matched multi-token mentions that start on the same token but end at a different one are accepted. We address this difference in F1 performance in the next section.

5.5 Analysis on Misclassifications

In this section, we review examples of misclassified concept mentions from the kdd09cma1 corpus in order to better understand the performance results. The misclassified multi-token examples are presented in Figure 5. In all examples, $SDOI_{CMI}$ detected the correct starting token, and often the final token

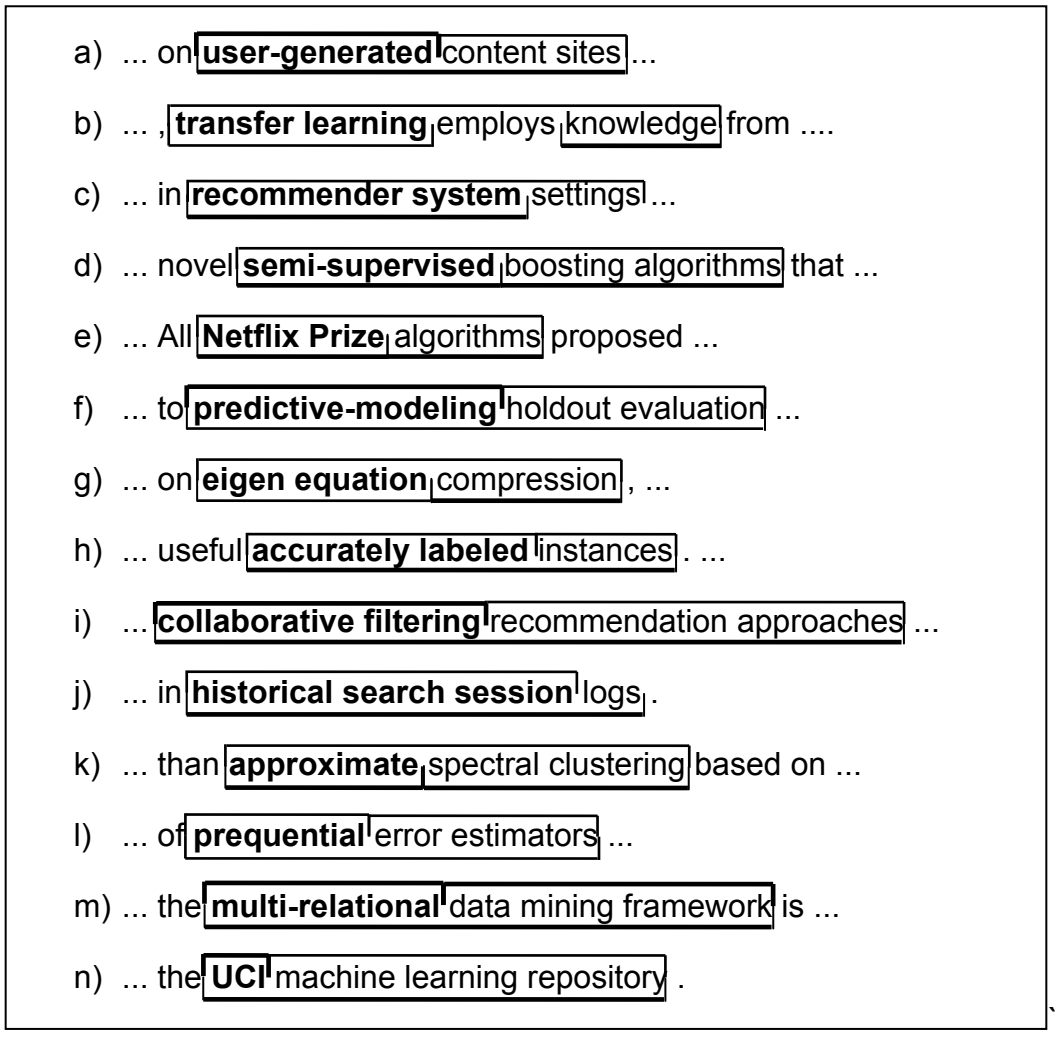
as well. A common pattern within these misclassified multi-token examples is that mentions are largely being divided at different spots by the annotator and the algorithm. These differently divided mentions clearly would affect the concept that it would be linked to, but they may still result in link to relevant concepts. For example, the manually annotated mention of "*user-generated content site*" is split by $\texttt{SDOI}_{\texttt{CMI}}$ into two concept mentions "*user-generated*" and "*content site*".

This pattern suggests an explanation of why the "partial match"-based comparison achieved such significantly higher F1 results in the learning curve analysis shown in Figure 4. Under an exact match criteria these mentions would often result in three penalties: either two false negatives (FN) and one false positive (FP) assessment, or one false negative (FN) and two false positives (FP). Under the partial match criteria, these mentions would results in one credit and one penalty: one true positive (TP) and either one false negative (FN) or one false positive (FP). The approach of a single penalty may be relevant to some settings. Imagine for example that a person had to review and 'fix' a mistake made by $\texttt{SDOI}_{\texttt{CMI}}$. Depending on the annotation interface, the examples below suggest that the annotator could more quickly determine and remedy the problem than have to both identify the start token and end token and then determine if the span of text also required to be further subdivided.

Under the strict exact-match criteria, the challenge appears to focus on correctly identifying whether the mention is to be divided and where the division occurs. We had envisioned that the inclusion of ontology-based features would provide a sufficiently strong signal to classifier to make this decision. Future work

may investigate the use of semi-Markov CRFs (Sarawagi & Cohen, 2004) in which labels are assigned to segments (not tokens). Further, it would be interesting to feedback the information about the more confident predictions by the linking and relation identification solutions that will be presented in the next two chapters.

Figure 5 - Sample passages from `kddcma1` that contain misclassified multi-token concept mentions. The underlined text represent the manual annotations; the lines above the text SDOI's predicted mention boundaries. The bolded tokens represent segments that are deemed as partial matches. The thicker lines represent linked anchor text while the thinner lines represent unlinked mentions (again, predicted on top and manual on bottom).



5.6 Conclusion

In this chapter, we presented the $\texttt{SDOI}_{\texttt{CMI}}$ supervised algorithm for the task for concept mention identification. The algorithm applies a supervised sequential tagging algorithm to the task based on eight proposed features drawn from the text and ontology. We evaluated the algorithm on the `kddcma1` dataset using a leave-one-document-out framework and demonstrated a significant performance improvement over a longest-matching token sequence baseline algorithm. We investigated some of the misclassification errors and that one of the main challenges is the identification of the splitting point of multi-token mentions.

Given identified concept mentions we can now turn to the challenge of linking the mentions to a concept in a relevant ontology.

6: FULLY-SUPERVISED CONCEPT MENTION LINKING: $SDOI_{CML}$

In this chapter, we transition our investigation from the supervised identification of concept mentions within a text to the task of linking identified concept mentions to a concept within an ontology, where possible. The chapter begins with a definition of our proposed algorithm for the task and concludes with a report on an empirical evaluation of the algorithm's performance. Our main contributions are: a continuation of the principled supervised approach to the task; a demonstration that when connected into a pipeline our proposed concept mention identification $\texttt{SDOI}_{\texttt{CMI}}$ and linking $\texttt{SDOI}_{\texttt{CML}}$ algorithms outperform baseline algorithms; and, also in this chapter we demonstrate that SDOI can reduce the amount of time required by annotator to annotate a document.

6.1 Proposed Approach

We propose the use of a fully-supervised classification algorithm to solve the concept mention linking task. The basic approach is to train a classification model on the training corpus and then use the model to predict the "class" (the ontology concept) of a concept mention in an unseen document. The use of a supervised algorithm to directly predict the ontology concept however is complicated by the very large number of classes (ontology concepts) that can be predicted and by the fact that many of the classes that will be encountered in test documents will not be represented in the training data. To overcome this

challenge we propose the transformation of the problem to a binary classification one in which the prediction is whether a given concept from the ontology represents the correct or incorrect (TRUE or FALSE) link for the concept mention. Finally, in special cases where a concept mention is associated with more than one concept that is predicted to be TRUE for it, we define a mechanism to select one of the concepts[32].

Our proposed approach enables the application of any one of the many successful state-of-the-art binary classification algorithm in existence, and also employs a feature space which can be expanded on in the future as additional information becomes available that appears to be relevant to the classification task. A challenge of the proposal however is the skewed dominance of FALSE labeled training examples over TRUE labeled ones. For this challenge we propose a filtering heuristic that substantially reduces the number of FALSE labeled examples with minimal culling impact on the TRUE labeled examples.

Procedurally the algorithm is structured as follows: 1) a set of candidate concepts is associated with each mention, 2) a set of features is associated with each candidate, 3) a binary classifier is trained with a supervised classification algorithm, and 4) a selection heuristic is applied when more than one TRUE prediction is made for a mention.

[32] The approach is akin to the supervised graph-edge prediction of (Al Hasan & al, 2006) that reviewed in Section 3.8. Recall however that their task did not require a single link/edge to be added between nodes; and they do not directly address the label skew challenge.

6.2 Training Example Filtering

While a concept mention could be linked to any one of the many concept in the ontology, we suggest that knowledge of the mention's anchor text can inform a filtering rule that significantly reduces the number of concepts to be realistically be considered as candidates for the evaluation – with low likelihood of discarding the correct (TRUE) concept to link to 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 could reasonably include concepts such as PREDICTOR FEATURE, APPLICATION FEATURE, and DATA TABLE ATTRIBUTE (one of which ideally is the correct concept for the mention), and could reasonably exclude irrelevant concepts such as: HIDDEN MARKOV MODEL, LINEAR DISCRIMINANT ANALYSIS, and thousands of other unrelated concepts. In this section, we define a filtering heuristic that determines which concepts in the ontology are considered as candidates for a given concept mention.

We refer to the set of concepts associated to the anchor text of a concept mention as its ***candidate set*** which is composed of zero or more distinct concepts from the ontology: $a_m \rightarrow C_m = \{\varnothing, o_{c'}, o_{c''}, \ldots\}$. The proposed heuristic is based on a set of eight (8) individual TRUE/FALSE tests that directly compare the concept mention's anchor text against some information about the concept $t_i(a_m, o_c)$. The overall heuristic accepts the union of all concepts that are accepted by any one of the selected tests. Note that while this section proposes eight tests, the actual subset of these tests that will be proposed for `SDOI` will be determined empirically (see Section 6.7.3).

The first test we propose, t_1, simply requires an exact match between the anchor text and the preferred name of the concept. 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 {t_1, t_2,} allow for the replication of the approach proposed in (Milne & Witten, 2008; Kulkarni & al, 2009).

In more specialized domains, that involve complex multi-token concept mentions and ontologies with incomplete synonym sets, however, these first two tests would result in a weak recall rate of the correct concept. An anchor text such as "*supervised learning of a sequential tagging model*" for example, would be very likely excluded. To increase the number of relevant candidates we define two additional candidacy tests for consideration.

Another test that we propose, t_3, probes into the documents in the training corpus to determine whether the anchor text was also linked to this concept. For example, if the training corpus has the phrase "*supervised learning of a sequential tagging model*" identified as a concept and linked to a concept then would detect the sequence again in a future test document.

Because many of the concept mentions and concept synonyms are composed of more than one token, the final primary test that we define, t_4, accepts a concept where any of the component tokens match. Table 8 summarizes the four *primary* tests of candidacy.

Table 8 – the primary tests used to determine whether concept (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 instead uses the stemmed versions of the text being compared. We denote these 'stemmer-based' tests as: t_{s1}, t_{s2}, t_{s3}, and t_{s4}.

6.3 Label Assignment

During training each candidate concept that passes the filter for a concept mention is assigned a TRUE or FALSE label. Because of the availability of manually annotated information the labeling procedure is straightforward. If the candidate concept associated with the mention matches the concept assigned by the annotator then a TRUE label is assigned; other a FALSE label is assigned. As a consequence, and as desired, mentions that the human annotator deemed not to have referent concept in the ontology will only have FALSE labels associated with every one of its candidate concepts.

6.4 Feature Space

This section describes the feature vector associated with each training case defined by a concept mention and candidate concept pairing. Table 9

illustrates the structure of the training data produced. Each of the seven feature categories is defined in the coming subsections.

The final category of features, referred to as “Collective Features”, differ substantially from the others in that they are recursively defined on the labelling decisions of a classifier. Thus, given the additional complexity of these features we describe their population (by means of an iterative classification procedure) in Section 6.5.

Table 9 – Illustration of the structure of the training data used for the linking task.

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
…	…	…	…	…	…	…	…	…	…	…	…	…	…	…		…

6.4.1 Anchor Text-based Features $f(a_m,o_c)$

Each of the eight tests defined for the filtering heuristic to determine concept candidacy are included as binary features. The intuition for their inclusion as features is that these tests signal how closely a mention’s anchor text matches some text associated with the concept in the ontology.

6.4.2 Text Window-based Features $f(t_m,o_c)$

Another source of features is text beside the concept mention's anchor text (its *text window*) and from the text used to describe the concept in the ontology. We propose a feature for 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 (Kulkarni & al, 2009)[33] and replicates the type of similarity measured proposed in (Lesk, 1986) for the word sense disambiguation task. We use the same text window size of five tokens used in (Kulkarni & al, 2009).

6.4.3 Document-based Features $f(d_m,o_c)$

Rather than limiting the analysis to the text window, information can also be drawn from the entire document can also inform the classification decision. Two proposed features drawn from the document are: 1) the cosine distance between the normalized bag-of-word vector representations of the document and the ontology description (also proposed in (Kulkarni & al, 2009)), 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. Table 10 summarizes these two features.

[33] they also include dot product and Jaccard similarity

Table 10 – Document-based features

FEATURE	DEFINITION
$\cos(d_m,o_c)$	The bag-of-word cosine similarity between the document and the concept description. Proposed in (Kulkarni & al, 2009)
$\text{tok}(a_m,d_m)$	Number of tokens between the start of the document and the first token in the mention.

6.4.4 Candidate Concept-based Features $f(o_c)$

The candidate concept 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, (Kulkarni & al, 2009) proposes the use of the frequency that a concept is linked to (its inlink count) as a feature. We include this count as a feature, $CI(o_c)$ and also include the count of internal links extending out of the concept (its outlink count), $CO(o_c)$. The first feature signals the popularity of the concept as a reference within the ontology. The second feature can signal whether the concept has received significant attention by the ontology engineers in the form of many (or few) links. Table 11 summarizes these two features.

Table 11 – Candidate concept-based features

FEATURE	DEFINITION
$CI(o_c)$	Cardinality of all internal links into o_c, i.e. $\|(o_c)\|$
$CO(o_c)$	Cardinality of all internal links from o_c, i.e. $\|O(o_c)\|$

6.4.5 Corpus Based-based Features $f(o_c,D)$

A novel data source of linking features that we propose is the use of training corpus. An intuition for the use of the training corpus is that if an anchor text in a corpus is frequently linked to a concept then this is more likely true in future unseen documents. Further, note that the t_3 candidacy test also signals the presence in the corpus of an identical anchor text. The corpus-based feature(s) are defined in Table 12.

An implementational challenge associated with the use of corpus-based features is the risk of creating different training and testing spaces. When calculating the corpus-based feature values for a training example the document that contains the associated mention must be excluded from the calculation. An intuition for this removal of the document is that it likely contains repetitions of the mention under consideration and these other mentions should not influence the calculation of a feature because a test document would not have the benefit of its repeated items from contributing.[34]

Table 12 – Corpus-based features

FEATURE	DEFINITION
CE (a_m, o_c, D)	Cardinality of all external links into o_c, i.e. $\|E(o_c, D')\|$, where $d_m \notin D'$

[34] The effect of not removing the document is particularly noticeable during cross-validation studies where a repeated concept mention provides significant information and would inflate the results.

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

Another novel feature type is based on the information about the size and membership of the candidate set of concepts associated with each anchor text. For example, it may be riskier to pick a concept from a large candidate set than from a candidate set composed of only two members. Table 13 summarizes these features.

Table 13 – Candidate Concept Set-based Features

FEATURE	DEFINITION
$CC(C_i)$	Cardinality of the set of candidate concepts. i.e. $\|C_i\|$
$\Sigma CI(C_i)$	Count of internal links into all candidate concepts. $CI(o_{j'}) + CI(o_{j''}) + \ldots$, for all $o_j \in C_i$
$RCI(o_j, C_i)$	Relative proportion of the internal links into the candidate concept relative to overall size. $CI(o_j) / \Sigma CI(C_i)$
$\Sigma CO(C_i)$	Count of internal links out from all candidate concepts. $CI(o_{j'}) + CI(o_{j''}) + \ldots$, for all $o_j \in C_i$
$RCO(o_j, C_i)$	Relative proportion of the internal links out from the candidate concept relative to overall size. $CI(o_j) / \Sigma CO(C_i)$
$\Sigma CE(C_i)$	Count of external links into all candidate concepts. $CE(o_{j'}) + CE(o_{j''}) + \ldots$, for all $o_j \in C_i$
$RCE(o_j, C_i)$	Relative proportion of the external links into the candidate concept relative to overall size. $CE(o_j) / \Sigma CE(C_i)$

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

(Milne & Witten, 2008) and (Kulkarni & al, 2010) focus much of their attention on "context" or "collective" features whose role is to inform the classifier about the other concepts that are mentioned in the document. With possession of some disambiguated links to the ontology, the ontology can provide some background knowledge into the classification decision for the remaining links. For

example, when faced with the ambiguous anchor text of "LDA" (which could refer to LATENT DIRICHLET ALLOCATION or LINEAR DISCRIMINANT ANALYSIS), the classifier could benefit from information about the presence, elsewhere in the document, of the TOPIC MODELING concept (which is directly linked to LATENT DIRICHLET ALLOCATION in the ontology) or the presence of the ANOVA concept (which has a direct link in the ontology to the LINEAR DISCRIMINANT ANALYSIS concept)[35].

In this section, we describe a set of features that are informed by the other concepts mentioned in the document. A challenge to calculating these features is that we do not know apriori which concepts are being mentioned (because it is the task that we are trying to solve). In this section, we describe the two-step heuristic approach proposed by (Milne & Witten, 2008) to resolve the challenge. In the following section, we describe an alternative principled approach to the calculation of the collective features.

The first set of collective features that we defined are the ones proposed by (Milne & Witten, 2008). Their proposal makes use of a "*relatedness*" measure, which in turn is based on (Cilibrasi & Vitanyi, 2007)'s Normalized Google Distance (*NGD*) metric used to assesses the dissimilarity between two sets.

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

Given two ontology concepts (o_a, o_b) their proposed relatedness measure defines two sets based on the links within the ontology into each of the concepts

[35] Other examples of ambiguous mentions include "wrapper", "boosting", and general terms such as "algorithm" or "task" which are used in an anaphoric role to a more specialized term mentioned earlier in the document.

A=*inlinks*(o_a), and *B*=*inlinks*(o_a). They also map NDG's function range of [0,∞] to a similarity metric by truncating the output to [0,1] and then subtracting the result from the value one (1).

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

We extend this feature space by including the components used in the calculation of *NDG*, and by including a Jaccard similarity function-based feature. Table 14 defines some relevant functions to collective analysis, while Table 15 defines the collective features. In the case where the context set of mentions is empty, such as initially when no mentions have been linked, these features all calculate to zero (0).

Table 14 – Functions used to define collective features based on the relatedness of two concepts.

FUNCTION	DEFINITION
Ca∩b(o_j, o_k)	The cardinality of I(o_j) ∩ I(o_k)
Ca∪b(o_j, o_k)	The cardinality of I(o_j) ∪ I(o_k)
max_ab(o_j, o_k)	The larger of |I(o_j)| or |I(o_k)|.
min_ab(o_j, o_k)	The smaller of |I(o_j)| or |I(o_k)|.
AMW08rel(o_j, O')	The average relatedness. of concept o_j and O', where $o_j \notin O'$.

Table 15 – Definition of the collective features.

FEATURE	DEFINITION
$CS(S_i)$	The cardinality of the set of context concepts. i.e. $\|S_i\|$.
$\Sigma IS(S_i)$	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. $Ca{\cap}b(o_i, S_i)$, where $o_k \in S_i$ and $o_i \neq o_k$.
$AMW08rel(o_j, S_i)$	The average weighted relatedness between the concept o_j and each of the concepts in S_i, as proposed in (Milne & Witten, 2008).
$\Sigma MW08\ (S_i)$	The sum of the relatedness between each concept in S_i. to the other concepts in S_i. This feature is proposed in (Milne & Witten, 2008) 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 concepts in S_i.

6.5 Collective Feature Handling via Iterative Classification

In this section, we propose an alternative method to the two-step heuristic proposed in (Milne & Witten, 2008) to the calculation of the "collective" features defined in Section 6.4.7. Recall that those features require that some portion of a document's concept mentions be already linked to the ontology. (Milne & Witten, 2008) accomplish this assignment by first identifying some mentions heuristically as "context" mentions that do not require disambiguation. (Kulkarni & al, 2009) accomplish this assignment by specifying a custom objective function that is then optimized by, for example, greedily committing to the next highest scoring mention.

We propose an incremental approach, as (Kulkarni & al, 2009), but one that is directed by a supervised learning algorithm rather than through the optimization of a custom function that requires advanced machine learning expertise to update for the domain. Our approach is to apply an iterative

classification algorithm similar to the one suggested by (Neville & Jensen, 2000) for relational classification settings. The intuition of their approach is to, during the application of the model on new data, identify the most confident predictions, select them, and based on these selected predictions recalculate the values of the collective features on all cases. The recalculated vectors are ready to repeat the cycle, and each with each iteration selecting a larger and larger proportion of the predictions until the final iteration selects all cases. Specifically, their proposal is to perform μ iterations of classification where $1/\mu$ of the predictions are selected. During the training phase they propose that the collective features of the training examples be based on complete knowledge of all the labels (in our case concepts) associated with them.

A challenge that we encountered with the direct application of this approach to our task and feature space definition is that $\mathtt{SDOI}_{\mathtt{CML}}$'s collective features are all zero (0) valued when no concepts have been committed to. Thus, feature vectors will differ dramatically (on the collective features) during the transition from the first iteration to the second one. We propose instead that on the 0^{th} iteration a second model be used, one that was trained on all but the collective features. In this way, the gap between the trained and tested feature vectors (on the collective features) is substantially reduced on the first (a likely critical) iteration.

Assume that the constant number of iterations, μ, is provided, and along with a set of N training instances[36]. Our proposed training and testing process for $\texttt{SDOI}_{\texttt{CML}}$ is presented in Figure 6. Steps 1 and 4b are novel additions.

Figure 6 – Proposed iterative classification algorithm.

1. Train model ($M_{\not{col}}$) without the collective features
2. Train a model (M_{col}) with the collective features (and oracle)
3. Create the test set (without collective feature values)
4. For each test iteration of ι from 1 to μ
   a. Calculate the value for the collective features (with selected links in *K*)
   b. Apply model $M_{\not{col}}$ to the test set, if ι is 1
      otherwise, apply model M_{col} to the test set.
   c. Select in *K* the κ most probable links, where $\kappa = N(\iota/\mu)$.
5. Output the final set of predictions on all mentions.

6.6 Mention-level Classification

The presented linking algorithm defined to this point trains a classifier that predicts whether a candidate concept from the ontology is the correct concept to associate to a given concept mention. When a mention has multiple candidates however, the classifier may assign the TRUE label to more than one of the candidates. Which to chose? The challenge arises because of our original conversion of a multiclass task into a binary one. We conclude the design of our proposed algorithm by defining the mechanism that will make the final selection of a single candidate when more than one TRUE prediction is made..

We propose two alternatives to mention-level classification. The first approach (which will be default approach during the evaluation) is to use a

[36] Looking ahead, the parameter μ will be set to 5, during the evaluation of $\texttt{SDOI}_{\texttt{CML}}$.

heuristic rule based on available information. The second approach trains a separate classifier to select the winning candidate.

The use of a heuristic rule to solve a multi-class problem with a binary classifier has been widely investigated in the literature (Rifkin & Klatau, 2004) and is the approach used by the related work of (Milne & Witten, 2008). Typically, the rule assumes the use of a binary classifiers, such as a kernel-based classifiers, a decision trees or a logistic functions, that also provide a score with each prediction, This score can generally be used to rank the predictions according to the likelihood of their correctness. The heuristic then, which we refer to a "*pick-largest-true-score*", selects the test example with the highest likelihood score – with ties broken randomly.

The "pick-largest-true-score" heuristic approach described above however does not account for dependencies that may exist between the candidate concepts. For example, in a scenario with the five candidate concepts, two of which are predicted to be TRUE, imagine further that neither is associated with a significantly larger confidence and that the one with the lower confidence is much more commonly linked to. In this type of situations it may be more reasonable to pick the candidate with the lower confidence. Further, it may occasionally be better to pick the most common class (which generally is the unlinked label). Based on this intuition, the alternative mechanism investigated is to trains a classifier at the mention level based on aggregated features from the candidate set. A further appeal of this approach is that it more closely approximates the ideal of making data-driven decisions.

We propose in Table 16 a small feature set that can be used to assess whether there is value in the modeling at the mention-level. The label associated with each mention is whether the default rule of picking the concept with the highest confidence is correct or not.

Table 16 – the features for the mention level classifier to be trained.

Feature Name	Definition
TruePreds	The number candidates predicted as TRUE in the candidate set
FalsePreds	The number candidates predicted as FALSE in the candidate set
HighTrueConf	The highest confidence value for a true prediction (if one existed).
HighFalseConf	The highest confidence value for a false predictions (if one existed).

Looking ahead, in Section 6.7.7 we report on the effectiveness of a trained model over the default "*pick-largest-true-score*" heuristic. The empirical results suggest that the simple heuristic rule is at least as accurate as the model-based approach. Thus, this approach to a collective-style decision when more than one case is labelled TRUE does not appear to be a worthwhile addition on this task.

6.7 Empirical Evaluation of Concept Mention Linking

In this section, we empirically evaluate the performance of the proposed concept-mention linking algorithm, $\texttt{SDOI}_{\text{CML}}$. For this evaluation, we used the `SVMlight`[37] package, with its default parameter settings, as the underlying supervised classification algorithm.

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

6.7.1 Evaluation Setup

Similar to the analysis performed for on concept mention identification, we used a leave-one-document-out method on the `kdd09cma1` corpus and compared against the accuracy attained by baseline algorithms. For each document we first train a model on the other (138) documents, apply the model to the held-out document, and evaluate the accuracy of the class (concept) assignment[38]. The overall accuracy is based on the average accuracy on the 139 holdout tests.

For this task, we also analyze performance on a separate corpora: `icdm09cma1`. Our approach is to apply the model trained on the `kdd09cma1` corpus to predict concept mention links on each of the document in the `icdm09cma1` corpus, and in this manner assess whether the annotation effort in one corpus is portable across to another. We also evaluate the ability of these predictions to reduce the amount of time required to annotate future documents.

In summary, the analysis indicates that $\texttt{SDOI}_{\text{CML}}$ outperforms the baseline both on the intrinsic performance measure of accuracy and on the extrinsic measure of annotation time saved. The analysis also shows that the benefits of the annotation effort on one corpus can be ported to a separate corpus, and that the accuracy attained is sufficient to speed-up the real-world effort of semantic annotation.

[38] Accuracy is calculated by counting the number of correct predictions and dividing by the number of predictions made.

6.7.2 Baseline Algorithm

The main baseline algorithm that we compare against for the linking task is the supervised approach proposed in (Milne & Witten, 2008). We reimplemented their proposed features (*RCI*, *AMW08rel*, and $\Sigma MW08$ as defined in Section 5) and their algorithm's two-step approach (as described in Section 6.4.7) to calculate their two collective 'context' features. Recall that the first step selects a set of 'context' concept mentions (S_{MW08}) whose links can be linked by an accurate heuristic, once these links are established the context features are calculated, and the binary classifier trained on the three features. We reimplemented their concept candidacy heuristic by accepting all candidate concepts that satisfy tests t_1+t_{1s}+t_2+t_{2s}, and use their "pick-largest-true-score" heuristic to select the winning concept.

A secondary baseline algorithm that we compared against is the naive unsupervised approach of selecting the concept with the larger number of internal inlinks in the ontology (CI_B). That is, when a candidate set has more than one concept link to the more "popular" one (popular within the ontology).

6.7.3 Candidacy Filter Finalization

Before proceeding to assessing SDOI's performance we first select, by empirical means, the subset of the eight candidacy test defined in Section 6.2 that filter which concepts to test against for each concept mention. Recall that the definition of the candidacy selection heuristic can affect performance: too restrictive a policy will limit the maximal attainable recall performance; too liberal

a policy could swamp the classifier with a large ratio of negative-to-positive training cases.

We empirically test the effect on accuracy of incrementally adding individual tests (primary and stemmed[39]) in the following simple sequential order: t_1+t_{1s}+t_2+t_{2s}+t_3+t_{3s}+t_4+t_{4s}. Table 17 summarizes the impact of sequentially adding each of the tests. Based on this empirical analysis, the candidacy test select for `SDOI` for the subsequent experiments was composed of tests: t_1+t_{1s}+t_2+t_{2s}+t_3+t_{3s}. Adding more tests beyond this point drops *F1* performance significantly, likely because the average number of training cases per mention increases from approximately 2.5 to 47 cases per mention on average. This large leap in training cases between t_3-based tests and the t_4-based tests (from 11,598 to 296,086 tests) suggests that some performance gain may be achieved as future work by investigating additional and more nuanced candidacy tests.

Table 17 – Effect of the candidacy test definition on linking performance. As the resulting filter becomes more inclusive, the maximum possible recall and number of training increases. The combination of tests selected for `SDOI` 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%

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

6.7.4 Linking Performance on `kdd09cma1`

To estimate algorithm performance, as mentioned earlier, 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 number of iterations for the iterative classifier used to handle the collective features was set to five (μ=5).

Table 18 reports the linking accuracy achieved from different combinations of (two) feature spaces and (two) candidate heuristics. The feature spaces tested are the complete feature space of $\texttt{SDOI}_{\texttt{CML}}$ and the one proposed by (Milne & Witten, 2008). The candidate heuristic tested is a baseline composed of t_1+t_{1s}+t_2+t_{2s} which most closely approximates the one used in (Milne & Witten, 2008) and (Kulkarni & al, 2009) and the heuristic selected for $\texttt{SDOI}_{\texttt{CML}}$ in the previous section which also includes the corpus-based candidacy tests of t_3+t_{3s}. In summary, both the additional features proposed for $\texttt{SDOI}_{\texttt{CML}}$ and the addition of the corpus-based candidacy tests (t_3+t_{3s}) equally help to significantly increase accuracy. All subsequent evaluations that we report on use $\texttt{SDOI}_{\texttt{CML}}$'s candidate concept selection heuristic.

Table 18 – Linking accuracy (on manually identified `kdd09cma1` mentions) for two different feature space and candidacy heuristic combinations.

Candidacy Heuristic	Feature Space	
	$\texttt{SDOI}_{\texttt{CML}}$	MW08
$\texttt{SDOI}_{\texttt{CML}}$	57.3%	44.7%
baseline	46.0%	33.8%

Next, we assess the performance when the concept mention identification and linking tasks are sequentially performed in a pipeline manner. Table 19

contrasts the linking accuracy achieved when the underlying anchor text was identified either by the human annotator or by one the algorithms presented in the previous chapter: either: $SDOI_{CMI}$ or the baseline heuristic approach that approximates the identification method used in (Milne & Witten, 2008) and (Kulkarni & al, 2009). In summary, as expected, overall accuracy is reduced when link predictions are made against predicted anchor text. Further, the relative difference in accuracy is more significant for the joint task of linking on algorithmically identified mentions. This larger drop in accuracy by the baseline is likely due to the multiplicative effects of performance of serially performing the identification and linking: The baseline linking algorithm received fewer opportunities to predict a correct link for a concept mention because the baseline identification method had failed to identify them to begin with.

Table 19 – Accuracy of the $SDOI_{CML}$ and (Milne & Witten, 2008) algorithms on the `kdd09cma1` linking task when based on true anchor text or those predicted by the $SDOI_{CML}$ and baselines algorithms described in Chapter 5:

anchor text identified by …	$SDOI_{CML}$	MW08
annotator	57.3%	44.7%
$SDOI_{CMI}$	45.4%	-
baseline	-	17.7%

Table 20 reports the effect loosening the definition of a “match” between the predicted and annotated concept. Rather than requiring an exact match we also test a “near match”-based accuracy that attempts to capture some of the subjectivity encountered during annotation, such as when there is ambiguity

about exactly which concept the writer meant to express in the text[40]. In summary, as expected, the loosened definition of a match increases the ensuing calculation of 'accuracy'. The difference between the two results of approximately 6% is larger than the approximately 1% difference encountered in the inter-annotator agreement analysis in Section 4.3.1. This larger difference suggests that the algorithmic approach is more likely to select unintended concepts – but that these unintended concepts are still "near" the intended one.

Table 20 – Accuracy of $SDOI_{CML}$ on kdd09cma1 where "accuracy" requires either an 'exact' match between predicted and annotated concept, or a 'near' match where direct links exist in the ontology between the concepts.

Exact Match	Near Match
57.3%	63.2%

6.7.5 Unsupervised Baseline Performance

Table 21 reports relative difference in accuracy[41] between $SDOI_{CML}$ (under different feature spaces configuration) and the unsupervised baseline approach, CI_B, that select the concept with the largest number of inlinks within the ontology. The results indicate a benefit of committing to all proposed features. They also show little lift from the sole use the (Milne & Witten, 2008) feature set. Recall that this feature set contains the commonness of the concept and two collective features. Since the commonness is captured by the CI_B, the small relative

[40] Recall that we defined a "near match" in Section 4.3.1 to refer to predictions in which the predicted concept is directly linked to the annotated concept, and the annotated concept is directly linked to the predicted concept.

[41] (SDOI accuracy – Unsupervised accuracy)/SDOI accuracy

difference of 1.2% (on the second row) is further evidence that collective features play a marginal role in this task[42].

Table 21 - Relative difference in the *accuracy* of $SDOI_{CML}$ (on five different feature sets) relative to the most common concept (CI_B) baselines.

CI_B	*FEATURE SET*
-4.0%	Anchor Text-based only
1.2%	(Milne & Witten, 2008)
3.5%	Anchor Text and Collective-based
14.6%	All except Collective-based
15.3%	All features

6.7.6 Analysis of Collective Features & Iterative Classification

This subsection further explores the effect of collective features, and analyzes the role of the iterative classifier defined in Section 6.5. We hypothesize that the collective features would offer more predictive power when fewer (non-collective) features are present that reduce the need for collective insight into the concepts mentioned in the text. Table 22 reports the accuracy attained by committing to the model created after each iteration, for two different feature spaces: all feature, or only Anchor-Text and Collective based ones. In summary, the collective features contribute more noticeably to the accuracy when fewer features are present (47.9%→50.2% versus 57.2%→57.3%).

[42] The weaker (-4.0%) performance when only anchor text-based features are used shows the importance of knowing which is the more popular concept.

Table 22 – 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%

6.7.7 Modeling the Multiple-TRUEs Decision

In Section 6.6, we defined a method to model the decision of which candidate to chose when more than one candidate is predicted as TRUE by the binary classifier. In this section, we compare the performance of the "*pick-largest-true-score*" heuristic approach and the performance attained by a second model trained[43] on the four features defined in Table 16. We also calculate the performance of making a random selection between the TRUE candidates. Table 23 summarizes the results.

In summary, the results suggest that the use of a trained model is no more accurate than the use of the "pick-largest-true-score". Some benefit is seen from making an informed choice over a random one.

Table 23 – Performance comparison between three different methods of selecting the final prediction when more than one candidate is predicted to be TRUE: highest likelihood rule, a trained as described in Section 6.6, and a random selection.

highest-likelihood	Trained Model	Random
57.3%	56.8%	55.6%

[43] For training this mention-level model we again used `SVMlight`.

6.8 Portability Analysis on `icdm09cma1`

Although when pipelined, the `SDOI` concept mention identification and linking algorithms have been shown to significantly outperform the baseline algorithms on the `kdd09cma1` corpus it would be helpful to assess whether once trained, the resulting model could be effectively applied outside of the training corpus. If it can be applied outside of its corpus then a significant amount of effort could be spared in creating a training set for the new corpus. In this section, we assess the portability of the models trained on `kdd09cma1` at identifying and linking concept mentions within `icdm09cma1` – a corpus from a difference source (different organizer and few overlapping authors) that is still within the data mining domain.

The accuracy of `SDOI` and the baseline algorithm with respect to the manually and collectively annotated abstract was analyzed and the results reported in Table 24. Performance on the second corpus is only slightly lower than the accuracy reported on the `kdd09cma1` corpus (reported earlier in Table 19) suggesting that the trained models are portable to other corpora.

Table 24 – Pipelined `SDOI` and baseline accuracy, trained on the *`kdd09cma1`*, and tested on the twenty-two ICDM'09 abstracts, for either exact or partially matched anchor text (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%

6.9 Time Savings Evaluation on `icdm09cma1`

We also use this opportunity to analyze `SDOI` from the perspective of its ability to aid in a real-world task of manual annotation. We explore whether `SDOI`'s predictions can reduce the amount of time required by a human annotator to manually annotate a document by working from a pre-annotated version (instead of the unprocessed version of the document). 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-impact journal publications to research ontologies, and of documented policy or agreements to official term definitions in business environments.

For this analysis, we selected the five human annotators who annotated three or more of the `kdd09cma1` abstracts. As they annotated each document, they would discover that the documents alternated from being pre-annotated and to being unprocessed[44]. We further asked the annotators to record the amount of time required for the three phases of identifying mentions, linking mentions, and revising their work (described in Section 4.3). Table 25 presents the averaged timing results.

[44] They were blind to whether their pre-annotated document used `SDOI` or the `MW08` pipelined solutions.

Table 25 – 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. Rather than spending approximately 34 seconds per annotated mention, they required 19 seconds per mention (44% less time). Interestingly, the use of the (Milne & Witten, 2008) based pipelined baseline resulted in no measureable savings in time spent. As much time was consumed by the annotator to fix the baseline's erroneously placed annotations than they saved from correct classifications. This extrinsic performance assessment suggests that SDOI has achieved sufficient accuracy to be of value on some tasks[45]. To the best of our knowledge, this type of timesavings evaluation has not been performed to date on a related task.

6.10 Conclusion

In this chapter, we presented the SDOI_{CML} supervised algorithm for the task of concept mention linking. We cast the multi-class problem to a binary one, propose a filtering heuristic to cope with the skew in FALSE labelled data, and propose a decision rule to enforce the constraint that a concept mention must only link to one concept in the ontology. We evaluated the algorithm on the

[45] Estimates of performance and relative benefits based on user-interaction studies are typically complicated by external factors. In our case, a different user interface may significantly reduce the observed benefit. As we noted in Section 4.3 however, the wiki-style environment used in our tests is widely used at present, such as in updates to Wikipedia.

kddcma1 dataset using a leave-one-document-out framework and demonstrated a lift in accuracy over (Milne & Witten, 2008) and naive baselines, on annotated and predicted concept mentions. SDOI achieved significantly higher accuracy on all assessments.

We further evaluated the models trained on the kddcma1 corpus on the icdm09cma1 corpus and noted that accuracy did not deteriorate, suggesting that in some cases, work on one corpus can be ported to another. Further, we used this opportunity to demonstrate that the models could significantly reduce the time required by an annotator to process a new document. While there remains a significant gap between human and algorithm annotation accuracy, our proposal presents a viable solution for the real-world application, and a principled solution for future comparison in the research arena.

We now turn to the final module of SDOI tasked with the identification of relations mentioned in the text.

7: SEMI-SUPERVISED RELATION MENTION IDENTIFICATION: $SDOI_{RMI}$

In this chapter, we switch from identifying and linking mentions of concepts to the final task defined in Chapter 2: of identifying mentions of relations. Recall that a "candidate relation mention" is any pair of concept mentions in a text, and the challenge of the task is to determine which pairs are contained in text that expresses a TRUE direct relation (of any type[46]) in the domain.

A further switch in this chapter is from the application of a fully-supervised classification algorithm to that of a semi-supervised one. This change in learning type is due to the absence of manually labelled (relation mention) data. Our task definition only requires labelled concept mention data. We propose the use of a labelling heuristic that exploits the information provided ontology: if a direct relation exists in the ontology between the two concepts then it labels the candidate as TRUE.

Finally, for the evaluation of task performance, we change the motivating real-world application from aiding with semantic annotation to aiding with ontology engineering. While automatically annotating text with relation mentions

[46] Recall from Chapter 2: that the type of the underlying relation does not play a role in our task definition of relation identification.

can be useful to applications such as: strategic reading[47], information retrieval, text summarization, textual entailment, and manual semantic annotation; our envisioned real-world application is to assist a knowledge engineer to process a new set of documents that present an opportunity to enhance the ontology. Instead of having to read meticulously through an entire new corpus, a knowledge engineer could instead receive a list of candidate relation mentions that are predicted to be TRUE relation mention candidates between concepts that do not (possibly yet) have a direct relation between them, and that is ranked according to some correctness likelihood score. With such a list, the knowledge engineer could dedicate more attention on comprehending the meaning of the passages that (very likely) contain high-quality relation mention candidates. For example, assume that the algorithm is requested to process the research paper (Chapelle & al, 2006) and, as a result produces, the following list[48]:

1) SUPERVISED LEARNING ALGORITHM + UNSUPERVISED LEARNING ALGORITHM
2) SEMI-SUPERVISED LEARNING ALGORITHM + SUPERVISED LEARNING ALGORITHM
3) SEMI-SUPERVISED LEARNING ALGORITHM + UNSUPERVISED LEARNING ALGORITHM

Based on the background knowledge the ontology engineer recognizes the possibility of an interesting direct relation between the concepts in one or (all three) pairings. They select (e.g. click) on one of the pairings and are taken to a passage in (Chapelle & al, 2006) that reads: "*Semi-supervised learning (SSL) is*

[47] Imagine, for example, a strategic reader selecting a concept mention of interest in a text and then seeing the related mentions become visually highlighted.

[48] The list presents the label for each linked concept mention. Note that this list assumes that the ontology does not yet contain direct relations between these concepts (possibly the concepts are still 'stubs').

halfway between supervised and unsupervised learning." Based on the passage and their background knowledge they could decide which of the three relation mentions are important enough to appear in the ontology. If they do enter a direct relation between the concepts into the ontology, the relation can be immediately linked to the supporting passage in (Chapelle & al, 2006). In this example, the three relations are rather commonplace in the data mining literature thus their appearance would suggest that the ontology contains many concept stubs.

The next two sections defined our proposed algorithm for relation mention identification: **SDOI**$_{\mathbf{RMI}}$, and then conclude with an empirical analysis of its performance on the datasets presented in Section 4.

7.1 Algorithm Overview

For the task of relation mention identification, we propose a semi-supervised algorithm inspired by the **TeGRR** text graph-based relation recognition algorithm proposed in (Melli & al, 2007). The algorithm first applies a labelling heuristic to unlabeled candidate relation mentions, and then trains a binary classification model. We were motivated to follow this approach used by **TeGRR** for the following reasons:

1) It is based on relation recognition approaches, such as (Jiang & Zhai, 2007), that achieve state-of-the-art performance (e.g. on benchmark tasks such as ACE[49]).

[49] ACE Relation Detection Recognition (RDR) task http://projects.ldc.upenn.edu/ace/annotation/

2) It is designed to recognize relation mentions that span beyond a single sentence (by the use of a text graph representation)

3) It exposes an extensible feature space (that can be extended with information drawn from our task's ontology).

4) It provides a natural path for the future support of tasks with labelled training data – possibly even labelled with the actual relation type.

One of the distinctive aspects of **TeGRR** is its representation of a document into a graph-based representation, where each concept mention or token in the text is mapped to an 'external' node in a graph, and which represents other syntactic and structural features of the text as internal nodes and edges between nodes. In Section 7.3 we define the text graph representation and its effect on the algorithm definition.

Given a document's text-graph, we can proceed to define a feature space for each relation mention candidate. Table 26 illustrates the structure of the training data and its feature space that we propose for $\mathbf{SDOI}_{\mathrm{RMI}}$. We divide the feature space into three information sources. An initial feature source is based on the shortest path between the concepts mentions, all of which have been proposed for **TeGRR** in (Melli & al, 2007). We also proposed to inherit the concept mention linking features (defined in Section 6) for each of the two concept mentions associated to a relation mention candidate. Finally, we also propose features that draw on information from the ontology. The formal feature space definition is found in Section 7.4

Table 26 – A high-level representation of training examples of a document's unique concept mention pairs (relation mention candidates). The label assignment procedure and the feature definitions are presented in the two coming subsections.

Relation Mention			TeGRR Text-Graph based	Concept Mention (CM) Linking based		Ontology based	label
doc_d	m_i	m_j		CM_a	CM_b		
							T
			Feature Space				F
							...

7.2 Label Assignment

Because of the lack of annotated relation mention data, the label assignment procedure for relation mention candidates differs significantly to the manually based assignment in the concept mention identification and linking tasks. We propose to use the ontology for the labeling decision. For each combination of concept mention pairs the heuristic automatically assign labels according to the following rule. If the concepts in the ontology associated with the relation mention share a direct internal link in the ontology in either direction then the training example is marked as TRUE; otherwise it is labeled as FALSE. For example, a candidate relation mention such as Semi-Supervised Learning Algorithm (SSL) + Supervised Learning Algorithm (SL) drawn from processing (Chapelle & al, 2006) would be labelled as TRUE if a direct link exists from, say, SSL to SL (and/or SL to SSL) in the ontology; while a candidate relation mention such as Semi-Supervised Learning Algorithm (SSL) + k-Nearest Neighbors Algorithm (kNN) would be labelled as FALSE if there is no direct link between the two concepts in the ontology.

This approach to labeling is similar to the one used by relation mention recognition task such as (Melli & al, 2007). In these tasks, annotators focus on identifying of TRUE relation mentions because these contain the information to be extracted into their target database. While, some candidate relation mentions they encounter are explicitly labeled as FALSE, the majority of the FALSE labeled relations are automatically generated by assuming closed-world assumption that all other combinations of concept mentions represent FALSE relations. This assumption can lead to erroneous assignments because, given the complexity of the annotation task, when not forced to decide whether a relation mention is FALSE, the annotator can more easily overlook TRUE relation mentions in the text.

Our proposal extends this automatic labeling approach for FALSE example labeling to also automatically label TRUE relation mentions. This approach is more likely to lead to erroneously mislabeled candidates. In many cases, the passages associated with a candidate relation mention that happens to refer to directly linked concepts in the ontology do not substantiate a direct semantic relation. In these cases, after reading the passage, an expert would instead conclude that a direct relation is not implied by the passage and would label the candidate relation mention as FALSE. Alternatively, the heuristic would label some relation mention candidates as FALSE simply because the relation did not yet exist in the ontology; while, upon manual inspection of the passage, the annotator would label the relation as a TRUE candidate.

Despite this recognition of noise in the generated labels, we hypothesize that this heuristic labeling approach provides a sufficient signal for the supervised

classification algorithm to detect many direct relation mentions with sufficient accuracy to be useful in some real-world tasks, such as ontological engineering.

7.3 Text Graph Representation

The **TeGRR** feature space is based on a graph representation of the document under consideration. The text graph representation is composed of the three types of edges: 1) Intra-sentential edges; 2) Sentence-to-sentence edges; and 3) Co-reference edges. Figure 7[50] presents a sample text graph that makes use of the three edge types.

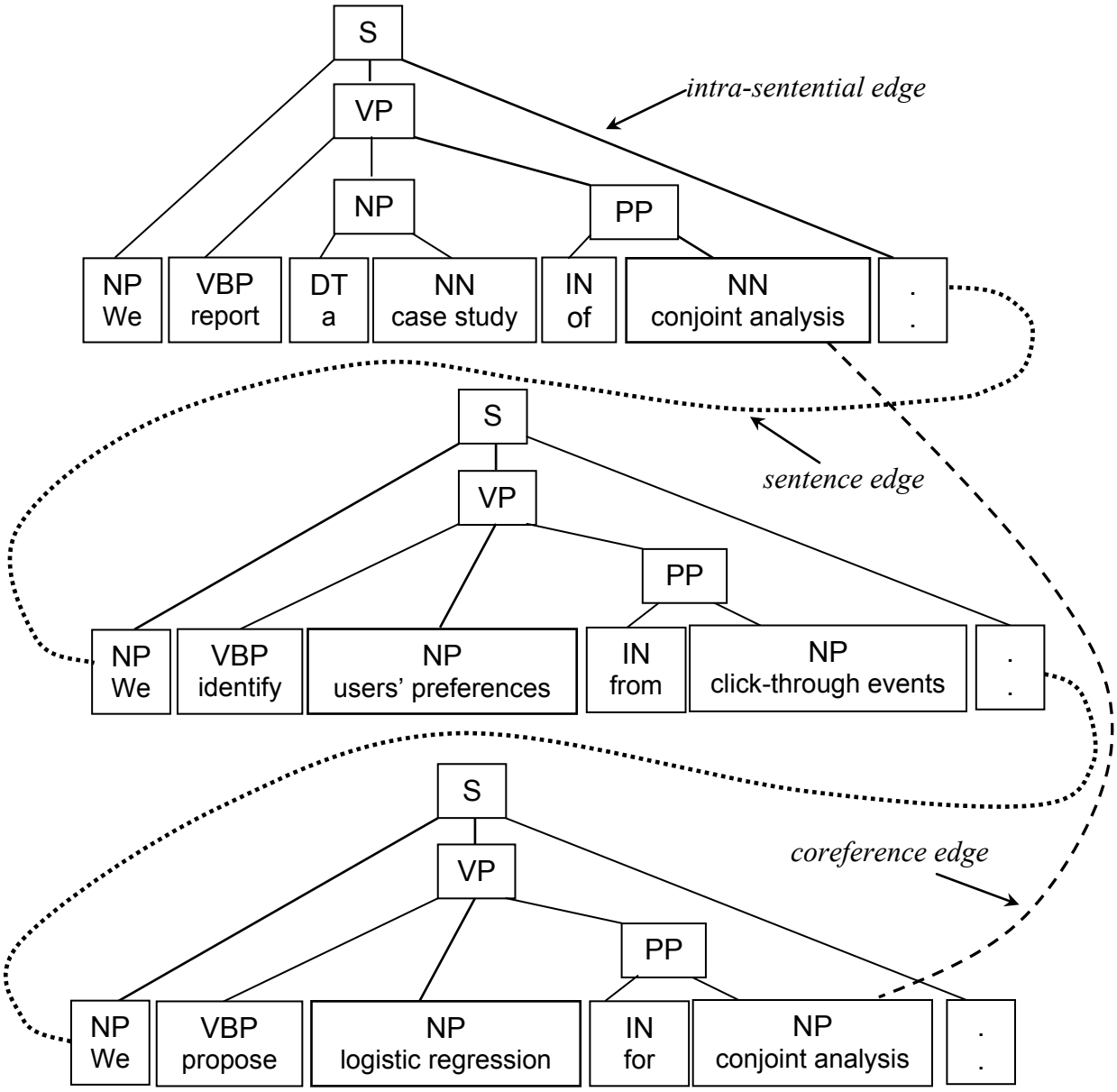
7.3.1 Intra-sentential Edges

Intra-sentential edges in a text-graph represent edges between nodes associated with tokens from the same sentence. In the literature, these edges can vary from being: word-to-word edges (Freitag & McCallum, 1999), shallow parsing edges (Zelenko & al, 2003), dependency parse tree edges (Suchanek & al, 2006), and phrase-structure parse tree edges (Zhang & al, 2006). We propose the use the phrase-structure parse tree as the source of intrasentential edges for two reasons. The choice of this data source over the others is the analysis by (Jiang & Zhai, 2007) that suggests that the phrase-structure parse tree is the best single source of information for relation detection. Secondly, all other proposed intra-sentential edge types can be derived, or approximated, from phrase-structure parse trees by means of transformations.

[50] Figure 1 in Section 1: can also help to visualize the three edge types.

A phrase-structure parse tree is composed of two types of nodes: leaf nodes and internal nodes. Leaf nodes (which map to our external nodes) are labelled with the text token (or concept mention), and with the part-of-speech role. Internal nodes contain the syntactic phrase-structure label.

Figure 7 - A sample of the text graph representation (for a highly summarized document) that $SDOI_{RMI}$ would use to create feature vectors for the talk of relation mention identification.



The text graph in Figure 7 contains 26 intrasentential edges connecting 12 internal nodes and 19 leaf nodes.

7.3.2 Sentence-to-sentence Edges

Edges in a text graph can also cross sentence boundaries. The first type of inter-sentential edge to be considered is the "sentence-to-sentence" edge that simply joins an end-of-sentence punctuation node with the first word of the sentence that follows. The intuition for this edge type is that a concept that is mentioned in one sentence can be in a semantic relation with a concept mention in the adjacent sentence, and that the likelihood of it being a relation increases as you reduce the number of sentences between the two entities. The text graph in Figure 7 contains two sentence-to-sentence edges.

7.3.3 Co-reference Edges

The other source of inter-sentential edges to be considered, also taken from (Melli & al, 2007), are based on concept mentions in the same document that are linked to (co-refer to) the same concept in the ontology. For example if "*hidden-Markov models*" is mentioned in one sentence, "*HMMs*" is mentioned in a subsequent one, and the pronoun "*they*" is used to refer to the concept further on in the document, then coreference edges would exist between "*hidden-Markov models*" and "*HMMs*", and between "*HMM*" and "*they*" (via the HIDDEN MARKOV MODELS concept). The intuition for this edge type is that concept mentions on separate sentences but that are near a some co-referent concept mention are more likely to be in a semantic relation than if that co-referent

mention did not exist. The text graph in Figure 7 contains a coreference edge between the mentions of to the CONJOINT ANALYSIS ALGORITHM (that were identified by $\mathbf{SDOI}_{CMI}$ and linked to the ontology by $\mathbf{SDOI}_{CML}$).

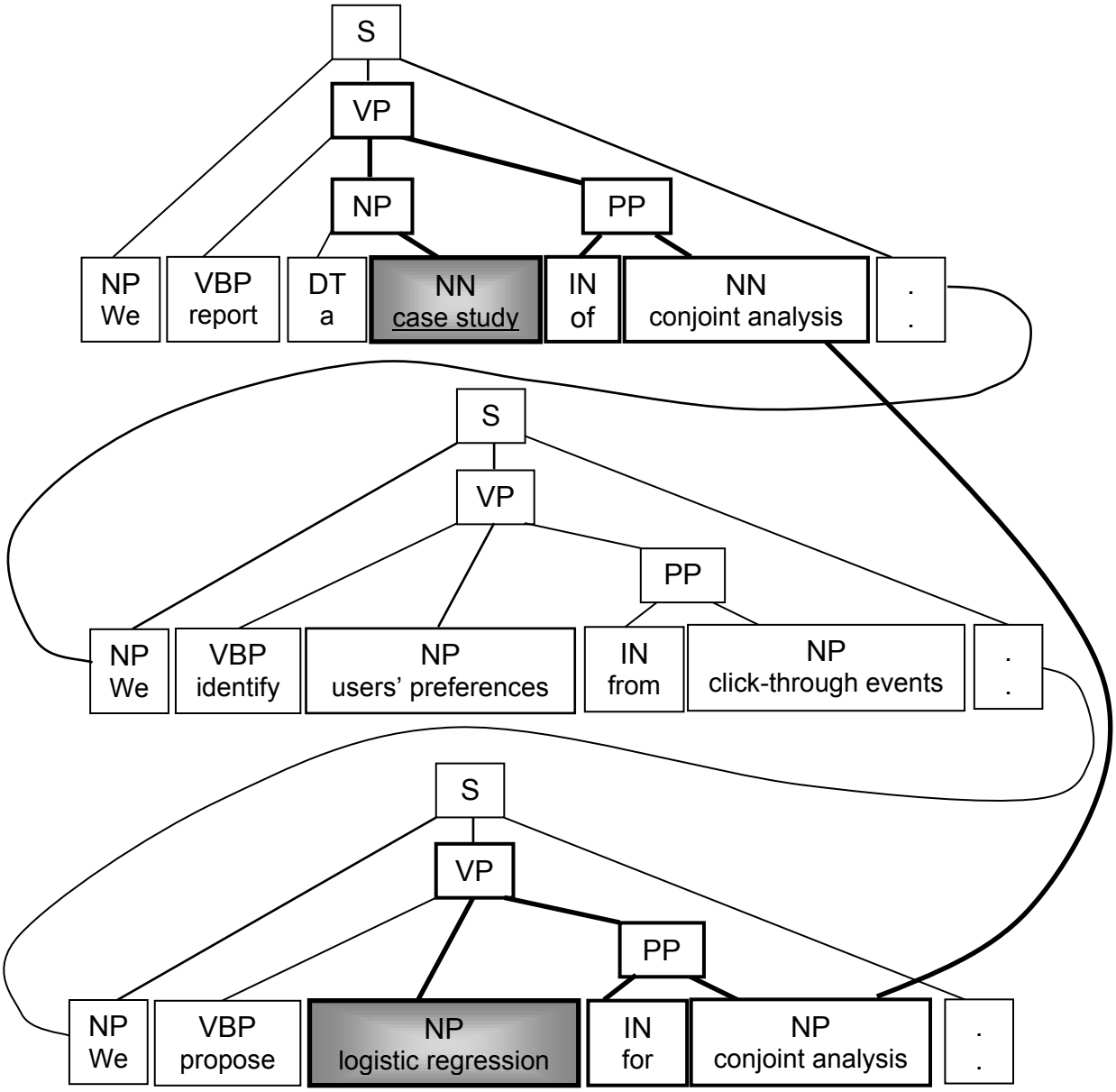
7.3.4 Text-Graph Properties

We describe properties of a text graph used to define $\mathbf{SDOI}_{RMI}$'s text-graph related features:

1) A test-graph is a connected graph: for every pair of nodes *n* and *v* there is a walk from *n* to *v*
2) A text-graph can be a cyclic graph, and such cycles must involve co-reference edges.
3) A text-graph, has at least one shortest path between any two nodes, *n* and *v*, and the number of edges between them is their *distance*.
4) A concept mention m_i is in a *p-shortest path* with concept mention m_j if there are only *p*-1 other concept mentions in a shorter shortest-path relation with m_i. The value of *p* can be interpreted as the rank of the proximity between the two concept mentions, e.g. 1st nearest, 2nd nearest, etc. If two alternate mention pairs are in equal *p-shortest path relation* then both are TRUE for the relation.
5) A path-enclosed subtree is the portion of the syntactic tree enclosed by the shortest-path between two leaf-nodes. This inner portion of a syntactic tree is predictive in relation extraction tasks (Jiang & Zhai, 2007).

Figure 8 highlights the nodes and edges that will be used in defining text-graph features for a candidate relation mention.

Figure 8 – An illustration of the subgraph based on the shortest-path between the two darkened concept mentions. The highlighted edges and nodes include the path-enclosed (IN preposition) tree nodes.



7.4 Relation Mention Identification Features

We begin the definition of the feature space with the text-graph based features that we retain from (Melli & al, 2007). We then proceed to describe the

ontology-based features, and conclude with the concept linking features inherited from the previous (concept linking) task.

7.4.1 Text-Graph based Features

This section describes the features that we directly inherit from `TeGRR`. We first describe the underlying text graph representation that is then used to define the associated features.

7.4.1.1 Path-Enclosed Shortest Path Features

From the path-enclosed shortest-path subgraph we identify all distinct subtrees with up to e edges as proposed in (Jiang & Zhai, 2007) to replicate the convolution-kernel approach of (Haussler, 1999). A feature is created for each possible *neighborhood* in the subgraph, where a neighborhood is defined by a subtrees with e edges, where e ranges from zero through to some upper limit on edges: $e \in [0, e_{max}]$. We retain the e proposed in (Jiang & Zhai, 2007) of e_{max}=2. Subtree-based features associated to the subtrees of size zero (e=0) simply summarize the number of nodes of a certain content type in either the entire relation mention graph, or one of its pairings. For example, one feature would count the number of **NP** (Noun Phrase) nodes in the relation mention graph, while another feature would count the number of times that the word "*required*" is present. Subtree-based features associated to the subtrees of size e>0 represent the number of times that a subgraph with e edges appears within the subgraph. For example, one feature would count the number of times that the triple IN – PP – NP appears in the graph.

7.4.1.2 Sentence Count:

This feature informs the classifier about the number of sentences that intervene between concept mentions. For example, the number of intervening sentences between the “case study” and “logistic regression” mention in the relation mention in Figure 8 is two (2) sentences. This information will help the classifier adjust its predictions based on the separation. The further apart the less likely that a relation mention is true.

7.4.1.3 Intervening Concept Mentions:

This pair of features informs the classifier about the number of concept mentions that intervene between two concept mention pairs. For example, in Figure 8 “*conjoint analysis*” is counted as one intervening concept mentions between “*case study*” and “*logistic regression*”. This information will help the classifier adjust its predictions based on how many other concept mention candidates exist; the greater then number of intervening concept mentions the less likely that a semantic relation between the two concept mentions is being stated.

7.4.2 Concept Mention Linking-based Features

A second source of features that we propose is to include the features for each concept mention defined for concept mention linking (in Section 6). We concatenate the two feature vectors in the following order: the concept mention that appears first in the text, followed by the other concept mention. These features provide signals of the context of each mention, such as even simply

what sentence it is locate on. In Figure 8, for example, the "*case study*" concept mention is located on the first sentence and the closer a mention is to the first sentence may affect the importance of the mention – as it can in other NLP tasks such as text summarization (Melli & al, 2006).

7.4.3 Ontology-based Features

We further propose four features based on information from the ontology – that differ from the ones inherited from the concept-mention linking task. These four features capture information signals from their pairing in the ontology: SHARED_OUTLINKS, SHARED_INLINKS, SHORTEST_GT1-EDGE_DISTANCE, and TF-IDF_CONCEPTS_SIMILARITY.

7.4.3.1 SHARED_OUTLINKS Feature

The SHARED_OUTLINKS feature counts the number of shared concept outlinks. The intuition for this feature is that two concepts that reference many of the same other concepts in the ontology are more likely to be themselves in a direct relation.

7.4.3.2 SHARED_INLINKS Feature

The SHARED_INLINKS feature counts the number of shared concept inlinks. The intuition for this feature is that two concepts that are referenced by many of the same other concepts in the ontology are more likely to be themselves in a direct relation.

7.4.3.3 SHORTEST_GT1-EDGE_DISTANCE Feature

The SHORTEST_GT1-EDGE_DISTANCE feature reports the shortest distance that is greater than one counts the number of edges that separate the two concepts. This feature is the one that introduces the risk of giving away the presence of a direct link between the two concepts in the candidate. An edge distance of one (1) versus any other edge distance would be a perfect predictor of the label. However, information about the distance of alternate paths can provide a signal that the two concepts should be (or are) linked.

7.4.3.4 TF-IDF_CONCEPTS_SIMILARITY Feature

The TF-IDF_CONCEPTS_SIMILARITY feature reports the TF-IDF bag-of-words similarity between the two concept descriptions in the ontology. The intuition is similar to that of the “Shared Outlinks” feature: two concepts that reference many of the same words are more likely to be themselves in a relation. Unlike the “Shared Outlinks” feature however, this feature normalizes for very common and uncommon words.

7.4.4 Corpus-based Features

A final source of information for features that we propose is the training corpus itself. As with the corpus-based features for concept linking (defined in Section 6.4.5), the use of cross-validation for performance estimation requires that the document associated with the training record does not inform these features. For this feature, the count is on “other” documents.

7.4.4.1 RELATION_MENTION_OTHER_DOC_COUNT Feature

The RELATION_MENTION_OTHER_DOC_COUNT feature counts the number of other documents in the corpus that contain the pair of linked concept mentions. For example, if one other document contains the two linked concept mentions (and thus contains the same candidate relation mention) this feature is set to one (1).

7.5 Empirical Evaluation of Relation Mention Identification

In this section, we empirically evaluate the performance of the proposed relation-mention identification algorithm: $\texttt{SDOI}_{\text{RMI}}$. For this evaluation, we again used the `SVMlight`[51] package with its default parameter settings, as the underlying supervised classification algorithm. For the syntactic parse trees, we use Charniak's parser[52] (Charniak, 2000).

7.5.1 Evaluation Setup

Similar to evaluation of `SDOI`'s two other component algorithms for concept mention identification and linking, we use a leave-one-document-out method on the `kdd09cma1` corpus. For each unseen document, we predict which of its binary relation mention candidates (with linked concept mentions) already exist in the ontology. Those relations that do not exist in the ontology are proposed candidates for addition to the ontology.

[51] http://svmlight.joachims.org/
[52] ftp://ftp.cs.brown.edu/pub/nlparser/

A challenge associated with this task, as found in the concept-mention linking task, is the highly skewed distribution of the labels. In this case, we do not propose a filtering heuristic to change the training data. Instead, we propose an algorithmic change by tuning `SVMlight`'s cost-factor parameter that multiplies the training error penalty for misclassification of positive examples. We set aside three documents to tune the parameter, and based on an analysis to optimize F1 we set the cost-factor to 8.

7.5.2 Analysis of Automatically Generated Labels

We briefly analyze characteristic of the training, particularly the behaviour of the labelling heuristic on the `kdd09cma1` corpus. Table 27 lists some of the more common candidate relation mentions derived from the corpus and whether the labelling heuristic would set associated relation mentions to TRUE or to FALSE. Most of these common candidate relations do not have a direct relation between them in the ontology (they have an indirect relation via intermediary concepts). In an unseen document, such pairings will also likely be predicted as FALSE because there are so examples in training data that that suggest that there is no direct relation between them. Even if a passage, for example, reads "*... the algorithm was tested on the dataset ...*"

Table 27 – A sampling of the more commonly mentioned concept pairs mentioned in the `kdd09cma1` corpus; and, whether the kddo1 ontology contains a direct link between the two concepts.

Concept mentioned	Concept mentioned	Abstracts that mention the pair	Present in Ontology
Research Paper	Algorithm	61	FALSE
Research Paper	Experiment Outcome	33	FALSE
Task	Algorithm	32	TRUE
Algorithm	Experiment Outcome	29	FALSE
Research Paper	Data Set	29	FALSE
Algorithm	Data Set	28	FALSE
Research Paper	Task	24	FALSE
Data Value Set	Algorithm	23	FALSE
Algorithm	Real-World Dataset	21	FALSE
Research Paper	Real-World Dataset	21	FALSE
Research Paper	Efficient Algorithm	20	FALSE
Experiment Outcome	Algorithm	19	FALSE
Algorithm	Efficient Algorithm	18	TRUE
Task	Experiment Outcome	18	FALSE
Data Mining Task	Algorithm	17	FALSE
Research Paper	Experimental Evaluation	17	FALSE
Algorithm	State-of-the-Art-Algorithm	13	TRUE
Dataset	Real-World Dataset	11	TRUE
Synthetic Dataset	Real-World Dataset	11	TRUE
…	…	…	…
Clustering Algorithm	Cluster Set	8	TRUE
Classification Model	Accuracy Measure	8	TRUE
…	…	…	…

Table 28 presents some of the key statistics for the `kdd09cma1` from the perspective of relation mention candidates. The corpus contains 44,896 relation mention candidates. Of these, which quantifies the task's data skew, only 3.55% of the mention candidates are found in the ontology.

Table 28 – Key statistics of the number of binary relation mentions in the `kdd09cma1` corpus, per abstract and for entire corpus. The final row reports the total number of concept pairings where, at the document-level, pairs to the same two concepts are consolidated.

	Binary Relation Mention Candidates	Positive Candidates	Proportion
Minimum (per abstract)	42.0	1.0	0.88%
Average (per abstract)	322.1	11.5	3.86%
Maximum (per abstract)	1,582.0	4.3	12.50%
Entire corpus	44,896.0	1,593.0	3.55%
Entire corpus (only distinct relations)	34,181.0	1,080.0	3.16%

7.5.3 Baseline Algorithm(s)

The baseline algorithm that we compare $\texttt{SDOI}_{\texttt{RML}}$'s performance against on the relation-mention identification task is an unsupervised co-occurrence-based algorithm that predicts all permutations of linked concept mention pairs regardless of distance between them. This is the baseline algorithm compared against in (Melli & al, 2007, and Shi & al, 2007). We refer to this algorithm as **`AllTrue`**.

We also include as a baseline a version of $\texttt{SDOI}_{\texttt{RML}}$ with a restricted feature space that contains the features originally proposed for `TeGRR`.

7.5.4 Intrinsic Performance Analysis

Table 29 presents the results of the leave-one out performance analysis. $\texttt{SDOI}_{\texttt{RML}}$ outperforms the baseline algorithm in terms of precision and F1. The proposed feature space for `SDOI` also outperforms the original feature space proposed for `TeGRR`.

Table 29 – Leave-one-out performance results on the relation mention identification task on the kdd09cma1 corpus (excluding the three tuning abstracts) by `SDOI`, `SDOI` with its feature space restricted to those originally proposed for `TeGRR` in (Melli & al, 2007), and the `AllTrue` baseline.

Algorithm	Feature Space	Precision	Recall	F1
SDOI	All	18.2%	24.3%	20.8%
	TeGRR	7.7%	41.8%	13.0%
AllTrue		3.7%	100.0%	7.1%

7.5.5 Extrinsic Performance Analysis

We analyze the performance on a real-world usage scenario where an ontology engineer receives the generated list of relation mention candidates predicted as TRUE for being a direct link, which upon inspection of the ontology does not exist. We manually analyzed the top 40 predicted relation mention candidates proposed for insertion into the `kddo1` ontology ranked on their likelihood score[53]. Table 30 reports a snapshot of these relation candidates. Of the 40 candidates 31 (77.5%) were deemed candidates for insertion into the ontology[54]. Given the high proportion of relation candidates worthy of insertion, this result illustrates some benefit to the ontology engineer.

[53] We used SVMlight's real-number predictions, and did not boost the selection based on whether more than two documents resulted in predictions for the concept pair.

[54] This task-based result is likely dependent on the maturity of the ontology.

Table 30 – A sample of candidate relations (and their source document) with high likelihood score predicted by SDOI as candidates for addition to the kddo1 ontology. The table groups candidates that refer to the same concept pairs.

Score	Binary Relation		Document
	Concept A	Concept B	
20.873	Computing System	Algorithm	doi:10.1145/1557019.1557112
…	…	…	…
15.975	Computing System	Algorithm	doi:10.1145/1557019.1557144
23.584	Conditional Probability	Marginal Probabilty	doi:10.1145/1557019.1557130
22.345	Conjoint Analysis	User Preference	doi:10.1145/1557019.1557138
22.075	Optimization Task	Gradient Descent Algorithm	doi:10.1145/1557019.1557129
20.349	Optimization Task	Gradient Descent Algorithm	doi:10.1145/1557019.1557100
21.788	Set	Pattern	doi:10.1145/1557019.1557071
19.849	Set	Pattern	doi:10.1145/1557019.1557077
21.047	Training Dataset	Performance Measure	doi:10.1145/1557019.1557144

7.5.6 Boostrapping Experiment

In practice, a common method of applying self-labelled learning is to treat the labelling heuristic as a means to seed a bootstrapped process where subsequent rounds of labelling are based on the most confident predictions by the newly trained model (Chapelle & al, 2006). Generally, evaluations of this approach have assumed high-accuracy seed labels - either from a small manually curated training set, such as in (Agichtein & Gravano, 2000), or with high-accuracy labelling patterns, such as in (Yarowsky, 1995). Each iteration sacrifices some precision for additional recall performance. In our case a bootstrapped process does not begin with high precision to sacrifice, because of our labelling heuristic does not start with high-precision predictions. However, we performed a bootstrap experiment by iteratively selecting the 10% of relation mentions that were predicted to be TRUE with the highest likelihood score, and

then labelled these candidates as TRUE in the subsequent iteration (even if no direct link existed in the ontology for the corresponding concept pair).

F1 performance dropped with each iteration. Some analysis can show that this deterioration in performance is unavoidably built into the process: with each iteration the supervised classifier trained models that were based on the increasingly false assumption that TRUE labelled training data were representative of direct links in the ontology. Ensuing models would begin to predict links that were by definition not in the ontology and would thus be evaluated as false positives.

Thus, we again manually inspected the top 40 predicted relations for the first two iterations. The precision dropped after each iteration. After the first iteration, 29 (72.5%) candidates were correct, and after the second iteration, 21 (52.5%) candidates were correct. During the manual review, we observed that predictions in subsequent iterations began to include some of the more common FALSE pairings listed in Table 27. Bootstrapping of $\texttt{SDOI}_{\texttt{RML}}$ does not improve the precision of the reported predictions, on the `kdd09cma1` benchmark task.

7.5.7 Observations

We include some observations of the original set of predictions reported in Table 30 of the leave-one-out evaluation on the `kdd09cma1` corpus The table includes some promising candidates for addition to the ontology. For example, because of this experiment we noted that the obvious missing direct relation

between a COMPUTING SYSTEM and an ALGORITHM[55]. The table also includes a more nuanced missing direct relation missing in the ontology between CONDITIONAL PROBABILITY and MARGINAL PROBABILITY[56].

Next, we observe that suggested relation mention candidates whose concept pairs are predicted within more than one document, such as COMPUTING SYSTEM + ALGORITHM, may be more indicative that the direct relation is indeed missing from the ontology than when only supported by a single document. However, as counter-evidence, some of the repeated pairs in Table 30 appear to be listed simply due to their frequent occurrence in the corpus. For example, the candidate relation between the concepts of SET and of PATTERN may simply be due to documents (abstracts) that often mention "*sets of patterns*". We would not expect the SET concept to be directly linked to every concept in the ontology that can be grouped into a set. This example however does suggest that PATTERN + SET may be a common and important concept in the data mining domain to deserve the addition of a PATTERN SET concept into the ontology. We note further that very frequent candidates reported in Table 27, such as RESEARCH PAPER + ALGORITHM, were not predicted; likely because the algorithm recognized that if such a commonplace relation is always false then it likely will be false in a new/unseen document. Thus, there is some evidence that the number of repetitions can indeed signify a more likely candidate. As future work, it would be

[55] The direct relation can naturally added in both directions "*an ALGORITHM can be implemented into a COMPUTING SYSTEM*" and "*a COMPUTING SYSTEM can implement an ALGORITHM*."

[56] Based on passage "…*assumption made by existing approaches, that the marginal and conditional probabilities are directly related...*." From [doi:10.1145/1557019.1557130] and due to the fact that, in `kddo1`, the two concept descriptions are briefly described (stubs).

worthwhile to attempt to train a second classifier that can use the number of referring documents as a feature[57].

A separate challenge that we observe from the predictions in Table 30 is illustrated by the OPTIMIZATION TASK + GRADIENT DESCENT ALGORITHM entry. While this seems like a reasonable candidate for addition at first glance, these two concepts are more likely indirectly related via the OPTIMIZATION ALGORITHM concept (*an* OPTIMIZATION TASK *can be solved by an* OPTIMIZATION ALGORITHM; *a* GRANDIENT DESCENT ALGORITHM *is an* OPTIMIZATION ALGORITHM.). The resolution of these situations could require additional background knowledge from the ontology, such as relation types, to inform the classifier that in some situations when the parents is linked to the concept then the child is not directly linked to.

Finally, the candidates reported in Table 30 that were considered to be candidates for the addition of a direct relation in the ontology illustrate the noise introduced by the labelling heuristic. When these candidates where used to train a model (when some other document was being held out) they were incorrectly labelled as FALSE. However, eventually, as the ontology becomes more complete, the labelling heuristic become more accurate. Still, it would be interesting as future work to perform a round of active learning (that repairs the labels on some of the most likely candidate relations from FALSE to TRUE) to investigate the algorithm's sensitivity to noisy labels.

[57] Although, as with the mention-level modeling that we performed on the linking task in Section 6.4, the classifier in this task may also already account for the frequency of the pair from, for example, the frequency of the pair which is signaled by the corpus-based features.

7.6 Conclusion

In this chapter, we presented the $\texttt{SDOI}_{\text{RMI}}$ semi-supervised algorithm for the task for relation mention identification in which the underlying concept mentions have been identified and linked to an ontology. To overcome the lack of annotated data, we propose a labelling heuristic based on information extracted from the ontology. We evaluated the algorithm on the `kdd09cma1` dataset using a leave-one-document-out framework and demonstrated an increase in F1 in performance over a co-occurrence based ALLTRUE baseline algorithm. An extrinsic evaluation of the predictions suggested a worthwhile precision on the more confidently predicted additions to the ontology, and could thus contribute to some real-world scenarios.

8: CONCLUSION AND FUTURE WORK

In this thesis, we introduced the task of document to ontology interlinking and proposed a composite supervised learning-based solution for it. Specifically, given a training corpus whose concept mentions are identified and linked to a given ontology, the solution trains predictive models that will, on unseen text: 1) identify the mentions of concepts that (could) exist in the ontology; 2) link these mentions to their referent concept in the ontology (if it exists); and, 3) identify the mentions of relations in the text that exist (or could exist) as direct relations in the ontology. In this chapter, we summarize the contributions of the thesis, and discuss some promising future research directions that address some of the known limitations.

8.1 Contributions

Our main contributions in this thesis are: a formal task definition that can ground the research area; a principled process SDOI composed of supervised learning algorithms ($\texttt{SDOI}_{\texttt{CMI}}$, $\texttt{SDOI}_{\texttt{CML}}$, and $\texttt{SDOI}_{\texttt{RMI}}$) that can become de facto baselines; their empirical evaluation and demonstration of benefit on intrinsic performance measures and on real-world tasks; and finally, a publicly available benchmark dataset that others can evaluate their solutions against and that can continue to grow and evolve. We now describe the contributions as they relate to each of the chapters in the thesis

In Chapter 2:, we formally define the task. One-day we believe that a substantial number of documents will be annotated in this manner; and, a formal task definition enables other research to focus on solutions by that can be constructively compared.

In Chapter 3, we review some of the literature that relates to the topic. Several diverse fields have addressed tasks related to ours, such as in natural language processing and information extraction. The most similar solutions that we review are the Wikipedia-centric proposals of (Milne & Witten, 2008) and (Kulkarni & al, 2009).

In Chapter 4, we describe the `kddo1` data mining ontology and the `kdd09cma1` and `icdm09cma1` annotated corpuses that we manually created to evaluate the task. This data may become a benchmark for future research. We do not know of another similar dataset within the field of Computing Science. Looking ahead, the ontology and corpora may continue to evolve if more conference proceeding abstracts are annotated and the ontology continually aligned.

In Chapter 5, we define a concept-mention identification algorithm that applies supervised sequential tagging classifier. Unlike prior solutions that use a dictionary-based longest-matching sequence heuristic, $\texttt{SDOI}_{\texttt{CMI}}$ will identify concept mentions from the domain that are not necessarily yet represented in the the ontology.

In Chapter 6, we define a concept mention linking algorithm that casts the multi-class problem into a binary one and defines a filtering heuristic to counteract the skew in FALSE labelled mentions. To support collective features, SDOI_{CML} performs iterative classification that circumvents the problematic transition in feature behaviour from the first to second iteration in (Neville & Jensen, 2000). In the evaluation phase the algorithm is shown to significantly outperform a reimplemented version of (Milne & Witten, 2008), and to also reduce by 44% the time required by an annotator to manually annotate a document.

Finally, in Chapter 7, we define a relation-mention identification algorithm SDOI_{CMI}, that applies a labelling heuristic based on information drawn from the ontology, and that extends the supervised binary classification approach proposed in (Melli & al, 2007) by adding features drawn from the corpus and ontology. For this phase, we demonstrated benefit in the real-world task of ontology enhancement based on the information in new documents.

8.2 Limitations and Future Work

While this thesis has made a significant contribution to the topic of interlinking documents and ontologies, as with most research that involves natural language, a significant amount of research lies ahead to achieve human levels of performance. To conclude, we discuss several of the research directions that we believe hold some promise in advancing the state-of-the-art in supervised document to ontology interlinking.

8.2.1 Performance Improvement through Pipeline Feedback

A promising method for performance improvement is through the introduction of feedback in the pipeline so that the predictions of the later stages of the pipeline, particularly the more confident predictions, can assist the decisions of the earlier stages. For example, in deciding between which of two overlapping anchor texts to identify as a concept mention, the anchor text that results in a more confident link to the ontology may be more likely to demarcate the correct anchor text; but `SDOI`'s concept mention identification algorithm currently does not have access to this information. This adjustment could help to reduce the gap between the exact matching performance and the partial match performance because the algorithm will make better decision about where to divide a multi-token concept mention. One possible means to create this feedback is to use an iterative classification method, as we did for the collective features of linking task, and then include the resulting likelihood score as a feature of the concept-mention identification task. Another method to more tightly couple the pipeline is through the use of joint inference models that model all decisions at once (Finkel & al, 2006; Miller & al, 2000).

8.2.2 Performance Improvement through Long-Range Features

Another promising direction for performance improvement is by explicitly including predictor features that inform the classifier of long-range dependencies to the presence of other concepts in a document or ontology. In the case of the mention linking task, information about other similar anchor text in the document could aid in better division predictions, such as to capture mentions that are

acting as anaphors (Culotta & al, 2007). For example, the linking of a general concept mention such as "*algorithm*" to a concept in the ontology can be affected by information in the text that lies outside of the mention's text window, or by concepts within the ontology that lie beyond the candidate one.

8.2.3 Performance Improvement through Semi-Supervised Learning

A source of additional features, for any of the three tasks, are the frequencies of the anchor text associated with a concept mention candidate in the literature, such as made available from Google's n-gram dataset[58]. For example, the fact that the phrase "*semi-supervised algorithm*" matches 406 documents, or that "*semi-supervised learner*" matches 53 documents on Google Scholar's search service likely could inform $\texttt{SDOI}_{\texttt{CMI}}$ about the likelihood that either of those two phrases is the anchor text for some concept mention, especially if the terms had not been encountered before in the labelled corpus.

8.2.4 Performance Improvement via Active Learning

As we noted in the learning curve analysis of the mention identification performance, it appeared that gathering more annotated data is a clear route to improved performance. Further, we noted in the empirical analysis on the `icdm09cma1` corpus in Section 6.9 that the annotation task was performed more quickly when it was based on the pre-annotated version of the documents. Thus, a possible enhancement to the process is the application of active learning methods that leverage trained models to reduce the labelling requirements.

[58] http://googleresearch.blogspot.com/2006/08/all-our-n-gram-are-belong-to-you.html

Indeed, one of the research directions in the natural language processing community to improve classification performance is to use active learning where the annotator is presented with the items to annotate[59] (Tang & al, 2001), and could be applied to our task as well.

8.2.5 Clustering and Classification of Relations

A foreseeable extension to the task is to extract additional structure about the extracted relation mentions. Recall that the task currently only identifies the presence of an internal link in the ontology and does not characterize the type of identified relations. One interesting topic is the ability to cluster relation mentions in order for an ontology engineer to determine whether some new type of relation should be added to ontology. Likely, these discovered relations types would be domain specific ones such as HEADQUARTER-LOCATION or PROTEIN-SUBCELLULAR-LOCALIZATION, rather than the domain independent ones of IS-A and PART-OF. For the later ones it would be interesting to attempt the direct classification of the relation type. Ideally, this extension could be achieved without the requirement of annotated relation data.

8.2.6 Real-World Application

Finally, while we have demonstrated that our proposal can lead to improved real-world performance, it would be helpful to more fully qualify this behaviour. Currently, someone wanting to apply `SDOI` will find little guidance in this thesis as to whether their environment is a good candidate. We plan to make

[59] pages.cs.wisc.edu/~bsettles/active-learning/alnlp2010/ Active Learning for NLP Workshop

progress in this direction by integrating the solution into the submission process of future conferences, such as ICDM-2010 and KDD-2011, in order to have the authors themselves validate and correct the pre-annotated versions of their abstracts. Such an assisted annotation system would be a living example of the future of deeply interlinked documents and ontologies that we foresee.

■

REFERENCE LIST

[1] Eugene Agichtein, and Luis Gravano. (2000). Snowball: Extracting Relations from Large Plain-Text Collections. In: Proceedings of the 5th ACM International Conference on Digital Libraries (DL 2000). [doi>10.1145/336597.336644]

[2] Mohammad Al Hasan, Vineet Chaoji, Saeed Salem, and Mohammed Zaki. (2006). Link Prediction Using Supervised Learning. In: Proceedings of SDM-2006 Workshop on Link Analysis, Counter-terrorism and Security.

[3] Erin L. Allwein, Robert E. Schapire, and Yoram Singer. (2001). Reducing Multiclass to Binary: a unifying approach for margin classifiers. In: Journal of Machine Learning Research, 1. [doi>10.1162/15324430152733133]

[4] Florence Amardeilh, Philippe Laublet, and Jean-Luc Minel. (2005). Document Annotation and Ontology Population from Linguistic Extractions. In: Proceedings of the 3rd International Conference on Knowledge Capture (K-CAP 2005). [doi>10.1145/1088622.1088651]

[5] Michele Banko, Michael J. Cafarella, Stephen Soderland, Matt Broadhead, and Oren Etzioni. (2007). Open Information Extraction from the Web. In: Proceedings of the International Joint Conference on Artificial Intelligence (IJCAI 2007).

[6] Michele Banko, and Oren Etzioni. (2008). The Tradeoffs Between Open and Traditional Relation Extraction. In: Proceedings of the 46th Annual Meeting of the Association for Computational Linguistics (ACL 2008).

[7] Michele Banko, Michael J. Cafarella, Stephen Soderland, Matt Broadhead, and Oren Etzioni. (2007). Open Information Extraction from the Web. In: Proceedings of the International Joint Conference on Artificial Intelligence (IJCAI 2007).

[8] 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.

[9] Tim Berners-Lee, James Hendler, and Ora Lassila. (2001). The Semantic Web. In: Scientific American, 284(5).

[10] Indrajit Bhattacharya, and Lise Getoor. (2004). Iterative Record Linkage for Cleaning and Integration. In: Proceedings of KDD Conference (KDD 2004).

[11] Beata Beigman Klebanov, and Eli Shamir. (2005). Guidelines for Annotation of Concept Mention Patterns. Technical Report 2005–8, Leibniz Center for Research in Computer Science, The Hebrew University of Jerusalem, Israel.

[12] Mikhail Bilenko, Sugato Basu, and Mehran Sahami. (2005). Adaptive Product Normalization: Using Online Learning for Record Linkage in Comparison Shopping. In: Proceedings of the 5th IEEE International Conference on Data Mining (ICDM-2005).

[13] David M. Blei, and John Lafferty. (2007). A Correlated Topic Model of Science. In: Annals of Applied Statistics, 1(1). [doi>10.1214/07-AOAS114]

[14] Leo Breiman, Jerome H. Friedman, Charles J. Stone, and R. A. Olshen. (1984). Classification and Regression Trees. Chapman & Hall/CRC. ISBN: 0412048418

[15] Paul Buitelaar, Philipp Cimiano, Peter Haase, and Michael Sintek. (2009). Towards Linguistically Grounded Ontologies. In: Proceedings of the 6th European Semantic Web Conference (ESWC 2009). [doi>10.1007/978-3-642-02121-3_12]

[16] Paul Buitelaar (editor), and Philipp Cimiano (editor). (2008). Ontology Learning and Population: Bridging the gap between text and knowledge. IOS Press, ISBN: 1586038184

[17] Paul Buitelaar, Philipp Cimiano, Anette Frank, Matthias Hartung, and Stefania Racioppa. (2008). Ontology-based Information Extraction and Integration from Heterogeneous Data Sources. In: International Journal of Human-Computer Studies, 66(11).

[18] Razvan C. Bunescu, and Marius Pasca. (2006). Using Encyclopedic Knowledge for Named Entity Disambiguation. In: Proceedings of EACL Conference (EACL 2006).

[19] Aljoscha Burchardt, Nils Reiter, Stefan Thater, and Anette Frank. (2007). A Semantic Approach To Textual Entailment: System Evaluation and Task Analysis. In: Proceedings of the ACL-PASCAL Workshop on Textual Entailment and Paraphrasing

[20] Claire Cardie. (1997). Empirical Methods in Information Extraction. AI Magazine, 18(4).

[21] 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]

[22] Venkatesan T. Chakaravarthy, Himanshu Gupta, Prasan Roy, and Mukesh Mohania. (2006). Efficiently Linking Text Documents with Relevant Structured Information. In: Proceedings of VLDB Conference (VLDB 2006).

[23] Olivier Chapelle (editor), Alexander Zien (editor), and Bernhard Schölkopf (editor). (2006). Semi-Supervised Learning. MIT Press. ISBN: 0262033585.

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

[25] Nitesh Chawla, Nathalie Japkowicz, Aleksander Kolcz. (2004). Editorial: Special issue on learning from imbalanced data sets. In: ACM SIGKDD Explorations Newsletter, 6(1). [doi>10.1145/1007730.1007733]

[26] Corinna Cortes, and Vladimir N. Vapnik. (1995). Support Vector Networks. In: Machine Learning, 20(3). [doi>10.1007/BF00994018]

[27] Jeremiah Crim, Ryan McDonald, and Fernando Pereira. (2005). Automatically Annotating Documents with Normalized Gene Lists. In: BMC Bioinformatics 2005, 6(Suppl 1):S13.

[28] D. Alan Cruse. (1986). Lexical Semantics. Cambridge University Press. ISBN: 0521276438

[29] Silviu Cucerzan. (2007). Large-Scale Named Entity Disambiguation Based on Wikipedia Data. In: Proceedings of EMNLP-CoNLL-2007.

[30] Aron Culotta, Michael Wick, Robert Hall, and Andrew McCallum. (2007). First-Order Probabilistic Models for Coreference Resolution. In: Proceedings of the the North American Chapter of the Association of Computational Linguistics (NAACL 2007).

[31] Ido Dagan, Oren Glickman, and Bernardo Magnini. (2006). The PASCAL Recognising Textual Entailment Challenge. In: Lecture Notes in Computer Science, 3944. Springer. [doi>10.1007/11736790_9]

[32] Thomas G. Dietterich, and Ghulum Bakiri. (1995). Solving Multiclass Learning Problems via Error-Correcting Output Codes. In: Journal of Artificial Intelligence Research, 2.

[33] Michael Erdmann, Alexander Maedche, Hans-Peter Schnurr, and Steffen Staab. (2000). From Manual to Semi-automatic Semantic Annotation: About Ontology-Based Text Annotation Tools. In: Proceedings of the COLING 2000 Workshop on Semantic Annotation and Intelligent Content.

[34] Oren Etzioni, Michele Banko, Stephen Soderland, and Daniel S. Weld. (2008). Open Information Extraction from the Web. In: Communications of the ACM, 51(12). [doi>10.1145/1409360.1409378]

[35] Oren Etzioni, Michele Banko, and Michael J. Cafarella. (2006). Machine Reading. In: Proceedings of the 21st AAAI Conference (AAAI 2006).

[36] Oren Etzioni, Michael J. Cafarella, Doug Downey, Ana-Maria Popescu, Tal Shaked, Stephen Soderland, Daniel S. Weld, and Alexander Yates. (2005). Unsupervised Named-Entity Extraction from the Web: An Experimental Study. In: Artificial Intelligence, 165(1).

[37] Jérôme Euzenat, and Pavel Shvaiko. (2007). Ontology Matching. Springer. ISBN: 3540496114

[38] Jenny Rose Finkel, Christopher D. Manning, and Andrew Y. Ng. (2006). Solving the Problem of Cascading Errors: Approximate Bayesian Inference for Linguistic Annotation Pipelines. In: Proceedings of the 2006 Conference on Empirical Methods in Natural Language Processing (EMNLP 2006).

[39] Dayne Freitag, and Andrew McCallum. (1999). Information Extraction with HMMs and Shrinkage. In: Proceedings of the AAAI 1999 Workshop on Machine Learning for Information Extraction.

[40] Katrin Fundel, Robert Kuffner, and Ralf Zimmer. (2007). RelEx - Relation Extraction Using Dependency Parse Trees. In: Bioinformatics, 23(3).

[41] Johannes Fürnkranz. (2002). Round Robin Classification. In: The Journal of Machine Learning Research, 2. [doi>10.1162/153244302320884605]

[42] Lise Getoor, and Christopher P. Diehl. (2005). Link Mining: A survey. In: SIGKDD Explorations, 7(2). [doi>10.1145/1117454.1117456]

[43] Thomas L. Griffiths, and Mark Steyvers. (2004). Finding Scientific Topics. In: Proceedings of the National Academy of Sciences (PNAS), 101(Suppl. 1). [doi>10.1073/pnas.0307752101]

[44] Ralph Grishman, and Beth Sundheim. (1996). Message Understanding Conference - 6: A Brief History. In: Proceedings of COLING Conference (COLING 1996).

[45] Thomas R. Gruber. (1993). A Translation Approach to Portable Ontology Specifications. In: Knowledge Acquisition, 2(5).

[46] Rob Hall, Charles Sutton, and Andrew McCallum. (2008). Unsupervised Deduplication Using Cross-field Dependencies. In Proceedings of SIGKDD Conference (KDD 2008).

[47] Joseph Hassell, Boanerges Aleman-Meza, and I. Budak Arpinar. (2006). Ontology-driven Automatic Entity Disambiguation in Unstructured Text. In: Proceedings of the 5th International Semantic Web Conference (ISWC 2006). [doi>10.1007/11926078]

[48] David Haussler. (1999). Convolution Kernels on Discrete Structures. Technical Report UCSC-CLR-99-10, University of California at Santa Cruz.

[49] Graeme Hirst. (2004). Ontology and the Lexicon. In: Steffen Staab, and Rudi Studer (Eds.) Handbook on Ontologies. Springer. ISBN: 3540408347

[50] Chu-ren Huang (editor), Nicoletta Calzolari (editor), Aldo Gangemi (editor), Alessandro Lenci (editor), Alessandro Oltramari (editor), Laurent Prevot (editor). (2010). Ontology and the Lexicon: A Natural Language Processing Perspective. Cambridge University Press. ISBN: 9780521886598

[51] Jing Jiang, and ChengXiang Zhai. (2007). *A Systematic Exploration of the Feature Space for Relation Extraction.* In: Proceedings of NAACL/HLT Conference (NAACL/HLT 2007).

[52] Christian Jacquemin, and Didier Bourigault. (2003). Term Extraction and Automatic Indexing. In: Ruslan Mitkov, editor. The Oxford Handbook of Computational Linguistics. Oxford University Press. ISBN: 019927634X

[53] Valentin Jijkoun, Mahboob Alam Khalid, Maarten Marx, and Maarten de Rijke. (2008). Named Entity Normalization in User Generated Content. In: Proceedings of the 2nd workshop on Analytics for Noisy Unstructured Text Data (AND 2008:23-30).

[54] Sayali Kulkarni, Amit Singh, Ganesh Ramakrishnan, and Soumen Chakrabarti. (2009). Collective Annotation of Wikipedia Entities in Web Text. In: Proceedings of the 15th ACM SIGKDD Conference (KDD 2009). [doi>10.1145/1557019.1557073]

[55] John D. Lafferty, Andrew McCallum, and Fernando Pereira. (2001). Conditional Random Fields: Probabilistic Models for Segmenting and Labeling Sequence Data. In: Proceedings of the 18th ICML Conference (ICML 2001).

[56] Yuangui Lei, Victoria Uren, and Enrico Motta. (2006). SemSearch: A Search Engine for the Semantic Web. In: Proceedings of the 15th International Conference on Knowledge Engineering and Knowledge Management Managing Knowledge in a World of Networks (EKAW 2006). [doi>10.1007/11891451_22]

[57] Michael Lesk. (1986). Automatic Sense Disambiguation Using Machine Readable Dictionaries: How to tell a pine cone from a ice cream cone. *In: Proceedings of the Fifth International Conference on Systems Documentation, (SIGDOC 1986). [doi>10.1145/318723.318728]*

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

[59] Bernardo Magnini, Emanuele Pianta, Octavian Popescu and Manuela Speranza. (2006). Ontology Population from Textual Mentions: Task Definition and Benchmark. In: Proceedings of the ACL 2006 Workshop on Ontology Population and Learning.

[60] Christopher D. Manning, Prabhakar Raghavan, and Hinrich Schütze. (2008). Introduction to Information Retrieval. Cambridge University Press. ISBN: 0521865719

[61] Gary Marchionini. (1997). Information Seeking in Electronic Environments, Part 5. Cambridge University Press, ISBN: 0521586747

[62] Gabor Melli, and Martin Ester. (2010) Supervised Identification of Concept Mentions and their Linking to an Ontology.In: Proceedings of the 19th ACM International Conference on Information and Knowledge Management (CIKM 2010). *forthcoming*

[63] Gabor Melli. (2010). Concept Mentions within KDD-2009 Abstracts (kdd09cma1) Linked to a KDD Ontology (kddo1). In: Proceedings of the 7th International Conference on Language Resources and Evaluation (LREC 2010).

[64] Gabor Melli, and Jerre McQuinn. (2008). Requirements Specification Using Fact-Oriented Modeling: A Case Study and Generalization. In: Proceedings of Workshop on Object-Role Modeling (ORM 2008)

[65] Gabor Melli, Martin Ester, and Anoop Sarkar. (2007). Recognition of Multi-sentence n-ary Subcellular Localization Mentions in Biomedical Abstracts. In: Proceedings of the 2nd International Symposium on Languages in Biology and Medicine (LBM 2007).

[66] Gabor Melli, Yang Wang, Yudong Liu, Mehdi M. Kashani, Zhongmin Shi, Baohua Gu, Anoop Sarkar and Fred Popowich. (2005). Description of SQUASH, the SFU Question Answering Summary Handler for the DUC-2005 Summarization Task. In: Proceedings of Document Understanding Workshop at the HLT/EMNLP Annual Meeting (DUC 2005).

[67] 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). [doi>10.1145/1321440.1321475]

[68] Scott Miller, Heidi Fox, Lance Ramshaw, and Ralph Weischedel. (2000). A Novel Use of Statistical Parsing to Extract Information from Text. In: Proceedings of NAACL Conference (NAACL 2000).

[69] Scott Miller, Michael Crystal, Heidi Fox, Lance Ramshaw, Richard Schwartz, Rebecca Stone, Ralph Weischedel, and the Annotation Group. (1998). Algorithms that learn to extract information BBN: Description of the SIFT system as used for MUC-7. In: Proceedings of MUC-7.

[70] 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]

[71] Einat Minkov, Richard C. Wang, and William W. Cohen. (2005). Extracting Personal Names from Email: Applying named entity recognition to informal text. In: Proceedings of Empirical Methods in Natural Language Processing (EMNLP 2005). [doi>10.3115/1220575.1220631]

[72] Ruslan Mitkov, editor. (2003). The Oxford Handbook of Computational Linguistics. Oxford University Press. ISBN: : 019927634X

[73] Alexander A. Morgan, Zhiyong Lu, Xinglong Wang, et al. (2008). Overview of BioCreative II gene normalization. In: Genome Biology 2008, 9(Suppl 2):S3. [doi>10.1186/gb-2008-9-s2-s3].

[74] Katharina Morik, Peter Brockhausen, and Thorsten Joachims. (1999). Combining Statistical Learning with a Knowledge-based Approach - A case study in intensive care monitoring. In: Proceddings of the 16th International Conference on Machine Learning (ICML 1999).

[75] David Nadeau, and Satoshi Sekine. (2007). A Survey of Named Entity Recognition and Classification. In: Lingvisticae Investigationes, 30(1).

[76] Roberto Navigli, and Paola Velardi. (2006). Ontology Enrichment Through Automatic Semantic Annotation of On-Line Glossaries. In: Proceedings of the 15th International Conference on Knowledge Engineering and Knowledge Management (EKAW 2006). [doi>10.1007/11891451_14]

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

[78] Patrick Pantel. (2005). Inducing Ontological Co-occurrence Vectors. In: Proceedings of the 43rd Annual Meeting of the Association for Computational Linguistics (ACL 2005). [doi>10.3115/1219840.1219856]

[79] Patrick Pantel, Eric Crestan, Arkady Borkovsky, Ana-Maria Popescu, and Vishnu Vyas. (2009). Web-Scale Distributional Similarity and Entity Set Expansion. In: Proceedings of EMNLP Conference (EMNLP 2009).

[80] Patrick Pantel, and Marco Pennacchiotti. (2006). Espresso: Leveraging Generic Patterns for Automatically Harvesting Semantic Relations. In: Proceedings of the 44th Annual Meeting of the Association for Computational Linguistics (ACL 2006).

[81] Patrick Pantel, and Marco Pennacchiotti. (2008). Automatically Harvesting and Ontologizing Semantic Relations. In: (Buitelaar & Cimiano, 2008)

[82] Patrick Pantel, Deepak Ravichandran, and Eduard Hovy. (2004). Towards Terascale Knowledge Acquisition. In: Proceedings of the 20th International Conference on Computational Linguistics (COLING 2004). [doi>10.3115/1220355.1220466]

[83] Marco Pennacchiotti, and Patrick Pantel. (2006). Ontologizing Semantic Relations. In: Proceedings of the 44th Annual Meeting of the Association for Computational Linguistics (ACL 2006).

[84] Lawrence Reeve, and Hyoil Han. (2005). Survey of Semantic Annotation Platforms. In: Proceedings of the 2005 ACM Symposium on Applied Computing [doi>10.1145/1066677.1067049].

[85] Ryan Rifkin, and Aldebaro Klautau. (2004). In Defense of One-Vs-All Classification. In: The Journal of Machine Learning Research, 5.

[86] Sunita Sarawagi. (2008). Information Extraction. FnT Databases, 1(3), 2008.

[87] Sebastian Schaffert. (2006). IkeWiki: A Semantic Wiki for Collaborative Knowledge Management. In: 1st International Workshop on Semantic Technologies in Collaborative Applications (STICA 2006).

[88] Bernhard Schölkopf (editor), Christopher J. C. Burges (editor), and Alexander J. Smola (editor). (1999). Advances in Kernel Methods - Support Vector Learning. MIT Press. ISBN: 0-262-19416-3

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

[90] 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]

[91] Urvi Shah, Tim Finin, Anupam Joshi, R. Scott Cost, and James Matfield. (2002). Information Retrieval on the Semantic Web. In: Proceedings of the 11th International Conference on Information and Knowledge Management (CIKM 2002) [doi>10.1145/584792.584868]

[92] Zhongmin Shi, Anoop Sarkar, and Fred Popowich. (2007). Simultaneous Identification of Biomedical Named-Entity and Functional Relation Using Statistical Parsing Techniques. In: Proceedings of NAACL/HLT Conference (NAACL/HLT 2007).

[93] Len Shustek. (2010). An Interview with Ed Feigenbaum. In: Communications of the ACM, 53(6). [doi>10.1145/1743546.1743564]

[94] Steffen Staab (editor), and Rudi Studer (editor). (2009). Handbook on Ontologies (2nd edition). Springer Verlag. ISBN: 3540709991

[95] Mark Stevenson, and Yorick Wilks. (2003). Word-Sense Disambiguation. In: (Mitkov, 2003).

[96] Tomek Strzalkowski (editor), and Sanda M. Harabagiu (editor). (2006). Advances in Open Domain Question Answering. Springer. [doi>10.1007/978-1-4020-4746-6] ISBN: 978-1-4020-47

[97] Fabian M. Suchanek, Georgiana Ifrim, and Gerhard Weikum. (2006). Combining Linguistic and Statistical Analysis to Extract Relations from Web Documents. In: Proceedings of ACM SIGKDD International Conference on Knowledge Discovery and Data Mining (KDD 2006). [doi>10.1145/1150402.1150492]

[98] Min Tang, Xiaoqiang Luo, and Salim Roukos. (2001). Active Learning for Statistical Natural Language Parsing. In: Proceedings of the 40th Annual Meeting on Association for Computational Linguistics (ACL 2001)

[99] Thorsten Joachims. (1999). Making Large-Scale SVM Learning Practical. In: (Schölkopf & al, 1999).

[100] Erik Tjong Kim Sang, and Jorn Veenstra. (1999). Representing Text Chunks. In: Proceedings of EACL Conference (EACL 1999).

[101] Erik Tjong Kim Sang. (2002). Introduction to the CoNLL-2002 Shared Task: Language-Independent Named Entity Recognition. In: Proceedings of CoNLL 2002. [doi>10.3115/1118853.1118877].

[102] Katrin Tomanek, and Fredrik Olsson. (2009). A Web Survey on the Use of Active Learning to Support Annotation of Text Data. In: Proceedings of the NAACL HLT 2009 Workshop on Active Learning for Natural Language Processing.

[103] Tzong-han Tsai, Wen-Chi Chou, Shih-Hung Wu, Ting-Yi Sung, Jieh Hsiang, and Wen-Lian Hsu. (2006). Integrating Linguistic Knowledge into a Conditional Random Field Framework to Identify Biomedical Named Entities. In: Expert Systems with Applications: An International Journal, 30(1). [doi>10.1016/j.eswa.2005.09.072]

[104] Thanh Tran, Philipp Cimiano, Sebastian Rudolph, and Rudi Studer. (2007). Ontology-Based Interpretation of Keywords for Semantic Search. In: Proceedings of tje 6th International Semantic Web Conference and the 2nd Asian Semantic Web Conference (ISWC 2007 + ASWC 2007). [doi>10.1007/978-3-540-76298-0_38]

[105] Peter D. Turney. (2000). Learning Algorithms for Keyphrase Extraction. In: Journal of Information Retrieval, 2(4). [doi>10.1023/A:1009976227802]

[106] Victoria Uren, Philipp Cimiano, José Iria, Siegfried Handschuh, Maria Vargas-Vera, Enrico Motta, and Fabio Ciravegna. (2006). Semantic Annotation for Knowledge Management: Requirements and a survey of the state of the art. In: Web Semantics: Science, Services and Agents on the World Wide Web, 4(1). [doi>10.1016/j.websem.2005.10.002]

[107] Gang Wu, and Edward Y. Chang. (2004). Aligning Boundary in Kernel Space for Learning Imbalanced Dataset. In: Proceedings of the Fourth IEEE International Conference on Data Mining (ICDM 2004) [doi>10.1109/ICDM.2004.10106]

[108] Adam Wyner. (2010). Towards Annotating and Extracting Textual Legal Case Elements. In: Proceedings of the 4th Workshop on Legal Ontologies and Artificial Intelligence Techniques (LOAIT 2010).

[109] Dmitry Zelenko, Chinatsu Aone, and Anthony Richardella. (2003). Kernel Methods for Relation Extraction. In: Journal of Machine Learning Research, 3.

[110] Min Zhang, Jie Zhang, and Jian Su. (2006). Exploring Syntactic Features for Relation Extraction using a Convolution Tree Kernel. In: Proceedings of HLT Conference (HLT 2006).

[111] 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]