Ontology Express: Statistical and Non-Monotonic Learning of Domain Ontologies from Text

Norihiro Ogata [1] and **Nigel Collier** [2]

Abstract. Text mining has an important role to play in aiding experts to construct domain specific ontologies by highlighting the important classes, properties and relations that occur within large text collections. In this paper we propose a systematic framework for discovery of ontological types using typing information complemented with statistical filtering. Preliminary experiments are conducted on three corpora in the domain of molecular biology and results show that the top level types we obtain closely meet the intuitions and expectations of domain experts.

1 Introduction

A *domain-specific ontology* (DSO), i.e. a set of domain-specific classes, properties, relations and instances, is important for information retrieval, information extraction, and management of domain-specific knowledge [14]. However, contrary to a *generic* ontology, the manual design and construction of a DSO is a process which is complicated by at least two factors. Firstly, a DSO can rarely be constructed based on a simple model or theory, in contrast to a generic ontology which has many established models or theories such as *Roget's thesaurus*, WORDNET [6], and some ontological investigations, therefore domain experts must define the criteria of the classification and structuring. Secondly, domain experts need to maintain the structure of the DSO during expansions and revisions of the domain knowledge. This second factor occurs as a result of changes in the nature of understanding of the domain by the expert and/or the community to which the expert belongs. For these reasons in the Ontology Express project we advocate the use of text mining as part of a machine-aided ontology discovery strategy cycle. Our approach however is not based on ad-hoc rules but a formal theory of ontology construction involving the processing of texts in the target domain.

Various approaches have been previously adopted to construct ontological structures such as traditionally clustering [21], classification using TFIDF [26], classification by linguistic patterns [2] and classification to a pre-defined framework such as [15]. However domain-specificity prevents experts from designing and constructing ontologies easily as they are forced to discover classes for entities, their names, and their interrelations that are special to each domain. In contrast to the above approaches our method concentrates on the *typing information* in a collection of domain-specific texts, which provides an explicit or implicit classification and structuring of domain-specific entities and concepts, and which can be formally specified using a constructive type-theory [16]. An advantage of our method is that it can discover not only semantic classes but also the names of semantic classes, treated as names of types, and easily combine background knowledge from domain experts, through *typing inference*. The basic extraction of typing and subtyping information is based on linguistic pattern matching which has been partially employed before in some projects such as SYNDIKATE [8], which uses apposition and exemplification patterns, and [9, 12, 19]. Most of these approaches require extensive post-processing to avoid poor precision [22, 3]. Additionally our method also aims to aid domain experts' maintenance of ontological structures by extracting unknown terms, constructing inter-term identities and inter-term associations, as well as integration of extracted information about terms of experts' background knowledge or pre-reference by *nonmonotonic inference of subtypes*.

Our overall goal in the Ontology Express project is to provide an enhancement to Open Ontology Forge (OOF) [5], an integrated ontology construction, content annotation and information extraction tool, to aid experts in synthesizing their intuitions about domain concepts, properties and relations. Ontology construction proceeds from a top-down specification of a core DSO by the domain expert supported by Ontology Express which discovers candidate concepts, relations and properties from a document collection. These candidates are filtered in two stages: firstly by statistical inference and then by non-monotonic reasoning. The expert then mediates in the ontology building process by validating the filtered concepts/relations/properties by attaching them to any existing part of the core DSO. In this paper we outline our initial experiments in the first of these filtering steps. Population of the ontology at the terminological level (i.e. finding instances of classes) is then expected to be done based on supervised machine learning from labelled examples by the expert which is already available in OOF.

At this stage we do not foresee full automation of the deeper aspects of DSO formation but rather an incremental approach that involves the expert in making the important decisions. The exception to this is in the acquisition of instances of classes and property values where significant progress has already been made by domain-based information extraction, e.g. [20, 1, 17, 4, 18, 23] This limitation to semi- rather than full-automation is necessary as much important information about the structure of a domain could be described as *common sense domain knowledge*, rarely given explicitly at the surface text level. Although this goal may seem modest we believe that computer-aided discovery can provide real benefits in helping experts make sense of the mass of evidence available to them.

[1] Faculty of Language and Culture, Osaka University email:ogata@lang.osaka-u.ac.jp

[2] National Institute of Informatics, 2-1-2 Hitotsubashi, Chiyoda-ku, Tokyo 101-8430 Japan email: collier@nii.ac.jp

2 Typing and Subtyping Information in Texts

We call *typing information* a bit of information which can be expressed in the form of 'x is of type of y'. Typing information which can be found in texts is classified as:

(i) typing information about *taxonomy*, i.e., the set of type names with a subtype structure and token names,

(ii) typing information about *mereology*, i.e., the set of type names and token names with a *informative* part-whole relation,

(iii) typing information about *synonymy*, i.e., the set of type names and token names with an equivalence relation.

(iv) typing information about *trope* and *trope type theory*, i.e., the set of trope type names and trope names.

(v) typing information about *eventuality* and *eventuality type theory*, i.e., the set of eventuality type names and eventuality names.

In this paper we focus mainly on taxonomic typing information due to the central position of taxonomy in ontology acquisition and we reserve discussion on other types for future work. The basic application though of statistical filtering and nonmonotonic reasoning are common to all types.

Typing information about taxonomy is contained in the following constructions and discourses:

(a) *definition* or *elaboration*

- taxonomic copula sentences – e.g., 'token name *is a* type name',
- naming constructions – e.g., '*a* type name *named/termed/called/designated* token name',
- appositions (e.g., 'type name+token name' where + is usually a blank),
- descriptive anaphoras – e.g., '*the* type name' (1a),

(b) *exemplification*

- exemplification constructions (e.g., 'token name *and other* type name,' 'type name *like/such as* token name'),
- exemplification sentences – e.g., '*Among* type name, token name *VP*.' see (1c),
- exemplification discourses (e.g., (1b)),

(c) *exception*

- exception constructions – e.g., type name *except/other than* token name',

(1) a. The rest of the technology sector scored gains amid good news about leading companies like *Intel*$_{\text{tokenname}}$. The *chip maker*$_{\text{typename}}$ gained 1-7/8 to 74-1/2.

b. In another market sector, *banking shares*$_{\text{type}_1}$ continued to *slide*$_{\text{type}_2}$ after two days of investor punishment. *Citicorp*$_{\text{token}_1}$ *lost*$_{\text{subtype}_2}$ 4-3/8 to 117-3/4, *Chase Manhattan*$_{\text{token}_1}$ *fell*$_{\text{subtype}_2}$ 4-3/8 to 104, *BankAmerica*$_{\text{token}_1}$ *tumbled*$_{\text{subtype}_2}$ 1-5/8 to 66, and *Dow component J.P. Morgan*$_{\text{token}_1}$ *shed*$_{\text{subtype}_2}$ 3-15/16 to 108-1/16 as investors worried about the sector's exposure to Asia.

c. Among other *technology stocks*$_{\text{type}}$, *IBM*$_{\text{token}}$ rose 3/8 to 96-7/8.

d. *Adaptec*$_{\text{tokenname}}$, *which makes computer subsystems*$_{\text{type}}$, software and chips, late Thursday warned that its fiscal third-quarter earnings would come in well below market expectations.

From (1a), we can extract typing information: chip maker:type and Intel:chip maker; from (1b), banking share:type, Citicorp:banking share, Chase Manhattan:banking share, BankAmerica:banking share, Dow component J.P. Morgan:banking share, sliding:type, losing:type, tumbling:type, falling:type, shedding:type, subtype(losing,sliding), subtype(falling,sliding), subtype(tumbling,sliding), and subtype(shedding,sliding); from (1c), technology stock:type and IBM:technology stock; and from (1d), maker of computer subsystems:type, which needs the transformation of present-tense verbs to their agentive derived nominals, and Adaptec:maker of compter subsystems.

Moreover, we call the information "x is a (proper) subtype of y" *subtyping information*, as in (2):

(2) a. Chipmakers are companies.

b. **subtype(chipmaker,company)**

(2a) has subtyping information represented by (2b). (2a) are treated as logical form '$\forall x.chipmaker(x) \rightarrow company(x)$' or in dependent type theory as proposition '$\Pi x : chipmaker.company(x)$'. That is, subtyping is not incorporated in the type theories of natural language. Furthermore, natural language has the notion of *proper subtyping* which is distinguished from the notion of subtyping. The subtyping information is typically contained in sentences such as 'Snakes are reptiles,' whereas the proper subtyping information is contained in the *taxonomic singular generic sentences* such as 'The snake is a reptile.' The difference between them is whether it is reflexive or not, as in (3):

(3) a. (singular reverse subtyping) The reptile is a snake. $\mapsto$ $\mathbf{subtype}(\mathbf{reptile}, \mathbf{snake})$, while singular proper subtyping is possible: e.g., The snake is a reptile (in generic sense). $\mapsto$ $\mathbf{properSubtype}(\mathbf{snake}, \mathbf{reptile})$

b. (singular proper subtyping) #The snake is a snake. (in generic sense) $\mapsto$ $\neg\mathbf{properSubtype}(\mathbf{snake}, \mathbf{reptile})$

c. (singular proper subtyping) Snake is a reptile. $\mapsto$ $\mathbf{properSubtype}(\mathbf{snake}, \mathbf{reptile})$

d. (singular proper subtyping) #Snake is a snake. $\mapsto$ $\neg\mathbf{properSubtype}(\mathbf{snake}, \mathbf{snake})$

e. (plural subtyping) Snakes are reptiles. $\mapsto$ $\mathbf{subtype}(\mathbf{snake}, \mathbf{reptile})$

f. (plural subtyping) Snakes are snakes. $\mapsto$ $\mathbf{subtype}(\mathbf{snake}, \mathbf{snake})$

g. (plural proper subtyping) Snakes are a reptile. $\mapsto$ $\mathbf{properSubtype}(\mathbf{snake}, \mathbf{reptile})$

h. (plural proper subtyping) #Snakes are a snake. $\mapsto$ $\neg\mathbf{properSubtype}(\mathbf{snake}, \mathbf{snake})$

As (3d) and (3e), (3f) and (3g), and (3j) and (3k), proper subtyping is irreflexive, whereas subtyping is reflexive.

The difference of the typing information and the (proper) subtyping information is shown by their relational property. The typing information is neither transitive nor continuous, whereas the (proper) subtyping information is transitive and almost continuous, as (4) shows:

(4) a. Draught bass is a bitter with a malty flavour and light

b. Bitter is an English draught ale served in pub.

c. Ale is a top-fermented beer.

d. Beer is a drink of fermented hops, malt and barley.

(4) has totally the following proper subtyping information:

(5) **properSubtype(draught bass, bitter)**,
properSubtype(bitter,English draught ale),
properSubtype(English draught,ale),
properSubtype(ale,beer),
properSubtype(beer,drink)

Basically typing information is information on hyponymy relation between types and their tokens, and therefore it is contained in a relation between common nouns and proper names, whereas basically subtyping information is information on hyponymy relation among common nouns, and so it is contained in a relation among common nouns.

3 Nonmonotonic Ontology Construction Rules

The nonmonotonic construction system of ontological hierarchies is defined in Prolog notation as follows:

(i) the domain experts' stable knowledge:

```
must_properSubtype(X,Y).
```

This assertion means that "X is a proper subtype of Y" is stable knowledge, and will not be revised according to users' knowledge.

(ii) the clustering rule based on type specifications:

```
member(Token,el(Type)) :-
ofType(Token,Type).
```

where `el(Type)` means the set of elements of type of `Type` and `ofType(Token,Type)` `Token` is of type `Type`,

(iii) the exclusion rules of metonymy:

```
ofType(Token,Type) :-
setof(Token0,may_ofType(Token0,Type),Tokens),
longestmorpheme(Token,Tokens).
```

For example, from {**Ford Motor Co.:stock**, **shares of Ford Motor Co.:stock**}, **Ford Motor Co.:stock** is abolished.

(iv) the exclusion rules of token-type misunderstanding:

```
properSubtype(X,Y):- may_typeOf(X,Y),
may_typeOf(Y,Z).
properSubtype(X,Y):- may_typeOf(X,Z),
may_typeOf(Z,Y).
```

For example, from {**Microsoft:softwaremaker**, **softwaremaker:maker**}, **softwaremaker:maker** is abolished.

(v) the construction rule of middle level type names based on analyzing morphological structures of type names:

```
may_properSubtype(Type,Middle):-
setof(Type1,Type1:type,TypeSet),
member(Type,TypeSet),
longest_shared_lemma(Middle,TypeSet).
```

For example, these rules guess a middle level type name **maker** from a set of type names {**automaker, chipmaker**}.

(vi) the trie structuring rule:

```
evidential_properSubtype(X,Y):-
may_type(X), may_type(Y),
trie_str_estimatable(X,Y).
```

`may_type(X)` means that `X` is the type name found statistically by trie structuring, and `trie_str_estimatable(X,Y)` means that 'X is a proper subtype of Y' is estimated in the trie structuring. (See section 4)

(vii) the nonmonotonic structuring rules of the type-hierarchy, i.e.,

1.
   ```
   properSubtype(X,Y):-
   must_properSubtype(X,Y).
   ```

2.
   ```
   properSubtype(X,Y):-
   evidential_properSubtype(X,Y),
   not(must_properSubtype(Y,X)).
   ```

3.
   ```
   properSubtype(X,Y):-
   may_properSubtype(X,Y),
   not(must_properSubtype(Y,X)),
   not(evidential_properSubtype(Y,X)).
   ```

4.
   ```
   evideltial_properSubtype(X,Y):-
   X:type, Y:type,
   el(X)=E1, El(Y)=E2,
   most(E1,E2), not(most(E2,E1)),
   not(must_properSubtype(Y,X)).
   ```

5.
   ```
   most(X,Y):-
   cardinality(X)∩cardinality(Y)
   >0.8×cardinality(X).
   ```

where `must_`φ expresses a *stable* proposition and is used as background knowledge and expert's prereference, `evidential_`φ is used as an expected proposition from the data, `may_`φ expresses a *non-stable* proposition and is used as an expected proposition by the rules, and `not`(φ) means that φ is not provable in the state of data, that is, `not` expresses non-monotonic negation (negation-as-failure). Therefore, if the data and background knowledge are updated or revised, then the constructable structure is naturally revised. That is, the credibility of the typing and subtyping information is ordered by the modalities as follows:

$$\mathtt{may\_}\varphi < \mathtt{evidential\_}\varphi < \mathtt{must\_}\varphi$$

4 Method

To test our assumptions about patterns derived from typing information we implemented several of them as regular expressions and ran these against corpora in the domain of molecular biology. The corpora were pre-processed using a variety of linguistic tools such as the Conexor FDG dependency parser [24] and a greedy chunker and lemmatiser based on the output of the parser.

The output candidates were then statistically filtered by grouping them according to the head of each noun phrase which was lemmatised. Some examples according to taxonomic typing information are given below.

taxonomic copula *Pit-1*$_{token}$ is a *pituitary-specific transcription factor*$_{type}$

taxonomic copula *IL-12*$_{token}$ is a *critical immunoregulatory cytokine*$_{type}$

the reverse taxonomic copula the *respective region*$_{type}$ is a *TCC-CCTCCCCT motif*$_{token}$

naming construction *latent transcription factors*$_{type}$ called *STAT*$_{token}$

naming construction an *inibitory subunit*$_{type}$ called *I kappa B alpha*$_{token}$

naming construction *degenerative neurologic syndrome*$_{type}$ termed *tropical spastic paraparesis*$_{token}$

descriptive anaphora *the kinase inhibitor*$_{type}$ *H-89*$_{token}$

descriptive anaphora *the glycoprotein hormone*$_{type}$ *erythropoietin*$_{token}$ (*Epo*$_{token}$)

descriptive anaphora *the protein*$_{type}$ *TBP*$_{token}$

exemplification *cytokines*$_{type}$ such as *TNF*$_{token}$

exemplification *cells*$_{type}$ such as *monocytes*$_{token}$ and *T cells*$_{token}$

exemplification *Sp1*$_{token}$ and two other *binding proteins*$_{type}$

At first glance we can see that the examples include a mixture of what domain experts would recognize as valid classes (e.g. *cells*, *hormone, proteins*) as well as subclasses (e.g. *kinase inhibitor, glycoprotein hormone, degenerative neurological syndrome*). Very often this class information is hidden within term constituents and for this reason we provided a filtering step based on statistical inference to filter out the classes from descriptive modifiers of those classes.

The next stage of our investigation involved constructing trie structures [7] based on the output of the typing information above. This was done in order to derive stable class expressions through statistical regularity. In other words, a string that starts from the right hand side of a class candidate and which is common to many different class candidates could be inferred to be stable and valid.

A trie is by definition a tree structure with one node for each common *prefix* in a string. In our experiments we worked from the *suffix* backwards so actually a trie represents the reverse of each class candidate string. We have supplemented this with counts of how many times a suffix is used in the trie.

By conflating candidate classes in this way we were able to obtain a much better idea about the distribution of class-indicating words. After conflating we counted according to frequency of the suffix word in its constituent members within the trie as shown in the example below for *oligonucleotide*.

1 OE:459||4||oligonucleotide||
2 OE:459.01||1||EOS||
3 OE:459.02||1||NF-AT||
4 OE:459.02.01||1||EOS||
5 OE:459.03||1||antisense||
6 OE:459.03.01||1||EOS||
7 OE:459.04—1||phosphorothioate||
8 OE:459.04.01||1||appropriate||
9 OE:459.04.01.01||1||EOS||

We then sorted on the suffix frequency (e.g. 4 for *oligonucleotide* at index OE:459 or 1 for *NF-AT* at index OE:459.02) to obtain a ranked list. In this way absolute frequency scores are less important than their rankings in our analysis.

This distribution though still does not reflect several of the intuitions that we hold about what should be a good ontology class. We therefore also included a measure of the 'fan out' of each suffix word and re-ranked class candidate words according to their suffix frequency multiplied by the number of words that they modify in the trie. For example in the case of *oligonucleotide* above, the fan out would be 3 because it is modifier by *NF-AT*, *antisense* and *phosphorothioate*. This allows us to model the fact that we generally regard a good class as being widely representative of many sub-classes or instances and a poor class is one which has a narrow membership. Clearly other intuitions could also be modelled using this method and we leave this for future work.

5 Data Sets

To show the application of typing information we studied three data sets. All the data sets come with labelled named entity classes which were removed from the data before doing any experiments. The labelled classes considered as a core DSO provide a useful reference point to check the results of our method against and our initial goal was to find clear evidence for each of the known classes using the method outlined above. The three data sets (Bio1, Bio2 and GENIA v3.02) come from the domain of molecular biology. A further line of comparison comes from comparing text types - two of the corpora (Bio1 and GENIAv3.02) are comprised of PubMed's MEDLINE abstracts [11] while the third (Bio2) is made from full EMBO journal articles. This is summarized in Table (1) below.

Corpus	Domain	Text type	#docs	#tokens
Bio1	biology	abstracts	100	28634
Bio2	biology	articles	50	386720
GENIA v3.02	biology	abstracts	2000	510321

Table 1. A summary of the data collections used in the experiments

The Bio1 collection [25] comes from a sub-domain of molecular biology that was formulated by searching under the terms *human, blood cell, transcription factor* in the PubMed database. From the retrieved abstracts 100 were randomly chosen for annotation by a human expert according to classes in a small core DSO.

The GENIA version 3.02 corpus [13] is similarly constructed to Bio1 with the most noticeable difference between Bio1 and GENIA 3.02 being in size (2000 MEDLINE abstract and 510321 tokens for GENIA and 100 abstracts and 28634 tokens for Bio1) and the number of annotated named entity classes (23 for GENIA and 10 for Bio1). For simplicity we have taken a subset of the taxonomy in GENIA v3.02 to give the same set of classes as in Bio1 [25] and Bio2. The overall breakdown of classes and frequencies of named entities (instances) can be see in Table (2). The table shows only a selected subset of named entities which happened to correspond in each of the corpora. It is worth noting that GENIA in particular specializes many of these name classes but for purposes of comparison we have taken the highest level common category between corpora. While the figures are not directly related to our method they are nevertheless indicative of the amount of evidence available for discovering each class.

The Bio2 corpus comprises 50 EMBO (European Molecular Biology Organization) Journal articles which have been hand annotated by domain experts for the same set of named entity classes used in Bio1. The articles were randomly selected from the period 2000 to 2002. Bio2 is particularly interesting because it shows a difference in text type which we expect may influence the classes that appear as well as their statistical distribution.

	Corpus			
Type	#Bio1	#Bio2	#GENIA	Description
PROTEIN	2125	21956	24404	proteins, protein groups ,families, complexes and substructures.
DNA	358	3231	8755	DNAs, DNA groups, regions and genes
RNA	30	1109	704	RNAs, RNA groups, regions and genes
SOURCE.cl	93	1257	3716	cell line
SOURCE.ct	417	1016	6292	cell type
SOURCE.mo	21	1296†	169	mono-organism
SOURCE.mu	64	1296†	1600	multiorganism
SOURCE.vi	90	703	1063	virus
SOURCE.sl	77	na	na	sublocation
SOURCE.ti	37	53	640	tissue

Table 2. A summary of annotated classes in the Bio1, Bio2 and GENIA corpora. Counts are for numbers of class instances.†The multi-celled and mono-celled organism classes are merged in Bio2.

6 Experiments and Results

From the three corpora we extracted a number of candidate classes using four regular expressions. While the raw counts are no indication of quality they do show some trend in the amount of evidence we can expect to capture and this is shown in Table (3) broken down according to pattern type and corpus.

Corpus	Pattern name	(A)†	(%A)	(B)‡	(%B)
Bio1	taxonomic copula	13	6.7	14	6.3
Bio1	naming construction	4 2.1	5	2.3	
Bio1	descriptive anaphora	169	87.6	194	87.8
Bio1	exemplification	7	3.6	8	3.6
		193	100	221	100
Bio2	taxonomic copula	103	5.1	120	3.9
Bio2	naming construction	33	1.6	41	1.4
Bio2	descriptive anaphora	1843	90.7	2823	92.9
Bio2	exemplification	53	2.6	54	1.8
		2032	100	3038	100
GENIA	taxonomic copula	248	8.9	304	8.1
GENIA	naming construction	45	1.6	65	1.7
GENIA	descriptive anaphora	2374	85.3	3239	86.7
GENIA	exemplification	116	4.2	127	3.5
		2783	100	3735	100

Table 3. Numbers of candidates obtained by each pattern for each data set. †denotes the number of candidates found after conflating duplicates, and ‡shows the number of candidates found before conflation.

These were then converted to suffix trie structures and the constituents of the trie were then sorted according to frequency and ranked. The rankings were then adjusted according to the fan out of each suffix string and a comparison made against known classes from the three molecular biology corpora as shown in Table (4). In several cases we could observe the class name directly from the highly ranked keywords (e.g. *protein*, *gene*, *virus*, *cell*), but several of the classes were far more difficult to find in the list (e.g. *cell line*, *tissue*) or non-existent (e.g. *mono-celled organism*, *sublocation*). One point that is interesting to note is that one of the most important classes called DNA actually had two strongly associated keywords *DNA* and *gene* which can be considered synonymous.

To explore further about the nature of the missing type keywords we looked through the list for keywords that should be strongly related in some way to known classes and a few results are shown in Table (5) along with the relationship to the type. Very often we could find reasonable keyword labels for subclasses of the missing class such as *human*, *host* etc. for the missing *multi-celled organism* and we could also discover known subtypes such as *receptor*, *promoter* or *enzyme* for protein and *family* which could be applied to either *DNA* or *protein*.

Finally we also found evidence for classes that were not given in the original class list such as *pathway*.

7 Discussion

The scope of these initial experiments has focussed narrowly on a set of four regular expressions that implemented some of the notions we have about typing and subtyping patterns in texts. Nevertheless, the initial experiments provide a good basis on which to proceed and seem to confirm some domain expert intuitions about what classes should occur in the molecular biology domain, at least at a high level conceptualization. They have shown that we can obtain important

	Order ranking of keyword in corpus by class before (and after) re-ranking with fan-out			
Keyword	Class	Bio1	Bio2	GENIA
protein	PROTEIN	6 (6)	1 (1)	3 (3)
gene	DNA	2 (2)	9 (7)	1 (1)
dna	DNA	29 (19)	12 (15)	23 (32)
rna	RNA	194 (137)	89 (53)	596 (328)
mrna	RNA	56 (26)	278 (118)	78 (62)
cell line	SOURCE.cl	26 (12)	208 (159)	24 (17)
cell	SOURCE.ct	30 (20)	3 (3)	5 (5)
-	SOURCE.mo	-	-	-
-	SOURCE.mu	-	-	-
virus	SOURCE.vi	108 (82)	29 (32)	38 (37)
-	SOURCE.sl	-	-	-
tissue	SOURCE.ti	-	1250 (786)	310 (239)

Table 4. Rankings of class indicative keywords obtained from the experiments after sorting tries based on the head word frequencies.

		Order ranking of keyword in corpus by class before (and after) re-ranking with fan-out			
Keyword	Rel†	Class	Bio1	Bio2	GENIA
receptor	ISA	PROT.	9 (7)	15 (11)	6 (8)
promoter	ISA	PROT.	3 (3)	7 (8)	2 (2)
enzyme	ISA	PROT.	-	72 (40)	117 (104)
motif	PW	PROT.	286 (196)	13 (16)	35 (23)
complex	PW	PROT.	4 (4)	5 (4)	8 (9)
terminal	PW	DNA	12 (37)	484 (100)	14 (42)
site	PW	DNA	1 (1)	11 (10)	4 (4)
promoter	PW	DNA	3 (3)	7 (8)	2 (2)
sequence	ISA	DNA	189 (134)	8 (6)	15 (11)
transcript	ISA	RNA	- (-)	524 (284)	308 (175)
region	PW	PROT.	8 (14)	6 (5)	13 (10)
complex	PW	PROT.	4 (4)	5 (4)	8 (9)
human	ISA	SRC.mu	- (-)	- (-)	- (-)
patient	ISA	SRC.mu	- (-)	- (-)	30 (27)
mouse	ISA	SRC.mu	297 (203)	409 (176)	186 (137)
yeast	ISA	SRC.mo	- (-)	225 (191)	- (-)
nucleus	PW	SRC.ct	275 (189)	79 (108)	44 (76)

Table 5. Interesting rankings for subclasses of known classes based on indicative keywords obtained from the experiments after sorting tries based on the head word frequencies.†Relation (Rel) is either ISA for taxonomic or PW for meronymy. Class names have been abbreviated as PROT for PROTEIN and SRC for SOURCE.

evidence to help guide experts in selecting core DSO classes and their relations to their children middle level classes.

We have said nothing so far about the effects of re-ranking but we have observed that this does indeed tend to 'push' keywords that we would like to see as class names higher in the list. There is clearly much more that could be done here in future work.

The results have shown that a variety of relations can be discovered between class indicating keywords such as taxonomy or meronym. More subtle distinctions also need to be explored further and may well be beyond the scope of shallow techniques to uncover such as the basis on which a taxonomy or meronymy is founded. In the molecular biology domain we can see that many classes are based on substance relations, e.g. motif is physically part of a protein, whereas other distinctions are also possible, e.g. based on functional aspects such as enzyme being a type of protein.

In terms of amount of text we believe that the results are relatively stable even when quite small amounts of text are used. The rankings observed for Bio1 and GENIA for example are quite similar although many lower ranking keywords are absent altogether from Bio1. The effects of text type are not clear at this time and again this will be left as future work for us to explore.

8 Conclusion

Text mining based extraction of types, relations and properties offers significant potential to aid humans in constructing ontologies from a mass of data. We aim from now to construct an ontology maintenance system called Ontology Express which will allow experts to interact with the extracted types and to add in knowledge from their own intuitions using non-monotonic inference. This will combine an extended version of the statistical filtering and non-monotonic reasoning approaches we have outlined here.

We expect that typing information will have an important role in the ontology construction process but since it is essentially a shallow technology this requires statistical filtering to obtain useful results. In many cases though it is clear that valid classes are missing, either because they are statistically not significant (such as *RNA*), or they are quite abstract and part of the common sense knowledge within the domain and never explicitly mentioned as was the case with *multi-celled organism*. This emphasizes the need to consider this technology as a supporting tool for human ontology construction and not as a replacement for human introspection.

REFERENCES

[1] M. Andrade and A. Valencia, 'Automatic extraction of keywords from scientific text: application to the knowledge domain of protein families', *BioInformatics*, **4**(7), (1998).

[2] O. Bodenreider, A. Burgun, and T. C. Rindflesch, 'Lexically-suggested hyponymic relations among medical terms and their representation in the UMLS', in *Proceedings of TIA'2001*, 11–21, (2001).

[3] S. Cederberg and D. Widdow, 'Using LSA and Noun Coordination Information to Improve the Precision and Recall of Automatic Hyponymy Extraction', in *Proceedings of CoNLL-2003, Edmonton, Canada*, 111–118, (2003.

[4] N. Collier, H.S. Park, N. Ogata, Y. Tateishi, C. Nobata, T. Ohta, T. Sekimizu, H. Imai, and J. Tsujii, 'The GENIA project: corpus-based knowledge acquisition and information extraction from genome research papers', in *Proceedings of the Annual Meeting of the European chapter of the Association for Computational Linguistics (EACL'99), Bergen, Norway*, (8–12th June 1999).

[5] N. Collier, K. Takeuchi, A. Kawazoe, T. Mullen, and T. Wattarujeekrit, 'A framework for integrating deep and shallow semantic structures in text mining', in *Proceedings of the Seventh International Conference on Knowledge-based Intelligent Information and Engineering Systems (KES'2003), University of Oxford, Oxford, UK*, (September 3–5 2003).

[6] C. Fellbaum, *WordNet: An Electronic Lexical Database*, The MIT Press, Cambridge, (1998).

[7] E. Fredkin, 'Trie memory', *Communications of the ACM*, **3**(9), 490–499, (1960).

[8] U. Hahn and M. Romacker, 'Content management in the syndikate system: How technical documents are automatically transformed to text knowledge bases', *Data and Knowledge Engineering*, **35**(2), 137–159, (2000).

[9] M. Hearts, 'Automatic acquisition of hyponyms from large text corpora', in *Proceedings of the Fourteenth International Conference on Computational Linguistics, Nantes, France*, pp. 539–545, (July 1992).

[10] Jerry R. Hobbs, William Croft, Todd Davies, Douglas Edwards, and Kenneth Laws, 'Commonsense metaphysics and lexical semantics', in *24th Annual Meeting of the Association for Computational Linguistics*, pp. 231–240. the Association of Computational Linguistics, (1986).

[11] MEDLINE. The PubMed database can be found at:, 1999. http://www.ncbi.nlm.nih.gov/PubMed/.

[12] N. Ogata, 'A Formal Ontology Discovery from Web Documents', in *Lecture Notes on Artificial Intelligence 2198: Web Intelligence: Research and Development, First Asia-Pacific Conference, WI 2001, Maebashi City, Japan*, Springer-Verlag, Berlin, 514–519, (October 2001)

[13] T. Ohta, Y. Tateishi, H. Mima, and J. Tsujii, 'The GENIA corpus: An annotated research abstract corpus in the molecular biology domain', in *Human Language Technologies Conference (HLT 2002)*, (??? 2002).

[14] Woojin Paik, Elizabeth D. Liddy, Edmund Yu, and Mary McKenna, 'Categorizing and standardizing proper nouns for efficient information retreival', in *Corpus Processing for Lexical Acquisition*, eds., Branimir Boguraev and James Pustejovsky, 61–73, The MIT Press, Cambridge, (1996).

[15] James Pustejovsky, *Generative Lexicon*, The MIT Press, Cambridge, 1995.

[16] Aarne Ranta, *Type-Theoretical Grammar*, Oxford University Press, Oxford, 1994.

[17] T. C. Rindflesch, L. Hunter, and A. R. Aronson, 'Mining molecular binding terminology from biomedical text', in *American Medical Informatics Association (AMIA)'99 annual symposium, Washington DC, USA*, pp. 127–131, (1999).

[18] T. C. Rindflesch, L. Tanabe, J. N. Weinstein, and L. Hunter, 'EDGAR: Extraction of drugs, genes and relations from the biomedical literature', in *Pacific Symposium on Bio-informatics (PSB'2000), Hawai'i, USA*, pp. 514–525, (January 2000).

[19] S. Rydin, 'Building a hyponym lexicon ith hierarchical structure', in *Proceedings of the SIGLEX Workshop on Unsupervised Lexical Acquisition, held at ACL2002, Philadelphia, Pennsylvania*, pp. 26–33, (2002).

[20] T. Sekimizu, H. Park, and J. Tsujii, 'Identifying the interaction between genes and gene products based on frequently seen verbs in medline abstracts', in *Genome Informatics*, pp. 62–71. Universal Academy Press, Inc., (1998).

[21] Padmini Srinivassan, 'Thesaurus construction', in *Information Retrieval: Data Structures and Algorithms*, eds., Willam B. Frakes and Ricardo Baeza-Yates, 161–218, Prentice-Hall, New York, (1992).

[22] H. Sundblad, 'Automatic acquisition of hyponyms and meronyms from question corpora', in *Workshop on Machine Learning and Natural Language Processing for Ontology Engineering, held at ECAI 2002, Lyon, France*, (July 22–23 2002).

[23] L. Tanabe and W. Wilbur, 'Tagging gene and protein names in biomedical text', *Bioinformatics*, **18**, 1124–1132, (2002).

[24] P. Tapanainen and T. Järvinen, 'A non-projective dependency parser', in *Proceedings of the 5th Conference on Applied Natural Language Processing, Washington D.C., Association of Computational Linguistics*, pp. 64–71, (1997).

[25] Y. Tateishi, T. Ohta, N. Collier, C. Nobata, K. Ibushi, and J. Tsujii, 'Building an annotated corpus in the molecular-biology domain', in *COLING'2000 Workshop on Semantic Annotation and Intelligent Content, Luxemburg*, (5th–6th August 2000).

[26] F. Xu, K. Daniela, P. Jakub, and S. Sven, 'A Domain Adaptive Approach to Automatic Acquisition of Domain Relevant Terms and their Relations with Bootstrapping', in *Proceedings of the 3rd International Conference on Language Resources an Evaluation (LREC'02)*, (May 29-31 2002).