Ontology Driven Information Extraction

Uwe Reyle
Institute for Computational Linguistics
University of Stuttgart
&
European Media Laboratory
Heidelberg
Uwe.Reyle@IMS.Uni-Stuttgart.de

Jasmin Šarić
European Media Laboratory
Heidelberg
Jasmin.Saric@eml.villa-bosch.de

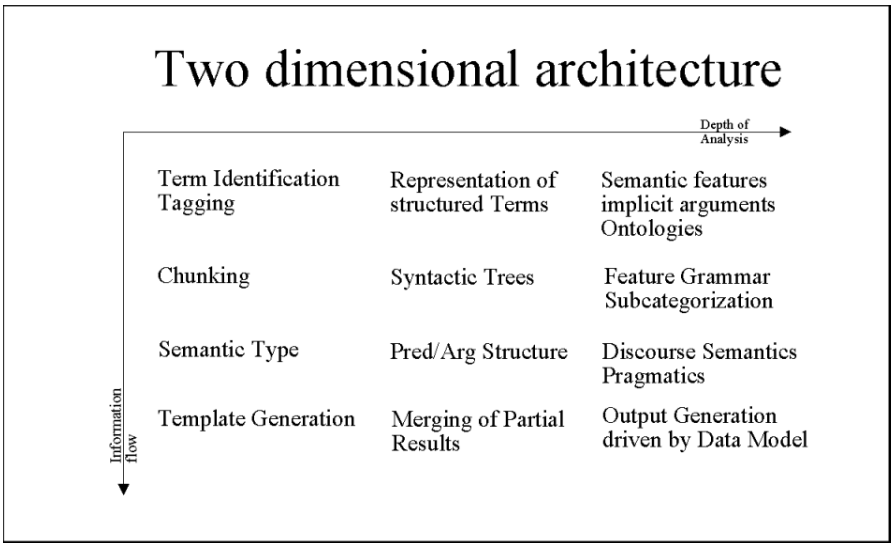
We describe the basic mechanism and architecture of a system (GenIE) that automatically extracts information about biochemical pathways from free text sources. We show, (i), that the extraction of information must be based on specialized lexica, semantic representations and ontologies, and (ii), that a systematic relationship between low-level and high-level representations is needed for the interleaving of shallow and deep processing.

1 Introduction

The analysis and interpretation of biochemical data is crucial for the understanding of biochemical phenomena and the discovery of new concepts. The organisation of the scientific data in a model that captures the concepts and relations involved can facilitate the understanding and reasoning over the data. This is the overall objective of the EML [1] *Data Alive* project. The primary goal of *Data Alive* lies in the development of a database which allows the handling and representation of the complex heterogeneous data about biochemical pathways. The creation of this database system has been done using an ontology for biochemical pathways, which includes concepts ranging from basic biochemistry to genomics.
Regarding the explosive growth of literature in this area the gains of computational linguistics and automatic information extraction technology have to be exploited to feed the databases from free text sources. Conventional extraction technology has been successfully applied to very restricted extraction tasks, like protein-protein interactions. The objective of the GenIE project [2]

[1]European Media Laboratory GmbH, Heidelberg
[2]GenIE abbreviates Genome Information Extraction. After a preparatory phase of one

is more complex. It will extract information about biochemical pathways and about sequences, structures and functions of genomes and proteins.
We will show that the extraction of information must be based on specialized lexica and parsers as well as semantic representations and ontologies. Only then can information extraction technology successfully be applied to more complex extraction tasks. Obviously the incorporation of these knowledge sources into the extraction system leads to the requirement to combine shallow and efficient NLP techniques - like tagging and chunking as well as template filling - with a deeper semantic interpretation in case more semantic insight is needed. This feature of 'depth if required' distinguishes our approach to information extraction from text understanding on the one hand and standard information extraction on the other.
In order to fulfill this requirement of interleaving shallow and deep processing there must be a systematic relationship between the representations used in shallow processing on the one hand and the representations contemplated in lingusitic theories one the other hand. This is best achieved by a methodology in which low-level and high-level representations are developed in conjunction and embedded into a comprehensive, computational architecture that guarantees possibilities for information flow as illustrated in the following picture.



year the project started the first of January 01. More information can be found at http://www.ims.uni-stuttgart.de/projekte/GenIE/main.html

Domain specific ontological and semantical knowledge can be exploited at any stage of morphological and syntactic processing, or vice versa. This two-dimensionality leaves sufficient leeway to accommodate combinations of shallow parsing as well as deep semantic analysis, and of full parsing and flat semantics. The two-dimensional architecture should also be shared by the morphological analyzer.
The extraction process itself is mediated by a semantic representation of the text which is then used to fill the data base. This semantic representation will make contextual information explicit and will serve as a basis for disambiguation tasks and contextual resolution. The semantic representations (at any level) are required not to overspecify, i.e. not to force any ambiguity the resolution of which is irrelevant for the extraction task at hand.
In Section 2 we illustrate our approach with a case analysis in which we will factor out the minimal semantic and ontological assumptions that are needed for the extraction task. Section 3 illustrate the basic need for a lexicon of biochemical terms and morphemes, and a morphological analyzer for derivationally and compositionally complex expressions. Section 4 presents the system components and including the domain-specific ontology which comprises a theory of the atomic predicates and relations used in the lexicon.

2 A (standard) example

A typical example for use of standard extraction techinques is shown by the following example. It is taken from "The Journal of Biological Chemistry" ([1]):

> *Phosphofructokinase (EC 2.7.1.11) (PFK) catalyzes the transfer of the gamma-phosphate from MgATP to fructose 6-phosphate (Fru-6P) in the first committed step of glycolysis.*

The following steps show the proceeding for the extraction of the relevant information.

1. **Word and sentence boundary detection**
   The first processing step involve Tokenization [3]. Tokenization mainly consists of recognition of token (word) boundaries and sentence boundaries.
   In our example we can follow the simple heuristic that a blank marks a word boundary. Except in one case: *fructose 6-phosphate* has to be

[3] In a very first step preprocessing has to take place. This means the conversion from HTML or PDF to plain text.

recognized as a multiword. This is done within the next processing step. Further information on our Tokenizer and POS-Tagger can be found in Section 5.

2. **Tagging and multiwords**
The second processing step concerns part-of-speech (POS) Tagging including lemmatization. The POS Tagger assigns an appropriate part-of-speech as well as the appropriate lemma for each token. The tagging process is followed by a correction step. Thereby multiword terms are recognized. The relevant information for composition of these terms comes from different tools, like an abbreviation detector (according to[2]). The abbreviation *(Fru-6P)* is a significant trigger for recognizing *fructose 6-phosphate* as a multiword term.

3. **Chunking**
The chunking step leads to the following result:

$[_c$ $[_{c0}$ $[_{np}$ Phosphofructokinase] $[_{vx}$ $[_{vbz}$ catalyzes]]]
  $[_{ng}$ $[_{nx}$ $[_{dt}$ the] $[_{nn}$ transfer]] $[_{of}$ of] $[_{nx}$ $[_{dt}$ the] $[_{nn}$ gamma-phosphate]]]
  $[_{pp}$ $[_{in}$ from] $[_{np}$ MgATP]]
  $[_{pp}$ $[_{to}$ to] $[_{nx}$ $[_{nn}$ fructose] $[_{nn}$ 6-phosphate]]]
  $[_{pp}$ $[_{in}$ in] $[_{ng}$ $[_{nx}$ $[_{dt}$ the] $[_{jj}$ first] $[_{jj}$ committed] $[_{nn}$ step]] $[_{of}$ of] $[_{nx}$ $[_{nn}$ glycolysis]]]]
]
[*sent* .]

4. **Word sense (/Semantic) tagging**
Within the step of semantic tagging, we annotate the chunks with semantic tags. We basically distinguish between three types of semantic annotation. The first type annotates concepts of the underlying ontology (which serves as a backbone for semantic operations and as a logical interface between different components of the system.) The second type associates semantic tags to closed class words of English (such as, *of*, *not*, *and*, etc.) And the third type of semantic annotations is context dependent. Here the context may be given by a particular syntactic environment, eventually including semantic tags or coreferent phrases (and their semantics). (This type of annotation thus takes place at some point in the two dimensional structure of our system.) We show the results for this step by placing a semantic tag as subscript at the end of the bracket[4]
:

[4] We are using the following abbreviatios as semantic tags: ENZ as abbreviation for Enzyme; CH-REACT classifies the kind of event as CHEMICAL REACTION; CH-ELEM as abbreviation for CHEMICAL ELEMENT; PATHW as abbreviation for PATHWAY; CO

$[_c\ [_{c0}\ [_{np}$ Phosphofructokinase$]_{ENZ}\ [_{vx}\ [_{vbz}$ catalyzes$]_{ENZ-REACT}]]$
$[_{ng}\ [_{nx}\ [_{dt}$ the$]\ [_{nn}$ transfer$]_{CH-REACT}]\ [_{of}$ of$]\ [_{nx}\ [_{dt}$ the$]\ [_{nn}$ gamma-phosphate$]_{CH-ELEM}]]$
$[_{pp}\ [_{in}$ from$]\ [_{np}$ MgATP$]_{CO}]$
$[_{pp}\ [_{to}$ to$]\ [_{nx}\ [_{nn}$ fructose$]\ [_{nn}$ 6-phosphate$]]_{FRU}]$
$[_{pp}\ [_{in}$ in$]\ [_{ng}\ [_{nx}\ [_{dt}$ the$]\ [_{jj}$ first$]\ [_{jj}$ committed$]\ [_{nn}$ step$]]\ [_{of}$ of$]\ [_{nx}\ [_{nn}$ glycolysis$]_{PATHW}]]]$
]
[*sent* .]

5. **Template detection with subcat information**
To get the relevant information for the filling of the templates subcat frames have to be detected. The lexicon provides the following two necessary entries:

catalyze	SUBJ: ENZ	OBJ_{acc}: CH-REACT	
transfer	PP_{of}: CH-ELEM	PP_{from}:CO	PP_{to}:FRU

The selectional restrictions are part of the lexical entries. Within this processing step the arguments have to retrieved and matched.

6. **Template filling**
After matching the arguments the relevant information can be extracted with help of template filling rules, like the following:

A:	@SUBJ:ENZ	$\Longrightarrow$	**Catalyze event** e_1
e_2:	@OBJ:CHEM-REACT		Catalyst: A
e_1:	catalyze:ENZ-REACT		Reaction: e_2

The extracted information can then be characterized by two events: the catalyze event e_1 and the transfer event e_2. The following table shows the filled template:

Catalyze event: e_1	**Transfer event**: e_2
Catalyst: Phosphofructokinase Reaction: e_2	Type: chem. reaction Transfer of: gamma-phosphate Transfer from: MgATP Transfer to: fructose 6-phosphate Transfer within: glycolysis

as abbreviation for ORGANIC COMPOUND and FRU abbreviates FRUCTOSE. The tags are arranged in a way to form a taxonomy that reflects the underlying ontology.

3 A Case for Semantic Tagging

The following two sentences were taken from "The Journal of Biological Chemistry" ([3]):

> *The serine protease CD26/dipeptidyl-peptidase IV (CD26/DPP IV) and chemo-kines are known key players in immunological processes. Surprisingly, CD26/DPP IV not only removed the expected Gly1-Pro2 dipeptide from the NH2 terminus of macrophage-derived chemokine (MDC) but subsequently also the Tyr3-Gly4 dipeptide, generating MDC(5-69).*

The first two columns of the following table represent the output of the first processing steps [5]: tokenizing and tagging[6]. We will focus here on the third step, the semantic annotation (or semantic tagging)[7]. The first column shows one lemmatized token per line. The second column gives the corresponding POS tag and the third column shows the result of annotating semantic tags to tokens.[8] If no arrow points to a tag, this means that it has been annotated at this stage of processing by direct projection from the lexicon and without exploitation of any kind of contextual knowledge. The tags with arrows pointing to them at least depend on the knowledge at the source of the arrows. They indicate that some steps in the direction of deeper processing had to be achieved in order to identify the correct tag. In particular the fact that CD26/DPP IV is an Enzyme can only be annotated after its contextual resolution to the occurrence of CD26/DPP IV in the first sentence; and that MDC abbreviates 'macrophage-derived chemokine' depends on the semantic tag of chemokine and the particular syntactic environment indicated by the dependency on PP. Note that we did not semantically annotate MDC(5-69) at this stage, simply because it cannot be reliably identified as an abbreviation of a protein. (The

[5] We only illustrate this for the second (relevant) sentence.

[6] Further information on our Tokenizer and POS-Tagger can be found in Section 5.

[7] Further abbreviatios as semantic tags: PEP as abbreviation for PEPTIDE; PEP-CH as abbreviation for PEPTIDE CHAIN; PRO as abbreviation for PROTEIN; PRO-ABBR as abbreviation for PROTEIN ABBREVIATION ; CC as abbreviation for CHEMICAL COMPOUND. The tags are arranged in a way to form a taxonomy that reflects the underlying ontology.

[8] The discussion of this section presupposes a unique, preferred output of the first processing steps. In principle we could give up this presupposition and reclaim the two-dimensional approach to include tagging as well as tokenizing. We didn't do this for two reasons. First, the results of the tokenizer and tagger are very good for standard English texts and will achieve this precision also for the biochemical texts once they are trained with the continuously increasing lexicon of biochemical terminlogy.

similarity between MDC(5-69) and MDC depends on the meaning of (5-69). But as the meaning of phrases in parenthesis is highly ambiguous nothing may be predicted at this stage.) The results of chunking are given in column 5.

Lemma	POS	STAG	Chunks			STAG
Surprisingly,	RB	-				
CD26/DPP IV	NP	ENZ	NP			
not	RB	-				
only	RB	-				
remove	VBD	-		VP		ENZ-REACT
the	DT	-				
expect	JJ	-				
Gly1-Pro2	NN	PEP	NP			PEP
dipeptide	NN	PEP				
from	IN	-				
the	DT	-	NP			
NH2 terminus	NN	PEP-CH				
of	IN	OF		NP		PEP-CH
macrophage-derive	JJ	-	PP			
chemokine	NN	PROT				
(MDC)	NP	PRO-ABR				
but	CC	-				
subsequently	RB	-				
also	RB	-				
the	DT	-				
Tyr3-Gly4	NN	PEP	NP			PEP
dipeptide	NN	PEP				
generate	VBG	-	VP			
MDC(5-69)	NP	-	NP	VP		PRO-ABR

A next, intermediate step has as its task a kind of semantic preprocessing. This step deals mainly with semantic annotation of complex chunks. The semantic tag for the complete chunk has to be calculated from its parts, their semantic tags and the way they are composed. This is illustrated by the NP *the NH2 terminus of macrophage-derived chemokine*, to which a rule of the form x:**PEP-CH OF** y:**PROT** $\rightsquigarrow$ x:**PEP-CH** $\wedge$ part_of(x,y) will be applied projecting the semantic tag of the head of the phrase and identifying the meaning of the preposition *of* as *part_of* relation. The meaning of the preposition depends

on the semantic tags of its arguments, which both are **PEP-CH**. (For y this must be derived from the ontology: proteins are peptide chains.)
This kind of meaning dependency also has to be exploited in order to calculate the last two semantic tags for the verbs *remove* and *generate* in the table above. The most frequent verbs in our corpus (see Section 3) are verbs that belong to standard English and have in general a broad variety of senses or meanings. And most of them are different, or more general than the meanings they typically have in the biochemical domain. As these meanings may only be determined on the basis of the semantic types of arguments these verbs subcategorize for their semantic tag depends on the calculation of the (semantic tags in the) subcategorization frame. The relevant subcategorization frame of the verb *remove* is 'NP_nom **remove** NP_acc **from** NP_dat'. Its slots will be filled with the semantically tagged representations of the arguments, which we denote as CD26/DPP_IV:**ENZ**, Gly1-Pro2:**PEP**, and MDC:**PROT**. If we make explicit the referential argument of the verb, i.e. the event e_1 it describes, in a DRT style manner [9] we get the following condition.

(1) e_1: remove(CD26/DPP_IV:**ENZ**,Gly1-Pro2:**PEP**,MDC:**PROT**)

That *remove* describes a chemical, and in particular an enzymatic reaction can now be derived on the basis of the semantic types of its arguments. In general a chemical reaction involves a change in which the atoms or molecules of two or more substances, the *substrates* of the reaction, are rearranged to form one or more additional substances, the reaction *products*. If this change is brought about by an enzyme, or catalyzed by a protein we have an enzymatic reaction. In addition *remove* implies that a molecular part y is taken away from a substrate z. We use lexical rules like the following to distinguish between different senses of verbs on the basis of their argument types.

$$\text{e:remove(x:}\mathbf{ENZ}\text{,y:}\mathbf{CC}\text{,z:}\mathbf{CC}) \rightarrow \text{ENZ-REACT(e)} \wedge \text{e: z} := \text{y} \oplus \text{z}\backslash\text{y}$$

We use the notation ':=' to indicate that the chemical compound z has changed its composition to $y \oplus z\backslash y$. Here the operator $\oplus$ denotes the mereological sum operator [10], and the operator $\backslash$ represents the result of cleavages.[11] Note that although we are using a variable e to denote the reaction event we do not explicitly specify the prestate and poststate of this event. The information that y is a molecular part of z during the prestate of e but no longer in the result state is implicitly given by the fact that z is a substrate of e but not a product. More precisely, a rule like

[9] A thorough Introduction can be found in [4]

[10] The theory of mereology can be found in [5].

[11] The authors of the article would represent MDC\Gly1-Pro2 instead as MDC(3-69), using parentheses the meaning of whose is highly ambiguous. We will come back to this shortly.

$$e\colon z := y \oplus z\backslash y \rightarrow z \in \mathrm{Substrates}(e) \wedge y \in \mathrm{Products}(e)$$

will allow us to derive the following information that may be (part of/or) added to an enzyme database.

Enzyme name	Substrates	Products
CD26/DPP IV	MDC	Gly1-Pro2 MDC\Gly1-Pro2

We now come to the cleavage of the Tyr3-Gly4 dipeptide which is expressed by an elliptical construction triggered by the phrase *not only ... but subsequently also*. We will not attempt do perform any syntax-based reconstruction of the elided phrase. We will instead use the semantic information that represents the source (and which is identified by the *not only* part of the trigger) in order to complete the semantics of the NP following *but subsequently also*.[12] Standard reconstruction of the elided phrase will extend the condition in (1) to the condition set in (2).

(2) e_1: remove(CD26/DPP_IV:**ENZ**,Gly1-Pro2:**PEP**,MDC:**PROT**)
e_1: remove(CD26/DPP_IV:**ENZ**,Try3-Gly4:**PEP**,MDC:**PROT**)

If we take into account the implications of the adverb *subsequently* we may consider $e_1 \circ\!\!\prec e_2$ as complex event consisting of the succession of e_1 and e_2. The properties of this complex event are determined by the general schema[13]

$$\begin{array}{l} e_1 \circ\!\!\prec e_2 \wedge e_1\colon z := y_1 \oplus z\backslash y_1 \wedge e_2\colon z := y_2 \oplus z\backslash y_2 \\ \quad \rightarrow \quad e_1 \circ\!\!\prec e_2\colon z := y_2 \oplus (y_1 \oplus z\backslash y_1)\backslash y_2 \end{array}$$

which allows us to derive

Enzyme name	Substrates	Products
CD26/DPP IV	MDC	Gly1-Pro2 MDC\Gly1-Pro2
	MDC\Gly1-Pro2	Tyr3-Gly4 MDC\Gly1-Pro2-Tyr3-Gly4

From this the substrate specificity of CD26/DPP IV can be computed automatically to contain Gly1-Pro2 and Tyr3-Gly4.
The last event that will be added to (2) is e_3: generate(CD26/DPP IV, MDC(5-69)). For the verb *generate* we have the lexical rule

[12] There are several approaches to a semantic based reconstruction of ellipses. [6], and others use higher order unification; [7] exploits parallelism; [8] (IMS, University of Stuttgart) has a very sophisticated algorithm

[13] The variables are interpreted dynamically, see [9].

$$\text{e:generate(x:}\mathbf{ENZ}\text{,y:}\mathbf{CC}\text{)} \rightarrow$$
$$\text{ChemReact(e)} \wedge \text{agent(e):PROT} \wedge \text{y} \in \text{Products(e)}$$

This event is understood as consisting of the sum of the two preceding events and thus producing the same result as they do, we may derive that MDC(5-69) must be equal to (MDC\Gly1-Pro2)\Tyr3-Gly4, or MDC\Gly1-Pro2-Tyr3-Gly4, and hence get an indication of what the parentheses in MDC(5-69) are supposed to mean.

4 Corpus Data and Lexicon Acquisition

Baayen ([10]) reports the fact that in every corpus there are as much hapax legomena, i.e. words that occur only once in a corpus, as there are other word types. This shows that one cannot analyze arbitrary texts without a reliable word formation component.

Within GenIE the GenIELex group has its main task in developing a biochemistry specific lexicon as well as an annotated corpus for the evaluation. The basic necessity for building up such lexica is illustrated by the following figures reflecting some properties of our JBC corpus [14], too.

Tokens [15]	$30 * 10^6$
Unknown Tokens [16]	$3.5 * 10^6$
Types	405 000
Hapax Legomena (1)	205 000
Unknown Types (2)	322 000
(1) ∩ (2)	172 500

As the amount of hapax legomena in the corpus is 204.775 of which 172.458, i.e. 85%, are unknown they are very likely to belong to the biochemical terminology.

Taking into account an average length of a sentence in our JBC corpus of 27 words and the average of every 9th token being unknown it is almost sure to have more than one unknown token per sentence. Even worse, three unknown token per sentence are the average. Even more, the amount of hapax legomena (most of them are biochemistry related technical terms) show the basic need for at least semi-automatic methods of lexicon generation.

[14] We built up a full text corpus from articles of the *Journal of Biological Chemistry* - www.jbc.org. The corpus has about $30*10^6$ token. This correspond to approximately 1/5 of the complete available Journal.

[15] We applied a tagger/lemmatizer that had been trained on the Brown Corpus to the biochemical corpus.

[16] The token that weren't recognized by the lemmatizer.

Lexicon There seems to be a general agreement within the literature on term extraction that not only terms but also their contexts must be taken into consideration in order to provide tools for automatic extraction of terms and in particular for automatic abstracting information on the semantic role of terms. The notions of context taken into consideration are:

- lexical, e.g. *association between* $\langle NP \rangle$ *and* $\langle NP \rangle$
- syntactic, e.g. ADJ+NN, NN+*of*-NN, or V+NN collocations,
- paralinguistic, e.g. parentheses, and
- so-called "knowledge rich" [17] textual contexts, which indicate important conceptual characteristics for the search term, and are thus used for the acquisition of domain knowledge and for the construction of ontologies.

For the detection (and also for the extraction) of such frames we use the IMS Corpus Query Processor (CQP). CQP is one of the tools from the IMS Corpus Workbench [18]. It is a specialized search engine for linguistic research. Information can be extracted from text corpora by the use of the CQP query language and by some additional commands and parameters [19].
The acquisition process is recursive. We start with semantically tagged morphemes and compositionally calculate the semantics of expressions like *phosphorylation*, *carboxamidomethylation*; *dephosphorylation*, or *5'-(de)phosphorylation*, *Galactosylation*, or *glucosylation*. (See [12]. Note that there are 576 one word term types (out of 43048 tokens) ending on *ylation* in our corpus.) This will then be used to enrich the lexicon and to enhance CQP work.

5 System Components

Tokenizing The main task of a tokenizer is to segment an input string into a sequence of token with sentential boundaries marked, where a token can either be a single word unit or a multiword unit. As we are using a Tokenizer developped by Helmut Schmid (IMS, University of Stuttgart), the period disambiguation is solved to a satisfying extent [20]. In the case of biochemistry related technical texts the crucial point is to determine the token boundaries, as many of the technical terms are multiword expressions. This task is mainly covered by GenIELex, as the GenIELex group already extracts multiword terms.

[17] 11

[18] The IMS Corpus Workbench is a set of tools for the manipulation of large, linguistically annotated text corpora.

[19] CQP assumes that corpora have been indexed beforehand.

[20] 13 reports an accuracy $> 99.5\%$.

Part-Of-Speech Tagging The process of going through a corpus of sentences and labelling each word in each sentence with its part of speech is called Part-Of-Speech Tagging. Within GenIE we are using the TreeTagger developped by Helmut Schmid [21] (IMS, University of Stuttgart). This tagger estimates transition probabilities using a decision tree. The implementation reached an accuracy of 96.36% on Penn-Treebank data. We regard this as reachable on thoroughly preprocessed biochemistry related corpora. The Part-Of-Speech Tagging suffers the same lexical gap as the Tokenizer, and will be captured by GenIELex, too.

Chunking Text chunking consists of dividing a text in syntactically correlated non-overlapping phrases (sentence parts). Text chunking is an intermediate step towards full parsing and is usually based on finite-state techniques. The considerable advantages a chunker has are that robustness and speed are primary design considerations. In addition precision is at a high level [22], too. Within GenIE we use Steven Abneys Cass partial parser [23]. It is an already approved system, that allows the user to modify easily the cascading set of rules and adjust the system to give it a seperate identity. In our case a biochemistry related identity.
As long as incomplete results suffice, we will prefer chunking instead of full parsing [24].

Full parsing LoPar is parser for probabilistic context-free grammars (PCFG) and head-lexicalised probabilistic context-free grammars (HPCFG) by Helmut Schmid [25] (IMS, University of Stuttgart). A PCFG is a context-free grammar which assigns a probability to each gammar rule. Such a grammar is able to rank different analysis of a sentence according to their probabilities. But, as PCFG fail to resolve syntactic ambiguities like PP-attachment, information about the lexical heads come into operation.
The parser is fed with a grammar and a corpus and as an output it generates a syntactic language model which allows for parsing and disambiguation of the relevant sentences. The precision is high (around 90%) given a good coverage of the lexicon.

[21] See [14]

[22] Precision of more than 90% are the rule.

[23] www.research.att.comãbney

[24] This reflects our two-dimensional approach, too. Where the leading idea is not to generate more structure than necessary.

[25] See [15]

Ontology An ontology is a description of concepts and relationships, that model a part of the world. What is important in this modelling process depends on what an ontology is for. One of our primary goals is the implementation of an ontology which models biochemical pathways and includes concepts ranging from basic biochemistry to genomics [26]. So, as an example the ontology describes the fact, that anything belonging to the concept *DNA kind* has an *Intron* and an *Exon* [27] etc., and it is a member of the concept *Nucleic acid kind*. All properties that are ascribed to the concept *Nucleic acid kind* are also inherited to the concept *DNA kind*. Hence, as a nucleic acid consists of nucleotides anything belonging to *DNA kind* consists of nucleotides, too.
Altogether the ontology has a manifold scope of duties:

- it establishes the backbone for inferences within the information extraction system (as illustrated already in Section 2),
- it represents joint terminology and logical interface between the lexicon, the database system and the extracted information, and
- it is relevant for the consistency of the extracted data and the database system.

The database system consists of (i) an Ontology Managemant System, i.e. a tool to build and manage ontologies based on a logic based language (KIF); (ii) a Deductive Database Generator, which generates a deductive database from the specification of the ontology and provides an Application Programming Interface (API); (iii) a deductive database which follows the semantics of the Ontology, is highly optimized, has a type system and allows for consistency checking.

[26] This work is being carried out together with Ontology Works, Hanover, Maryland U.S.A..
[27] Procaryotic DNA is regarded as a special case where Introns have legth 0.

6 References

1. Paul Proost, Sofie Struyf, Dominique Schols, Ghislain Opdenakker, Silvano Sozzani, Paola Allavena, Alberto Mantovani, Koen Augustyns, Gunther Bal, Achiel Haemers, Anne-Marie Lambeirand Simon Scharp, Jo Van Damme, and Ingrid De Meester. Truncation of macrophage-derived chemokine by cd26/ dipeptidyl-peptidase iv beyond its predicted cleavage site affects chemotactic activity and cc chemokine receptor 4 interaction. *The Journal of Biological Chemistry*, 24:3828–3835, 1995.
2. Youngja Park Roy. Hybrid text mining for finding abbreviations and their definitions.
3. W. M. Byrnes, W. Hu, E. S. Younathan, and S. H. Chang. A chimeric bacterial phosphofructokinase exhibits cooperativity in the absence of heterotropic regulation. *The Journal of Biological Chemistry*, 274:3988–3993, 1999.
4. Hans Kamp and Uwe Reyle. *From Discourse to Logic*. Kluwer, 1993.
5. Achille C. Varzi. Parts, wholes, and part-whole relations: The prospects of mereotopology. *Data and Knowledge Engineering*, 20:259–286, 1996.
6. S. M. Shieber M. Dalrymple and F. C. N. Pereira. Ellipsis and higherorder unification. *Linguistics and Philosophy*, 14:399–452, 1991.
7. Nicholas Asher. *Reference to Abstract Objects in Discourse*. Kluwer Academic Publisher, 1993.
8. Michael Schiehlen. *Semantikkonstruktion*. PhD thesis, Universität Stuttgart, 1999.
9. Jeroen Groenendijk and Martin Stockhof. Dynamic predicate logic. *Linguistics and Philosophy*, 14:39 – 100, 1991.
10. Harald Baayen. *Word frequency distributionse*. Kluwer Academic Publishers, 2001. to appear.
11. Ingrid Meyer. Extracting knowledge-rich contexts for terminography: A conceptual and methodological framework. In Marie-Claude L'Homme Didier Bourigault, Christian Jacquemin, editor, *Recent advances Computational Terminology*, chapter 14, pages 279–302. Kluwer, 2001.
12. Ciprian Gerstenberger. Semantische analyse von namen organischer verbindungen, 2001. to appear.
13. Helmut Schmid. Unsupervised learning of period disambiguation for tokenisation, 2000.
14. Helmut Schmid. Probabilistic part-of-speech tagging using decision trees. In *International Conference on New Methods in Language Processing*, Manchester, UK, 1994. unknown.
15. Helmut Schmid. Lopar: Design and implementation, 2000.