# Partiality and Underspecification in Information, Languages, and Knowledge

# Partiality and Underspecification in Information, Languages, and Knowledge

Edited by
Henning Christiansen,
M. Dolores Jiménez-López,
Roussanka Loukanova
and Lawrence S. Moss

Cambridge
Scholars
Publishing

Partiality and Underspecification in Information, Languages, and Knowledge

Edited by Henning Christiansen, M. Dolores Jiménez-López, Roussanka Loukanova and Lawrence S. Moss

This book first published 2017

Cambridge Scholars Publishing

Lady Stephenson Library, Newcastle upon Tyne, NE6 2PA, UK

British Library Cataloguing in Publication Data
A catalogue record for this book is available from the British Library



ISBN (10): 1-4438-7947-9
ISBN (13): 978-1-4438-7947-7

# Contents

# Preface

This book includes chapters based on selected and extended versions of papers presented at the special sessions on *Partiality, Underspecification, and Natural Language Processing*, PUaNLP 2015 and PUaNLP 2016, of the 7th and 8th International Conferences on Agents and Artificial Intelligence, ICAART 2015 and ICAART 2016, which took place in Lisbon, Portugal, 10–12 January 2015, and Rome, Italy, 24–26 February, 2016, respectively. These chapters have been complemented by other invited contributions in order to give a more representative characterization of the topics of the special sessions PUaNLP. We begin by a brief motivation of the increasing interest in partiality and underspecification, especially in computational approaches to natural language.

## 1 Intelligent, Computerized Systems

In recent years, there has been a proliferation of technological developments that incorporate processing of human language. Hardware and software can be specialized for designated subject areas, and computational devices are designed for a widening variety of applications. At the same time, new areas and applications are emerging by demanding *intelligent* technology enhanced by processing of *human language*. These new applications often perform tasks which handle information, and they have a capacity to reason, using both formal and human language. Many sub-areas of Artificial Intelligence (AI) demand integration of Natural Language Processing (NLP), at least to some degree. Furthermore, technologies require coverage of known as well

as unknown agents, and tasks with potential variations. All of this takes place in environments with unknown factors.

Adequate, computerized processing of human language encounters difficult problems related to its ambiguity. Many researchers in NLP regard ambiguity in human language as a drawback for computerization. But the fact is that ambiguity reflects an important feature of language: it enables us to use one utterance to communicate many things simultaneously. One might even take ambiguity to be a design feature of natural language rather than a bug. The same may be said of uncertainty in natural language; in a sense, uncertainty is a generalization of ambiguity.

In general, partiality, underspecification, and context-dependency are signature features of information in nature and natural languages (NLs). Furthermore, humans and computational systems interact by exchanging and processing information. They engage in information transfer as producers, conveyors, and interpreters.

## 2 Scope of the Book and Potential Readers

The book covers theoretical work, advanced applications, approaches, and techniques for computational models of information, reasoning systems, and presentation in language. The book promotes work on intelligent natural language processing and related models of information, thought, reasoning, and other cognitive processes. The topics covered by the chapters prompt further research and developments of advanced systems in the areas of logic, computability, computational linguistics, cognitive science, neuroscience of language, robotics, artificial intelligence, etc.

Potential readers include researchers and developers working on theory and applications in areas such as: Natural Language Processing (NLP); reasoning that involves human language; mathematical foundations of NLP and reasoning; computational approaches to formal and human language; AI that covers NLP or uses NLP, or is otherwise related to human language; and other related areas.

# 3 The Chapters

**Chapter I: A Natural Logic for Natural-Language Knowledge Bases.** This chapter addresses logical representation of information and corresponding reasoning, in knowledge bases built on fragments of natural language. The target is the development of a Natural Logic, as a formal system, which combines the readability of natural language with the rigorous computational reasoning of a formal logic. The work takes the perspective of a practical compromise between language expressiveness, systems' complexity, and computational tractability. As a Natural Logic, the approach builds upon stylized fragments of natural language. It deals computationally with restricted forms of affirmative sentences of natural language. It employs deductive reasoning that uses intuitive, logical rules. The idea is to avoid complex formal logics, either first-order predicate calculi or advanced type-theories, but still to provide a rigorous approach. In a word, the goal is to be simple and natural, while also rich. The chosen Natural Logic determines the semantic range by restricting it to partial coverage of the considered sentences. The methodology in the chapter uses a top-down process governed by the Natural Logic. The sentences of the logic are represented by graph structures, which are joined into a formal ontology.

**Chapter II: Paradoxes of Material Implications in Minimal Logic.** The authors investigate paradoxes of material implication and classify them with respect to minimal logic, by providing proofs of equivalence and also semantic models. They also separated various paradoxes. The equivalences and separation results reveal enhanced logical power, i.e., taking paradoxes as axioms bring in additional proof-theoretic strength. They show that a number of equivalent groups collapse with unrestricted use of double negation elimination. The principle *ex falso quodlibet* supports minimal logic for reasoning in the possible presence of inconsistency, and in this chapter *ex falso quodlibet* is distinguished as a very interesting principle.

Interest in material implication stems from the philosophical origins of the topic, and it extends to applications in linguistics and computer science. For example, databases and other information systems accumulate inconsistencies, and so there is a need to develop logical systems which can handle them appropriately. Another source of potential applications is reasoning systems coming from AI.

**Chapter III: Guesswork? Resolving Vagueness in User-Generated Software Requirements.** The chapter represents ongoing work at the Collaborative Research Center "On-The-Fly (OTF) Computing", on techniques and processes for automatic ad-hoc configuration of individual services. The services operate on customer-specific requirements expressed in natural language. As always, the language includes uncertainties. The authors discuss the challenges of user-generated software requirements, related definitions, and explanations. Then, they give an overview of general approaches to disambiguation. The aim of the work is to learn patterns from the requirements expressed by users in natural language, and then to use those patterns to detect ambiguity, vagueness, missing information, and underspecification, The authors' approach includes an interactive component, with the user and the system as participants. The focus is on resolution of vagueness.

**Chapter IV: Partiality, Underspecification, Parameters, and Natural Language.** This chapter introduces linguistic concepts that play crucial roles in various areas of computational linguistics. It discusses the mathematical foundations of such concepts and related technological advances. The presentation is based on an overview of the Constraint-Based Lexicalized Grammar (CBLG) approach to computational grammar of natural language. CBLG has established solid linguistic knowledge and continues to be an active, open area of research and technical developments. Prominently, various CBLG grammars, e.g., HPSG, integrate partiality and underspecification, in lexicon, syntax, semantics, and in interfaces between these modules. CBLGs have potentials for integration with data-oriented, machine learning methods.

**Chapter V: Typed Theory of Situated Information and its Application to Syntax-Semantics of Human Language.** The chapter introduces a formal language for a type-theory of situated information. The formal expressions represent situation-theoretic objects taking place in time and 3-dimensional space. Constrained computations are formalized by specialized terms representing mutual recursion, which can be complemented by sub-terms for constraints. Some language terms designate networks of parameters that are simultaneously constrained to satisfy restrictions.

The chapter gives an outlook for applications to the representation of semantic content of human language, via syntax-semantics in Constraint Based Lexicalized Grammar.

**Chapter VI: All and Only.** As the title suggests, the chapter is on the words "all" and "only", although it is more of a contribution to natural logic than to lexical semantics. It takes as a point of departure the standard formulations of the "relational syllogistic", the basic logical language for dealing with relations. In linguistic terms, this language is important because in it, one can express very simple reasoning patterns about transitive verbs. The chapter in this volume re-formulates this relational syllogistic, starting with the word "only" instead of "all." This leads to new work on natural logic, which is the contribution of this chapter.

**Chapter VII: Formalization of Many-Valued Logics.** Classical Montague semantics is a development in classical logic, and so it is based on two-valued logic. The chapter departs from two-valued logic. It thus extends the underlying logic in semantics to various kinds of many-valued logics, thereby allowing for a more fine-grained partial semantics for propositional attitudes like assertion, knowledge, and belief. The proof assistant Isabelle is used as formalization of semantics. This allows for a concise presentation and brings in verification results.

**Chapter VIII: Comparative performance analysis of selected VSM and Ontology-based Text Classification Methods.** The chapter addresses the challenging tasks of categorizing large collections of texts. Based on work with test-sets, it compares two different approaches — the Vector Space Model (VSM) and ontology-based solutions — with respect to accuracy, computing speed, and how processing flow affects the classification results. The work employs the algorithms of VSM with enhancements, e.g., TFIDF (Term Frequency Inverse Document Frequency). Problems related to the large dimensionality plague most common reduction algorithms, i.e., Latent Semantic Indexing (LSI) and Random Projection (RP). The authors address this with GPU-based acceleration of their techniques. They performed a series of tests by comparing the methods and corresponding algorithms on test-sets of 2.8 million Wikipedia articles. The work revealed important properties of the algorithms and their accuracy.

The conclusion is that the ontology-based method outperformed others, with respect to the number of categories which can be processed. On the other hand, classifying using with support vector machines (SVMs) is much faster and performs well when appropriately trained.

**Chapter IX: Dynamic Semantic Update for Underspecified Information: A Context-based Interpretation of Ellipsis Constructions in Mandarin.** The chapter presents a structural representation of a range of elliptical constructions in Mandarin, such as Null Object Construction, English-like VP ellipsis construction, *shi*-support elliptical construction, gapping construction, and fragmentary constructs in dialogue. The authors take an interpretative perspective. An elliptical construction is analyzed as a pro-form (a structural generalization of pronominal forms) with underspecified content. The underspecified information is updated interactively by a structural context and pragmatics. The approach builds upon the framework of Dynamic Syntax, by a technical definition of the nation of a context consisting of tree structures. Dialogue fragments can be represented by extending the record of appropriately-defined structural trees. The authors show that syntactic and pragmatic processes interactively determine the underspecified content of the considered elliptic constructions in Mandarin. This analysis provides a formal characterization of a variety of elliptical constructions, including a representation of their structural characteristics, e.g., ambiguities between the strict and sloppy reading.

## 4 Prospects for Future Work

The last decades of research and concurrent technological progress have seen significant achievements in NLP. At the same time, we are somewhat disappointed, especially regarding adequate and intelligent processing of human language by computational systems. Many, if not most, of the difficulties are related to ambiguities, vagueness, and context dependence in human language, in all aspects, and how the corresponding computational theories deal with related issues in the entire language strata — phonology, phonetics, morphology, lexicons, syntax, semantics. The primary difficulties are rooted in semantics and its interrelations with all other aspects of of human language, including its role in reasoning.

We hope that the chapters of this book will promote more work on their topics, and in related areas, too. We eagerly anticipate new results and applications.

## Acknowledgments

We are grateful to the contributing authors who made this book possible, and to Cambridge Scholars for publishing it and providing help during the process.

The Editors

# Chapter I

# A Natural Logic for Natural Language Knowledge Bases

Troels Andreasen
Henrik Bulskov
Per Anker Jensen
Jørgen Fischer Nilsson

*For better or worse, most of the reasoning that is done in the world is done in natural language.*
G. Lakoff: Linguistics and Natural Logic, *Synthese*, **22**, 1970.

## 1 Introduction

This chapter[1] addresses representation of information in knowledge bases. Our focus is on a logical representation language enabling reasoning with knowledge base information. The overall ambition is to

[1] The chapter is an improved and much extended version of [4] and a continuation of earlier work published in [2, 3, 20].

unify the readability of natural language with the rigorous computational reasoning of formal logic in a knowledge base system. Specifically, we endeavour to reach a practical compromise between language expressivity, systems complexity and computational tractability.

Traditionally, computational reasoning with information given in natural language is carried out by conducting a translation of sentences into an appropriate formal logic. A common choice is first order predicate logic, see e.g., [11,15], or derivatives of predicate logic such as a description logic dialect, as in [10, 27]. There are also advanced approaches which call on forms of logical type theory, thereby taking advantage of an assumed compositional semantics for natural language drawing on higher order denotations [12]. Further, there are natural logic approaches which extend syllogistic proof systems [22] as discussed in Section 6.

Our approach relies on appropriate forms of natural logic. Natural logics are stylized fragments of natural language in which the deductive logical reasoning uses simple, intuitive rules. That is to say, we dispense with predicate-logical reasoning systems such as resolution. Natural logic originates from the traditional Aristotelian categorial syllogistic logic [14, 20, 28], which became further developed and refined in late medieval times. Various extensions of syllogistic logics and proof systems are analyzed in [22]. However, in the course of the late 19th century, forms of logic coming close to natural language were largely deemed obsolete by Peirce's and Frege's introduction of the more general, mathematically inclined, quantifier-based predicate logic. As is well-known, the latter subsequently prevailed throughout the 20th century.

Here we pursue the idea of choosing natural logic as a target language for dealing computationally with appropriately constrained and regimented, yet rich, forms of affirmative sentences in natural language. The purported methodological advantage of the present approach lies in the proximity of natural logic to natural language, very much in contrast to predicate logic. The predicate logical edifice of quantified variables and function-argument structure aligns well with mathematical requirements, but is far from the pervasive subject-predicate structure of natural language assertions, see also [18, 24].

Indeed then, when adopting natural logic, the translation of the considered natural language fragments becomes a partial recasting of the considered sentences into an even smaller formal language fragment, namely natural logic sentences. Thus, the chosen natural logic

determines and confines the semantic range for partial coverage of the considered sentences. The relationship between natural language found in scientific corpora and natural logic is handled in a top-down process governed by the natural logic. The natural logic sentences are internally decomposed into graph structures which are joined into a formal ontology.

## 2 Semantic Framework

In our framework, the applied natural logic forms express binary relationships between classes. Thus, our semantic framework comprises a selection of stated classes of entities together with binary relationships between the classes akin to the popular entity-relationship models. The entity classes are classes of physical entities as well as abstract entities such as events, processes and causes. Classes here are conceived intensionally, meaning that classes may be distinct even though they contain the same members, unlike sets. Moreover, in our strong intensional view, a pair of classes are recorded as distinct in the knowledge base even if they comprise the same subordinate and superordinate classes in the knowledge base. This strong form of intensionality, sometimes called hyper-intensionality, is achieved not by possible world notions, but rather by introduction of a meta-logic in which the developed knowledge base language is encoded. However, in the underlying predicate logical explication of the natural logic, given below, eventually classes become reduced to sets in the usual model-theoretic explication.

Among the class-class relations, first of all, there is the fundamental isa subclass relationship known from formal ontologies and typically expressed as copula sentences. In addition, class-class relationships may be introduced according to need, as discussed in [9, 23, 25, 30]. The subclass relationship is illustrated in Figure 1 by a fragment of an ontology of given classes named by lexical items in natural language, saying e.g., beta_cell isa cell, insulin isa hormone, etc. Thus, from the viewpoint of natural logic, this ontological graph is simply a collection of copula sentences.

It is a key feature of our natural logic approach that the given lexicalized classes may be used generatively to form subclasses *ad libitum* by restriction with relationships to other classes in the natural logic. As an example illustrated in Figure 2, given the classes cell and hormone, we may attach a relation like produce to form the subclass

cell that produce hormone. This is a phrase in the applied natural logic forming a new class which is a subclass of cell. The restrictions allow a potentially recursive structure reflecting the recursive structure of natural language phrases and natural logic phrases. An early attempt to formalize ontological generativity by means of grammatical generativity, by way of recursion, is described in [5].

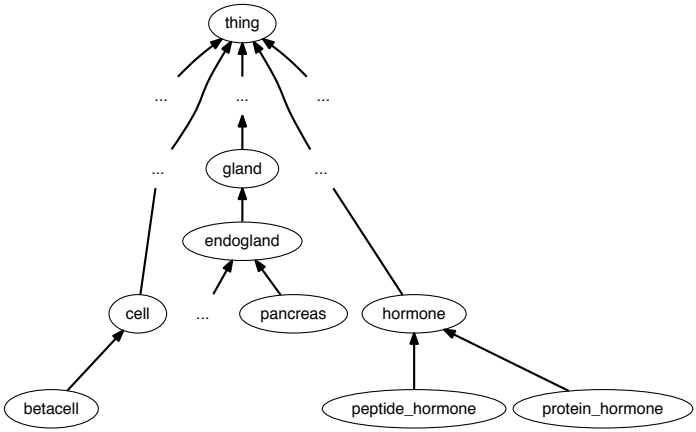


Figure 1: A core ontology fragment from a life science domain.

Unlike what is the case in predicate logic, in our framework, the entities belonging to the classes are not dealt with explicitly. Individual entities may, however, if necessary, be dealt with as stipulated singleton classes. In our setup, such singleton classes possess no subclasses, not even the empty class. We assume all classes mentioned in sentences in the knowledge base to be non-empty, since the empty class makes no sense in an empirical domain.

Classes may also emerge by natural language derivational processes such as nominalization, so that for instance a verb *secrete* may have an associated class *secretion*. Such an association may be handled as a relation between the two terms at the metalogical level.

So far we have dealt with the nominal parts of copula sentences and their representation as graphs cf. Figure 1 and 2. Further, it is a crucial feature of our approach that arbitrarily complex sentences in our natural logic are represented as graphs in a manner which preserves the semantics of the sentences. This calls for introduction

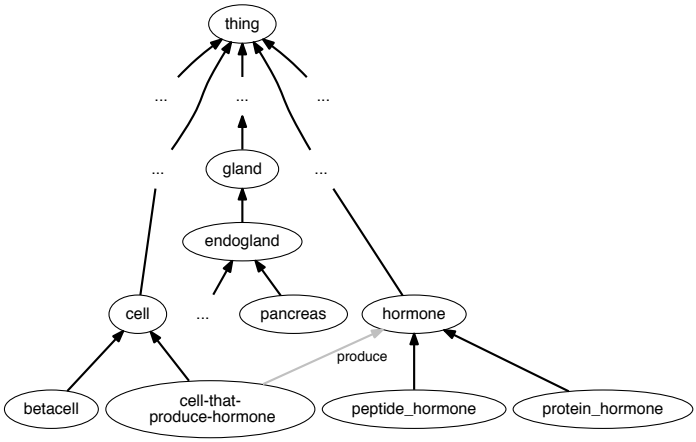


Figure 2: The subclass cell that produce hormone coming about by means of a relation produce.

of new class terms as exemplified above by cell-that-produce-hormone. This issue leads to the notion of core natural logic addressed below in Section 3.

This semantic framework, while being application-neutral at the outset, appears to be particularly useful for applications within the bio-sciences as discussed in [23, 25] as well as in our [1–3, 6, 20].

Generally speaking, natural language descriptions in the empirically oriented natural sciences abound with classes, let us just mention the historical Linnean taxonomies and contemporary chemical and medical taxonomies. However, the present framework enriches and transcends the traditional taxonomic classifications by the presence of multiple other relations. In particular it supports partonomies by introduction of partonomic class-class relationships, cf. [5, 26].

# 3 Core Natural Logic

Having introduced the semantic framework, we turn next to the logical sentences supporting the mentioned class-relationship setup. We consider a natural logic which has the general form:

$$Q_1\ Cterm'\ R\ Q_2\ Cterm''$$

In this sentence template

- $Q_i$ are quantifiers (determiners) every/all, some/a,
- the grammatical subject term $Cterm'$ and the grammatical object term $Cterm''$ are class expressions, and
- $R$ is a relation name.

In linguistic parlance, the $Cterm$s are noun phrases, and $R$ is a transitive verb. In the simplest cases, $Cterm$s are just class names (nouns) $C$. Below we describe how class restrictions, as exemplified in Figure 2, can be represented by $Cterm$s.

Among the quantifier options here, we focus on the quantifier structure

$$\text{every } Cterm' \ R \text{ some } Cterm''$$

This general form is pivotal in our treatment because it represents the default interpretation of sentences like *betacells produce insulin* in ordinary descriptive language. A general explication of this form in predicate logic with quantifier structure $\forall x \exists y$ is straightforward, cf. our references above and, therefore, not repeated here. However, let us look at a particular example every betacell produce some insulin and its relation to predicate logic:

$$\forall x(betacell(x) \rightarrow \exists y(produce(x, y) \wedge insulin(y)))$$

The linguistically inherent, structural ambiguity corresponding to the scope choice $\forall x \exists y$ versus $\exists y \forall x$ is overcome by stipulating the $\forall x \exists y$ reading, which is the useful one in practice.

Our natural logic notations rely on the convention that if no quantifiers are mentioned explicitly, the interpretation follows the scope pattern $\forall x \exists y$. Thus, the sentence betacell produce insulin is semantically equivalent to our sample sentence above, every betacell produce some insulin. Note further that we persistently use uninflected forms of nouns and verbs rather than morphologically correct forms in our natural logic expressions.

Observe that we are not going to translate the sentences into their predicate logical form with individual variables $x$ and $y$. Rather, it is a crucial feature of our approach that we decompose the sentences into simpler constituents forming a graph without variables as, explained in Section 3.4.

### 3.1 Subclass through Copula Sentence

In the above natural logic affirmative sentence template $Q_1\ Cterm'\ R\ Q_2\ Cterm''$, there is an extremely important sub-class, namely, the copula sentence form:

$$Cterm'\ \mathsf{isa}\ Cterm''$$

as a shorthand for $\mathsf{every}\ Cterm'\ \mathsf{isa\ some}\ Cterm''$, generalizing the categorial syllogistic proposition $\mathsf{every}\ C'\ \mathsf{isa}\ C''$. We say that the class $Cterm'$ specializes the class $Cterm''$, and, conversely, that $Cterm''$ generalizes the class $Cterm'$. As explained in [19, 20], the copula sentence form may actually be understood as a special case of the general sentence template with the relation being equality. For example, the sentence $\mathsf{betacell\ isa\ cell}$ is predicate logically construed as

$$\forall x(betacell(x) \rightarrow \exists y(x = y \wedge cell(y)))$$

giving in turn

$$\forall x(betacell(x) \rightarrow cell(x))$$

Note once again that we are not using these predicate logical forms in our reasoning with natural logic. The categorial syllogistic negative proposition $\mathsf{no}\ C'\ \mathsf{isa}\ C''$ is not available in our setup, since class disjointness is assumed initially for pairs of classes by default, cf. [20], which also discusses the relation to the well-known square of opposition in traditional logic. The upshot of our convention is that two classes are disjoint unless one is stipulated as a subclass of the other, or that they have a common subclass introduced by the copula form. Recall that the empty class is absent. This default convention on disjointness conforms with the use of classes in scientific practice, as reflected in formal ontologies. However, one may observe that the convention deviates from the description logic principle, which follows predicate logic with the open world assumption.

### 3.2 Simple and Compound Class Terms

Compound terms $Cterm$ in the natural logic take the form of a class name $C$ optionally modified by various forms of restrictions giving rise to a virtually unlimited number of subclasses of $C$. Linguistically, this "generativity" is provided by constructions like restrictive relative clauses and adnominal prepositional phrases (PPs).

Accordingly, in the present context, we consider $Cterm$ in the form of a class name (noun) $C$ optionally followed by

- a stylized relative clause: that $R$ $Cterm$
  or optionally by
- a PP in the logical form $R_{prep}$ $Cterm$, in turn optionally followed by a relative clause.

The relation $R_{prep}$ is to be provided by the pertinent preposition in the applied vocabulary.

Sample class terms illustrating these patterns are:

cell
cell that produce hormone
cell in pancreas
cell in pancreas that produce hormone

Ontologically, these four classes form a (trans-hierarchical) diamond by the isa subclass relation as shown in Figure 3.

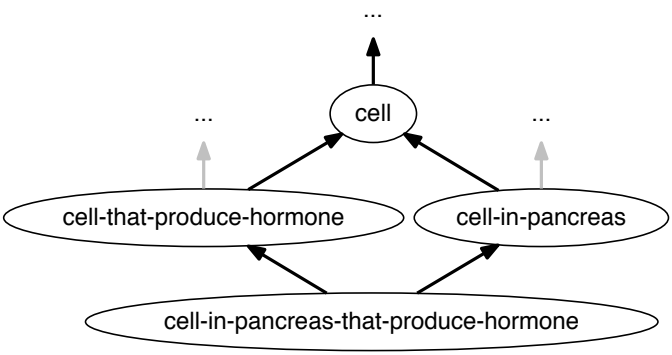


Figure 3: A trans-hierarchical classification by way of compound terms.

The sample class term cell in pancreas that produce hormone is structurally aligned as in cell (in pancreas) (that produce hormone).

We disregard the more tricky restrictions provided by adjectives (even when assumed to behave restrictively), noun-noun compounds[2], and genitives. This is because these constructs, unlike the case of relative clauses, do not explicitly single out a specific relation $R$, cf. the discussions in [13,29]. As described below, we also admit conjoined constructions with the conjunction *and* in class terms. See also [2] for various extensions of the natural logic.

[2]Some class names (given terms) in an application may consist of more than one word, but are still to be considered as simple, fixed terms.

### 3.3 Atomic Natural Logic

In our framework, the core natural logic introduced above is decomposed into atomic natural logic devoid of compound class names, that is, *Cterm* is simply a class name $C$. The natural logic sublanguage where *Cterm* is simply a class name we call *atomic natural logic*. The decomposition into atomic natural logic is accomplished by introduction of fresh, internal class names such as cell-that-produce-hormone, which are formally conceived of as class names. The term cell-that-produce-hormone is defined by two atomic natural logic sentences

cell-that-produce-hormone isa cell

cell-that-produce-hormone produce hormone.

The knowledge base of the decomposed sentences may be viewed as one single labeled graph whose nodes are uniquely labeled with class names.

### 3.4 Reasoning with Natural Logic

The key inference rules for the considered natural logic are the so-called monotonicity rules [28]. They very intuitively admit specialization of the grammatical subject class and generalization of the grammatical object class. Accordingly, given $A$ isa $B$ and [every] $B$ $R$ [some] $C$, one may derive [every] $A$ $R$ [some] $C$, that is, as inheritance inference rule

$$\frac{A \text{ isa } B \quad [\text{every}]\ B\ R\ [\text{some}]\ C}{[\text{every}]\ A\ R\ [\text{some}]\ C}$$

This rule provides inheritance of properties to subclasses, as known from the object-oriented paradigm. Moreover, given [every] $B$ $R$ [some] $C$ and $C$ isa $D$, one may derive [every] $B$ $R$ [some] $D$, that is, as generalization inference rule

$$\frac{[\text{every}]\ B\ R\ [\text{some}]\ C \quad C \text{ isa } D}{[\text{every}]\ B\ R\ [\text{some}]\ D}$$

In particular, these rules provide transitivity of the isa subclass relationship with $R$ being isa. Figures 4 and 5 provide examples of the inheritance and generalization rules, respectively. Figure 4 shows that given pancreas is a endogland and endogland produce hormone it may be inferred that pancreas produce hormone. Similarly for Figure 5 and generalization.

In addition, we provide a subsumption inference rule which makes the properties assigned to classes act restrictively as detailed with an

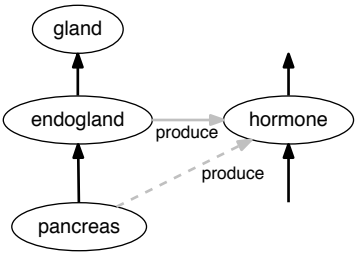


Figure 4: Sample use of the inheritance rule.

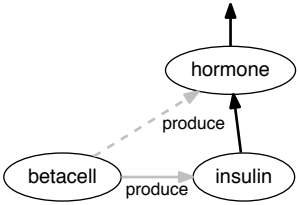


Figure 5: Sample use of the generalization rule.

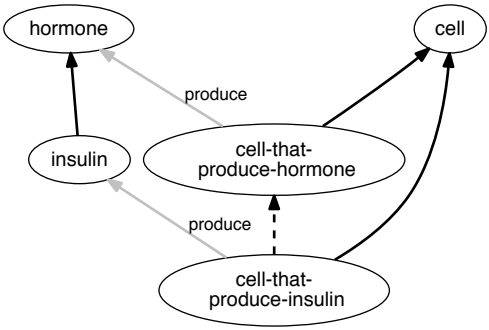


Figure 6: Inferred inclusion by subsumption.

algorithm in [3]. By way of example, the subsumption rule ensures that

cell that produce insulin

is recognized as a specialization of

cell that produce hormone,

given that insulin isa hormone, cf. Figure 6.

These rules are very intuitive and are applied in daily life, common sense reasoning. Formally, they can be justified by reduction of the involved sentences to first order logic.

Except for those relationships that follow from transitivity, we make sure that all valid isa relationships between nodes are materialized in the graph by the subsumption rule, so that, for instance,

cell-that-produce-insulin isa cell-that-produce-hormone

is recorded. As such, the graph appears as an extended formal ontology with the isa relationship forming the skeleton, as it were.

# 4 Extending Core Natural Logic

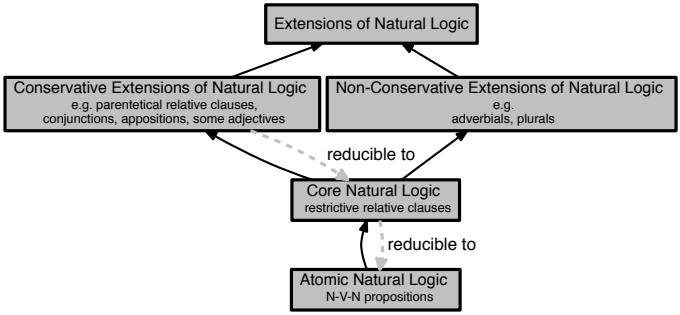


Figure 7: Natural logic classes and their reducibility to Atomic Natural Logic.

In the previous sections, we have introduced the core natural logic and the sublanguage atomic natural logic. We now turn to extensions of core natural logic in order to capture larger fragments of natural language within the considered scientific domains. The proposed extensions are conservative and, thus, do not increase the semantic range of the core natural logic as stipulated above. The point is, however, that the extensions capture paraphrases commonly encountered in natural language. These extensions, when taken jointly, form an extended natural logic coming closer to free natural language formulations, remaining, however, within the confines of the semantics of core natural logic. The extensions facilitate translation from select domain specific natural language constructs into core natural logic.

### 4.1 Extension with Conjunctions

Let us first consider conjunctions of *Cterm*s including the linguistic conjunction *and* assuming distributive (in contrast to collective) readings. A conjunction in the grammatical object

$Cterm'\ R\ Cterm''_1$ and $Cterm''_2$

as in

pancreas contain betacell and alphacell

straightforwardly gives rise to the decomposition:

$Cterm'\ R\ Cterm''_1$
$Cterm'\ R\ Cterm''_2$

A conjunction in the grammatical subject

$Cterm'_1$ and $Cterm'_2\ R\ Cterm''$

as in

betacell and alphacell is-contained-in pancreas

is conventionally interpreted as disjunction rather than overlap of the two classes and, therefore, decomposed into

$Cterm'_1\ R\ Cterm''$
$Cterm'_2\ R\ Cterm''$

These conventions are justified by the underlying predicate logical explication of core natural logic. Our general logical construal here is

$\forall x(Cterm'_1(x) \vee Cterm'_2(x) \rightarrow \exists y(R(x,y) \wedge Cterm''(y)))$

which is logically equivalent to

$\forall x(Cterm'_1(x) \rightarrow \exists y(R(x,y) \wedge Cterm''(y))) \ \wedge$
$\forall x(Cterm'_2(x) \rightarrow \exists y(R(x,y) \wedge Cterm''(y)))$.

Now we turn to linguistic disjunctions. The linguistic disjunction *or* in the linguistic subject seems irrelevant from the point of view of the considered domains. It may be considered a case where predicate logic covers more than needed.

More interesting are disjunctions in the grammatical object, viz.,

$Cterm'\ R\ Cterm''_1$ or $Cterm''_2$

which cannot simply be decomposed into two core natural logic sentences. One approach is to appeal to a common general term $Cterm_{sup}$ for $Cterm''_1$ and $Cterm''_2$ if one is available in the KB. More precisely, one seeks a $Cterm_{sup}$, such that

$Cterm''_1$ isa $Cterm_{sup}$
$Cterm''_2$ isa $Cterm_{sup}$

and such that for all different $Cterm_x$ having these properties $Cterm_{sup}$ isa $Cterm_x$. This supremum requirement seems reasonable

in cases where the considered disjunction is pragmatically relevant at all. For instance, left-eye or right-eye has supremum eye since left-eye isa eye and right-eye isa eye.

Notice that all of the above reductions of conjunctions endorse the desired commutativity and associativity properties.

It goes without saying that the presence of conjunctions together with relative clauses and prepositions gives rise to structural ambiguities. Introduction of appropriate default readings and/or addition of auxiliary parentheses are the simplest ways to eliminate these.

Collective, i.e., non-distributive, readings such as co-presence of $A$ and $B$ cause $C$ call for separate treatment, which goes beyond the scope of the present approach.

### 4.2 Extension with Appositions and Parenthetical Relative Clauses

Let us consider natural logic sentences

$Cterm'$ $R$ $Cterm''$

extended with appositions bounded by commas

$Cterm'$ , [a|an] $Cterm_{appo}$ , $R$ $Cterm''$

as in

insulin, a hormone, affect metabolism

This is paraphrased into the pair

$Cterm'$ $R$ $Cterm''$

$Cterm'$ isa $Cterm_{appo}$

as in

insulin affect metabolism

insulin isa hormone

Analogously, we add sentences with apposition in the grammatical object, $Cterm''$, as in

betacell produce insulin, a peptide hormone

which is spelled out as

betacell produce insulin

insulin isa peptide hormone

In our extended natural logic, the pronoun that is formally set off for restrictive relative clauses as accounted for above. By contrast, in parenthetical relative clauses, we use which together with commas

$Cterm'$ , which $R_{par}$ $Cterm_{par}$ , $R$ $Cterm''$

as in

insulin, which isa peptide hormone, affect metabolism

Retaining logical equivalence, this can be paraphrased into the expressions

$Cterm'\ R\ Cterm''$

$Cterm'\ R_{par}\ Cterm_{par}$

which can be spelled out in an identical manner to the sentences containing appositions, cf.

betacell affect metabolism

insulin isa peptide hormone

This applies similarly and recursively for $Cterm''$ and $Cterm_{par}$.

### 4.3 Non-conservative Extensions of Natural Logic

On our agenda for non-conservative extensions of core natural logic are passive voice verb forms, nominalisation, and plural formation. As far as negation is concerned, we rely throughout on the closed world assumption in the query answering.

Scientific texts abound with constructions of verbs or verb phrases modified by adverbials. Adverbials are notoriously difficult, if not impossible, to cope with within first order predicate logic. However, our natural logic can be extended with relations between relations, with appropriate inference rules, without taking resort to higher order notions. By way of example, verbs should be allowed extensions with adverbial PPs yielding restricted relations, $Rterm$, for plain $R$ as in

$A$ produce in pancreas $B$

with variants $A$ produce $B$ in pancreas and in pancreas $A$ produce $B$, with obvious additional structural ambiguity problems.

Of course, there are numerous genuine (that is, non-conservative) language extensions which go beyond the semantic range of core natural logic, even within the given scope of monadic and dyadic relations in affirmative sentences. Thus, admission of anaphora as in the infamous donkey sentences breaks the boundaries, as mentioned in [14]. An example of this is seen in *every cell that has a nucleus is-controlled-by it*. The point is that in the applied natural logic, the subject noun term and the object noun term are independent, connected solely by the relator verb and unconnected by anaphora.

## 5 Extracting and Applying Natural Logic

Extraction of natural logic from natural language sentences is conceived of as a partial recasting of the considered natural language into a smaller formal language fragment, namely natural logic sentences.

### 5.1 Extracting Natural Logic

Methodologically, rather than the usual forward or bottom-up translation following the phrase structure, we devise a top down processing governed by the natural logic. In this process, we try to cover as much as possible of the considered sentence, in a (partial) "best fit" process.

In the present approach, we simply process the input sentence by sentence. Thus, sentential context is not exploited. Each input sentence is preprocessed for markup, and the result is further processed by a parser that also functions as a natural logic generator. During the preprocessing, the sentence is tokenised into a list of lists, where each word from the sentence is represented by a list of possible lexemes specifying the base form and category (part of speech) of each word. A lexeme is included as possible if it matches the lemma of the input form of the word. Finally, the preprocessing applies a domain specific vocabulary to identify multiword expressions in the input sentence. To ensure that multiwords are treated as inseparable units they are replaced by unique symbols. A preprocessing of the sentence: *Cells that produce insulin are located in pancreas* returns the following tagged and lemmatised word list, where the word sequence *'are located in'* is replaced by the symbol *locatedin*:

{{cell/NOUN}, {that/DETERMINER, that/ADVERB, that/CONJUNCTION}, {produce/VERB}, {insulin/NOUN}, {locatedin/VERB}, {pancreas/NOUN}}

Each possible lexeme for a word (and multiword) $W_i$ is included as an element $L_{ij}/C_{ij}$ if $W_i$ has lemma $L_{ij}$ for category $C_{ij}$.

Our approach to recognizing and deriving natural logic expressions from natural language texts can be considered a knowledge extraction task where the goal is to extract expressions that cover as much as possible of the meaning content from the source text. The search is guided by a natural logic grammar, and the aim is to create propositions that comprise well-formed natural logic expressions. The "best fit" approach is basically a guiding principle aiming for largest possible coverage of the input text. Thus, if an expression, that covers the full input sentence can be derived, it would be considered the "best", and if not, the aim is a partial coverage where larger means "better".

As far as parsing is concerned, and as already mentioned, our approach is a top down processing governed by the natural logic. The word list given as input is ambiguous due to possible multiple categories assigned to each word. Thus, the parser should be able to

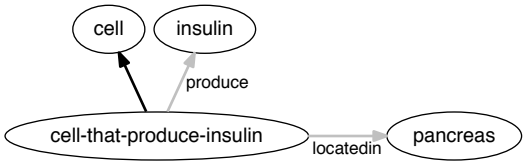


Figure 8: Graph representation of the sentence *cells that produce insulin are located in pancreas.*

recognize an input proposition if such exists for at least one combination of possible lexemes of the input words. Therefore, in addition to processing the grammar (given below), the parser must ensure that all combinations are tried before failing the recognition of a proposition.

Core natural logic, as described in Section 3, without the extensions sketched in Section 4, can be specified by the following grammar.

*Prop* ::= *Cterm R Cterm*
*Cterm* ::= NOUN [*RelClauseterm* | *Prepterm*]
*RelClauseterm* ::= [that|which|who] *R Cterm*
*R* ::= VERB
$R_{Prep}$ ::= PREPOSITION
*Prepterm* ::= $R_{Prep}$ *Cterm*

Notice that terminals are specified as either specific words or word categories.

**Example 1:** The preprocessing result of the example sentence *cells that produce insulin are located in pancreas* is shown above. In this case, the full sentence can be recognized as a proposition with the given grammar, and the following atomic natural logic propositions can be derived from the parse tree:

| | | |
|---|---|---|
| cell-that-produce-insulin | isa | cell |
| cell-that-produce-insulin | produce | insulin |
| cell-that-produce-insulin | locatedin | pancreas |

The corresponding graph is shown in graph form in Figure 8.

**Example 2:** Preprocessing the sentence *insulin is a peptide hormone produced by betacells in the pancreas* leads to: