An anatomy of knowledge representation and a theory of meaning

Maurice Gittens <maurice at gittens dot nl>

22nd December 2002

Abstract

An understanding knowledge and its representation entails an understanding of meaning, any unit of information with an associated meaning represents some unit of knowledge. Similarly any unit of information which represents a fact in some particular context by definition has a meaning in the designated context. This paper introduces a theory of meaning together with an associated vehicle for knowledge representation. The knowledge representation vehicle presented, is basically a generalization of the knowledge representation vehicle used for representing knowledge in intensional type logic. Additionally, a language allowing the expression of arbitrarily complex claims, is introduced together with definitions for the truth-value and the significance of expressions in this language. Finally the notion of a logical document based upon which a formalization of a knowledge acquisition process in introduced. The subject matter presented in this paper will serve as the formal foundation for a system for discourse representation.

Contents

Preface

This paper presents an anatomy of knowledge and its representation together with a definition of a notion of the meaning of a logical expression called the significance of said expression. Based on this the notion of a logical document is defined. Logical documents are intended to serve as the formal foundation of a discourse representation system. The first chapter provides a definition of a knowledge representation model together with associated terminology and concepts. The second chapter introduces a language allowing the expression of arbitrarily complex claims, together with a definition of the truth-value and significance of expressions in this language. Chapter 3 illustrates how first order predicate logic, extensional type logic and intensional type logic relation to the knowledge representation system defined in chapter one. The fourth chapter considers how concepts from Category Theory can be represented by the knowledge representation models introduced in chapter one. The fifth Chapter defines the notion of a logical documents and discourse grammars. Finally the sixth chapter introduces a model for knowledge acquisition called tutoring grammars.

Acknowledgments

Sometimes one book makes a significant difference. In my case “Logica voor alfa’s en informatici” written in the Dutch language by Jan van Eijck and Elias Thijse is such book. Though I have not read many books about formal logic, I have spent many years reading through this book simply because it was lots of fun and there seemed to always be a reason to reread parts of some chapters. This one book together with my own experience with other formalisms has provided me with much of the background background in formal logic on which I base this paper.

Changes

- December 22 2002; OK, thanks to Rene Jansen, I found out about Category Theory. It turns out that morphisms (or arrows) and objects from Category Theory are readily mapped to associators and elements of the Universe of Discourse in the knowledge representation models defined in this paper. So, I added a section on this.

- December 20 2002; Oops, it seems I forgot all about assignments in the definition of the significance of significance of expressions. I forgot about them at some points in the definition of the truth-value of these expressions as well. Fix these problems quickly before anyone notices. Also there were some other issues with this definition that I clean up a bit as well.
- December 18 2002; Unsatisfied with the semantics of significance of expressions of the form $\neg\varphi$. I decided to massage the definition of the Universe of Discourse so that semantics, without the Closed World Assumption, for significance of expressions of the form $\neg\varphi$ is obtained.
- December 17 2002; Change Tutoring Grammars to speak of assert and retract statements instead of assertions and retractions to avoid overloading the definition of an assertion.
- December 15 2002; Define the notion of inheritance for associators. Define *claims* as synonyms of *assertions,* fix inaccuracies.
- December 5 2002; Add the section Über Sinn und Bedeutung. Add examples illustrating intensional constructs, add tutoring grammars; fix some more inaccuracies
- November 11 2002; Add fixes suggested by Elena Mauro; new definition of knowledge representation language including the concept of significance.
- Rewritten the abstract of the document
- September 19 2002. Corrected many inaccuracies, there are probably more still waiting to be found.
- Added section first order predicate logic
- February 25 2002. Finished first definition of the valuation function; much editing
- February 18 2002. First version of the document

Chapter 1

A model for knowledge representation

1.1 Introduction

Let us recognize that, fundamentally, any unit of information, with an associated meaning, represents a unit of knowledge. Elaborating on this chain of thought allows three key concepts to be recognized, namely functions or associators f which associate units of information r to other unit of information i. This recognition motivates the following definition of a model for knowledge representation.

1.2 Definitions

1.2.1 The Universe of Discourse

1.2.1.1 Definition

A *Universe of Discourse* I is defined as or set of strings over an alphabet Σ with the following properties:

1. Each string $i \in I$ has as an associated string, $i^{-1} \in I$ called the inversion of i, for which it holds that $i = (i^{-1})^{-1}$.

2. Associated with each $i \in I$ is an integer called the *address* of i denoted by $address(i)$. It is understood that for no two elements i_1, i_2 of I, it is true that $address(i_1) = address(i_2)$. Additionally is holds that $address(i) = -address(i^{-1})$.

3. $address(\varepsilon) = 0$ where ε represents the empty string. For the inversion of the empty string it holds that $\varepsilon^{-1} = \varepsilon$

1.2.1.2 Terminology

Let I be a Universe of Discourse over an alphabet $\sum$. Each $i \in I$ is called an *identity*. Such an identity is said to represent a *unit of meaning* or an *identifier*. The set of integers representing addresses of elements in I is called the *address space* of M. The address space A of I is denoted by $addressspace(M)$. The *identity function* for I is defined as a function $identity : A \to I$ mapping the address of an element $i \in I$ to the element i such that $i = identity(address(i))$ and $i^{-1} = identity(-address(i))$.

1.2.1.3 Tuples, Sets and Functions in the Universe of Discourse

Let I be a Universe of Discourse over an alphabet $\sum$. When a tuple $t = (t_1, t_2, ..., t_n)$ is an element I, it is understood that t is represented as a string over the form $(address(t_1), address(t_2), ..., address(t_n))$, where each t_i $(1 \le i \le n)$. The inverse t^{-1} is defined as an element in I of the form $(-address(t_1), -address(t_2), ..., -address(t_n))$. Similarly when a mathematical set $s = \{e_1, e_2, ..., e_k\}$ is an element of I, s is represented in I as a string of the form: $\{address(e_1), address(e_2), ..., address(e_k)\}$, where each e_i $(1 \le i \le k)$. The inverse s^{-1} is an element of I of the form: $\{-address(e_1), -address(e_2), ..., -address(e_k)\}$. Where $A = addressspace(I)$, a function $f : A \to A$ is called an *associator*. When an associators f is an element of I the inverse of f is an associator representing the inverse relation of f represented.

[1]An associator is said to define a *semantic perspective* of a universe of discourse. The codomain of an associator f is said to represent the *domain of discourse* of f.

1.2.1.4 Significance tokens

Let I represent a universe of discourse over an alphabet $\sum$. Every $i \in I$ is called *universe specific significance token*. The address of every $i \in I$ is called a *normalized significance token*. A reference to a *significance token* is a reference to a universe specific significance token or a normalized significance token, as determined by the context.

1.2.2 Knowledge representation models

A knowledge representation model M is defined as a quintuple $(\sum, F, I, T, P)$ where:

- $\sum$ represents a set of symbols.
- I Universe of Discourse over Σ
- F subset of I, represents a set of associators
- T subset of I represents a set of tuples tuples of arbitrary arity.
- P subset of I is a set of sets where each $p \in P$ is a called a *tuple set* or a *property set*. Each property set $p \in P$ of cardinality k can be written as $\{address(t_1), address(t_2), ..., address(t_k)\}$ while each $t_i \in T$ $(1 \le i \le k)$ is a tuple of the same arity.

[1]By virtue of the fact that sets of units of meaning are also units of meaning it can always be insured that inverse relation of an associator $f : A \to A$ is also a unary function $f^{-1} : A \to A$ and thus also an associator.

1.3 Some terminology

1.3.1 Referers, assertions, claims and units of knowledge

Let $M = (\sum, F, I, T, P)$ represent a knowledge representation model. Where $f \in F$ and $r, i \in addressspace(M)$, the triple $(address(f), r, i)$ is called an *assertion* or a *claim*. The *inverse assertion* of $(address(f), r, i)$ denoted $(address(f), r, i)^{-1}$ is defined as $(address(f^{-1}), -r, -i)$. An assertion$(address(f), r, i)$ is said to be *true* or *valid* if and only if $f(r) = i$, an assertion is said to be *false* or *invalid* otherwise. A valid assertion is defined to represent a *unit of knowledge*. A unit of knowledge is also called an *association*. If for any $f \in F$, $f(r) = i$ is a unit of knowledge then r is called a *referer* in M, while i is said to be the reference of r according to f.

1.3.2 The significance of assertions and knowledge representation models

Let $M = (\sum, F, I, T, P)$ represent a knowledge representation model. The *knowledge content* or *significance* S of a knowledge representation M, denoted $S(M)$ is defined as the set of units in knowledge of M. Where $f \in F$ and $r, i \in addressspace(M)$, and $(address(f), r, i)$ is an assertion, the *significance* of the assertion $(address(f), r, i)$, denoted $S(address(f), r, i))$ is defined as: $S(address(f), r, i) = \{(address(f), r, i)\}$, which is a set of one element containing the assertion.

1.3.3 Synonyms, synonym sets and homonyms

Consider the knowledge representation model $M = (\sum, F, I, T, P)$. Let R represents the set of all referers in M while r_1, r_2 are two distinct referers in M. $r_1$is said to be a *synonym* of r_2 if there exists associators $f_1, f_2 \in F$ for which it holds that $f_1(r_1) = f_2(r_2)$. The *synonym set* of a referer $r \in R$ is defined as the set of all synonyms of r which exist in M. If $f_1(r_1) = f_2(r_1)$ for any $f_1, f_2 \in F$ then r_1 is called a *homonym* in M.

1.3.4 Semantic resolution, semantic diversity, scope and size

Let $M = (\sum, F, I, T, P)$ be a knowledge representation model. The *semantic resolution* of a knowledge representation model M is defined as the cardinality of the set of referers in M. The *semantic resolution* of an association $f \in F$ is defined as the cardinality of the domain of f. The cardinality of the universe of discourse is called the *scope* of M. The cardinality of the set of associators is called the *semantic diversity* of M. Thus the semantic diversity of M is defined as the number of semantic perspectives defined in M. The *size* of M is defined as the cardinality of the set of units of knowledge.

1.3.5 Composing associators; the inheritance intuition

Let $M = (\sum, F, I, T, P)$ be a knowledge representation model. Let the domain of an associator $f \in F$ be denoted $Dom(f)$. When $f_1, f_2 \in F$ are associators then a string f

of the form: $(f_1 : f_2)$ is also an associator. The associator f is defined as:

$$f(r) \quad = \quad \begin{cases} f_1(r) & \text{if } r \in Dom(f_1) \\ f_2(r) & \text{otherwise} \end{cases}$$

1.3.6 Derivations, extensions and translations

Let $M = (\sum, F, I, T, P)$ be a knowledge representation model. Let $f' \in F$, f' is said to be a *derivation* of f if and only if each referer r in the domain of f is also a referer in the domain of f'. If $f'$is a derivation of f, it is said that f' is an *extension* of f, if and only if all units of knowledge u defined by f are also defined by f', while the cardinality of the domain of f is smaller than the cardinality of the domain of f'. If $f'$is derived from f, it is said that f' is an *translation* of f, if and only if, the cardinality of the domain of f equals the cardinality of the domain of f'.

1.3.7 The notion of a sub universe, a parallel universe and a super universe

1.3.7.1 Defining a sub universe

Let $M = (\sum, F, I, T, P)$ and $M' = (\sum', F', I', T', P')$ knowledge representation models, while the address space of M is represented by A. M is said to be a *sub universe* of M' if and only if the following holds for M and M':

- the knowledge content of M is smaller than the knowledge content of M'
- the semantic diversity of M is smaller than the semantic diversity of M'
- where $f \in F$ and $r, i \in A$, each assertion $(address(f), r, i)$ that is a unit of knowledge in M must also be a unit of knowledge in M'

1.3.7.2 Defining a parallel universe

Let $M = (\sum, F, I, T, P)$ and $M' = (\sum', F,' I', T', P')$ knowledge representation models, while the address space of M is represented by A. M' is said to be a *parallel discourse universe* of M if and only if the following holds for M and M':

- the knowledge content of M equals the knowledge content of M'
- the semantic diversity of M equals the semantic diversity M'
- Where $f \in F$ and $r, i \in A$, each assertion $(address(f), r, i)$ that is a unit of knowledge in M must also be a unit of knowledge in M'

It is equally valid to state that M and M' are parallel universes if and only if the significance of M equals $S(M)$ the significance of M'.

1.3.7.3 Defining a super universe

Let $M = (\sum, F, I, T, P)$ and $M' = (\sum', F', I', T', P')$ knowledge representation models, while the address space of M is represented by A'. M is said to be a *super universe* of M' if and only if the following holds for M and M':

- the knowledge content of M is greater than the knowledge content of M'
- the semantic diversity of M is greater than the semantic diversity of M'
- where $f \in F'$ and $r, i \in A'$, each assertion $(address(f), r, i)$ that is a unit of knowledge in M' must also be a unit of knowledge in M.

Chapter 2

About the truth-value and the significance of logical expressions

2.1 Introduction

In the previous chapter a definition was provided for the concept of a claim relative to a knowledge representation model. Using the principle of semantic compositionality this chapter will pursue the following avenues:

- The definition of the truth-value of expressions in a language for intensional higher order logic.

- The definition of the significance of expressions in language for intensional higher order logic. The concept of the the significance of logical expression is deemed a necessity by recognition of the fact that it is logically incorrect to assume that an assertion has no meaning if the assertion in question is false. The fact that sentences which are logically false can be translated from one natural language to another serves as a confirmation of this fact.

- A mapping of Frege's notions of the *Sinn* and the *Bedeutung* of expressions unto the models for knowledge representation and meaning introduced in this paper.

This chapter thus introduces a language for logic together with a definition of the truth-value and significance of expressions in this language.

2.2 A theory of meaning

2.2.1 Significance expressions

Let $M = (\sum, F, I, T, P)$ be a knowledge representation model. Let $f \in F$, $r, i \in I$ and $t_1, ..., t_k \in I$. A *significance expression* of for M is an expression defined as:

- $f'(r') = i'$ and $f' : r'(t'_1, ..., t'_k)$ are significance expressions where $f' = f$ is f' or a variable; $r' = r$ or $r'$is a variable, $i' = i$ or $i'$is a variable. Additionally $t'_1, ..., t'_k$ are variables or elements of the universe of discourse I.
- if φ is a significance expression $\neg\varphi$ is also a significance expression
- Where φ, ψ are significance expressions $(\varphi \wedge \psi), (\varphi \vee \psi)$ and $(\varphi - \psi)$ are also a significance expressions
- If φ is a significance expression then $\exists v\varphi$ and $\forall v\varphi$ are also significance expressions. where v is variable
- Nothing else is a significance expression

2.2.2 Assignments and assignment sets

An *assignment* b for a model $M = (\sum, F, I, T, P)$ is a function which maps variables to objects in I. So the expression $y = b(x)$ denotes that $y \in I$ is the identity bound to the variable x by the assignment b. An assignment b' is a member of the *assignment set* for an assignment b and an identity $d \in D$ denoted by $b(x|d)$ if and only if the assignment b relates to the assignment b' as in the following:

- $b'(y) = b(y)$ when $d \neq y$
- $b'(y) = d$ when $d = y$

2.2.3 The value assignment function

The value assignment function $W_{M,f,b}$ for a model $M = (\sum, F, I, T, P)$ of a language L, an associator $f \in F$ and an assignment b is defined as follows:

- $W_{M,f,b}(r) = f(r)$ if r is a constant
- $W_{M,f,b}(r) = b(r)$ if r is a variable

2.2.4 The valuation function

Let L represent the set of significance expressions for a knowledge representation model $M = (\sum, F, I, T, P)$. The valuation function $V_{M,b}$ for M and an assignment b is a function from L to $\{true, false\}$. The valuation function $V_{M,b}$ is defined as:

- $V_{M,b}((f(r) = i)) = true$ if and only if $f'(r') = i'$ where $f' = W_{M,f,b}(f)$, $r' = W_{M,f,b}(r)$ and $i' = W_{M,f,b}(i)$. To put another way: $(address(W_{M,f,b}(f)), address(W_{M,f,b}(r)), address(W_{M,f,b}$ must be a unit of knowledge in M.

- $V_{M,b}((f : r(t_1,...,t_k))) = true$ if and only if $f'(r') = i'$ where $f' = W_{M,f,b}(f)$, $r' = W_{M,f,b}(r)$ and $i' = W_{M,f,b}(i)$. Furthermore $address(p) = i'$ and $p \in P$ and $(address(W_{M,f,b}(t_1)),....,address(W_{M,f,b}(t_k))) \in p$.

- $V_{M,b}((\neg\varphi)) = true$ if and only if $V_{M,b}(\varphi) = false$

- $V_{M,b}((\psi \vee \varphi)) = true$ if and only if $V_{M,b}(\psi) = V_{M,b}(\varphi) = true$

- $V_{M,b}((\psi \wedge \varphi)) = true$ if and only if $V_{M,b}(\psi) = true$ or $V_{M,b}(\varphi) = true$

- $V_{M,b}((\psi - \varphi)) = true$ if and only if $V_{M,b}(\psi) \neq V_{M,b}(\varphi) = true$.

- $V_{M,b}(\exists x(\varphi)) = true$ if and only if there exists an $i \in I$ for which it holds that $V_{M,b(x|i)} = true$

- $V_{M,b}(\forall x(\varphi)) = true$ if and only if for all $i \in I$ it holds that $V_{M,b(x|i)} = true$

2.2.5 The significance, true significance and meaning

2.2.5.1 The significance of a significance expression

The significance of a significance expression is defined as a set of assertions defined in the following section. Let L represent the set of significance expressions for a knowledge representation model $M = (\sum, F, I, T, P)$. Let $A = addressspace(M)$ represent the address space of M. The significance function $S_{M,b}$ for M and an assignment b is a function from L to $POW(A \times A \times A)$, where $POW(A \times A \times A)$ represents the power set of the A^3. $S_{M,b}$ is defined as:

- $S_{M,b}((f(r) = i)) = \{(address(f'), address(r'), address(i'))\}$ where $f' = W_{M,f,b}(f)$, $r' = W_{M,f,b}(r)$ and $i' = W_{M,f,b}(i)$.

- $S_{M,b}((f : r(t_1,...,t_n)) = \{(address(f'), address(t'), address(r'))\}$ where $f' = W_{M,f,b}(f)$, $t' = (address(W_{M,f,b}(t_1)),...,address(W_{M,f,b}(t_n)))$ and r' is a addresses $\{r_1,...,r_k\}$ where $address(W_{M,f,b}(r)) \in r'$.

- $S_{M,b}((\neg\varphi)) = S_{M,b}(\varphi)^{-1}$, where $S_{M,b}(\varphi)^{-1}$ is $S_{M,b}((\varphi))$ with all assertions (f, r, i) in $S_{M,b}((\varphi))$ are replaced by their inverse assertions $(-f, -r, -i)$.

- $S_{M,b}((\psi \vee \varphi)) = S_{M,b}(\psi) \cap S_{M,b}(\varphi)$

- $S_{M,b}((\psi \wedge \varphi)) = S_{M,b}(\psi) \cup S_{M,b}(\varphi)$

- $S_{M,b}((\psi - \varphi)) = S_{M,b}(\psi) - S_{M,b}(\varphi)$ which represents the difference between $S_{M,b}(\psi)$ and $S_{M,b}(\varphi)$.

- $S_{M,b}((\exists x\varphi)) = S_{M,b(x/i)}(\varphi)$ if and only if there exists an $i \in I$ for which it holds that $V_{M,b(x|i)}(\varphi) = true$, otherwise $S((\exists x\varphi)) = \emptyset$

- $S_{M,b}((\forall x \varphi)) = \bigcup_{i=1}^{i=|n|} S(V_{M,b(x|identity(i))})$ if and only if for all $i \in I$ it holds that $V_{M,b(x|i)} = true$ and n is the cardinality of I.

The *meaning* of a significance expression φ for an assignment b is defined as $S_{M,b}(\varphi)$, which is a set of ternary tuple and as such $S_{M,b}(\varphi) \in I$.

2.2.5.2 The true significance of significance expressions

Let φ represent a significance expression for knowledge representation model $M = (\sum, F, I, T, P)$. The *true significance* of φ, denoted $T(\varphi)$ is defined as the sub set of the significance of φ which are units of knowledge in M. In general the true significance $T(\varphi)$ of a significance expression φ for a model M represents a definition of a sub model of M.

2.3 Capturing intensional phenomena

2.3.1 Introduction

The purpose of this section is to illustrate by example how certain intensional phenomena can be captured by the theory of meaning presented in this chapter.

2.3.2 Temporal and modal logic

The knowledge representation system introduced in chapter one supports the representation of temporal knowledge. This fact is quickly recognized when one considers a knowledge representation model $M = (\sum, F, I, T, P)$ for which all units of knowledge in M have the form $(address(f), address((r,t)), address(i))$ where $f, r, t, i \in I$ and t represents a the in time and the tuple $(r,t) \in I$. Let φ be a significance expression for M. Examples of valid significance expression for M are provided in the following table.

Significance expression	Description
$\exists x(f((r,t)) = i)$	*According to f, r refers to i at time t*
$\exists x(x((r,t)) = i)$	*there exists a world x in which r refers to i at the moment t time*
$\forall x(x((r,t)) = i)$	*in all possible worlds r refers to i at the moment t in time*
$f((r,t)) = i$	*According to f, r believes i at the moment t in time*.Where $S_{M,b}((\varphi)) = i$

2.4 Über Sinn und Bedeutung

This section presents an attempt to map the concepts of the *Sinn* (the sense) and the *Bedeutung* (the reference) as introduced by the German logician and mathematician Frege unto the knowledge representation framework introduced in this paper.

2.4.1 Mapping *Sinn* and *Bedeutung*

Let $M = (\sum, F, I, T, P)$ be a knowledge representation model and (f, r, i) is a unit of knowledge in M. At first, one might be inclined to associate Frege's notion of *Sinn* of a name with the referer r of a unit of knowledge. However this would be in conflict with Frege's idea that a particular *Sinn* always has the same *Bedeutung.* To be in harmony with Frege it is thus chosen that the concept of the *Sinn* corresponds with $f(r)$ representing the application of the function f to the referer r. Frege's concept of the *Bedeutung* readily allows its self to be associated with the reference i of the the unit of knowledge.

2.4.2 More on the mapping of *Sinn* and *Bedeutung*

According to Gamut [ref 1] Frege associated the concepts of *Sinn* and *Bedeutung* not only with names but also with sentences. To be able capture this some more machinery is needed. Let $M = (\sum, F, I, T, P)$ be a knowledge representation model and $S_{M,b}$ be the significance function for M. To be able to capture Frege's *Sinn* and *Bedeutung* for sentences we agree that for this purpose of this section the following holds:

- the recognition of the fact that the significance of all significance expressions are also elements of I, because the significance of a significance expression is a set of tuples.
- all significance expressions (as defined above) are elements of the universe of discourse I of M.
- For x, y for which it holds that the significance $S_{M,b,}(x) = y$ it holds that there exists a particular associator $f \in F$ for which it holds that $f(address(x)) = address(y)$.

Harmony with Frege concepts of the *Sinn* and *Bedeutung* for sentences are thus obtained in a conceptually pure way.

Chapter 3

What is the relation with knowledge representation in predicate logic?

3.1 Introduction

In the first chapter of this paper a general model for knowledge representation was introduced. The purpose of this chapter to to illustrate that the knowledge representation model introduced in chapter one is more expressive than the knowledge representation models used in when in the definition of semantics for intentional type logic. In fact the knowledge representation model introduced in this paper may be viewed as a generalization of the knowledge representation models of predicate logic, inspired by chapter 1 of ref [1].

3.2 Essential components of first order predicate logic

3.2.1 Introduction

For the purpose of this section the essentials of first order predicate logic are summarized as follows:

- A set of objects D called the *domain of discourse*, a finite set of predicates P, and a set of constants C.
- A function I the domain of which is $(P \cup C)$ which assigns elements of C to elements D and assigns properties $p \in P$ to subsets of $D, D^2, D^3, \ldots, D^k$ for a natural number k. So that any n tuple $(c_1, \ldots, c_n) \in D^n$ can be assigned a property p, and any constant c can refer to an element $d \in D$. In addition it holds that no property $p \notin D$. A *model* M is defined as a pair (D, I). I is called the *interpretation function* of M.

- A language L for a model $M = (D, I)$ allowing properties of objects in the model to be queried and asserted.
- An assignment function b which maps variables in L to elements of D.
- A valuation function which given an assignment b defines if an utterance e in the language L is true or false for a model $M = (D, I)$. This function is said to define the semantics of first order predicate logic relative to L,b and M.

3.2.2 Mapping first order predicate logic models

This section illustrates how knowledge represented in models of first order predicate logic can be represented by the models introduced in chapter one. Let $M_o = (\sum, F, I_o, T_o, P_o)$ represent a knowledge representation model. Let $M_p = (D_p, I_p)$ represent a model for a first order predicate language. Let C_p represent the set of constants which form a subset of the domain of I_p, while P_p represents the subset of the domain the of $I_p$which are predicates. The universe of discourse $I_o$of M is partitioned into four disjoint subsets F, D_o, P_o, C_o, such that there respectively exists a one to one, onto mapping from D_o, P_o and C_o to D_p, P_p and C_p. The interpretation function I_p of M_p corresponds to an associator $f \in F$, where it is understood that f is the only associator in M_o. When $I_p(s) = d$ then $f(address(s)) = address(d))$, where $s \in (P_o \cup C_o))$ and $d \in D_o$.

3.3 Essential components of extensional type logic

3.3.1 Introduction

The basic intuition of extensional type logic knowledge is to generalize first order predicate logic by adding predicate constants to the domain of discourse of first order predicate logic, so that it becomes possible to state facts about predicates. For the purposes of this paper the essentials of extensional type logic are summarized as follows:

- A set of objects D called the *domain of discourse*, a finite set of predicates P, and a set of constants C, where $P, C \subset D$.
- A function I the domain of which is $(P \cup C)$ which assigns elements of C to elements D and assigns properties $p \in P$ to subsets of $D, D^2, D^3, ..., D^k$ for a natural number k. So that any n tuple $(t_1, ..., t_n) \in D^n$ can be assigned a property p, and any constant c can refer to an element $d \in D$. A *model* M is defined as a pair (D, I). I is called the *interpretation function* of M. The *distinguishing property* for extensional logic relative to first order predicate logic is that for all predicates p and constants c it holds that $p, c \in D$.
- A language L for a model $M = (D, I)$ allowing properties of objects in M to be queried and asserted. Since properties are also elements of the domain of discourse L must also provide means to assert and query properties of properties.
- An assignment function b which maps variables in L to elements of D.

- A valuation function which, given an assignment b, defines if an utterance e in the language L is true or false for a model $M = (D,I)$. This function is said to define the semantics of extensional logic relative to L,b and M.

Thus as a result of the way in which models for extensional logic are defined, extensional logic allows us to make statements about objects, about relations between objects and also about the relations themselves.

3.3.2 Mapping extensional type logic models

This section illustrates how knowledge represented in models of extensional type logic can be represented by the models introduced in chapter one. Let $M_o = (\sum, F, I_o, T_o, P_o)$ represent a knowledge representation model. Let $M_p = (D_p, I_p)$ represent a model for a extensional type logic language. Let C_p represent the set of constants which form a subset of the domain of I_p, while P_p represents the subset of the domain the of $I_p$which are predicates. The universe of discourse $I_o$of M is partitioned into four disjoint subsets F, D_o, P_o, C_o, such that there respectively exists a one to one, onto mapping from D_o, P_o and C_o to D_p, P_p and C_p. The interpretation function I_p of M_p corresponds to an associator $f \in F$, where it is understood that f is the only associator in M_o. When $I_p(s) = d$ then $f(address(s)) = address(d))$, where $s \in (P_o \cup C_o))$ and $d \in D_o$.

3.4 Essential components of intensional logic

3.4.1 Introduction

Models $M = (D,I)$ for first order predicate logic and extensional type logic are defined using a single domain of discourse D and interpretation function I. For intensional logic the existence of multiple interpretation functions is allowed. So statements of fact can now be made from different perspectives. For the purposes of this paper the essentials of intensional logic allow themselves to be summarized as follows:

- A set of objects D called the *domain of discourse,* a finite set of predicates P, and a set of constants C, where $P, C \subset D$.

- A set I of interpretation functions the domain of which is $(P \cup C)$, which assign elements of C to elements D and assign properties $p \in P$ to subsets of $D, D^2, D^3, ..., D^k$, for a natural number k. So that any n tuple $(t_1, ..., t_n) \in D^n$ can be assigned a property p, and any constant c can refer to an element $d \in D$. A *model M* is defined as the pair (D,i) $i \in I$. i is called the *interpretation function* of M.

- A language L for a model $M = (D,I)$ allows properties of objects in the model to be queried and asserted relative to a particular interpretation function $i \in I$. Since properties of elements are also in the the domain of discourse, L must also provide means to query and make assertions about properties of properties.

- An assignment function b which maps variables in L to elements of I.

- A valuation function which given an assignment b defines whether an utterance e in the language L is true or false for a model $M = (D, I)$. This function is said to define the semantics of extensional logic relative to M and L.

As as result of the way in which models for intensional logic are defined, intensional logic allows us to make statements about objects, relations and also about relationships between objects relative to different contexts.

3.4.2 Mapping intensional type logic models

This section illustrates how knowledge represented as models for intensional type logic can be represented by the models introduced in chapter one. Let $M_o = (\sum, F, I_o, T_o, P_o)$ represent a knowledge representation model. Let $M_p = (D_p, I_p)$ represent a model for a intensional type logic language. For each interpretation function in $i \in I_p$ there exists a corresponding associator $f \in F$ such that for all s, d it holds that if $i(s) = d$ then $f(address(s)) = address(d))$ where $d \in D$ o $s, d \in I_o$.

3.5 What are the unique properties of the knowledge representation model?

The previous sections of this chapter have substantiated that knowledge representation for first order predicate logic, extensional type logic and intensional type logic can be represented by the knowledge representation model defined in chapter one. This section considers distinguishing properties of the knowledge representation models defined in chapter one. Let $M_o = (\sum, F, I_o, T_o, P_o)$, while $M_p = (D_p, I_p)$ represents a knowledge representation model for intentional type logic. Unique properties of the knowledge representation model presented in this paper relative to knowledge representation models in intentional type logic are enumerated here:

- all associators $f \in F$ are functions from the $A \to A$ where A represents the address space of M_o which allows any element $i \in I_o$ be a referer and also a reference, this is not possible in models for intentional type logic M_p.

- associators (or interpretation functions in the terminology of predicate logic) are part of the universe of discourse. This makes it possible to make assertion about the possible worlds as identified by their respective interpretation functions. This is not possible in models for intentional type logic M_p

- since it is possible that an associator $f \in I$ while $f \notin F$ it is possible to assign properties to associators (or worlds) which do not function as such in M_o (because they are not elements of F).

- tuples and tuple sets are part of the universe of discourse M_o. In the case of models for intentional type logic M_p this not the case

Chapter 4

How does the knowledge representation model relate to Category Theory?

4.1 Defining categories

According to the Stanford Encyclopedia of Philosophy define a category Category theory is a generalized mathematical theory of structures. One of its goals is to reveal the universal properties of structures of a given kind via their relationships with one another. [1] Formally, a category C can be described as a collection O, the objects of C, which satisfy the following conditions:

- For every pair $a, b \in O$, there is a collection $Mor(a,b)$, namely, the *morphisms* from a to b in C. When f is a morphism from a to b, we write $f : a \rightarrow b$.

- For every triple $a, b, c \in O$, there is a partial operation from pairs of morphisms in $Mor(a,b) \times Mor(b,c)$ to morphisms in $Mor(a,c)$, called the *composition* of morphisms in C. When $f : a \rightarrow b$ and $g : b \rightarrow c$, $(g \circ f) : a \rightarrow c$ is their composition.

- For every object $a \in O$, there is a morphism $identity_a \in Mor(a,a)$ called the *identity* on a.

- Finally, morphisms have to satisfy two axioms:

  - Associativity: if $f : a \rightarrow b$, $g : b \rightarrow c$ and $h : c \rightarrow d$, then $h \circ (g \circ f) = (h \circ (g \circ f)$.
  - Identity: if $f : a \rightarrow b$, then $(identity_b \circ f) = f$ and $(f \circ identity_a) = f$.

[1] The definition of Categories presented here is quite literally taken from this Encyclopedia.

4.2 Mapping categories

Mapping Category Theory unto the knowledge representation models introduces in Chapter one is a matter defining a knowledge representation model $M = (\sum, F, I, T, P)$ which respects the requirements of Category Theory. The concept of a morphism in Category Theory allows its self to me mapped to associators. For every pair $a, b \in I$, there is a collection $Mor(a, b)$, namely, the *morphisms* or the *associator set* from a to b in a category C. Since knowledge representation models place no restrictions on the properties of associators (or morphisms if you prefer), it is clearly to that knowledge representation models can be crafted which about the requirements of Category Theory.

Chapter 5

About Discourse Grammars and Logical Documents

5.1 Introduction

This chapter introduces the definition of a discourse grammar. The definition used here is basically a dotted grammar (as in ref [31]), where the set of terminal symbols of the grammar corresponds to the universe of discourse of a knowledge representation model. Practical systems based discourse grammars would likely add the notions of grammar variables or place-holders as found in *two level grammars, attribute grammars* or *affix grammars*.[1]

5.2 Discourse grammar

A *discourse grammar*, G_M for a knowledge representation model $M = (\sum_m, F_m, I_m, T_m, P_m)$ is defined as a quadruple (M, N, T, S, P) where

- N is a non empty finite set of nonterminal symbols
- T is a non empty finite set of terminal symbols, such that each $t \in T$ is a normalized significance token in for M.
- S element of N is a distinguished symbol called the start symbol
- P is a set of rewriting rules called production rules of the form:
  $\omega_1 \cdot A\omega_2 \rightarrow \omega_3 \cdot \omega_4 \qquad A \in N$

$\omega_1, \omega_2, \omega_3$, and ω_4 are elements $(N \cup T)*$. The symbol A is called the subject of the production rule $\omega_1 \cdot A\omega_2 \rightarrow \omega_3 \cdot \omega_4$. $\omega_1 \cdot A\omega_2$ is called the left-hand side of the production rules while $\omega_3 \cdot \omega_4$ represents the right-hand side.

[1] My current experimental implementation uses recording grammars [ref 31] and place-holders similar to those in attribute grammars.

5.3 Documents and document types for discourse grammars

A *document* x for a discourse grammar $G_M = (N, T, S, P)$, $M = (\sum_m, F_m, I_m, T_m, P_m)$ is an element of $T*$ for which there exists a finite sequence ω_1 , ω_2, ... ω_n $(n > 1)$, of strings in$(N \cup T) * \cdot (N \cup T)*$, defined as:

$\omega_1 = \cdot S$

$\omega_i = B_i C_i \cdot D_i E_i F_i$

$\omega_n = x \cdot$

For $\omega_{i+1}$the following holds:

1. $D_i \in N$ then $\omega_{i+1} = B_i I_i \cdot J_i F_i$ and $p_i = C_i \cdot D_i E_i \rightarrow I_i \cdot J_i$ is a production rule where $(1 \leq i \leq n-1)$.
2. $D_i \in T$ then $w_{i+1} = B_i C_i D_i \cdot E_i F_i$where $(1 \leq i \leq n-1)$.

The set of documents which is generated by a discourse grammar D is called the *document type* defined by G_M.

Chapter 6

Modeling knowledge acquisition

6.1 Introduction

This chapter introduces a formalism which is an attempt to model the process of learning. The formalism presented combines discourse grammars and knowledge representation models together with some additional machinery intended to allow simple but sophisticated learning models to be developed. [1]

6.2 Token expressions

Let $M = (\sum, F, I, T, P)$ be a knowledge representation model, while φ represents a significance expressions for M. A *token expression* for M is defined as follows:

- all normalized the significance tokens in M are token expressions
- when t represents a token expression for M, $+t$ and $-t$ also represent token expressions called *assert statements* and *retract statements* respectively

6.3 Tutoring grammars

A *tutoring grammar* G_M, for a knowledge representation model $M = (\sum_m, F_m, I_m, T_m, P_m)$ is defined as a quadruple (N, T, S, P) where

- N is a non empty finite set of nonterminal symbols
- T is a non empty finite set of terminal symbols, such that each $t \in T$ is a token expression for M.
- S element of N is a distinguished symbol called the start symbol

[1] The material introduced here may be viewed as a formalization of the *assert* and *retract* statements the reader may be familiar with from the Prolog logical programming language.

- P is a set of rewriting rules called production rules of the form:
  $\omega_1 \cdot A\omega_2 \rightarrow \omega_3 \cdot \omega_4 \quad A \in N$

$\omega_1, \omega_2, \omega_3$, and ω_4 are elements $(N \cup T)*$. The symbol A is called the subject of the production rule $\omega_1 \cdot A\omega_2 \rightarrow \omega_3 \cdot \omega_4$. $\omega_1 \cdot A\omega_2$ is called the left-hand side of the production rules while $\omega_3 \cdot \omega_4$ represents the right-hand side.

6.4 Lessons and courses for tutoring grammars

A *lesson* x for a tutoring grammar $G_M = (N, T, S, P)$, $M = (\sum_m, F_m, I_m, T_m, P_m)$ is an element of $T*$ for which there exists a finite sequence ω_1 , ω_2, ... ω_n $(n > 1)$, of pairs $(m, (N \cup T) * \cdot (N \cup T)*)$ where m is a knowledge representation model and $(N \cup T) * \cdot (N \cup T)*$is a sentential form of G_M, defined as:

$$\omega_1 = (M, \cdot S)$$

$$\omega_i = (M_i, B_iC_i \cdot D_iE_iF_i)$$

$$\omega_n = (M_n, x\cdot)$$

For $\omega_{i+1}$the following holds:

1. $D_i \in N$ then $\omega_{i+1} = (M_i, B_iI_i \cdot J_iF_i)$ and $p_i = C_i \cdot D_iE_i \rightarrow I_i \cdot J_i$ is a production rule where $(1 \leq i \leq n-1)$.
2. $D_i \in T$ then the following holds for w_{i+1}:
   (a) If D_i is an assert statement $+address(S(\varphi))$ then $w_{i+1} = (M_i', B_iC_iD_i \cdot E_iF_i)$ where $(1 \leq i \leq n-1)$ and is M_i' is a knowledge representation model for which it holds that $S(M_i') = S(M_i) \cup S(\varphi)$.
   (b) If D_i is a retract statement $-address(S(\varphi))$ then $w_{i+1} = (M_i', B_iC_iD_i \cdot E_iF_i)$ where $(1 \leq i \leq n-1)$ and $M_i' = S(M_i) - S(\varphi)$ which is the difference between the knowledge content of M_i and the significance of $S(\varphi)$.
   (c) otherwise $w_{i+1} = (M_i, B_iC_iD_i \cdot E_iF_i)$.

The set of lessons which is generated by a tutoring grammar D is called the *course* defined by G. A set of courses is called a *curriculum*.

Chapter 7

Conclusion

By generalizing the knowledge representation model of first order predicate logic as introduced by Alfred Tarski, this paper has introduced a knowledge representation vehicle supporting a formal representation of intensional phenomena. A language for representing knowledge was introduced together with definitions for the truth-value and significance of expressions in this language. Based on the significance of logical expressions, discourse grammars and logical documents where introduced. Finally a formalism for modeling knowledge acquisition was introduced.

Bibliography

[1] L.T.F. Gamut, [1991] “Logic, Language and Meaning” volume 2, University of Chicago press

[2] J. van Eijck, E. Thijsse, [1989], Logica voor alfa’s en informatici,Academic Service

[3] Samuel Guttenplan [1986] "The languages of logic", Blackwell Publishers

[4] James Allen [1995] "Natural Language Understanding" The Benjamin/Cummings Publishing Company, Inc.

[5] R. Book [1973] edited by A. Aho "Currents in the theory of computing" chapter 1. Prentice-Hall, Inc.

[6] S. Ginsburg, S. A. Greibach [1966] "Mappings which preserve context sensitive languages" Information and Control 9, 563-582 .

[7] D. Grune, C. Jacobs [1990] "Parsing Techniques". Ellis Horwood Limited.

[8] D. Harel [1989] "Algoritmiek" Academic Service.

[9] S.-Y Kuroda [1964] "Classes of Languages and Linear-Bounded Automata", Information and Control 7, 207-223.

[10] J. Loeckx [1970] "Parsing for general phrase structure grammars". Information and control 16, 443-464.

[11] G. Matthews [1964] "A note on asymmetry in phrase structure grammars", Information and control 7, 360-365.

[12] T. A. Sudkamp [1991] "Languages and Machines" Addison-Wesley.

[13] D. Ullman , J. Hopcroft [1969] "Formal languages and their relation to automata". Addison-Wesley.

[14] R. Uzgalis, J. Cleaveland [1977] "Grammars for Programming Languages" Elsevier Computer Science Library.

[15] V.J. Rayward-Smith [1988] "Inleiding in de theorie van formele talen" Academic Service.

[16] R. Book [1973] "On the structure of context sensitive grammars". International Journal of Computer Sciences Vol 2, No. 2. pages 129 - 138.

[17] Robert Sedgewick [1988] "Algorithms" Addison-Wesley.

[18] S. Dik, J. Kooij, [1991], "Algemene taal wetenschap", Uitgeverij het spectrum

[19] F. McCabe, [1992], "Logic and Objects", Prentice Hall International (UK) Ltd

[20] A. Aho, R.Sethi, J. Ullman, [1986], "COMPILERS Principles, Techniques and tools", Addison-Wesley.

[21] P. Brown, [1981], "Writing Interactive Compilers and Interpreters", John Wiley & Sons

[22] V.J. Rayward-Smith, [1985], "Inleiding in de berekenbaarheidstheorie", Academic Service

[23] C.A. Koster and various co-authors, [1992], "1992/1992", University of Nijmegen Dept. Informatics (The Netherlands).

[24] Kent Beck, [1994], "Patterns and Software development", Dr. Dobbs Journal

[25] M.T. Rosetta [1994] "ROSETTA Compositional Translation", Kluwer Academic Publishers

[26] Arie Sturm en Fred Weerman [1983] "Generatieve Syntaxis", Martinus Nijhoff/Leiden

[27] H. van Riemsdijk and Edwin Williams [1986] "Introduction to the theory of grammar", The Massachusetts Institute of Technology.

[28] J. Glenn Brookshear [1989] "Formal Languages, Automata, and Complexity", The Benjamin/Cummings Publishing Company, Inc.

[29] R. Sommerhalder/S.C. van Westrhenen [1987] "Introduction to the theory of Computability of Programs, machines, effectiveness and feasibility", Technische Universiteit Delft

[30] R. Plasmeijer, M van Eekelen [1993] "Functional Programming and Parallel Graph Rewriting", Addison-Wesley Publishing Company.

[31] M.H.A Gittens [2002] Generating efficient table driven parsers for non context free languages, http://www.gittens.nl/grammar.html

[32] C.J. Date, Huge Darwen [2000] Foundation for Future Database Systems, Addison-Wesley Publishing Company.

[33] Edited by Christiane Fellbaum 1998, WORDNET An electronic lexical database, MIT Press Cambridge

[34] Stanford Encyclopedia of Philosophy; section on Modal logic.

[35] Seymour Lipschuts, Discrete Mathematics, McGraw-Hill Book Company

[36] Stanford Encyclopedia of Philosophy; section on Category Theory

[37] Rossiter, Nelson, Heather [1994], The Categorial Product Data Model as a Formalism for Object-Relational Databases