Proceedings of the Symposium on Logic and Algorithms in Computational Linguistics 2018 (LACompLing2018)

Stockholm, 28–31 August 2018
Department of Mathematics and Department of Philosophy, Stockholm University, Stockholm, Sweden

Editors:
Krasimir Angelov, University of Gothenburg, Sweden
Kristina Liefke, Goethe University Frankfurt, Germany
Roussanka Loukanova, Stockholm University, Sweden
Michael Moortgat, Utrecht University, The Netherlands
Satoshi Tojo, School of Information Science, JAIST, Japan

Publisher: Stockholm: Stockholm University, 2018, DiVA Portal for digital publications

September 23, 2018

Table of Contents

Part I

Abstracts of Talks and Short Biographies

Lasha Abzianidze (University of Groningen, Nederlands) Invited Talk

Compositional Semantics in the Parallel Meaning Bank

(joint work with Johan Bos)

Abstract: The Parallel Meaning Bank (PMB) is a corpus of translations annotated with shared, formal meaning representations. The principle of compositionality lies at the heart of the corpus as it drives a derivation process of phrasal semantics and enables cross-lingual projection of meaning representations. The talk will present the PMB annotation pipeline and show how it leads to the formal, compositional semantics of translations. As a highlight, compositional treatment of several challenging semantic phenomena in English will be shown.

Short Biography: Lasha Abzianidze is a postdoc researcher at the University of Groningen. His research interests span meaning representations and natural language inference. Currently he works on the Parallel Meaning Bank project where His research focuses on semantic annotation and compositional semantics for wide-coverage texts. He obtained his PhD, titled “A natural proof system for natural language”, at Tilburg University. In his PhD research, he developed a tableau-based theorem prover for a Natural Logic which operates and solves textual entailment problems.

Krasimir Angelov (University of Gothenburg and Digital Grammars AB, Sweden) Invited Talk

A Parallel WordNet and Treebank in English, Swedish and Bulgarian

Abstract: We present a work in progress about a parallel WordNet-like lexicon and a treebank for English, Swedish and Bulgarian. The lexicon uses the Princeton WordNet senses but in addition incorporates detailed morphological and syntactic information. Words accross languages with the same sense which are moreover frequent mutual translations are grouped together via a language-independent identifier. These features make the lexicon directly usable as a library in GF applications. As part of the development we also converted all examples from WordNet to a treebank parsed with the GF Resource Grammars. Thanks to that the examples are translated to Swedish and Bulgarian.

Short Biography: Krasimir Angelov is an Associate Professor in computer science at the University of Gotheburg. His interests are in formal and natural languages, functional programming, machine translation and natural language parsing and generation. He is also one of the developers of Grammatical Framework (GF). The later is a programming language for developing hybride rule-based and statistical natural language applications. He is also one of the founders of Digital Grammars AB, a company which offers reliable language technologies, i.e. solutions where quality is prioritised usually in exchange of coverage.

Rasmus Blanck and Aleksandre Maskharashvili (CLASP, FLOV, University of Gothenburg, Sweden)

From TAG to HOL Representations of AMRs via ACGs

Abstract: We investigate a possibility of constructing an Abstract Categorial Grammar (ACG) that relates Tree Adjoining Grammar (TAG) and Higher

Order Logic (HOL) formulas which encode Abstract Meaning Representations (AMRs). We also propose another ACG that relates TAG and HOL formulas expressing the neo-Davidsonian event semantics. Both of these encodings are based on the already existing ACG encoding of the syntax-semantics interface where TAG derivations are interpreted as HOL formulas representing Montague semantics. In particular, both of these encodings share the same abstract language coming from the ACG encoding of TAG with Montague semantics, which is second-order. For second-order ACGs, problems of parsing and generation are known to be of polynomial complexity. Thus we get the natural language generation and parsing with TAGs and HOL formulas modeling AMR for free.

Robin Cooper (University of Gothenburg, Sweden) Invited Talk

How to Play Games with Types

(joint work with Ellen Breitholtz)

ABSTRACT: This talk will discuss how the kind of game theory (GT) presented in the course by Heather Burnett and E. Allyn Smith at ESSLLI 2017 (https://www.irit.fr/esslli2017/courses/6) and Burnett's paper "Signalling Games, Sociolinguistic Variation and the Construction of Style" (http://www.heatherburnett.net/uploads/9/6/6/0/96608942/burnett_smgs.pdf) could be connected to work on TTR, a type theory with records, and Ginzburg's KOS, a formal approach to conversational semantics. Here are some points I will consider:

1. Recasting GT in TTR. They both talk about types (of action) and when GT talks about possible worlds it is really what TTR would call types of situations. (The same holds of the use of the term "possible worlds" in probability theory). I will sketch an example of how it might look.

2. But what might doing (1) add to a linguistic theory? KOS/TTR might provide a framework for dealing with issues like choosing which games to play, misunderstandings between two agents about what game is being played or accommodating a game on the basis of another agent's behaviour. There is a notion of game in my paper "How to do things with types" (https://www.cisuc.uc.pt/ckfinder/userfiles/files/TR%202014-02.pdf). There is more detail in my book draft (https://sites.google.com/site/typetheorywithrecords/drafts) and also in Ellen Breitholtz's work on enthymemes and topoi in her thesis and book in preparation. Ginzburg's work on genre and conversation types is related. The games in this literature are very simple from the perspective of GT. They are defined in terms of a string type for a string of events on the gameboard which is traversed by an agent trying to realize the types. We have nothing to say about how you would make choices in a non-deterministic game, but GT would add that. It could be extremely productive to embed game theory in a theory of dialogue — one even begins to imagine metagames, games you play about concerning which game to play. We can perhaps supply a way of connecting GT to dialogue and grammar in a formal setting.

3. We could view this as making a connection between games and a general theory of action along the lines of "How to do things with types". The assumption seems to be that you compute utility and then perform the action that has

highest utility for you. But you could think of other strategies: e.g. cooperative (make the move that has the highest utility irrespective of player), altruistic (maximize the utility of the other player). If you think of games as assigning utilities to event types at a given state of play, perhaps exploiting techniques from our work on probabilistic TTR (http://csli-lilt.stanford.edu/ojs/index.php/LiLT/article/view/52) you could have a superordinate theory of action which would tell you what you might do depending on which strategy you are using.

Short Biography: Robin Cooper is Senior Professor at the University of Gothenburg, where he was previously Professor of Computational Linguistics. He is currently conducting research within the Centre for Linguistic Theory and Studies in Probability (CLASP) at Gothenburg. He has an undergraduate degree from the University of Cambridge and a PhD in Linguistics from the University of Massachusetts at Amherst. He has taught prevsiouly at the following universities: Universität Freiburg, University of Texas at Austin, University of Massachusetts at Amherst, University of Wisconsin at Madison, Stanford University, Lund University and Edinburgh University. He has held a Mellon Postdoctoral Fellowship and a Guggenheim Fellowship and has been a fellow at the Centre for Advanced Study in the Behavioral Sciences at Stanford. He is a Fellow of the British Academy and the Royal Society of Arts and Sciences in Gothenburg and a member of Academia Europaea. He holds an honorary doctorate from Uppsala. His main research interests are semantics (both theoretical and computational), dialogue semantics and computational dialogue systems. Currently he is working on a type theoretical approach to language and cognition.

Hercules Dalianis (DSV-Stockholm University, Sweden) Invited Talk

HEALTH BANK — A Workbench for Data Science Applications in Healthcare

Abstract: Healthcare has many challenges in form of monitoring and predicting adverse events as healthcare associated infections or adverse drug events. All this can happen while treating a patient at the hospital for their disease. The research question is: When and how many adverse events have occurred, how can one predict them? Nowadays all information is contained in the electronic patient records and are written both in structured form and in unstructured free text. This talk will describe the data used for our research in HEALTH BANK — Swedish Health Record Research Bank containing over 2 million patient records from 2007–2014. Topics are detection of symptoms, diseases, body parts and drugs from Swedish electronic patient record text, including deciding on the certainty of a symptom or disease and detecting adverse (drug) events. Future research are detecting early symptoms of cancer and de-identification of electronic patient records for secondary use.

Short Biography: Hercules Dalianis has Master of Science in engineering (civilingenjör) with speciality in electrical engineering, graduated in 1984 at the Royal Institute of Technology, KTH, Stockholm, Sweden, and received his PhD/Teknologie doktor in 1996 also at KTH. Since 2011 he is Professor in Computer and Systems Sciences at Stockholm University, Sweden. Dalianis was post doc researcher at University of Southern California/ISI in Los Angeles 1997–98. Dalianis was also post doc researcher (forskarassistent) at NADA KTH 1999–2003, moreover he held a three year guest professorship at CST, University of Copenhagen during 2002–2005, founded by Norfa, the Nordic Council of Ministers. Dalianis was on a sabbatical stay at CSIRO/Macquire University, Sydney, Australia 2016–17 compiling a text book with the title Clinical text mining: Secondary use of electronic patient records, that will be published open access at Springer in April 2018. Dalianis works in the interface between industry and university and with the aim to make research results useful for society. Dalianis has specialized in the area of human language technology, to make computer to understand and process human language text, but also to make a computer to produce text automatically. Currently Dalianis is working in the area of clinical text mining with the aim to improve healthcare in form of better electronic patient record systems, presentation of the patient records and extraction of valuable information both for clinical researchers but also for lay persons as for example patients.

Philippe de Groote (Directeur de Recherche, Inria, France) Invited Talk

New Progress in Continuation-Based Dynamic Logic

Abstract: In this talk, we revisit the type-theoretic dynamic logic introduced by de Groote (2006) and developed by Lebedeva (2012). We show how a slightly richer notion of continuation allows new dynamic connectives and quantifiers to be defined in a systematic way.

Short Biography: Dr. Philippe de Groote received his PhD degree in engineering from the Université Catholique de Louvain in March 1991. After a postdoc at the University of Pennsylvania, he joined Inria in September 1992, initially as Chargé de Recherche and then Directeur de Recherche. His research interests include mathematical logic, type-theory, proof-theory, computational linguistics, and natural language formal semantics.

Marie Duzi (VSB-Technical University of Ostrava, Czech Republic) Invited Talk

Negation, Presupposition and Truth-Value Gaps

Abstract: There are many kinds of negation and denial. Perhaps the most common is Boolean negation ‘not’ that applies to propositions-in-extension, i.e. truth-values. The others are, inter alia, the property of propositions of not being true which applies to propositions; the complement function which applies to sets; privation which applies to properties; negation as failure applied in logic programming; negation as argumentation ad absurdum, and many others. I am going to deal with negation of propositions that come attached with a presupposition that is entailed by the positive as well as negated form of a given proposition. However, there are two kinds of negation, namely internal and external negation, which are not equivalent. I will prove that while the former is presupposition-preserving, the latter is presupposition-denying. This issue has much in common with the difference between topic and focus articulation within a sentence. Whereas articulating the topic of a sentence activates a presupposition, articulating the focus frequently yields merely an entailment. While the

Russellian wide-scope (external) negation gets the truth-conditions of a sentence right for a subject occurring as a focus, Strawsonian narrow-scope (internal) negation is validly applicable for a subject occurring as the topic. My background theory is Transparent Intensional Logic (TIL). It is an expressive logic apt for the analysis of sentences with presuppositions, because in TIL we work with partial functions, in particular with propositions with truth-value gaps. Moreover, procedural semantics of TIL makes it possible to uncover the hidden semantic features of sentences, make them explicit and logically tractable.

Short Biography: Marie Duzi is a professor of Computer Science at VSB-Technical University of Ostrava. She graduated from mathematics and her main professional interests concern mathematical logic, Transparent Intensional Logic and natural-language processing. She is also a visiting professor at the Faculty of Informatics, Masaryk University of Brno where she closely cooperates with the group of computational linguists in the Centre for natural language processing.

Tim Fernando (Trinity College Dublin, Ireland)

Intervals and Events with and without Points

Abstract: Intervals and events are examined in terms of strings with and without the requirement that certain symbols occur uniquely. Allen interval relations, Dowty's aspect hypothesis and inertia are understood against strings, compressed into canonical forms, describable in Monadic Second-Order logic. See: https://www.scss.tcd.ie/Tim.Fernando/stock.pdf

Short Biography: Tim Fernando is a lecturer in computer science at Trinity College Dublin. He is interested in semantics and particularly finite-state methods.

Annie Foret (IRISA - University of Rennes 1, France) Invited Talk

On Categorial Grammatical Inference and Logical Information Systems

Abstract: We shall consider several classes of categorial grammars and discuss their learnability. We consider learning as a symbolic issue in an unsupervised setting, from raw or from structured data, for some variants of Lambek grammars and of categorial dependency grammars. In that perspective, we discuss for these frameworks different type constructors and structures, some limitations (negative results) but also some algorithms (positive results) under some hypothesis.

On the experimental side, we also consider the Logical Information Systems approach, that allows for navigation, querying, updating, and analysis of heterogeneous data collections where data are given (logical) descriptors. Categorial grammars can be seen as a particular case of Logical Information System.

Short Biography: Annie Foret is an associate-professor of computer science in Rennes 1 university, France. She belongs to the SemLIS research team (on "Semantics, Logics, Information Systems for Data-User Interaction") in the Data and Knowledge Management department at IRISA. Her general research interests are on logic, language and computation. Her current research interests include grammatical inference and categorial grammars. Previously, she studied mathematics and computer science at Ecole normale supérieure, and non-classical logics and rewriting in her PHD under the supervision by G. Huet. She then joined IRISA and Rennes 1 where she completed her habilitation on "some classes of type-logical grammars that model syntax".

Jonathan Ginzburg (Laboratoire de Linguistique Formelle, Université Paris-Diderot and Laboratoire d'Excellence LabEx-EFL, France) Invited Talk

Combining Verbal and Non-Verbal Interaction in Dialogue

ABSTRACT: The talk will provide detailed motivation, contrary to received wisdom until recently, as to the mutual interaction between non-verbal social signals such as laughter, smiling, frowning etc and content emanating from verbal material. In particular, I will argue that such non-verbal social signals bear propositional content and can participate in own and other communication management (e.g., clarification requests and corrections). I will show how the content emanating from non-verbal social signals can be integrated in type theoretic accounts of dialogue interaction by combining work in existing frameworks with psychological and computational approaches to emotion appraisal and to common sense reasoning.

SHORT BIOGRAPHY: Jonathan Ginzburg is Professor of Linguistics at Université Paris-Diderot (Paris 7). He has held appointments at the Hebrew University of Jerusalem and King's College, London. He is one of the founders and associate editors of the journal Dialogue and Discourse. His research interests include semantics, dialogue, language acquisition, and musical meaning. He is the author of Interrogative Investigations (CSLI Publications, 2001, with Ivan A. Sag) and The Interactive Stance: meaning for conversation (Oxford University Press, 2012).

Justyna Grudzinska (University of Warsaw, Poland) Invited Talk

Taking Scope with Continuations and Dependent Types

(joint work with Marek Zawadowski)

ABSTRACT: Dependent type theoretical frameworks have been used to model linguistic phenomena of central importance, e.g., unbound anaphora (Ranta 1994, Cooper 2004, Bekki 2014, Grudzinska et al. 2014), lexical phenomena such as selectional restrictions and coercions (Asher 2011, Luo 2012), adjectival and adverbial modification (Luo et al. 2017). Continuations have been used for an influential in situ analysis of quantifier scope ambiguities (Barker 2002). In my talk I will present a semantic system combining continuations and dependent types (joint work with Marek Zawadowski) that is sufficient to account for a broad range of existing readings for multi-quantifier sentences, including simple sentences and more complex syntactic environments such as inverse linking.

SHORT BIOGRAPHY: Justyna Grudzinska obtained a Ph.D. in philosophy at the University of Warsaw. Her research interests are formal semantics and philosophy of language, and her current main focus is on the use of dependent type theories to the study of natural language semantics (plural unbound anaphora, long-distance indefinites, in situ semantics for scope ambiguities, possessive and Haddock definites). She is also a coordinator of the Cognitive Science Programme at the University of Warsaw.

M. Dolores Jiménez López (GRLMC-Research Group on Mathematical Linguistics, Universitat Rovira i Virgili, Tarragona, Spain) Invited Talk

Complexity, Natural Language and Machine Learning

Abstract: The talk focuses on linguistic complexity. Are all languages equally complex? Does it make sense to compare the complexity of languages? Can languages differ in complexity? Complexity is a controversial concept in linguistics. Until recently, natural language complexity has not been widely researched and and it is still not clear how complexity has to be defined and measured. It is necessary to provide an objective and meaningful method to calculate linguistic complexity. In order to reach this goal, an interdisciplinary solution — where computational models should be taken into account — is needed. Linguistics must propose tools for the analysis of natural language complexity, since the results obtained from these studies may have important implications both from a theoretical and from a practical point of view.

Short Biography: M. Dolores Jiménez-López is an Associate Professor at Departament de Filologies Romaniques at the Universitat Rovira i Virgili, Tarragona, Spain. She has a PhD degree in linguistics. She worked for two years, as a pre-doctoral fellow, at the Computer and Automation Research Institute of the Hungarian Academy of Sciences in Budapest, Hungary. Her post-doctoral training includes a three-year stay at Department of Computer Science in University of Pisa, Italy. Application of formal models to natural language analysis is one of her main research topics.

Ron Kaplan (Stanford University, US) KeyNote Talk

An Architecture for Structured Ambiguity Management

Abstract: A pipeline for full-fledged natural language understanding consists of components that deal with information at different levels of remove from the elements that make up an utterance. Computing across the full pipeline is difficult because complex patterns (at all levels) may overlap in different ways, giving rise to ambiguities that feed from one component to the next. A typical approach is to apply probabilistic or heuristic preferences within each component so as to reduce the number of candidates that it feeds forward to the next. This has an obvious disadvantage: ambiguity resolution based on local information may eliminate the only candidate that gives the best result when all later components are taken into account. An alternative approach is to organize representations so as to "manage" the way ambiguous structures are propagated rather than attempting to resolve ambiguity at each level. The final result can then be globally optimal with respect to the whole pipeline. The trick is to do this without blowing up the computation.

Short Biography: Ron Kaplan is an Adjunct Full Professor of Linguistics at Stanford University. He served previously as a Vice President of Amazon and the Chief Scientist of Amazon Search Technologies. He was founder and director of the Natural Language and Artificial Intelligence Laboratory at Nuance Communications, with a focus on dialog and the conversational user interface. Before Nuance, he managed the Semantic Initiatives and Natural Language Platform teams for the Bing search engine. He also served as Chief Technology Officer and

Chief Scientific Officer at Powerset, a deep semantic-search company acquired by Microsoft. Powerset was a spin-out of the (Xerox) Palo Alto Research Center based on technology developed by the natural language and artificial intelligence research group that Ron directed at PARC for many years. He is known for his influential contributions to computational linguistics and linguistic theory, particularly for the development of Lexical Functional Grammar and for the mathematical underpinnings and implementation of finite-state morphology.

Ron is a past President and Fellow of the Association for Computational Linguistics, a co-recipient of the 1992 Software System Award of the Association for Computing Machinery, a Fellow of the ACM, and a Fellow of the Cognitive Science Society. He received his Ph.D. in Social Psychology from Harvard University and was awarded an honorary doctorate by the Faculty of Humanities of Copenhagen University.

Yusuke Kubota (University of Tsukuba, Japan) Invited Talk

Type-Logical Grammar and Natural Language Syntax

Abstract: In this talk, I will first briefly sketch my recent work, which focused on developing a particular version of Type-Logical Grammar with emphasis on linguistic application. I will then speculate on what (I think) is still missing in my own research and what still needs to be done and whether now is a good time to start addressing these issues seriously. While I believe that my previous work has revealed some interesting points of comparison between Type-Logical Grammar and mainstream Chomskian syntax, it has also raised (or at least made me aware of) many issues pertaining to the relationship between theoretical linguistics and computational linguistics. I will touch on these issues and speculate on future directions.

Short Biography: Yusuke Kubota has received Ph.D at the Department of Linguistics at Ohio State University in 2010 and is currently an Assistant Professor at the University of Tsukuba. His main research interests are natural language syntax and semantics and mathematical linguistics. Together with Robert Levine, he has been developing a version of Type-Logical Grammar called Hybrid Type-Logical Categorial Grammar. Some of the results of this work, mainly dealing with empirical issues in the domain of coordination and ellipsis, have recently appeared in major linguistics journals including Linguistics and Philosophy, Linguistic Inquiry, and Natural Language Linguistic Theory.

Shalom Lappin (University of Gothenburg, Sweden) Invited Talk

Towards a Computationally Viable Framework for Semantic Representation

Abstract: Most formal semantic theories proposed since Montague (1974) employ possible worlds to model intensions and modality. Classical theories of knowledge representation also use worlds to represent epistemic states and reasoning. If worlds are construed as equivalent to ultrafilters in a lattice of propositions (maximal consistent sets of propositions), then they pose serious problems of tractable representability. In addition, traditional worlds-based semantic theories are unable to accommodate vagueness, which is a pervasive feature of predication. They also do not explain semantic learning, and it is not clear how they could be naturally extended to incorporate such an explanation. To offer a cognitively plausible system for interpreting expressions in natural language a semantic theory should generate tractable representations, handle vagueness of predication, and provide the basis for an account of semantic learning. In this paper I discuss the problem of computational tractability of semantic representation. I suggest a probabilistic Bayesian alternative to classical worlds-based semantics, and I indicate how it can deal with intensions, modality, vagueness, epistemic states, and semantic learning.

Short Biography: available at
https://www.kcl.ac.uk/artshums/depts/philosophy/people/staff/associates/emeritus/lappin/index.aspx

Hans Leiß (Ludwig-Maximilians-Universität München, Germany) Invited Talk
Predication with Sentential Subject in GF

Abstract: The resource grammar library of the Grammatical Framework of Ranta et al. distinguishes binary or ternary verbs with nominal or prepositional objects from verbs whose objects have the form of a sentence, a question or an infinitive. No such distinction is made for the subject position of verbs. We introduce syntactic categories for verbs, adjectives and verb phrases with sentential subjects and extend the predication grammar of Ranta (EACL, 2014) so that sentential subjects can only be combined with verb phrases of appropriate types (which may arise by passivizing verbs with sentential objects). We also report on the price in computational complexity that has to be paid for the gain in linguistic accuracy.

Short Biography: Hans Leiß has studied mathematics and computer science at the University of Bonn and wrote his doctoral thesis in model theory. After a few years in theoretical computer science at the Technical University of Aachen he joined Siemens AG in Munich, working on object-oriented programming, hardware verification, and parsing. He switched to computational linguistics (CIS) at the University of Munich (LMU) in 1990. His reseach interests are in formal language theory, type theories for programming languages, parsing and grammar development for natural languages, semantics of natural language. He retired from LMU in 2017.

Zhaohui Luo (Royal Holloway, University of London, UK) Invited Talk
Formal Semantics in Modern Type Theories: An Overview

Abstract: I'll give an overview, and report some recent developments, of Formal Semantics in Modern Type Theories (MTT-semantics for short). MTT-semantics is a semantic framework for natural language, in the tradition of Montague's semantics. However, while Montague's semantics is based on Church's simple type theory (and its models in set theory), MTT-semantics is based on dependent type theories, which we call modern type theories, such as Martin-Lof's type theory (MLTT) and the Unifying Theory of dependent Types (UTT). Thanks to recent development, MTT-semantics has become not only a full-blown alternative to Montague's semantics, but also a very attractive framework with a promising future for linguistic semantics.

In this talk, MTT-semantics will be explicated, and its advantages explained, by focussing on the following:

1. The rich structures in MTTs, together with subtyping, make MTTs a nice and powerful framework for formal semantics of natural language.

2. MTT-semantics is both model-theoretic and proof-theoretic and hence very attractive, both theoretically and practically.

By explaining the first point, we'll introduce MTT-semantics and, at the same time, show that the use and development of coercive subtyping play a crucial role in making MTT-semantics viable. The second point shows that MTTs provide a unique and nice semantic framework that was not available before for linguistic semantics. Being model-theoretic, MTT-semantics provides a wide coverage of various linguistic features. Being proof-theoretic, its foundational languages MTTs have proof-theoretic meaning theory based on inferential uses (appealing philosophically and theoretically) and it establishes a solid foundation for practical reasoning in natural languages based on proof assistants such as Coq (appealing practically). Altogether, this strengthens the argument that MTT-semantics is a promising framework for formal semantics, both theoretically and practically.

Short Biography: Zhaohui Luo is Professor of Computer Science at Royal Holloway, University of London. He is an expert in dependent type theory and its applications. In the last decade, he has worked on, among other things, formal semantics in modern type theories, applying type theory to linguistic semantics. His publications include "Computation and Reasoning", a monograph on type theories ECC/UTT that was published by OUP in 1994, and "Formal Semantics in Modern Type Theories", a forthcoming book (jointly with S. Chatzikyriakidis) to be published by Wiley/ISTE Science Publishing Ltd.

Mehdi Mirzapour, Jean-Philippe Prost, and Christian Retoré (LIRMM, Montpellier University CNRS, 161 Rue Ada, France)

Categorial Proof Nets and Dependency Locality: A New Metric for Linguistic Complexity

Abstract: This work provides a quantitative computational account of why a sentence has harder parse than some other one, or that one analysis of a sentence is simpler than another one. We take for granted Gibson's results on human processing complexity, and we provide a new metric which uses (Lambek) Categorial Proof Nets. In particular, we correctly model Gibson's account in his Dependency Locality Theory. The proposed metric correctly predicts some performance phenomena such as structures with embedded pronouns, garden pathing, unacceptability of center embedding, preference for lower attachment and passive paraphrases acceptability. Our proposal extends existing distance-based proposals on Categorial Proof Nets for complexity measurement while it opens the door to include semantic complexity, because of the syntax-semantics interface in categorial grammars.

Aarne Ranta (University of Gothenburg and Digital Grammars AB, Sweden)
Invited Talk

Concept Alignment for Compositional Translation

ABSTRACT: Translation between natural languages is not compositional in a naive word-to-word sense. But many problems can be solved by using higher-level concepts, implementable as abstract syntax constructors in type theory together with compositional linearization functions in Grammatical Framework (GF). The question then arises: what are these constructors for a given set of languages? A whole spectrum of possibilities suggests itself: word senses (as in WordNet), multiword phrases (as in statistical machine translation), predication frames (as in FrameNet), syntactic deep structures (as in GF Resource Grammar Library), and lexico-syntactic constructions (as in Construction Grammar). The talk will study the problem in the light of experiences for building a cross-lingual lexicon of concepts in the General Data Protection Regulation (GDPR) in five languages. We have identified over 3000 concepts of varying complexity. A lot of manual work has been needed in the process, but some ideas have emerged toward a computational approach that generates concept alignment candidates by automated analysis.

SHORT BIOGRAPHY: Aarne Ranta is Professor of Computer Science at the University of Gothenburg as well as CEO and co-founder of Digital Grammars AB. Ranta's research was initially focused on constructive type theory and its applications to natural language semantics. It evolved gradually to computational applications, leading to the implementation of GF (Grammatical Framework). The mission of GF is to formalize the grammars of the world and make them available for computer applications. It enables the processing of natural language with the same precision as programming languages are processed in compilers.

Frank Richter (Goethe University Frankfurt a.M., Germany) Invited Talk

Computational Semantics: Representations and Reasoning

ABSTRACT: Computing with classical meaning representations of formal semantics encounters two major problems (with many sub-problems): How do we compose logical representations for natural language expressions in a computationally feasible grammar, and how do we actually reason with the sophisticated logical representations that theoretical linguists devise? This talk revisits the construction of logical representations in a few empirically and theoretically challenging areas of grammar, and presents a treatment of formulae of higher-order logic which makes it possible to use first order model builders and theorem provers to reason with them, with special attention to the emerging overall architecture.

SHORT BIOGRAPHY: Frank Richter is Privatdozent and senior lecturer at the Institut für England und Amerikastudien at Goethe Universität Frankfurt a.M., Germany, since 2014. After studying general linguistics, computer science and psychology in Tübingen and a year at the University of Massachusetts at Amherst, he earned his PhD in general and computational linguistics at Tübingen University. He worked as researcher, lecturer and visiting professor at the University of Tübingen, University of Stuttgart and University of Düsseldorf. His publications are on the formal foundations of constraint-based grammar, he is co-inventor of the framework of Lexical Resource Semantics (with Manfred

Sailer), and he published on sentential negation, negative concord, idiomatic expressions, polarity items and syntactic and semantic grammar implementations.

Mehrnoosh Sadrzadeh (Queen Mary University of London, UK) Invited Talk
Lambdas, Vectors, and Dynamic Logic

(This is joint work with Reinhard Muskens and is supported by a Royal Society International Exchange Award.)

ABSTRACT: Vector models of language are based on the contextual aspects of language, the distributions of words and how they co-occur in text. Truth conditional models focus on the logical aspects of language, compositional properties of words and how they compose to form sentences. In the truth conditional approach, the denotation of a sentence determines its truth conditions, which can be taken to be a truth value, a set of possible worlds, a context change potential, or similar. In the vector models, the degree of co-occurrence of words in context determines how similar the meanings of words are. In this talk, we put these two models together and develop a vector semantics based on the simply typed lambda calculus models of natural language. We provide two types of vector semantics: a static one that uses techniques familiar from the truth conditional tradition of Montague and a dynamic one based on a form of dynamic interpretation inspired by Heim's context change potentials. We show how the dynamic model revokes a dynamic logic whose implication can be applied to admittance of a sentence by a corpus, and provide examples.

SHORT BIOGRAPHY: I got a BSc and an MSc from Sharif University, Tehran, Iran and a PhD with joint supervision at UQAM and Oxford. I held an EPSRC PDRF, EPSRC CAF, and a JRF in Wolfson College, Oxford; at the moment I am a senior lecturer in Queen Mary University London, where I teach NLP and mathematics for engineers. I have worked on algebra and proof theory for multi-agent systems and on vector composition and algebras for distributional semantics. I have done recurrent PC and PC chair work in conferences and workshops of the field and have edited volumes

Manfred Sailer (Goethe University Frankfurt a.M., Germany) Invited Talk
Contraint-Based Underspecified Semantic Combinatorics

ABSTRACT: In this talk, I will review a number of challenges of the syntax-semantics interface for a standard concept of compositionality. Such phenomena include: scope ambiguity, negative concord, discontinuous semantic contribution, polyadic quantification, and incomplete utterances. I will argue that a constraint-based underspecified semantic combinatorics, as pursued in Lexical Resource Semantics (LRS), allows for a natural and interesting analysis of such phenomena. A system like LRS combines insights and techniques of computational and formal semantics and, as such, continues the tradition of fruitful interaction between computational and theoretical linguistics.

SHORT BIOGRAPHY: Manfred Sailer is professor of English Linguistics at Goethe-University Frankfurt a.M. He studied general linguistics, computer science and psychology at Universität Tübingen (Master 1995, Promotion 2003) and received his postdoctoral degree (Habilitation) in English and General Linguistics at Göttingen University (2010). His main areas of research are the syntax-semantics interface, formal phraseology, negation, and the interaction of regularity and irregularity in language.

Satoshi Tojo (School of Information Science, Japan Advanced Institute of Science and Technology (JAIST), Japan) Invited Talk

Linear Algebraic Representation of Knowledge State of Agent

Abstract: We first propose a linear algebraic representation for the frame property, that is the accessibility in possible worlds as adjacency matrix. We show that the product between an adjacency matrix and a column vector of valuation results in possibility modality, and translate also the necessity modality, employing Boolean operations. Then, we apply the method to agent communication; we represent the belief change of agents by dynamic epistemic logic (DEL), and show that the belief change can also be shown by a sequence of linear transformation on accessibility matrix. Finally, we discuss the requirements for the formal presentation of 'who knows what at which time'.

Short Biography: Satoshi Tojo received a Bachelor of Engineering, Master of Engineering, and Doctor of Engineering degrees from the University of Tokyo, Japan. He joined Mitsubishi Research Institute, Inc. (MRI) in 1983, and the Japan Advanced Institute of Science and Technology (JAIST), Ishikawa, Japan, as associate professor in 1995 and became professor in 2000. His research interest is centered on grammar theory and formal semantics of natural language, as well as logic in artificial intelligence, including knowledge and belief of rational agents. Also, ha has studied the iterated learning model of grammar acquisition, and linguistic models of western tonal music.

Adrià Torrens Urrutia (Universitat Rovira i Virgili, Tarragona, Spain)

A Proposal to Describe Fuzziness in Natural Language

Abstract: In this presentation, we propose formal models that consider grammaticality as a gradient property instead of the categorical view of grammaticality defended in theoretical linguistics. Given that deviations from the norm are inherent to the spontaneous use of language, linguistic analysis tools should account for different levels of grammaticality.

Christian Wurm (University of Düsseldorf, Germany) Invited Talk

Reasoning with Ambiguity

Abstract: Ambiguity is often considered to be a nemesis of logical reasoning. Still, when addressing natural language semantics with formal logic, we somehow have to address it: we can "lose it in translation" by saying all ambiguity is syntactic and we interpret unambiguous syntactic derivations; we can use meta-formalisms in order to represent it; but the fact remains that humans usually can perfectly reason with ambiguous statements. Hence it seems to be an interesting idea to include ambiguity into logic itself. In this talk, I will present the results of my pursuit of this idea, which are partly very surprising and odd, but in the very end (I hope) provide us with a deeper understanding of ambiguity and maybe even the nature of meaning.

Short Biography: I completed my PhD in Bielefeld with Marcus Kracht and Greg Kobele, the topic being what I called "metalinguistics", that is the construction of language as an infinite object. My main interests are accordingly formal languages, automata and substructural logic. Currently, I am a lecturer at the University of Düsseldorf and focus on the analysis of ambiguity, by means of logic but also machine learning techniques.

Yuan Xie (Utrecht University, The Netherlands)

Referential Dependencies in Chinese: A Syntax- Discourse Processing Model

Abstract: I am proposing a syntax-discourse processing model for the representation and interpretation of referential dependencies in Chinese. Chinese referentially dependent expressions (e.g. pronouns, reflexives, certain full noun phrases) are different from those in many indo-European languages and rely more on discourse (e.g. using bare noun phrases to express definiteness–lacking overt article the; sentence-free reflexive ziji (self-N)– referring to the speaker), for this reason, this model, taking both the morphosyntactic and discourse features of the referentially dependent expressions into consideration, reflects the view that referentially dependent nominal expressions and their antecedents are information units that are stored in our working memory system and the referential dependencies are established through the interactions of those information units in our working memory system.

Robert Östling (Stockholm University, Sweden) Invited Talk

Language Structure from Parallel Texts

Abstract: Some texts have been translated into thousands of languages, a fact that allows us to compare the structures of language from a bird's-eye view. This information can then be used to study the evolutionary forces driving language change. I will discuss some of our results in this area, as well as current models for formalizing the phenomenon of human language on a global scale.

Short Biography: I am a researcher in computational linguistics, (un)focusing on a variety of topics including machine translation, language modeling, computational approaches to linguistic typology and sign language, multilingual natural language processing, and language learning. I am currently employed at the Department of Linguistics, Stockholm University, but have also worked at the University of Helsinki.

Part II

Contributed Papers

From TAG to HOL Representations of AMRs via ACGs

Rasmus Blanck and Aleksandre Maskharashvili

CLASP, FLOV, University of Gothenburg, Sweden
{Rasmus.Blanck and Aleksandre.Maskharashvili}@gu.se

Abstract. We investigate the possibility of constructing an Abstract Categorial Grammar (ACG) that relates Tree Adjoining Grammar (TAG) and Higher Order Logic (HOL) formulas encoding Abstract Meaning Representations (AMRs). We also propose another ACG that relates TAG and HOL formulas expressing the neo-Davidsonian event semantics. Both of these encodings are based on the already existing ACG encoding of the syntax-semantics interface where TAG derivations are interpreted as HOL formulas representing Montague semantics. In particular, both of these encodings share the same abstract language coming from the ACG encoding of TAG with Montague semantics, which is second-order. For second-order ACGs, problems of parsing and generation are known to be of polynomial complexity. Thus we get the natural language generation and parsing with TAGs and HOL formulas modeling AMR for free.

1 Introduction

Abstract Meaning Representations (AMRs) [2] have been subject to the interest of the computational linguistics community as they offer meaning representations of natural language expressions (sentences, noun phrases, etc.) without explicitly referring to morpho-syntactic features of a particular natural language. Several works were proposed to make use of AMRs for natural language (semantic) parsing [1] as well as for (sentence) generation [10]. To provide a logical setting for AMR semantics, recently the following two approaches were offered: [3], which provides translations of AMRs into First Order Logic (FOL) formulas, and [19], which translates AMRs into Higher Order Logic (HOL) formulas that express neo-Davidsonian event semantics.

It has been claimed that the Tree Adjoining Grammar (TAG) formalism [12] [13] is beneficial for modelling natural language syntax as TAGs can express various phenomena (such as encoding long-distance dependencies) and at the same time, polynomial parsing algorithms exist for TAG. Various approaches have been developed for natural language parsing and generation using TAGs, not only at the sentential level [11] but also for discourse [13] [5].

Abstract Categorial Grammars (ACGs) [7] present a grammatical framework, designed in the spirit of type-logical grammars. ACGs proved to be capable of encoding various grammatical formalisms, including TAG. Moreover, ACGs allow one to model the syntax-semantics interface where the syntactic part comes from a TAG grammar [15]. Importantly, ACGs constructed for encoding TAG with semantics belong to the class of ACGs that enjoy polynomial parsing and generation algorithms [14].

An approach with a compositional treatment of event semantics, by interpreting syntactic trees of sentences into formulas expressing event semantics, is offered by [4]. In order to obtain event semantic interpretations from syntactic descriptions of sentences, ACG were employed ACGs in [20]. Neither of these two works, however, uses TAGs for their syntax.

One of the main problems of a compositional approach to event semantics is related to quantification. Following [9], which studies interactions of quantifiers and events in a type-logical setting, (1) is a typical example challenging compositional approaches to event semantics: while the syntactic scope of *every woman* is inside that of *kissed* (i.e. of an event), a part of the semantic interpretation of *every woman* scopes over *kissed* (i.e., $\exists k.(\mathsf{kiss}\, k))$ and it operates inside the scope of *kissed* (i.e., $arg_1\ k\ x$).

(1) John kissed every woman.
$\forall x(\mathsf{woman}\, x \supset \exists k(\mathsf{kiss}\ k) \wedge (arg_0\ k\ \mathsf{john}) \wedge (arg_1\ k\ x))$

The present work offers an approach to the syntax-semantics interface where syntax comes from TAG and the semantics is neo-Davidsonian. We follow the ACG encoding of TAG with Montague semantics given in [17], but we provide neo-Davidsonian semantics instead of Montagovian.

2 AMR

Banarescu et al. [2] introduce AMRs as a means of representing basic meaning of natural language phrases to facilitate producing uniform semantic annotations across various languages. An AMR is a directed, acyclic graph, with a unique root and labelled nodes and edges. These graphs can be represented in PENMAN notation (2.a), or as a FOL formula (2.b).

(2) a.
$$(w/want_{01} : arg_0(b/boy) \\ \quad : arg_1(g/go_{01} : arg_0\, b))$$

b.
$$\exists w \exists g \exists b\, (instance(w, want_{01}) \wedge instance(g, w) \wedge \\ \quad instance(b, boy) \wedge arg_0(w, b) \wedge arg_1(w, g) \wedge arg_0(g, b))$$

Graph nodes represent entities and events in a neo-Davidsonian style, while edges represent relations among them. Leaves can only be labelled with concepts, so that, for example, (b/boy) refers to an instance b of the concept *boy*. AMRs do not contain information about tense, aspect, number and articles, etc. AMRs do not express universal quantification either; rather, such quantifiers are treated as modifiers of the nouns they are quantifying over. To overcome these problems,[1] Stabler [19] suggests an augmentation (AAMR) of AMR in which decorations such as *want*$_{01}$*.pres* and *boy.sg* are used to express tense and number, and where quantification is given a more general treatment. AAMRs are mapped to HOL formulas, in which quantifiers always outscope the event existential quantifier, thus generating the basic surface order reading. To accomplish this, AAMR graphs are transformed into trees, where roles are encoded as node labels, contrary to the original AMR representation where they are encoded as arc labels. This allows AAMRs to have a standard term representation to which tree transducers can

[1] Bos [3] also deals with the restrictions of AMRs related to universal quantification, but he uses FOL instead of HOL. Here, we are interested in HOL translations and therefore focus on [19].

be applied, yielding HOL formulas encoding intended meanings that the initial AMRs are not able to express. For example, (3.a) is an AAMR in PENMAN notation, whereas (3.b) is its translation into HOL that represents event semantics for the sentence *Most boys do not walk*.

(3) a. $walk(:instance(walk_{01}.pres),$
$:arg_0(b(:instance(boy.pl), :quant(most))),$
$:polarity(-))$

b. $\mathsf{most}(\mathsf{boy.pl}, \lambda b \neg \exists w (\mathsf{walk}_{01}.\mathsf{pres}(w) \wedge :arg_0(w, b)))$

3 Tree Adjoining Grammars (TAG)

TAG is a tree generating formalism. A TAG *derived tree* language is obtained by combining elementary trees, which are either *initial* or *auxiliary*. Conceptually, an initial tree models domain of locality (e.g. verbs and their arguments), whereas auxiliary trees enable one to recursively expand (e.g. adverbs, adjectives) a syntactic tree. TAG express that by allowing initial trees to *substitute* only frontier nodes of a tree, whereas auxiliary trees can substitute internal nodes of a tree - this is called *adjunction*.[2] A node that is being substituted or adjoined should have the same *label* (usually modelling a category such as NP, VP, S, etc.) as the root node of the substituted or adjoined tree. Such nodes are called *substitution* and *adjuction* sites of a tree. For example, γ_{kissed}, γ_{John} and γ_{Mary} are initial trees, whereas $\gamma_{passionately}$ is an auxiliary one (see Figure 1). We can substitute γ_{John} and γ_{Mary} into γ_{kissed} on the frontier nodes labeled with np and adjoin $\gamma_{passionately}$ into γ_{kissed} on the node with label vp, we obtain the derived tree depicted in Figure 2(a).

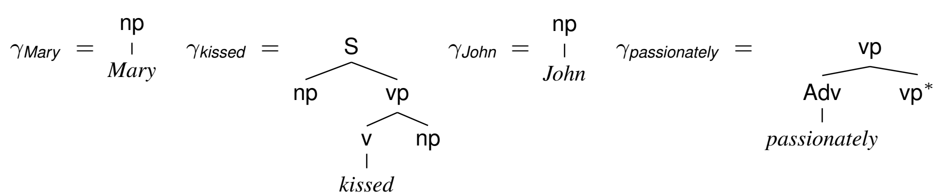


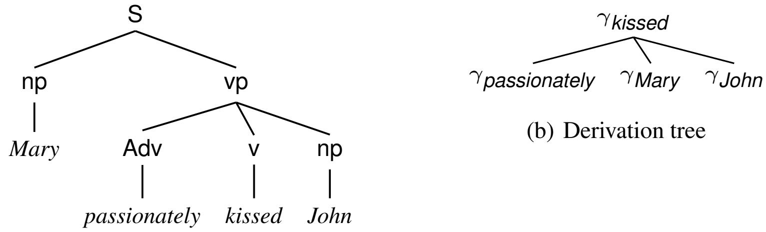
Fig. 1. TAG trees

The process of the production of the derived tree 2(a) is recorded by the corresponding *derivation* tree, which is represented as a tree 2(b).

4 The Syntax-Semantics interface for TAG using ACGs

An abstract categorial grammar (ACG) defines two languages, the *abstract* and *object* ones (the tecto and pheno grammatical levels, respectively, à la Curry). The object language is a homomorphic image (translation à la Montague) of the abstract one [7]. To

[2] Since (by definition) an internal node n of a tree γ has got some children, they would be left orphan as a result of adjoining of an auxiliary tree β on n. TAG has a solution for that: an(y) auxiliary tree β has a frontier node, marked with $*$, which has the same label as the root of β (and thus the same label as n). This frontier node, called the *foot* node of β, becomes mother to the children of the node n into the resultant tree of adjoining β into γ on n.



(a) Derived tree

(b) Derivation tree

Fig. 2. A TAG derived tree and the corresponding derivation tree for *Mary passionately kissed John*

define ACGs, let us first define the notion of a *higher-order linear signature*: it is a triple $\Sigma = \langle A, C, \tau \rangle$, where A is a finite set of atomic types, C is a finite set of constants, and τ is a mapping from C to $\mathscr{T}_A$, where $\mathscr{T}_A$ is the set of types built on A: $\mathscr{T}_A ::= A | \mathscr{T}_A \multimap \mathscr{T}_A$. $\Lambda(\Sigma)$ denotes the set of λ-terms[3] built on Σ. To denote that $M \in \Lambda(\Sigma)$ is of type α, we write $M :_{\Sigma} \alpha$ or just $M : \alpha$.

Definition 1. *An* ACG *is a quadruple* $\mathscr{G} = \langle \Sigma_1, \Sigma_2, \mathscr{L}, s \rangle$ *where:*

1. Σ_1 *and* Σ_2 *are two higher-order linear signatures, called the* abstract vocabulary *and the* object vocabulary*, respectively.*
2. $\mathscr{L} : \Sigma_1 \longrightarrow \Sigma_2$ *is called a lexicon of the ACG* $\mathscr{G}$. $\mathscr{L}$ *is a homomorphic mapping of types and terms built on* Σ_1 *to types and terms built on* Σ_2*, defined as follows:*
   (a) *If* $\alpha \multimap \beta \in \mathscr{T}_{\Sigma_1}$ *then* $\mathscr{L}(\alpha \multimap \beta) = \mathscr{L}(\alpha) \multimap \mathscr{L}(\beta)$.
   (b) *If* $\lambda x.M, (M\ K) \in \Lambda(\Sigma_1)$ *then* $\mathscr{L}(\lambda x.M) = \lambda x.\mathscr{L}(M)$ *and* $\mathscr{L}(M\ K) = \mathscr{L}(M)\ \mathscr{L}(K)$.
   (c) *For any constant* $c :_{\Sigma_1} \alpha$ *of* Σ_1 *we have* $\mathscr{L}(c) :_{\Sigma_2} \mathscr{L}(\alpha)$.
3. $s \in \mathscr{T}_{\Sigma_1}$ *(i.e., s is a type of the abstract vocabulary) is the* distinguished type *of the grammar* $\mathscr{G}$.

The abstract language *of* $\mathscr{G}$ *is defined as:* $\mathcal{A}(\mathscr{G}) = \{M \in \Lambda(\Sigma_1) \mid M :_{\Sigma_1} s \text{ and } M \text{ is closed}\}$
The object language *of* $\mathscr{G}$ *is* $\mathcal{O}(\mathscr{G}) = \{M \in \Lambda(\Sigma_2) \mid \exists u \in \mathcal{A}(\mathscr{G}).\ M =_{\beta\eta} \mathscr{L}_G(u)\}$

ACGs enable one to encode TAG derivation trees within the grammar: they are modelled as the abstract language [7]. Derived trees are modelled as the object language. One defines the following signatures and lexicons: a signature Σ_{TAG}, where TAG derivation trees are encoded; a signature Σ_{trees} that encodes TAG derived trees; a lexicon $\mathscr{L}_{d\text{-}ed\ trees} : \Sigma_{TAG} \longrightarrow \Sigma_{trees}$ that maps derivation trees to derived trees; the signature Σ_{Log} where one defines HOL terms encoding Montague semantics; and $\mathscr{L}_{Log} : \Sigma_{TAG} \longrightarrow \Sigma_{Log}$ that maps derivation trees to Montague semantics [15], [17].

Σ_{TAG}: Its atomic types include S, vp, np, S_A, $\mathsf{vp}_A \ldots$, where the X types stand for the categories (i.e. labels) X of the nodes where a substitution can occur, while the X_A types stand for the categories X of the nodes where an adjunction can occur. For each elementary tree $\gamma_{lex.\ entry}$, Σ_{TAG} contains a constant $C_{lex.\ entry}$ whose type encodes the adjunction and substitution sites of $\gamma_{lex.\ entry}$: every X-adjunction (resp. X-substitution)

[3] As a notational convention, we may use $\lambda x\, y.K$ instead of $\lambda x.\lambda\, y.K$. Instead of $\mathscr{L}(K) = M$, we write $K := M$ if it does not cause confusion.

site is modelled by an argument of type X_A (resp. X) of $C_{\textit{lex. entry}}$. Σ_{TAG} additionally contains constants $I_X : X_A$ that are meant to provide a *fake auxiliary tree* in the cases where no adjunction actually takes place in a TAG derivation. Since arguments of a $C_{\textit{lex. entry}}$ can be only atomic ones (any X_A and/or X is atomic), Σ_{TAG} is a *second-order* signature.

Here we are interested in semantic interpretations.[4] Constants of the semantic vocabulary Σ_{Log} are shown in Table 1. We have two atomic types in Σ_{Log}, e for entities and t for propositions. The lexicon $\mathscr{L}_{Log}$ from Σ_{TAG} to Σ_{Log} is provided in Table 2. The distinguished type of the ACGs for encoding TAG with semantics is S.

john, mary	$: e$	certainly	$: t \to t$
woman, important, walk	$: e \to t$	kiss, love	$: e \to e \to t$
passionately, fast	$: t \to t$	$\neg$	$: t \to t$
$\Rightarrow, \vee, \wedge$	$: t \to t \to t$	$\exists, \forall$	$: (e \to t) \to t$

Table 1. Constants in the semantic vocabulary Σ_{Log} for encoding Montague semantics

Constants of Σ_{TAG}	Their interpretations by $\mathscr{L}_{Log}$
$C_{\textit{woman}} : \mathsf{n}_A \multimap \mathsf{np}$	$\lambda D.\lambda q\,.D(\lambda x.\mathsf{woman}\,x)q$
$C_{\textit{smart}} : \mathsf{n}_A \multimap \mathsf{n}_A$	$\lambda D.\ \lambda n\,.\lambda q\,.\,D\ (\lambda\,x.\,(\mathsf{important}\ x) \wedge (n\ x))q$
$C_{\textit{every}}\,, C_{\textit{each}} : \mathsf{n}_A$	$\lambda\,P\,Q\,.\forall\,x.\,(P\,x)\ \supset\ (Q\,x)$
$C_{\textit{some}}\,, C_a : \mathsf{n}_A$	$\lambda\,P\,Q\,.\exists\,x.\,(P\,x)\ \wedge\ (Q\,x)$
$C_{\textit{passionately}} : \mathsf{vp}_A \multimap \mathsf{vp}_A$	$\lambda adv_v\ pred.\ adv_v\ (\lambda x.\ \mathsf{passionately}\ (pred\,x))$
$C_{\textit{kissed}} : \mathsf{S}_A \multimap \mathsf{vp}_A \multimap \mathsf{np} \multimap \mathsf{np} \multimap \mathsf{S}$	$\lambda adv_s\ adv_v\ sbj\ obj.$ $adv_s\ (sbj\ (\lambda x.(obj\ (adv_v(\lambda y.\mathsf{kiss}\,x\,y)))))$
$I_X : X_A$	$\lambda x.x$
S	t

Table 2. Interpretations by $\mathscr{L}_{Log}$

$$
\begin{aligned}
M_0 &= C_{\textit{kissed}}\ I_{\mathsf{S}}\ (C^{\mathsf{vp}}_{\textit{passionately}}\ I_{\mathsf{vp}})\ C_{\textit{Mary}}\ C_{\textit{John}}\ : \mathsf{S}\\
\mathscr{L}_{\textit{d-ed trees}}(M_0) &= \mathsf{S}_2\ (\mathsf{vp}_2\ (\mathsf{np}_1\ \textit{Mary})\ (\mathsf{vp}_2\ \textit{passionately}\ (\mathsf{v}_1\ \textit{kissed})))\ (\mathsf{np}_1\ \textit{John})\\
\mathscr{L}_{Log}(M_0) &= \mathsf{passionately}\,(\mathsf{kiss\,mary\,john})
\end{aligned}
$$

For instance, the term M_0 models the TAG derivation tree on Figure 2(b). By mapping M_0 with $\mathscr{L}_{\textit{d-ed trees}}$, one obtains the term representation of the derived tree shown on Figure 2(a); and by mapping M_0 with $\mathscr{L}_{Log}$, one gets a Montague style HOL formula, which expresses semantics of *Mary passionately kissed John*.

5 From TAG Derivation trees to AMR Style Formulas

Our goal is to interpret terms modeling TAG derivation trees as HOL formulas that are close to the standard AMR representations. While we focus on the declarative sentences

[4] For the details of mapping terms modelling TAG derivation trees into ones modelling derived trees, we refer the reader to [17].

and their interpretations, what we propose also includes a compositional approach to noun phrases and other expressions.[5]

We add a variable for every verb that denotes an event. For instance, a predicate signalled by an intransitive verb, such as *go*, becomes $\lambda x.\lambda h.\exists g\,(\mathsf{go}\,g) \wedge (arg_0\,g\,x) \wedge (h\,g)$ instead of $\lambda x.\mathsf{go}\,x$. The former term is of type $e \to (v \to t) \to t$. This treatment is inspired by Champollion's [4] approach to neo-Davidsonian semantic interpretations. While he chooses not to make a difference between arguments and adjuncts, we would like to encode arguments (of type e) within the semantics of an event predicate in order to be close to the AMR semantic representations, where core relations (arguments) are licensed by a verb frame. However, note that in AMRs, there is *only one* kind of internal node to represent both nouns and verbs. To reflect this in our encodings, all of these entities should have the type v instead of type e, and this change makes predicates signalled by intransitive verbs and nouns of type $v \to (v \to t) \to t$. This suggests that we should encode nouns as follows: $\lambda x.\lambda f.\exists n\,(\mathsf{noun}\,n) \wedge (\mathsf{instance}\,n\,x) \wedge (f\,n)$ instead of $\lambda x.\mathsf{noun}\,x$. This decision results in rather implausible interpretations such as $\exists x\,\exists m(\mathsf{man}\,m \wedge (\mathsf{instance}\,m\,x) \wedge \exists g\,(\mathsf{go}\,g) \wedge (arg_0\,g\,x))$ corresponding to *a man goes*. Even more problematic to interpret would be the formula encoding the semantics of *every man goes* : $\forall x\,(\exists m \mathsf{man}\,m \wedge (\mathsf{instance}\,m\,x) \supset \exists g\,(\mathsf{go}\,g) \wedge (arg_0\,g\,x))$. We deal with this shortcoming in the next section.

In order to encode a negation, let us note that in TAG it is modelled by an auxiliary tree that adjoins on the vp node of a tree. We model this fact by a constant of type vp_A in the vocabulary Σ_{TAG} with the following interpretation: $\lambda V x h.\neg(V x h)$. In words, it means that the negation scopes over the existentially closed event formula, but still allows for further continuations (which has been argued for in [4]).

john, mary	$: v$	certainly	$: t \to t$
woman, important, walk	$: v \to t$	kiss, love	$: v \to t$
passionately, fast	$: t \to t$	$\neg$	$: t \to t$
$\Rightarrow, \vee, \wedge$	$: t \to t \to t$	$\exists, \forall$	$: (v \to t) \to t$
arg_0, arg_1, arg_2	$: v \to v \to t$	$True$	$: v \to t$

Table 3. Constants in the semantic vocabulary Σ_{AMR}

We model trees anchored by nouns, adjectives and determiners (quantifier words and phrases), verbs, etc. as before in Σ_{TAG} but attribute to them different semantic interpretations. These new interpretations are shown in Table 4. To be precise, we create a new vocabulary Σ^E_{TAG} for encoding derivation trees by adding to Σ_{TAG} one more type T and a constant *Closure* of type $\mathsf{S} \multimap \mathsf{T}$. We need them in order to *close* a sentence (S), i.e., to model that there is no more content to come (no more continuations). In semantics, we interpret *Closure* by applying an interpretation of a sentence that is looking for a continuation to a *vacuous* continuation, which one models as $True : v \to t$, where $(True\,x) \wedge p$ is equivalent to p (for any $x : v$). Now, T will be our distinguished type. It is straightforward to map the new vocabulary, Σ^E_{TAG}, to the old one, Σ_{TAG}. We map T

[5] Pogodalla [16][17] shows how to encode other kind of sentences in the same principled way and thus one can easily give an account to them here as well.

to S, and $Closure$ to $\lambda x.x$; the rest of Σ^E_{TAG} is exactly the same as Σ_{TAG}, i.e., we map any ξ (being a constant or a type) from Σ^E_{TAG} to ξ in Σ_{TAG}.

To define the semantic interpretation with events, we create a new signature, called Σ_{AMR}, shown in Table 3. We construct $\mathscr{L}_{dere\text{-}amr} : \Sigma^E_{TAG} \longrightarrow \Sigma_{AMR}$ provided in Table 4.

In (4), we list the examples and their encodings in $\Lambda(\Sigma^E_{TAG})$ that we use here and onwards to exemplify our interpretations as we maintain Σ^E_{TAG} as the abstract vocabulary and therefore these terms will be in reused again.

(4) a. Every smart woman walks.
$M_1 = \textit{Closure}(C_{walks}\, I_{\mathsf{S}}\, I_{\mathsf{vp}}\, (C_{woman}\, (C_{smart}\, C_{every})))\ : \mathsf{T}$

b. John does not walk.
$M_2 = \textit{Closure}(C_{walks}\ I_{\mathsf{S}}\ C_{does\ not}\ C_{john})\ : \mathsf{T}$

c. Every smart woman walks fast.
$M_3 = \textit{Closure}(C_{walks} I_{\mathsf{S}}(C_{fast}\, I_{\mathsf{vp}})(C_{woman}\, (C_{smart}\, C_{every})))\ : \mathsf{T}$

d. Certainly, every smart woman walks.
$M_4 = \textit{Closure}(C_{walks}(C_{certainly}\, I_{\mathsf{S}}) I_{\mathsf{vp}}(C_{woman}\, (C_{smart}\, C_{every})))\ : \mathsf{T}$

For instance, consider (4)(a). It is modelled by the term M_3 of type T, which we can then interpret using a lexicon. Table 5 shows its interpretation by $\mathscr{L}_{dere\text{-}amr}$.

$$
\begin{array}{ll}
\mathsf{S} & := (v \to t) \to t \qquad\qquad \mathsf{T} := t \\
Closure & := \lambda P.P\,True\ :\ ((v \to t) \to t) \to t \\
C_{john} & := \lambda P.\ P\,\mathsf{john} \\
C_{walks} & := \begin{array}{l}\lambda adv_s\, adv_v\, subj. \\ adv_s\,(subj\,(adv_v(\lambda x.\lambda h.\exists w.\,(\mathsf{walk}\,w) \land (arg_0\,w\,x) \land (h\,w))))\end{array} \\
C_{smart} & := \lambda D.\lambda n.\lambda q.\lambda f.D(\lambda x h.(n\,x\,h) \land (\mathsf{smart}\,x))q\,f \\
C^{\mathsf{n}_A}_{every} & := \lambda p.\lambda q.\lambda f.\forall x.(p\,x\,f) \supset (q\,x\,f) \\
C^{\mathsf{n}_A \multimap \mathsf{np}}_{woman} & := \lambda D.D(\lambda\,x\,h.(\mathsf{woman}\,x \land h\,x)) \\
C^{\mathsf{S}_A \multimap \mathsf{S}_A}_{certainly} & := \lambda m.\,\lambda V.\ m\,(\lambda h.V(\lambda v.(\mathsf{certainly}\,v) \land (h\,v)) \\
C^{\mathsf{vp}_A \multimap \mathsf{vp}_A}_{fast} & := \lambda m.\,\lambda V.\ m\,(\lambda x.\lambda h.V\,x(\lambda v.(\mathsf{fast}\,v) \land (h\,v))) \\
C^{\mathsf{vp}_A}_{does\ not} & := \lambda V x h.\neg(V\,x\,h)
\end{array}
$$

Table 4. Interpretations by $\mathscr{L}_{dere\text{-}amr}$

$$
\begin{array}{l}
M_1 := \forall x(\mathsf{woman}\,x \land \mathsf{smart}\,x \supset \exists w\,(\mathsf{walk}\,w) \land (arg_0\,w\,x)) \\
M_2 := \neg\exists w\,(\mathsf{walk}\,w) \land (arg_0\,w\,\mathsf{john}) \\
M_3 := \forall x(\mathsf{woman}\,x \land \mathsf{smart}\,x \land \mathsf{fast}\,x \supset \exists w(\mathsf{walk}\,w) \land (arg_0\,w\,x) \land (\mathsf{fast}\,w)) \\
M_4 := \forall x\,(\mathsf{woman}\,x \land \mathsf{smart}\,x \land \mathsf{certainly}\,x \supset \exists w(\mathsf{walk}\,w) \land (arg_0\,w\,x) \land (\mathsf{certainly}\,w))
\end{array}
$$

Table 5. Interpretations of M_1, M_2, M_3 and M_4 by $\mathscr{L}_{dere\text{-}amr}$

As Table 5 shows, we obtain the desired interpretations for M_1 and M_2, but not for M_3 and M_4. This is due to the failure to distinguish between event entities and discourse entities. By this uniform treatment, the event continuation is applied not only to the verb phrase but also to the noun phrase, and that is the source of the incorrect results (e.g. in M_4 above, we obtain the subformula $(\mathsf{certainly}\, x)$, in words *certainly woman*, which is clearly not what we want). One concludes that in event semantics with continuations, nouns and predicates should not be treated in the same manner: event continuations should not operate on discourse referents, but on event ones. In the next section, the current approach is modified to make a proper distinction between event entities and other components of discourse.

6 From AMRs to Montague style HOL and to Neo-Davidson HOL

Thanks to the polynomial reversibility properties of second-order ACGs, we obtain from HOL formulas encoding AMRs the ones encoding Montague semantics, which are HOL formulas that do not incorporate a notions of event. For that we construct two ACGs sharing the abstract vocabulary encoding TAG derivation trees. In addition, in order to obtain a translation of HOL formulas encoding AMRs into HOL formulas encoding event semantics, we define yet another lexicon from Σ^E_{TAG} into a new signature Σ_{evhol}, shown in Table 6. Figure 3 shows interpretations of Σ^E_{TAG} into Σ_{evhol} (event semantics).

$\mathsf{john}, \mathsf{mary}$	$: e$	$\mathsf{kiss}, \mathsf{love}$	$: v \to t$
$\mathsf{woman}, \mathsf{important}, \mathsf{walk}$	$: e \to t$	$\exists, \forall$	$: (e \to t) \to t$
$\mathsf{passionately}, \mathsf{fast}$	$: v \to t$	$\neg$	$: t \to t$
$\Rightarrow, \vee, \wedge$	$: t \to t \to t$	$\exists^v$	$: (v \to t) \to t$
arg_0, arg_1	$: v \to e \to t$	Arg_0, Arg_1	$: v \to v \to t$

Table 6. Constants in the semantic vocabulary Σ_{evhol}

While we interpret constants encoding trees anchored by verbs and adverbs with the same terms as in the previous section, their types are now different. We denote this new lexicon with $\mathscr{L}_{evhol} : \Sigma^E_{\mathit{TAG}} \longrightarrow \Sigma_{\mathit{evhol}}$. Moreover, constants modelling quantifiers (e.g. C_{every}) are now of type $(e \to t) \to (e \to (v \to t) \to t) \to (v \to t) \to t$. This means that our encoding is asymmetric, whereas the standard one is symmetric: its type is $(e \to t) \to (e \to t) \to t$. This is explained by our choice not to use *event continuations* for noun phrases but only for events. In this setting, we obtain the following, correct interpretations of both M_3 and M_4, which were problematic in the previous section:

$$\mathscr{L}_{evhol}(M_3) = \forall x(\mathsf{woman}\, x \wedge \mathsf{smart}\, x \supset \exists^v w(\mathsf{walk}\, w) \wedge (arg_0\, w\, x) \wedge (\mathsf{fast}\, w))$$
$$\mathscr{L}_{evhol}(M_4) = \forall x(\mathsf{woman}\, x \wedge \mathsf{smart}\, x \supset \exists^v w(\mathsf{walk}\, w) \wedge (arg_0\, w\, x) \wedge (\mathsf{certainly}\, w))$$

Coreference: AMRs are useful for representing coreferences (for example, in the case of raising verbs such as *wants*, (5a)), but this property gets lost in Stabler's transformation of AMRs into trees. To encode coreferences, we follow [17]. In TAG, *wants* anchors an auxiliary tree, whereas *to sleep* anchors an initial tree. For instance, to derive the sentence (5a) in TAG, one substitutes the tree for *John* into the one for *wants* and the resultant tree adjoins into the S-labeled node of the initial tree *to sleep*. So, we

$$
\begin{array}{ll}
\Omega & \equiv_{\mathsf{def}} (v \to t) \to t \\
C_{\textit{john}} & := \lambda P.\, P\, \mathsf{john} : (e \to \Omega) \to \Omega \\
C_{\textit{walks}} & := \begin{array}{l} \lambda adv_s\, adv_v\, subj\, .adv_s \\ (subj\, (adv_v(\lambda x.\lambda h.\exists^v w.\, (\mathsf{walk}\, w) \wedge (arg_0\, w\, x) \wedge (h\, w)))) \end{array} \\
C_{\textit{kissed}} & := \begin{array}{l} \lambda adv_s\, adv_v\, subj\, obj\, .\, adv_s\, (subj(\lambda x. \\ (obj\, (adv_v(\lambda y.\lambda h.\exists^v w\, (\mathsf{kiss}\, w) \wedge (arg_0\, w\, x) \wedge (arg_1\, w\, y) \wedge (h\, w)))))) \end{array} \\
C_{\textit{every}} & := \lambda PQ.\lambda h.\forall x(Px \supset Qxh) : (e \to t) \to (e \to \Omega) \to \Omega \\
C_{\textit{a}} & := \lambda PQ.\lambda h.\exists x(Px \wedge Qxh) : (e \to t) \to (e \to \Omega) \to \Omega \\
C_{\textit{smart}} & := \lambda D.\lambda n.\lambda q.\lambda f.D(\lambda x.(n\, x\,) \wedge (\mathsf{smart}\, x))q\, f \\
C_{\textit{woman}} & := \lambda D.D(\lambda\, x.\mathsf{woman}\, x) \\
C_{\textit{certainly}} & := \lambda m.\, \lambda V.\, m\, (\lambda h.V(\lambda v.(\mathsf{certainly}\, v) \wedge (h\, v)) \\
C_{\textit{fast}} & := \lambda m.\, \lambda V.\, m\, (\lambda x.\lambda h.V\, x(\lambda v.(\mathsf{fast}\, v) \wedge (h\, v))) \\
C_{\textit{does not}} & := \lambda V x h.\neg(V\, x\, h) \\
\mathsf{n}_A & := (e \to t) \to (e \to \Omega) \to \Omega \qquad\qquad \mathsf{n} := e \to t \\
\mathsf{np} & := (e \to \Omega) \to \Omega \qquad\quad \mathsf{vp}_A := (e \to \Omega) \to e \to \Omega \\
\mathsf{S}_A & := \Omega \to \Omega \qquad\quad \mathsf{S} := \Omega \qquad\quad \mathsf{T} := t
\end{array}
$$

Fig. 3. Interpretation of $\Sigma^E_{\textit{TAG}}$ by $\mathscr{L}_{evhol}$

introduce the constants and then interpret them (see Figure 4). Note that even in the case of coreference and universal quantification, we obtain the desired results (e.g. (5b)).[6]

$C_{\textit{wants}}$: $\mathsf{S}_A \multimap \mathsf{vp}_A \multimap \mathsf{np} \multimap \mathsf{S}'_A$	$C_{\textit{to-sleep}}$: $\mathsf{S}'_A \multimap \mathsf{S}$

$$
\begin{array}{ll}
C_{\textit{wants}} := & \lambda adv_s\, adv_v\, subj.\lambda Pred.adv_s\, (subj(adv_v.\lambda x\, h. \\
& \exists^v w((\mathsf{want}\, w) \wedge (h\, w) \wedge (arg_0\, w\, x) \wedge Pred(\lambda Q.Q\, x)(\lambda r.\, Arg_1\, w\, r)) \\
C_{\textit{to-sleep}} := & \lambda cont.cont(\lambda subj.subj(\lambda x.\lambda f.\exists^v u.(\mathsf{sleep}\, u) \wedge (arg_0\, u\, x) \wedge (f\, u)) \\
\mathsf{S}'_A := & (((e \to \Omega) \to \Omega) \to \Omega) \to \Omega
\end{array}
$$

Fig. 4. Types and constants for modeling raising verbs and their interpretations

(5) a. John wants to sleep.
$M_5 = \textit{Closure}(C_{\textit{to-sleep}}\,(C_{\textit{wants}}\, I_{\mathsf{S}}\, I_{\mathsf{vp}} C_{\textit{john}}))\ : \mathsf{T}$
$\exists^v w(\mathsf{want}\, w) \wedge (arg_0\, w\, \mathsf{john}) \wedge (\exists^v u(\mathsf{sleep}\, u) \wedge (Arg_1\, w\, u) \wedge (arg_0\, u\, \mathsf{john}))$

b. Every boy wants to sleep.
$M_6 = \textit{Closure}(C_{\textit{to-sleep}}\,(C_{\textit{wants}}\, I_{\mathsf{S}}\, I_{\mathsf{vp}}(C_{\textit{boy}} C_{\textit{every}})))\ : \mathsf{T}$
$\forall x(\mathsf{boy}\, x \supset \exists^v w(\mathsf{want}\, w) \wedge (arg_0\, w\, x) \wedge (\exists^v u.(\mathsf{sleep}\, u) \wedge (Arg_1\, w\, u) \wedge (arg_0\, u\, x)))$

[6] ACG files encoding grammar and examples provided in Section 6 can be found at the following link: `https://www.dropbox.com/s/g2c58yq0ulp7a3j/AMR-TAG_ACG.zip?dl=0`.

7 Future Work and Conclusion

To encode certain kind of complex interactions between events and quantification, second-order ACGs may not suffice. For instance, consider (6) (from [20]). In semantics, *everyday* quantifies over times of events of *kissing*, but in syntax, *everyday* is an S-modifier of a sentence. To model this kind of complex scope interactions, one may invent new arguments of verbs that can be their modifiers in syntax whereas they play special roles in semantics. However, such an approach would deviate from the generic approach of the ACG encoding of TAG. Another way is to use higher-order ACGs. [17] presents a generic way of overcoming scoping problems of a similar kind. His approach leads to third-order ACGs, for which one cannot guarantee the polynomial parsing property (there is a third-order ACG generating an NP-complete language [18]).

(6) John kisses Mary everyday.
$\forall x\,(\mathsf{day}\,x) \supset \exists^v w(\mathsf{kiss}\,w) \wedge (arg_0\,w\,\mathsf{John}) \wedge (arg_1\,w\,\mathsf{Mary}) \wedge (\mathsf{time}\,w\,x)$

Stabler suggests to use de Groote and Lebedeva's approach [8] to pronouns and the definite article while dealing with AMRs in HOL. The dynamic setting where their approach is developed is a type-logical one, which is very close to the idea of ACGs as one can make distinctions between two levels of grammars. This makes us believe that the encoding proposed in this paper could be beneficial for that. In addition, ACGs were employed to study some discourse formalisms based on TAGs [6]. Thus, further extending the current approach with an aim of integrating it within already existing discourse encoding of TAG could be done as future work.

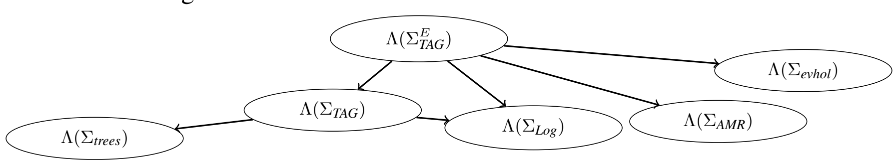


Fig. 5. ACG architecture for a syntactic and several semantic interpretations

To sum up, the current work makes it explicit that one can obtain AMR style semantic formulas compositionally from a grammar. With the same grammar, one obtains Montagovian HOL semantic representations. Again, the same grammar is employed in order to obtain HOL representations modelling neo-Davidsonian event semantics where negations and quantifiers, including a universal one, interact with events so that one obtains correct interpretations. Since all these encodings are done within second-order ACGs, one can draw correspondences between these interpretations using an algorithm of polynomial complexity. This makes the ACG architecture we constructed (depicted in Figure 5) beneficial for natural language generation/parsing tasks with AMRs and TAGs.

References

1. Y. Artzi, K. Lee, and L. Zettlemoyer. Broad-coverage CCG semantic parsing with AMR. In *Proceedings of the 2015 Conference on Empirical Methods in Natural Language Processing*, pages 1699–1710. Association for Computational Linguistics, 2015.

2. L. Banarescu, C. Bonial, S. Cai, M. Georgescu, K. Griffitt, U. Hermjakob, K. Knight, P. Koehn, M. Palmer, and N. Schneider. Abstract meaning representation for sembanking. In *Proceedings of the 7th Linguistics Annotation Workshop & Interoperability with Discourse*, pages 178–186, Sofia, Bulgaria, 2013.
3. J. Bos. Expressive power of abstract meaning representations. *Computational Linguistics*, 42(3):527–535, 2016.
4. L. Champollion. The interaction of compositional semantics and event semantics. *Linguistics and Philosophy*, 38(1):31–66, Feb 2015.
5. L. Danlos. G-tag: A lexicalized formalism for text generation inspired by tree adjoining grammar. In A. Abeillé and O. Rambow, editors, *Tree Adjoining Grammars: Formalisms, Linguistic Analysis, and Processing*, pages 343–370. CSLI Publications, 2000.
6. L. Danlos, A. Maskharashvili, and S. Pogodalla. Interfacing sentential and discourse TAG-based grammars. In *Proceedings of the 12th International Workshop on Tree Adjoining Grammars and Related Formalisms (TAG+12)*, pages 27–37, 2016.
7. P. de Groote. Towards abstract categorial grammars. In *Proceedings of ACL*, pages 148–155, 2001.
8. P. de Groote and E. Lebedeva. Presupposition Accommodation as Exception Handling. In R. Fernandez, Y. Katagiri, K. Komatani, O. Lemon, and M. Nakano, editors, *The 11th Annual Meeting of the Special Interest Group on Discourse and Dialogue*, SIGDIAL 2010 Conference, pages 71–74, Tokyo, Japan, 2010. Association for Computational Linguistics.
9. P. de Groote and Y. Winter. A type-logical account of quantification in event semantics. In *Logic and Engineering of Natural Language Semantics 11*, Tokyo, Japan, Nov. 2014.
10. J. Flanigan, C. Dyer, N. A. Smith, and J. Carbonell. Generation from abstract meaning representation using tree transducers. In *Proceedings of the 2016 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 731–739. Association for Computational Linguistics, 2016.
11. C. Gardent and L. Perez-Beltrachini. RTG based surface realisation for TAG. In *Proceedings of the 23rd International Conference on Computational Linguistics (COLING 2010)*, pages 367–375, Beijing, China, August 2010. Coling 2010 Organizing Committee.
12. A. K. Joshi, L. S. Levy, and M. Takahashi. Tree adjunct grammars. *Journal of Computer and System Sciences*, 10(1):136–163, 1975.
13. A. K. Joshi and Y. Schabes. Tree-adjoining grammars. In G. Rozenberg and A. Salomaa, editors, *Handbook of formal languages*, volume 3, chapter 2. Springer, 1997.
14. M. Kanazawa. Parsing and generation as Datalog queries. In *Proceedings of the 45th Annual Meeting of the Association of Computational Linguistics (ACL)*, pages 176–183, 2007.
15. S. Pogodalla. Computing semantic representation: Towards ACG abstract terms as derivation trees. In *Proceedings of TAG+7*, pages 64–71, 2004.
16. S. Pogodalla. Advances in Abstract Categorial Grammars: Language Theory and Linguistic Modeling. ESSLLI 2009 Lecture Notes, Part II. 2009.
17. S. Pogodalla. A syntax-semantics interface for Tree-Adjoining Grammars through Abstract Categorial Grammars. *Journal of Language Modelling*, 5(3):527–605, 2017.
18. S. Salvati. *The Mathematics of Language: 10th and 11th Biennial Conference*, chapter A Note on the Complexity of Abstract Categorial Grammars, pages 266–271. Springer Berlin Heidelberg, 2010.
19. E. Stabler. Reforming AMR. In A. Foret, R. Muskens, and S. Pogodalla, editors, *Formal Grammar*, pages 72–87. Springer Berlin Heidelberg, 2018.
20. Y. Winter and J. Zwarts. Event semantics and abstract categorial grammar. In M. Kanazawa, A. Kornai, M. Kracht, and H. Seki, editors, *The Mathematics of Language*, pages 174–191. Springer Berlin Heidelberg, 2011.

Intervals and Events with and without Points

Tim Fernando*

Computer Science Department, Trinity College Dublin
Tim.Fernando@tcd.ie

Abstract. Intervals and events are examined in terms of strings with and without points, represented by symbols that occur uniquely in strings. Allen interval relations, Dowty's aspect hypothesis and inertia are understood against strings, compressed into canonical forms, describable in Monadic Second-Order logic. That understanding is built around a translation of strings that replaces stative predicates by their borders.

1 Introduction

To analyze temporal relations between events, James Allen treats intervals as primitive (not unlike [7]), noting

> There seems to be a strong intuition that, given an event, we can always "turn up the magnification" and look at its structure. ... Since the only times we consider will be times of events, it appears that we can always decompose times into subparts. Thus the formal notion of a time point, which would not be decomposable, is not useful. [1, page 834].

In place of an indivisible point, an arbitrarily decomposable interval t might be conceived as a box filled by a predicate such as *rain* that is *homogeneous* in that it holds of t iff it holds of any pair of intervals whose union is t, illustrated by the equivalence between (a) and (b).

(a) It rained from 8 am to midnight.
(b) It rained from 8 am to noon, and from 10 am to midnight.

David Dowty has famously hypothesized that

> the different aspectual properties of the various kinds of verbs can be explained by postulating a single homogeneous class of predicates — *stative predicates* — plus three or four sentential operators or connectives. [2, page 71].

Dowty's investigation of his hypothesis in terms of intervals and worlds is reformulated in [3] using strings of finite sets of homogeneous predicates. A simple

* My thanks to three anonymous referees for helpful comments.

example of such a string is the representation of the Allen *overlap* relation between intervals a and a' as the string

$$\boxed{}\boxed{a}\boxed{a,a'}\boxed{a'}\boxed{} \tag{1}$$

of length 5, starting with an empty box $\boxed{}$ for times before a, followed by $\boxed{a}$ for times in a but not a', followed by $\boxed{a,a'}$ for times in a and a', followed by $\boxed{a'}$ for times in a' but not a, followed by $\boxed{}$ for times after a'.[1] In (1), the intervals a and a' are identified with predicates interpreted as the subsets

$$U_a = \{2,3\} \text{ and } U_{a'} = \{3,4\}$$

of the set $\{1,2,3,4,5\}$ of string positions where a and a' (respectively) occur.

In general, a string $s = \alpha_1 \cdots \alpha_n$ of n subsets α_i of a set A specifies for each $a \in A$, a subset of the set

$$[n] := \{1,\ldots,n\}$$

of string positions, namely, the set

$$U_a \ := \ \{i \in [n] \mid a \in \alpha_i\}$$

of positions where a occurs. If we repackage s as the model

$$Mod(s) \ := \ \langle [n], S_n, \{U_a\}_{a\in A}\rangle$$

over $[n]$ with successor relation

$$S_n \ := \ \{(i,i+1) \mid i \in [n-1]\}$$

then a theorem due to Büchi, Elgot and Trakhtenbrot says the regular languages over the set 2^A of subsets of A are given by the sentences φ of MSO_A as

$$\{s \in (2^A)^+ \mid Mod(s) \models \varphi\}$$

where MSO_A is Monadic Second-Order logic over strings with unary predicates labelled by A (e.g., [8]). The Büchi-Elgot-Trakhthenbrot theorem is usually formulated for strings over the alphabet A (as opposed to 2^A above), but there are at least two advantages in using the alphabet 2^A. First, for applications such as (1), it is convenient to put zero, one or more intervals in boxes for a simple temporal construal of succession. Second, for any subset $A' \subseteq A$ of A, a string $s = \alpha_1 \cdots \alpha_n \in (2^A)^+$ need only be intersected componentwise with A' to capture the *A'-reduct of $Mod(s)$* by the string

$$\rho_{A'}(\alpha_1 \cdots \alpha_n) \ := \ (\alpha_1 \cap A') \cdots (\alpha_n \cap A').$$

[1] Boxes are drawn instead of $\emptyset$ and curly braces $\{\cdot\}$ so as not to confuse, for example, the empty language $\emptyset$ with the string $\square$ of length one.

For example, returning to (1) with $A' = \{a\}$,

$$\rho_{\{a\}}(\fbox{ }\fbox{a}\fbox{a,a'}\fbox{a'}\fbox{ }) \;=\; \fbox{ }\fbox{a}\fbox{a}\fbox{ }\fbox{ }.$$

Only elements of A' are observable in A'-reducts. To expand what can be observed (and turn up, as Allen puts it, the magnification), A must be enlarged (not reduced). On (1), for instance, a third interval a'' may come into view, overlapping both a and a', as depicted by the string

$$\fbox{ }\fbox{a''}\fbox{a,a''}\fbox{a,a',a''}\fbox{a,a'}\fbox{a'}\fbox{ }.$$

Its $\{a\}$-reduct

$$\rho_{\{a\}}(\fbox{ }\fbox{a''}\fbox{a,a''}\fbox{a,a',a''}\fbox{a,a'}\fbox{a'}\fbox{ }) \;=\; \fbox{ }\fbox{ }\fbox{a}\fbox{a}\fbox{a}\fbox{ }\fbox{ }$$

is, like the $\{a\}$-reduct $\fbox{ }\fbox{$a$}\fbox{$a$}\fbox{ }\fbox{ }$ of (1), just another representation of $\fbox{ }\fbox{$a$}\fbox{ }$ inasmuch as any box α of homogeneous predicates is decomposable to α^n for any positive integer n. With this in mind, let us define a *stutter of* a string $\alpha_1 \cdots \alpha_n$ to be a box α_i such that $\alpha_i = \alpha_{i+1}$. To remove stutters, we apply *block compression* bc, defined by induction on the string length n

$$\mathit{bc}(\alpha_1 \cdots \alpha_n) \;:=\; \begin{cases} \alpha_1 \cdots \alpha_n & \text{if } n < 2 \\ \mathit{bc}(\alpha_2 \cdots \alpha_n) & \text{else if } \alpha_1 = \alpha_2 \\ \alpha_1\, \mathit{bc}(\alpha_2 \cdots \alpha_n) & \text{otherwise} \end{cases}$$

so that $\mathit{bc}(s)$ has no stutter, and

$$s \text{ has no stutter} \quad \iff \quad s = \mathit{bc}(s).$$

The finite-state approach to temporality in [4, 5] identifies a string s of sets of homogeneous predicates with its stutterless form $\mathit{bc}(s)$.

But can we assume a string representing an event is built solely from homogeneous predicates? It is not clear such an assumption can be taken for granted. The event nucleus of [9], for instance, postulates not only states but also events that can be extended or atomic, including points. Given a string s over the alphabet 2^A, let us agree an element $a \in A$ is an *s-point* if it occurs exactly once in s — i.e.,

$$\rho_{\{a\}}(s) \;\in\; \fbox{ }^*\fbox{a}\fbox{ }^* \tag{2}$$

Just as a string of statives can be compressed by removing stutters through bc, a string s of points can be compressed by deleting all occurrences in s of the empty box $\Box$ for $d_\Box(s)$. Line (2) above simplifies to the equation

$$d_\Box(\rho_{\{a\}}(s)) \;=\; \fbox{a}.$$

We shall see that for an s-interval a, the corresponding equation is

$$d_\Box(b(\rho_{\{a\}}(s))) \;=\; \fbox{$l(a)$}\fbox{$r(a)$}$$

for a certain function b on strings that associates a with a left border $l(a)$ and right border $r(a)$. The precise details are spelled out in section 2, where the set of interval relations from [1] are analyzed from the perspective of MSO_A through formulas such as

$$(\forall z)(P_a(z) \equiv (x < z \wedge z \leq y))$$

saying a occurs exactly at positions $> x$ and $\leq y$ (where P_a is the unary predicate symbol in MSO_A labeled by a, and $<$ and $\leq$ are defined from the successor relation via monadic second-order quantification). Applications to events are taken up in section 3, where the set of predicate labels is expanded in a constrained manner and the map b is inverted to expose a notion of inertia and force. A synthesis of bc and $d_\square$ is presented, suited to strings with or without points.

2 Strings of Points and Allen Relations

The key notion in this section is projection between strings, for which it is useful to define the *vocabulary* of a string $\alpha_1 \cdots \alpha_n$ of sets α_i to be the union

$$voc(\alpha_1 \cdots \alpha_n) \ := \ \bigcup_{i=1}^{n} \alpha_i$$

(making $voc(s)$ the $\subseteq$-least set A such that $s \in (2^A)^*$). We can then say *s projects to s'* if deleting all occurrences of the empty box $\square$ from the $voc(s')$-reduct of s yields s'

$$d_\square(\rho_{voc(s')}(s)) = s'$$

(recalling that $d_\square(\alpha_1 \cdots \alpha_n)$ is what remains after deleting each $\alpha_i = \square$). The $\mathrm{MSO}_{\{a\}}$-sentence

$$(\exists x)(\forall y)(P_a(y) \equiv x = y)$$

states there is a position where a occurs and nowhere else, as asserted in (2). It follows immediately that

Proposition 1. *The following are equivalent, for any $a \in A$ and $s \in (2^A)^*$.*

(i) $Mod(s) \models (\exists x)(\forall y)(P_a(y) \equiv x = y)$
(ii) $\rho_{\{a\}}(s) \in \square^* \boxed{a} \square^*$
(iii) *s projects to* $\boxed{a}$

Turning to (bounded) intervals, we define the string function b mentioned in the introduction relative to a set A, with which we associate a set

$$A_\bullet \ := \ \{l(a) \mid a \in A\} \cup \{r(a) \mid a \in A\}$$

formed from two 1-1 functions l and r, under the assumption that the three sets

$$A, \ \ \{l(a) \mid a \in A\} \ \text{ and } \{r(a) \mid a \in A\}$$

are pairwise disjoint. (Think of $l(a)$ and $r(a)$ as terms — bits of syntax — rather than say, numbers.) Now, let the function

$$b_A : (2^A)^* \to (2^{A_\bullet})^*$$

map a string $\alpha_1 \cdots \alpha_n$ of subsets α_i of A to a string $\beta_1 \cdots \beta_n$ of subsets β_i of $A_\bullet$ as follows

$$\beta_i \ := \ \begin{cases} \{r(a) \mid a \in \alpha_n\} & \text{if } i = n \\ \{l(a) \mid a \in \alpha_{i+1} - \alpha_i\} \ \cup \ \{r(a) \mid a \in \alpha_i - \alpha_{i+1}\} & \text{if } i < n. \end{cases}$$

For example, for $a, a' \in A$,

$$b_A(\boxed{}\boxed{a}\boxed{a,a'}\boxed{a'}\boxed{}) \ = \ \boxed{l(a)}\boxed{l(a')}\boxed{r(a)}\boxed{r(a')}\boxed{} \ .$$

To simplify notation, we will often drop the subscript A on b_A. The idea behind b is that P_a is an interval iff it is the half-open interval $(l(a), r(a)]$ with open left border $l(a)$ and closed right border $r(a)$. For an interval analog of Proposition 1, recall the MSO-formula

$$\text{bounded}_a(x, y) \ := \ (\forall z)(P_a(z) \equiv (x < z \wedge z \leq y))$$

mentioned in the introduction, and observe that

$$b(\rho_{\{a\}}(s)) \ = \ \rho_{\{l(a),r(a)\}}(b(s)).$$

Proposition 2. *The following are equivalent, for any $a \in A$ and $s \in (2^A)^*$.*

(i) $Mod(s) \models (\exists x)(\exists y)(x < y \wedge bounded_a(x, y))$
(ii) $\rho_{\{a\}}(s) \in \boxed{}^+ \boxed{a}^+ \boxed{}^*$
(iii) $b(\rho_{\{a\}}(s)) \in \boxed{}^* \boxed{l(a)}\boxed{}^* \boxed{r(a)}\boxed{}^*$
(iv) $b(s)$ *projects to* $\boxed{l(a)}\boxed{r(a)}$

Focussing on strings s over the alphabet $2^{A_\bullet}$ (as opposed to 2^A), let us agree that a is an *s-interval* if s projects to $\boxed{l(a)}\boxed{r(a)}$ (as Proposition 2 suggests), and also that *s demarcates* A if each $a \in A$ is an s-interval. We show next how to generate the strings that demarcate a finite set A. The plan is to map any finite sequence $a_1 \cdots a_n$ into a finite set $\mathcal{L}_\bullet(a_1 \cdots a_n)$ of strings establishing

Proposition 3. *For any n distinct $a_1, \ldots, a_n$, a string s demarcates $\{a_1, \ldots, a_n\}$ iff s projects to some string in $\mathcal{L}_\bullet(a_1 \cdots a_n)$.*

The languages $\mathcal{L}_\bullet(a_1 \cdots a_n)$ are defined by induction on n. Writing ϵ for the string of length 0, we set

$$\mathcal{L}_\bullet(\epsilon) \ := \ \epsilon$$

conflating a string s as usual with the language $\{s\}$. The inductive step is

$$\mathcal{L}_\bullet(a_1 \cdots a_n a_{n+1}) \ := \ \mathcal{L}_\bullet(a_1 \cdots a_n) \ \&^\square \ \boxed{l(a_{n+1})}\boxed{r(a_{n+1})}$$

for a certain operation $\&^\square$ defined as follows. Given two strings $\alpha_1 \cdots \alpha_k$ and $\alpha'_1 \cdots \alpha'_k$ of sets with the same length k, we form their componenwise union for their superposition

$$(\alpha_1 \cdots \alpha_k) \ \& \ (\alpha'_1 \cdots \alpha'_k) \ := \ (\alpha_1 \cup \alpha'_1) \cdots (\alpha_k \cup \alpha'_k).$$

We lift & to languages L and L' stringwise

$$L \ \& \ L' \ := \ \bigcup_{k \geq 0} \{s \& s' \mid s \in L_k \text{ and } s' \in L'_k\}$$

where L_k is the set of strings in L of length k, and similarly for L'_k. Next, we collect strings $d_\square$-equivalent to s and s' in $d_\square^{-1} d_\square(s)$ and $d_\square^{-1} d_\square(s')$ respectively, which we superpose for

$$s \ \&_\square \ s' \ := d_\square^{-1} d_\square(s) \ \ \& \ \ d_\square^{-1} d_\square(s')$$

and then reduce to the finite set

$$s \ \&^\square \ s' \ := \{d_\square(s'') \mid s'' \in s \ \&_\square s'\}$$

and finally lift to languages L, L' stringwise

$$L \ \&^\square \ L' \ := \ \bigcup_{s \in L} \bigcup_{s' \in L'} s \ \&^\square s'.$$

Proposition 3 is proved by induction on $n \geq 1$. The case $n = 1$ is immediate

$$\mathcal{L}_\bullet(a_1) \ = \ \epsilon \ \&^\square \ \boxed{l(a_1)}\boxed{r(a_1)} \ = \ \boxed{l(a_1)}\boxed{r(a_1)} \ .$$

For the inductive step $n+1$, appeal to $a_{n+1} \notin \{a_1, \ldots, a_n\}$, the induction hypothesis, and

Lemma 4. *If $voc(s) \cap voc(s') = \emptyset$, then every string in $s \&^\square s'$ projects to $d_\square(s)$ and $d_\square(s)'$.*

When $a \neq a'$, a routine calculation shows

$$\mathcal{L}_\bullet(aa') \ = \ \{\mathfrak{s}_R(a, a') \mid R \in \mathcal{AR}\}$$

where the 13 interval relations in [1] constitute the set

$$\mathcal{AR} \ := \ \{<, >, \text{d,di,f,fi,m,mi,o,oi,s,si}, =\}$$

and for each $R \in \mathcal{AR}$, $\mathfrak{s}_R(a, a')$ is the string with vocabulary

$$\{l(a), r(a), l(a'), r(a')\}$$

given in Table 1 such that for $s \in (2^A)^*$ (as opposed to $(2^{A_\bullet})^*$),

$$s \models aRa' \quad \Longleftrightarrow \quad b(s) \text{ projects to } \mathfrak{s}_R(a, a').$$

R	aRa'	$\mathfrak{s}_R(a,a')$	R^{-1}	$\mathfrak{s}_{R^{-1}}(a,a')$
$<$	a before a'	$\boxed{l(a)}\boxed{r(a)}\boxed{l(a')}\boxed{r(a')}$	$>$	$\boxed{l(a')}\boxed{r(a')}\boxed{l(a)}\boxed{r(a)}$
m	a meets a'	$\boxed{l(a)}\boxed{r(a),l(a')}\boxed{r(a')}$	mi	$\boxed{l(a')}\boxed{r(a'),l(a)}\boxed{r(a)}$
o	a overlaps a'	$\boxed{l(a)}\boxed{l(a')}\boxed{r(a)}\boxed{r(a')}$	oi	$\boxed{l(a')}\boxed{l(a)}\boxed{r(a')}\boxed{r(a)}$
s	a starts a'	$\boxed{l(a),l(a')}\boxed{r(a)}\boxed{r(a')}$	si	$\boxed{l(a),l(a')}\boxed{r(a')}\boxed{r(a)}$
d	a during a'	$\boxed{l(a')}\boxed{l(a)}\boxed{r(a)}\boxed{r(a')}$	di	$\boxed{l(a)}\boxed{l(a')}\boxed{r(a')}\boxed{r(a)}$
f	a finishes a'	$\boxed{l(a')}\boxed{l(a)}\boxed{r(a),r(a')}$	fi	$\boxed{l(a)}\boxed{l(a')}\boxed{r(a),r(a')}$
$=$	a equal a'	$\boxed{l(a),l(a')}\boxed{r(a),r(a')}$	$=$	

Table 1. Allen interval relations as strings of points[2]

Given a set A of interval names and a specification $f : (A \times A) \to 2^{\mathcal{AR}}$ of sets $f(a,a')$ of Allen relations possible for pairs (a,a') from A, is there a string s that meets that specification in the sense of (3) below?

$$\text{for all } a,a' \in A, \text{ there exists } R \in f(a,a') \text{ such that } s \models aRa' \tag{3}$$

A popular tool from [1] is the transitivity table $T : (\mathcal{AR} \times \mathcal{AR}) \to 2^{\mathcal{AR}}$ mapping a pair (R,R') from $\mathcal{AR}$ to the set $T(R,R')$ of relations $R'' \in \mathcal{AR}$ such that for some intervals X, Y and Z,

$$\text{X } R \text{ Y and Y } R' \text{ Z and X } R'' \text{ Z.}$$

A function $f : (A \times A) \to 2^{\mathcal{AR}}$ is a *T-consistent labeling of A* if for all $a,a',a'' \in A$,

$$f(a,a'') \subseteq \bigcup_{R \in f(a,a')} \bigcup_{R' \in f(a',a'')} T(R,R').$$

T-consistency falls short of true consistency; Figure 5 in [1, page 838] provides a T-consistent labelling f of a set A of 4 intervals for which there is *no* string s of subsets of A satisfying (3) above. But for A of 3 or fewer intervals, every T-consistent labeling of A has a string s making (3) true. By Proposition 3, we can compute $T(R,R')$ by searching the language $\mathcal{L}_\bullet(a_1a_2a_3)$ for strings that satisfy a_1Ra_2 and $a_2R'a_3$

$$\begin{aligned} T(R,R') \;=\; \{R'' \in \mathcal{AR} \mid (\exists s \in \mathcal{L}_\bullet(a_1a_2a_3))\ & s \text{ projects to} \\ & \mathfrak{s}_R(a_1,a_2),\ \mathfrak{s}_{R'}(a_2,a_3) \text{ and } \mathfrak{s}_{R''}(a_1,a_3)\}\ . \end{aligned} \tag{4}$$

Implicit in (4) is a generate-and-test approach, which we can improve by refining the superposition $\&^\Box$ underlying $\mathcal{L}_\bullet(a_1 \cdots a_n)$ to an operation $\&_p$ such that for all $s,s' \in (2^A - \{\Box\})^*$,

$$s \;\&_p\; s' \;=\; \{s'' \in (s \;\&^\Box\; s') \mid s'' \text{ projects to } s \text{ and } s'\} \tag{5}$$

[2] The strings $\mathfrak{s}_R(a,a')$ can be derived from strings $\mathfrak{s}_R^\circ(a,a')$ over the alphabet $\{a,a'\}$ by the equation $\mathfrak{s}_R(a,a') = b(\Box\mathfrak{s}_R^\circ(a,a'))$. For example, $\mathfrak{s}_<^\circ(a,a') = \boxed{a}\boxed{\ }\boxed{a'}$ and $\mathfrak{s}_{\mathrm{m}}^\circ(a,a') = \boxed{a}\boxed{a'}$. A full list of $\mathfrak{s}_R^\circ(a,a')$, for every Allen relation R, can be found in Table 7.1 in [4, page 223].

(noting from Lemma 4 that for strings in the superposition of s with s' to project to $d_\Box(s)$ and $d_\Box(s')$, we assumed s and s' have disjoint vocabularies). To define $\&_p$, we first construct a family of 3-ary relations $\&_{\Sigma,\Sigma',\Sigma''}$ on strings over the alphabet 2^Σ, indexed by subsets Σ' and Σ'' of Σ. We proceed by induction, with base case

$$\frac{}{\&_{\Sigma,\Sigma',\Sigma''}(\epsilon,\epsilon,\epsilon)}$$

superposing ϵ with itself to get itself, and 3 rules which given $\&_{\Sigma,\Sigma',\Sigma''}(s,s',s'')$, extend s'' by a symbol added to s

$$\frac{\&_{\Sigma,\Sigma',\Sigma''}(s,s',s'') \qquad \alpha \subseteq \Sigma - \Sigma''}{\&_{\Sigma,\Sigma',\Sigma''}(\alpha s, s', \alpha s'')}$$

or to s'

$$\frac{\&_{\Sigma,\Sigma',\Sigma''}(s,s',s'') \qquad \alpha' \subseteq \Sigma - \Sigma'}{\&_{\Sigma,\Sigma',\Sigma''}(s, \alpha' s', \alpha' s'')}$$

or to both (in part)

$$\frac{\&_{\Sigma,\Sigma',\Sigma''}(s,s',s'') \qquad \alpha,\alpha' \subseteq \Sigma \qquad \alpha \cap \Sigma'' \subseteq \alpha' \qquad \alpha' \cap \Sigma' \subseteq \alpha}{\&_{\Sigma,\Sigma',\Sigma''}(\alpha s, \alpha' s, (\alpha \cup \alpha') s'')}$$

subject in all cases to certain conditions on the symbol added, expressed through $\Sigma, \Sigma', \Sigma''$. The case $\Sigma' = \Sigma'' = \Box$ gives $\&^\Box$

$$\&_{\Sigma,\Box,\Box}(s,s',s'') \iff s'' \in (s \,\&^\Box\, s')$$

for all $s, s', s'' \in (2^\Sigma)^*$. More generally, however, the point of non-empty Σ' and Σ'' is to constrain the superposition according to

Proposition 5. *For all* $\Sigma', \Sigma'' \subseteq \Sigma$ *and* $s, s', s'' \in (2^\Sigma - \{\Box\})^*$,

$$\begin{aligned} \&_{\Sigma,\Sigma',\Sigma''}(s,s',s'') \iff\ & s'' \in s\&^\Box s' \text{ and} \\ & s'' \text{ projects to } d_\Box(\rho_{\Sigma'}(s)) \text{ and } d_\Box(\rho_{\Sigma''}(s')). \end{aligned}$$

Now, for $\&_P$, let Σ be $A_\bullet$, and Σ' be the vocabulary of s, and Σ'' be the vocabulary of s'

$$\&_P(s,s') \ := \ \{s'' \mid \&_{A_\bullet, voc(s), voc(s')}(s,s',s'')\}.$$

By Proposition 5, (5) holds for all $s, s' \in (2^{A_\bullet} - \{\Box\})^*$. We can then sharpen the computation of $T(R,R')$ by (4) to the set of relations $R'' \in \mathcal{AR}$ such that

$$\mathfrak{s}_R(a_1,a_2) \,\&_P\, \mathfrak{s}_{R'}(a_2,a_3) \text{ has a string that projects to } \mathfrak{s}_{R''}(a_1,a_3).$$

Also, to check if a labeling f of A that specifies singleton sets $\{R^{a,a'}\} = f(a,a')$ has a string satisfying (3), we $\&_P$-superpose together each $\mathfrak{s}_{R^{a,a'}}(a,a')$. Apart from transitivity tables and (3), $\&_P$ applies to the constrained generation of strings in or out of $\mathcal{L}_\bullet(a_1 \cdots a_n)$, with projection constraints beyond intervals.

3 Expansions, Inertia and Events

The requirement that $l(a)$ and $r(a)$ mark the left and right borders of a can be expressed with the help of certain $\mathrm{MSO}_{\{a\}}$-formulas over a free variable x. Let $\chi_{l(a)}(x)$ say P_a holds right after x but not at x

$$\chi_{l(a)}(x) \ := \ \neg P_a(x) \wedge (\exists y)(xSy \wedge P_a(y))$$

and $\chi_{r(a)}(x)$ say P_a holds at x but not right after

$$\chi_{r(a)}(x) \ := \ P_a(x) \wedge \neg(\exists y)(xSy \wedge P_a(y)).$$

We can then interpret $P_{l(a)}$ and $P_{r(a)}$ in terms of P_a according to the set

$$\Phi(A_\bullet) \ := \ \{(\forall x)(P_t(x) \equiv \chi_t(x)) \mid t \in A_\bullet\}$$

of $\mathrm{MSO}_{A \cup A_\bullet}$-sentences equating $P_t(x)$ with $\chi_t(x)$. Given a string s of subsets of $A_\bullet$, the strings over the alphabet 2^A that b maps to s can be collected in the set

$$b^{-1}s \ = \ \{\rho_A(s') \mid s' \in s\&(2^A)^* \text{ and } (\forall \varphi \in \Phi(A_\bullet)) \ \mathit{Mod}(s') \models \varphi\}$$

in 3 steps

Step 1: expand with labels from A, superposing s with $(2^A)^*$
Step 2: constrain by $\Phi(A_\bullet)$
Step 3: reduce by ρ_A (for A-reduct).

To compute b from Steps 1-3 above, it suffices to replace A by $A_\bullet$ in steps 1 and 3. The difference between b and b^{-1} comes down to the subalphabet added in Step 1 and preserved in Step 3. $\Phi(A_\bullet)$ is, however, arguably more in sync with b than with b^{-1}, grounding, as it does, $P_{l(a)}$ and $P_{r(a)}$ in P_a. The inverse b^{-1} invites us to consider the reverse:

(Q) how do we interpret P_a, given interpretations of $P_{l(a)}$ and $P_{r(a)}$?

Answering (Q) is an instructive exercise, pointing to forces and events.

Our answer to (Q) comes in two parts, assuming $P_{l(a)}$ and $P_{r(a)}$ are interpreted as subsets $U_{l(a)}$ and $U_{r(a)}$ (respectively) of the set $[n]$ of positions of a string of length n. The first part is an inductive construction

$$U_a = \bigcup_{i \geq 0} U_{a,i} \tag{6}$$

of the interpretation U_a of P_a according to the definitions

$$\begin{aligned} U_{a,0} \ &:= \ U_{r(a)} \\ U_{a,i+1} \ &:= \ U_{a,i} \cup \{k \in [n-1] \mid k+1 \in U_{a,i} \text{ and } k \notin U_{l(a)}\} \end{aligned}$$

suggested by the implications

$$\begin{aligned} (\forall x)(P_{r(a)}(x) \supset P_a(x)) \\ (\forall x)(\forall y)((xSy \wedge P_a(y) \wedge \neg P_{l(a)}(x)) \supset P_a(x)) \end{aligned} \tag{7}$$

from $\Phi(A_\bullet)$. The second part of our answer consists of two conditions

$$U_{l(a)} \cap U_a = \emptyset \tag{8}$$

$$\{i+1 \mid i \in U_{l(a)}\} \subseteq U_a \tag{9}$$

expressed by the implications

$$\begin{aligned} &(\forall x)(P_{l(a)}(x) \supset \neg P_a(x)) \\ &(\forall x)(P_{l(a)}(x) \supset (\exists y)(xSy \wedge P_a(y))) \end{aligned}$$

implicit in $\Phi(A_\bullet)$. (8) and (9) hold precisely if $l(a)$ and $r(a)$ are positioned properly under $U_{l(a)}$ and $U_{r(a)}$ — i.e., there is a string in

$$(\epsilon + \boxed{r(a)})(\boxed{l(a)}\boxed{r(a)})^*$$

to which the string s corresponding to the MSO$_{A_\bullet}$-model $\langle [n], S_n, \{U_{t \in A_\bullet}\}\rangle$ projects

$$d_\square(\rho_{\{l(a),r(a)\}}(s)) \in (\epsilon + \boxed{r(a)})(\boxed{l(a)}\boxed{r(a)})^*. \tag{10}$$

Proposition 6. *For every* $s \in (2^{A_\bullet})^*$,

$$b^{-1}s \neq \emptyset \iff \text{(10) holds for every } a \in A.$$

Moreover, if $Mod(s) = \langle [n], S_n, \{U_t\}_{t \in A_\bullet}\rangle$ *then for every* $s' \in (2^A)^*$ *such that* $b(s') = s$,

$$Mod(s') = \langle [n], S_n, \{U_a\}_{a \in A}\rangle$$

where U_a *is given by (6) above from the sets* $U_{l(a)}$ *and* $U_{r(a)}$ *in* $Mod(s)$*. That is, under* b*,* P_a *is definable from* $P_{l(a)}$ *and* $P_{r(a)}$ *according to*

$$P_a(x) \ \equiv (\exists X)(X(x) \wedge a\text{-}path(X))$$

where a-$path(X)$ *is the conjunction*

$$\forall x(X(x) \supset P_{r(a)}(x) \vee \exists y(xSy \wedge X(y))) \ \wedge \ \neg\exists x(X(x) \wedge P_{l(a)}(x))$$

saying X *is an* S*-path to* $P_{r(a)}$ *that avoids* $P_{l(a)}$.

A crucial ingredient of the analysis of b^{-1} described by Proposition 6 is the implication (7) underlying the inductive step $U_{a,i+1}$. That step lets a spread to the neighboring left box unless $l(a)$ is in that box. This property of $l(a)$ can be isolated in a label $f(a)$ constrained by the implication

$$(\forall x)(\neg P_a(x) \wedge (\exists y)(xSy \wedge P_a(y)) \ \supset \ P_{f(a)}(x)) \tag{11}$$

which (without the converse of (11)) falls short of reducing $P_{f(a)}$ to $P_{l(a)}$. For the sake of symmetry, we also introduce labels $\overline{a}$ and $P_{f(\overline{a})}$ subject to

$$(\forall x)(P_{\overline{a}}(x) \equiv \neg P_a(x))$$

making $\overline{a}$ the negation of a, and the implication

$$(\forall x)(P_a(x) \wedge (\exists y)(xSy \wedge \neg P_a(y)) \ \supset \ P_{f(\overline{a})}(x)). \tag{12}$$

tracking, with (11), any changes in $a/\overline{a}$. The intuition is that $f(a)$ and $f(\overline{a})$ mark the applications of forces for and against a (respectively). The slogan behind (11) and (12) is

no change unless forced

or, in one word, *inertia*. To save that principle from vacuity, let us be careful *not* to identify $f(a)$ with $l(a)$ or $f(\overline{a})$ with $r(a)$. Indeed, insofar as clashing forces are commonplace and merit logical scrutiny (rather than dismissal), there is nothing illegitimate about a box containing both $f(a)$ and $f(\overline{a})$. By contrast, $l(a)$ and $r(a)$ are mutually exclusive under (10). Over any given stretch of time, any number of forces can be at play, some of which may be neutralized by competition. A force in isolation may have very different effects with company. That said, there is no denying that we detect and evaluate forces by the state changes they effect.

Stative predicates labelled by $a \in A$ differ significantly from non-stative predicates labelled by $l(a), r(a), f(a)$ and $f(\overline{a})$ in how strings built from them compress to canonical forms. Recall from the Introduction the link between homogeneity and block compression bc, deleting stutters

$$bc^{-1}bc(s) = \alpha_1^+ \cdots \alpha_n^+ \quad \text{if } bc(s) = \alpha_1 \cdots \alpha_n$$

just as $d_\square$ deletes $\square$

$$d_\square^{-1} d_\square(s) = \square^* \alpha_1 \square^* \cdots \square^* \alpha_n \square^* \quad \text{if } d_\square(s) = \alpha_1 \cdots \alpha_n.$$

Proposition 7. *For every $s \in (2^A)^*\square$, $b(bc(s)) = d_\square(b(s))\square$ and $bc(s)$ is the unique string over 2^A in the set $b^{-1}(d_\square(b(s))\square)$.*[3]

Underlying both notions of compression, $d_\square$ and bc, is the slogan

[3] $\square$ is put after $(2^A)^*$ and after $d_\square(b(s))$ to reconcile a difference between a's left and right borders, $l(a)$ and $r(a)$; whereas $r(a)$ is in a, $l(a)$ is outside a. This gives rise to a wrinkle in Proposition 2, line (ii), $\rho_{\{a\}}(s) \in \square^+ \boxed{a}^+ \square^*$. The Kleene star $\square^*$ becomes a plus in Proposition 7, with $s \in (2^A)^*\square$ necessitating a $\square$ after $d_\square(b(s))$.

no time without change.

But while bc represents that change in terms of decomposable intervals/statives, $d_\Box$ employs non-decomposable points/borders (not to mention forces). The function $d_\Box$ is simpler than bc, and provides a pointed twist on the interval-based approach in [3] to Dowty's aspect hypothesis.

An obvious question is how to compress a string s of sets consisting of labels for stative and non-stative predicates alike (as in Step 2 above). Let us collect labels of non-homogeneous predicates in a set C. A simple synthesis of $d_\Box$ and bc is the function $\mathit{bc}^C(s)$ defined by induction on the length of s as follows. Let $\mathit{bc}^C(\epsilon) := \epsilon$ and

$$\mathit{bc}^C(\alpha s) \; := \; \begin{cases} \mathit{bc}^C(s) & \text{if } \alpha = \Box \text{ or } (\alpha \cap C = \emptyset \text{ and } s \text{ begins with } \alpha) \\ \alpha\, \mathit{bc}^C(s) & \text{otherwise.} \end{cases}$$

It is easy to see that

$$\mathit{bc}^C(s) \; = \; d_\Box(s) \;\; \text{if } s \in (2^C)^*$$

while for any label $\smiley \not\in C \cup \mathit{voc}(s)$,

$$\mathit{bc}(s) \; = \; d_{\smiley}(\mathit{bc}^C(i_{\smiley}(s))) \;\; \text{if } \mathit{voc}(s) \cap C = \emptyset$$

where $i_{\smiley}$ inserts $\smiley$

$$i_{\smiley}(\alpha_1 \cdots \alpha_n) \; := \; (\alpha_1 \cup \{\smiley\}) \cdots (\alpha_n \cup \{\smiley\})$$

and $d_{\smiley}$ deletes $\smiley$

$$d_{\smiley}(\alpha_1 \cdots \alpha_n) \; := \; (\alpha_1 - \{\smiley\}) \cdots (\alpha_n - \{\smiley\}).$$

Any void $\Box$ is filled with ambient noise $\smiley$, which we may otherwise ignore.

4 Conclusion

We can summarize the work above as follows, recalling the passages from [1] and [2] quoted in the Introduction. We "turn up the magnification" by inverting A-reducts ρ_A, and analyze homogeneous statives through block compression bc, reconstructed according to Proposition 7 through a border translation b and $\Box$-removal $d_\Box$. Working with strings s, we form A-canonical representations by compressing $\rho_A(s)$ according to

$$s\Box s' \; \approx \; ss' \quad \text{(leading to } d_\Box)$$

and

$$s\alpha\alpha s' \; \approx \; s\alpha s' \;\; \text{if } \alpha \cap C = \emptyset \quad \text{(leading to } \mathit{bc})$$

where C collects labels of non-homogeneous predicates (including forces). Among the labels in C are s-points that (as defined in section 2) describe *particulars*, in contrast to labels that occur in more than one position in s (describing *universals*). Handling granularity through A and A-reducts is a hallmark of *institutions* ([6]), where models and sentences are organized around *signatures* for variable granularity. To view MSO as an institution, we pair a set A up with a subset B of A for a signature (A, B); a model of signature (A, B) is then a string $s \in (2^A)^*$ such that each element of B is an s-point, which serves as a first-order variable to express MSO predication in a sentence of signature (A, B). That said, the set C of non-homogeneous predicates is not restricted to s-points. Indeed, line (10) in section 3 allows $r(a)$ and $l(a)$ to occur more than once in a string s for which $b^{-1}s$ is non-empty. The advance over [4, 5] that the present work may claim has less to do with the particular notion, s-point, than with the deletion $d_\Box$ of $\Box$, simplifying block compression bc. Compression $d_\Box$ and bc carve out two sides of a coin, the border translation b that yields s-points and more.

References

1. J. Allen. Maintaining knowledge about temporal intervals. *Communications of the ACM*, 26(11):832–843, 1983.
2. D.R. Dowty. *Word Meaning and Montague Grammar.* Reidel, Dordrecht, 1979.
3. T. Fernando. Dowty's aspect hypothesis segmented. In *Proceedings of the 19th Amsterdam Colloquium*, pages 107–114. 2013.
4. T. Fernando. The semantics of tense and aspect: A finite-state perspective. In S. Lappin and C. Fox, editors, *Handbook of Contemporary Semantic Theory*, pages 203–236. Wiley-Blackwell, second edition, 2015.
5. T. Fernando. On regular languages over power sets. *Journal of Language Modelling*, 4(1):29–56, 2016.
6. J.A. Goguen and R.M. Burstall. Institutions: abstract model theory for specification and programming. *Journal of the ACM*, 39(1):95–146, 1992.
7. C.L. Hamblin. Instants and intervals. *Studium Generale*, 24:127–134, 1971.
8. L. Libkin. *Elements of Finite Model Theory.* Springer, 2010.
9. M. Moens and M. Steedman. Temporal ontology and temporal reference. *Computational Linguistics*, 14(2):15– 28, 1988.

Towards a Computationally Viable Framework for Semantic Representation*

Shalom Lappin

University of Gothenburg
shalom.lappin@gu.se

Abstract. Classical theories of formal semantics employ possible worlds to model intensions and modality. If worlds are construed as corresponding to ultrafilters in a lattice of propositions (maximal consistent sets of propositions), then they pose serious problems of tractable representability. Moreover, these theories cannot accommodate vagueness, which is a pervasive feature of predication in natural language. It is also unclear how they can be extended in a straightforward way to explain semantic learning. A cognitively plausible account of interpretation should generate computationally tractable representations of meaning. It must also deal with vagueness and semantic learning. A probabilistic Bayesian approach to natural language semantics provides a more promising approach to these issues. It can also cover epistemic states and inference, in a tractable way. This framework offers the promise of a robust, wide coverage treatment of natural language interpretation that integrates meaning and information update.

Keywords: possible worlds · tractable representations · Bayesian semantics · intensions · modality · epistemic states · information update

1 Introduction

Since [35] a mainstream view among formal semanticists has depended on possible worlds to model the meanings of natural language expressions. Montague imported possible worlds into his model theory through his use of Kripke frame

* The research reported in this paper was supported by grant 2014-39 from the Swedish Research Council, which funds the Centre for Linguistic Theory and Studies in Probability (CLASP) in the Department of Philosophy, Linguistics, and Theory of Science at the University of Gothenburg. Earlier versions were presented in the Syntax Reading Group of University College London and the Colloquium of the Centre for Logic, Language, and Mind at the University of Stockholm, both in October 2017. I am grateful to the participants in these forums for useful discussion and feedback. I would also like to thank Jean-Philippe Bernardy, Rasmus Blanck, Stergios Chartzikyriakidis, Robin Cooper, Simon Dobnik, Staffan Larsson, Per Martin-Löf, Peter Pagin, and Daniel Rothschild for helpful comments and suggestions on some of the ideas presented here. Of course I bear sole responsibility for any errors that may remain.

semantics ([27, 28]) for modal logic. This approach to intensions and modality is anticipated in [5]'s characterisation of intensions as functions from state descriptions to extensions.

Possible worlds have played a central role in the characterisation of belief ([45]) and the related field of epistemic reasoning (see, for example [19]). Dynamic semantics ([17, 18]), and, more recently, Inquisitive Semantics ([6, 7]) use possible worlds to incorporate epistemic elements into formal semantics. They characterise sentence meanings as functions from discourse contexts to discourse contexts. From this perspective speakers use sentences to communicate information by modifying their hearers' representation of a discourse context.[1]

There are, in fact, serious computational problems of representability for worlds. Moreover, specifying intensions as functions from worlds to extensions does not respect important fine-grained distinctions of meaning. I discuss these issues in detail in [30, 31]. In Section 2 I summarise the representability problems raised there, and argue that these must be solved in order to develop a cognitively viable semantics for natural language.

It is possible to "de-modalise" intensions by characterising them as computable functions. This involves borrowing the difference between operational and denotational meaning from the semantics of programming languages and applying it to the meanings of natural language expressions ([30, 31, 12]). In Section 3 I review this approach to representing intensions.

An operational treatment of intensions might solve the representability problem for (some) natural language meanings, and provide the basis for a fine-grained semantics. However, it leaves the treatment of modality and epistemic states untouched. At first glance it would seem that there is no alternative but to invoke worlds to model possibilities, knowledge, and beliefs. But if we are forced to reintroduce worlds to handle these concepts, then we have not solved the representability problem, and so we have not grounded semantics on cognitively viable foundations.

In Section 4 I suggest an approach to this question that avoids worlds. It involves characterising both modality and epistemic states through probability distributions over situations, rather than complete worlds. On his account probabilities are assigned to possible, as well as to actual situations. However, it is not necessary to represent or enumerate the complete class of possible situations, which, as we argue in Section 2, is even more problematic than representing a complete world, or the set of worlds. It is sufficient to specify those situations to which probabilities are assigned, and the situations expressed by the conditions on which a probability assignment depends. Large subclasses of probability models can be efficiently represented, and tractability problems with computing probability distributions or complex sampling spaces can frequently be overcome by estimation and approximation. The probabiistic accounts of modality

[1] It may be possible, at least in principle, to develop versions of Inquisitive Semantics and Dynamic Semantics which do not rely on possible worlds. However, they are an integral element of the current theories.

and epistemic states proposed in this Section constitute the main contribution of the paper.

I offer an overview of some current related work in Section 5, and I briefly compare these approaches to the framework proposed here.

Finally, in Section 6 I present some conclusions, and I briefly indicate several problems to be addressed in future work on these questions.

2 A Repesentability Problem with Worlds

In Kripke frame semantics a model $M = \langle D, W, F, R \rangle$, where D is a non-empty set of individuals, W is a non-empty set of worlds, F is an interpretation function that assigns intensions to the constants of a language, and R is an accessibility relation on W. Formal semanticists have expanded M to include additional indices representing elements of context, such as sets of points in time, and sets of speakers. The elements of W are points at which a maximal consistent set of propositions are satisfied.[2]

There is a one to one correspondence between the elements of W and the elements of the set of maximal consistent sets of propositions. [13, 14, 42] use this correspondence to formally represent worlds as the set U of ultrafilters in the prelattice of propositions. On this approach a proposition p holds at a world w_i iff $p \in u_i$, where $u_i \in U$. The question of how to represent W reduces to the representability of U

To simplify the problem considerably, I assume that the the prelattice on which the elements of U are defined encodes classical Boolean propositional logic. This system is complete and decidable, and so minimal in expressive power. To identify any $u_i \in U$ we need to specify all and only the propositions that hold at u_i. As u_i is an ultrafilter, for any $p_i \in u_i$, all of the propositions that p_i entails are also in u_i, and so it will be an infinite set. We can enumerate the elements of an infinite set if there is an effective procedure (a finite set of rules, an algorithm, a recursive definition, etc.) for recognising its members. It is not clear what an effective procedure for enumerating the propositions of u_i would consist in.

Simplifying further, let's assume that we are able to generate u_i from a finite set P_{u_i} of propositions, where each $p \in P_{u_i}$ is in *Conjunctive Normal Form* (CNF). A proposition in CNF is a conjunction of disjunctions of literals (elementary propositional variables or their negations). The propositions in P_{u_i} can be conjoined in a single formula p_{u_i} that is itself in CNF. For p_{u_i} to hold it is necessary to determine a distribution of truth-values for its literals that renders the entire formula true. Determining the complexity of this satisfaction problem is an instance of the kSAT problem, where k is the number of literals in p_{u_i}. If $3 \leq k$, then the satisfiability problem for p_{u_i} is, in the general case, NP-complete, and so intractable.[3]

[2] In fact [5, 23, 27] originally characterised worlds as maximal consistent sets of propositions.

[3] See [39] for a discussion of the complexity properties of kSAT classes.

Given that this formula is intended to express the finite core of propositions from which the entire ultrafilter u_i is derived, it is not plausible to limit it to two literals, and it is reasonable to allow it to contain a large number of distinct elementary propositional constituents, each corresponding to a "core" fact that holds in u_i. It will also be necessary to include law like statements expressing regular relations among events that hold in a world (such as the laws of physics). These will be expressed as conditionals $A \rightarrow B$, which are encoded in a CNF formula by disjunctions of the form $\neg A \vee B$.

Therefore, even given the generous simplifying assumptions that we have made concerning the enumeration of u_i, specifying the ultrafilter of propositions that corresponds to an individual world is, in general, a computationally intractable problem. It follows that it is not possible to compute the set of worlds W efficiently.[4]

There are (at least) three ways in which one might try to evade this problem. First, we could follow Montague in claiming that formal semantics is a branch of mathematics rather than psychology. It involves the application of model theory, or, on the perspective adopted here, algebraic, and specifically, lattice theoretic methods, to develop formal models of meaning in natural languages. If this is the case, questions of efficient computability and representability are not relevant to the theoretical constructions that it employs. This move raises the obvious question of what formal semantics is explaining. If it seeks to account for the way in which people interpret the expressions of a natural language, then one cannot simply discard issues of representation. To do so is to ignore the cognitive aspect of meaning, which risks eliminating the empirical basis for assessing semantic theories.

A weaker form of this approach acknowledges that using and interpreting natural language is indeed a cognitive process, but it invokes the competence-performance distinction to insulate formal semantic theory from computational and processing concerns. On this view formal semantics offers a theory of semantic competence, which underlies speakers' linguistic performance.

[40] seems to suggest a move of this kind in distinguishing between semantic and psychological facts. But this is simply a version of the competence-performance distinction applied to semantics. Interestingly, this distinction is not generally regarded as granting immunity from the requirement of tractable representation in other areas of linguistic representation. So, for example, if a class of grammars (more precisely, the languages that they generate) is shown to be intractable for the recognition/parsing task, it is generally regarded as unsuitable for encoding the syntax of a natural language. Consequently, the full

[4] One might seek to treat propositions as unstructured, and worlds as ontologically primitive. It is unclear how either move could alleviate the representability problem. Literals are unstructured in the sense that they are elementary propositional variable or their negations. To banish the additional logical structure necessary to construct propositions in CNF would deprive propositions of any content at all. Taking worlds as primitive begs the question of how we identify and distinguish them. The conclusion that there is a one-to-one correspondence between a world and the ultrafilter of propositions that hold in it seems inescapable.

class of Context Sensitive Grammars, which, in some cases, require exponential time to decide membership in a context sensitive language, is regarded as too powerful to model NL syntax. Instead, the weaker subclass of Mildly Context Sensitive Grammars, for which the recognition problem is polynomial, is preferred. Consistency requires that tractability of representation also apply to semantic theories, even when these are taken to be abstract models of linguistic competence.

The difficulty here is that unless one provides an explicit account of the way in which competence drives processing and behaviour, then the distinction becomes vacuous. The notion of competence remains devoid of explanatory content.[5] We cannot simply set aside questions of effective computability if we are interested in semantic theories that are grounded on sound cognitive foundations.

A second strategy for dealing with the representability problem for possible worlds is to invoke the method of stratification. This technique stratifies a class of intractable problems into subclasses in order to identify the largest subsets of tractable tasks within the larger set.[6] So, for example, work on the tractable subclasses of kSAT problems is an active area of research. Similarly, first-order logic is undecidable (the set of its theorems is recursively enumerable, but the set of its non-theorems is not). However, many efficient theorem provers have been developed for subsets of first-order logic that are tractably decidable. We could focus on identifying the largest subsets of each $u_i \in U$ that can be tractably specified.

The problem with using stratification here is that, by definition, a world is (corresponds to) a maximal set of consistent propositions, an ultrafilter in a prelattice. If we specify only a proper subset of such an ultrafilter (a non-maximal filter), then it is not a world in the intended sense. It is no longer identified by all and only the propositions that hold at that world. In fact, in principle, several distinct worlds could share the same set of efficiently representable subsets of propositions, in which case they would not be efficiently distinguishable.[7]

Note that one cannot avoid this problem by claiming that, in principle, a "clever" algorithm could be devised to identify the ultrafilter of propositions that corresponds to a world. Unless one specifies such a procedure and shows that it efficiently identifies the set of worlds needed for a semantic theory, asserting

---

[5] See [33] for a detailed critical discussion of the difficulties raised by using the competence-performance distinction to protect syntactic theories from responsibility for handling a wide range of observed phenomena concerning speakers' syntactic judgments.

[6] See [8] on stratification of classes of grammars as a way of dealing with complexity in the context of computational learning theory for natural language grammars.

[7] [44] seems to have partial worlds in mind when he characterises worlds as elements in a partition of logical space, where such partitions are dependent on context. The problem with Stalnaker's suggestion is that he does not provide procedures for identifying partitions in logical space or their elements. In the absence of these it is not clear how such worlds/possibilities are to be represented or enumerated. Therefore, it does not offer a solution to the representability problem.

the mere possibility that one might be devised adds nothing of substance to the discussion.

Finally, a third approach to the problem of representability is to substitute possible situations for possible worlds. As situations are partial worlds, one may think that they are easier to represent. This is indeed be the case for individual situations, which are non-maximal, and for certain sets of situations.[8] However, it is not the case for the complete set of possible situations.[9]

For any given u_i corresponding to a world w_i, a situation $s_i \subseteq u_i$. The set of situations S_i for u_i is $\mathcal{P}(u_i)$, the power set of u_i. As $|u_i| = \aleph_0$, by Cantor's theorem on the cardinality of power sets, $|S_i|$ is uncountably infinite. Therefore S_i is not recursively enumerable. The set of all possible situations $S = \bigcup S_i$, and S inherits non-recursive enumerability from its constituent S_is. The representability problem for the set of possible situations is, then, even more severe than the one that we encounter for the set of possible worlds.

It may be possible to avoid this difficulty if we do not invoke the entire set of possible situations, but limit ourselves to subsets that we can specify effectively as we require them for particular analyses. This is, in effect, a form of stratification. But as situations are not maximal in the way that worlds are, it might be a viable method when applied to situations. In order for this method to work, it is necessary to show that we do, in fact, have effective procedures for representing the situations that we need for our theories. I will explore this approach in greater detail in Section 4.

3 Operational and Denotational Meaning

In the formal characterisation of programming languages it is common to distinguish between the operational and the denotational semantics of a program.[10] Operational meaning corresponds (roughly) to the sequence of state transitions that occur when a program is executed. It can be identified with the computational process through which the program produces an output for a specified input. The denotational meaning of a program is the mathematical object that represents the output which it generates for a given input. The operational and denotational meanings of the constituents of a program can be understood compositionally in terms of the contributions that they make to determining the state transitions performed by the program, and the value that it yields, respectively.

We can illustrate this distinction with two simple examples. First, it is possible to construct a theorem prover for first-order logic using either semantic tableaux or resolution.[11] Both theorem provers use proof by contradiction, but

[8] See [2] for the basic ideas of situation semantics.

[9] [21, 29, 26], for example, use the set of possible situations instead of the set of possible worlds to develop intensional semantic analyses.

[10] See, for example, [46] on these two types of meaning for expressions in programming languages.

[11] See [4] for tableaux and resolution theorem provers implemented in Prolog, and applied as part of a computational semantic system for natural language.

they employ alternative formal methods, and they are implemented as different computational procedures. They exhibit distinct efficiency and complexity properties. Consider the two predicates $TheoremTableaux$, which is true of the elements of the set of classical first-order theorems that a tableaux theorem prover produces, and $TheoremResolution$ that is true of the members of the set of classical first-order theorems that a resolution prover identifies. The predicates are intensionally distinct, but they are provably equivalent in their extensions.

The second example involves two functions from fundamental frequencies to the letters indicating musical notes and half tones. The first takes as its arguments the pitch frequency waves of the electronic sensor in a chromatic tuner. The second has as its domain the pitch frequency graphs of a spectrogram. Assume that both functions can recognise notes and half tones in the same range of octaves, to the same level of accuracy. Again, their operational semantics are distinct, but they are denotationally equivalent. The pairs of corresponding classifier predicates for these functions, $\langle A_{ChromTuner}, A_{SpecGram}\rangle$, $\langle A\#_{ChromTuner}, A\#_{SpecGram}\rangle$, $\ldots$, $\langle G_{ChromTuner}, G_{SpecGram}\rangle$, are intensionally distinct but denotationally equivalent. Both classifiers in a pair select the same set of notes, each through a different method.

We can apply this distinction to natural languages by taking the operational meaning of an expression to be the computational process through which speakers compute its extension, and its denotational meaning to be the extension that it generates for a given argument. We identify the intension of an expression with its operational meaning. This view of intension avoids the intractability of representation problem that arises with possible worlds.

It is also allows us to solve the difficulty of fine-grained intensionality (sometimes referred to as hyperintensionality). This issue arises because logically equivalent expressions are not, in general, inter-substituable in all contexts in a way that preserves the truth-value of the matrix sentence in which the expressions are exchanged. But if logically equivalent expressions have the same denotations in all possible worlds and intensions are functions from worlds to denotations, then these expressions are identical in intension. The following example illustrates the problem.

(1) a. If $A \subseteq B$ and $B \subseteq A$, then $A = B$. $\Leftrightarrow$
b. A prime number is divisible only by itself and 1.

(2) a. Mary believes that if $A \subseteq B$ and $B \subseteq A$, then $A = B$. $\not\Leftrightarrow$
b. Mary believes that a prime number is divisible only by itself and 1.

(1)a and b are both mathematical truths, but they are not inter-substitutable in the complement of *Mary believes that* in (2). However, if we identify intensions with operational meaning, then (1)a and b are intensionally distinct. (1)a is a theorem of set theory, while (1)b is a theorem of number theory. Their proofs are entirely different, and so they encode distinct objects of belief. The operational notion of intension permits us to individuate objects of propositional attitude with the necessary degree of fine-grained meaning.[12]

This solution to the issue of hyperintensionality is a secondary consequence of the operational account of intensions. Its primary motivation is to avoid the representability problem posed by possible worlds. [38] and [36] suggest related solutions, which retain possible worlds. See [30] for discussion of these proposals.

We have eliminated the dependence of intensions on possible worlds, and with it the representability problem for meanings, to the extent that the interpretation of an expression can be expressed as a procedure for computing its denotation. However, this only takes us part of the way to solving the cognitive plausibility problem for natural language semantics. We still need to develop an approach to modality and epistemic states which does not require possible worlds.

4 Modality and Epistemic States

Consider the following modal statements.

(3) a. Necessarily if $A \subseteq B$ and $B \subseteq A$, then $A = B$.
b. Possibly interest rates will rise in the next quarter.
c. It is likely that the Social Democrats will win the next election in Sweden.

In possible worlds semantics modal operators are construed as generalised quantifiers (GQs) on worlds. Necessity is a universal quantifier, possibility an existential quantifier, while *likely* is a variant of the second-order GQ *most*.[13] Let α, β, γ be the propositions to which the modal adverbs *necessarily, possibly* and *likely* apply in (3)a-c, respectively. The truth conditions of the sentences in (3) would be given by (something like) the following.

(4) a. $\|\Box\alpha\|^{M,w_i} = t$ iff $\forall w_{\in W} \|\alpha\|^{M,w} = t$.
b. $\|\Diamond\beta\|^{M,w_i} = t$ iff $\exists w_{\in W} \|\beta\|^{M,w} = t$.
c. $\|\mathit{Likely}\ \gamma\|^{M,w_i} = t$ iff for an appropriately defined $W' \subseteq W$, $|\{w_j \in W' : \|\gamma\|^{M,w_j} = t\}| \geq \epsilon$, where ϵ is a parametric cardinality value that is greater than 50% of W'.

[12] [11], Chapter 6 proposes an account of modality and propositional attitudes which dispenses with possible worlds, within the framework of Type Theory with Records, an intensional theory of types as judgements classifying situations. Some of Cooper's suggestions run parallel to the account proposed here. However, it is not clear how TTR solves the problem of complexity in representing the full set of record types. Moreover, it is not obvious that type membership in TTR is decidable.

[13] [25] presents a treatment of modalised degree modifiers that posits an ordering of possible worlds for similarity to a normative world. [22] discuss problems with this account and offer an alternative, which uses epistemically possible worlds. Given their reliance on a classical notion of possible world, neither theory avoids the representability problem.

On an alternative approach, we can reformulate modal statements as types of probability judgments. As a prelude it will be useful to review some basic ideas of probability theory.[14] A probability model M consists of a sample space of events with all possible outcomes given, and a probability distribution over these outcomes, specified by a function p. So, for example, a model of the throws of a die assigns probabilities to each of its six sides landing up. If the die is not biased towards one or more sides, the probability function will assign equal probability to each of these outcomes, with the values of the sides summing to 1.

Probability theorists often refer to the set of possible outcomes in a sample space as possible worlds. In fact this is misleading. Unlike worlds in Kripke frame semantics, outcomes are non-maximal. They are more naturally described as situations, which can be as large or as small as required by the sample space of a model. Therefore, in specifying a sample space it is not necessary to distribute probability over the set of all possible situations. In fact one need not even represent all possible situations of a particular type. One estimates the likelihood of an event of a particular type on the basis of observed occurrences of events, either of this type, or of others that might condition it. If we are working with Bayesian models, then we compute the posterior probability of an event A (the hypothesis) given observed events B (the evidence) with Bayes' Rule, where $p(B) \neq 0$.

(5) $p(A|B) = \dfrac{p(B|A)p(A)}{p(B)}$

Computing the full set of such joint probability assignments is, in the general case, intractable. However, there are efficient ways of estimating or approximating them within a Bayesian network.[15] It is, then, possible to efficiently represent a large subset of probability models, and to compute probability distributions for the possible events in their sample spaces.

Returning to the modal statements in 3, we can construct the following alternatives to 4, where M is a probability model, and p is the probability function in M.

(6) a. $\|Necessarily\ \alpha\|^{M,p} = t$ iff for all models $M' \in R, p_{\in M'}(\alpha) = 1$, where R is a suitably restricted subset of probability models.
b. $\|Possibly\ \beta\|^{M,p} = t$ iff $p(\beta) > 0$.
c. $\|Likely\ \gamma\|^{M,p} = t$ iff $p(\gamma) > \epsilon$, where ϵ is a parametric probability value that is greater than 0.5.

(6)a expresses universal necessity. Notice that to demonstrate this necessity it is sufficient to prove that assuming a probability model $M' \in R$ in which $p(\alpha) \neq 1$ produces a contradiction. If we are limiting ourselves to an appropriate

[14] See [20] for a particularly clear introduction to probability theory, that is relevant to some of the issues discussed here.
[15] See [41, 37, 20, 24] on Bayesian networks.

class of probability assertions and models, an efficient theorem prover may be available for such a result.[16] (6)b identifies possibility in a model with non-nil probability of occurrence. (6)c characterises likelihood in a model with a high degree of probability. These probabilistic characterisations of the modal adverbs *necessarily, possibly* and *likely* do seem to identify core aspects of their meanings in many of their common uses.[17]

In general we may use stratification to identify classes of probability models that can be efficiently represented, and we might invoke approximation techniques to estimate at least some of the others which are not. This is in contrast to individual worlds and sets of worlds. The maximality of worlds and the absence of any apparent procedure for generating their representations seem to exclude the application of these methods to possible worlds of the kind that figure in the formal semantics of natural language.

Let's consider how we might extend the probability-based approach proposed here for modality to epistemic states. Within a possible worlds framework knowledge and belief have traditionally been characterised along the following lines. Let W_B be the set of worlds (understood as ultrafilters of propositions) compatible with an agent a's beliefs. Take F_B to be a possibly non-maximal filter such that $F_B \subseteq \bigcap W_B$, where for every proposition $\phi \in F_B$, a regards ϕ as true. Let w_{actual} be the actual world. a's knowledge is contained in $F_K \subseteq F_B \cap w_{actual}$.[18]

As an alternative to this account we can use a probability model to encode an agent's beliefs. The probability distribution that this model contains expresses the agent's epistemic commitments concerning the likelihood of situations and events. One way of articulating the structure of causal dependencies implicit in these beliefs is to use a Bayesian network as a model of belief.[19]

---

[16] I am grateful to Robin Cooper for correcting a mistake in an earlier version of (6)a. One might be tempted to think that (6)a expresses a metaphysical concept of necessity, while (6)b,c correspond to epistemic modalities. In fact this is not the case. (6)a characterises necessity as a generalised quantifier over a suitably restricted set of probability models, each of which specifies a probability distribution over a number of events. These distributions constitute an agents' perception of the likelihood of certain events in the world. Therefore (6)a is not less of an epistemic specification of modality than (6)b,c.

[17] In order for this approach to modality to succeed, it will be necessary to develop accounts of the full class of modal expressions, including auxiliary verbs, other modal adverbs, and a variety of modal modifiers within the framework presented here. This is an important task for future work, but it is well beyond the scope of this paper. My objective here is programatic. I wish to show the viability of a probabilistic view of modality as an alternative to the traditional possible worlds treatment. Therefore, I have limited myself to the modal expressions that have been highlighted in the classical theories.

[18] [19] presents a version of this view.

[19] [34] considers the connection between conditional statements of the form $A \rightarrow B$ and the conditional probability $p(B|A)$. While this is an important issue, it is tangential to my concerns here. I am seeking a way of characterising epistemic states that does not invoke possible worlds.

Formally a Bayesian network is a Directed Acyclic Graph (DAG) whose nodes are random variables, each of whose values is the probability of one of the set of possible states that the variable denotes. Its directed edges express dependency relations among the variables. When the values of all the variables are specified, the graph describes a complete joint probability distribution (JPD) for its random variables.

The Bayesian network given in Fig 1, from [43], contains only boolean random variables, whose values are T (true) and F (false). In general, a discrete random variable X may have values $X_1, \ldots, X_n$ for any $n > 1$. Random variables may also be continuous.

The values of the instances of a variable depend directly only on the value of the variable of its parent. The dependency of a variable V on a non-parent ancestor variable A is mediated through a sequence of dependencies on the variables in the the path from V to A.

The only observable event for the network in Fig 1 is if the weather is cloudy or not, and the variable whose probability value we seek to determine is the likelihood of the grass being wet. We do not know the values of the random variables corresponding to rain, and to the sprinkler being on. Both of these events depend on whether the weather is cloudy, and both will influence the probability of the grass being wet. Sample conditional probabilities are given for each variable at each node of the network. The probability of the event C (cloudy) corresponding to the variable at the root of the graph is not conditioned, and its T and F instances are given equal likelihood.

We can compute the marginal probability of the grass being wet (W = T) by marginalising out the probabilities of the other variables on which W conditionally depends, either directly, or through intermediate variables. As we have seen, this involves summing across all the joint probabilities of their instances.

(7) $p(W = T) = \sum_{s,r,c} p(W = T, S = s, R = r, C = c)$

As we have a complete JPD for the variables of this network, it is straightforward to compute $p(W = T)$ using the chain rule for joint probabilities, together with the independence assumptions encoded in the network, which gives us (8).

(8) $p(W = T) = \sum_{s,r,c} p(W = T|S = s, R = r)p(S = s|C = c)p(R = r|C = c)p(C = c)$

In principle we could model an agent's beliefs as a single integrated Bayesian network. This would be inefficient, as it would be problematic to determine the dependencies among all of the random variables representing event types that the agent has beliefs about, in a way that sustains consistency. Moreover, the complexity involved in determining the conditional probabilities for the instances of each variable in such a global network would be daunting. It is more computationally manageable, and more epistemically plausible to construct local Bayesian networks to encode an agent's a's beliefs about a particular domain of situations. A complete collection of beliefs for a will consist of a set of such local

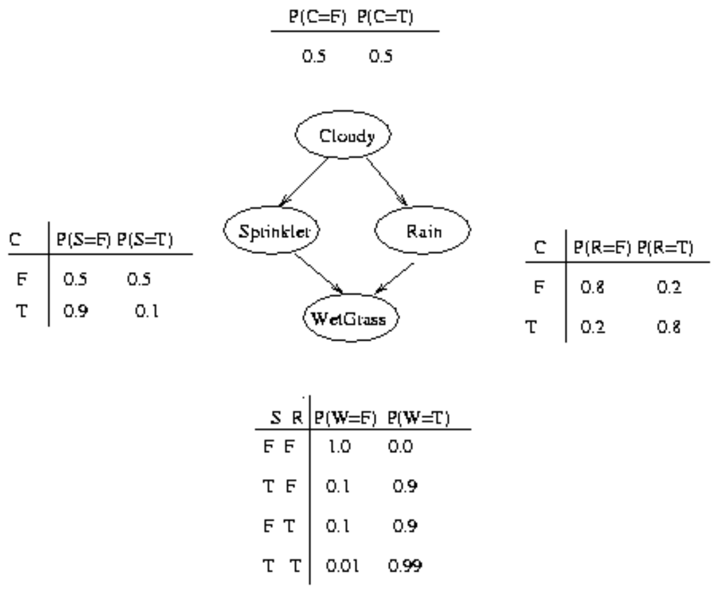


Fig. 1. Example of a Bayesian Network ([43])

networks, where each element of this set expresses a's beliefs about a specified class of events.

Two graphs G_i and G_j are *isomorphic* iff they contain the same number of vertices, and there is a bijection from the vertices of G_i to the vertices of G_j and vice versa, such that the same number of edges connect each vertex v_i to G_i and v_j to G_j, through identical corresponding paths.[20] For isomorphic DAGs this condition entails that the edges going into v_i and coming from it are of the same directionality as the edges going into and coming out of v_j, and vice versa.

Let's say that two subgraphs of two Bayesian networks *match* iff they are isomorphic, and the random variables at their corresponding vertices range over the same event instances, with the same probability values. Let BN_B be the

[20] [1] presents an algorithm for solving the graph isomorphism problem in quasi-polynomial time. An error was discovered in Babai's proof for this result. He subsequently repaired the proof in 2017, and posted the fix on his personal website at `http://people.cs.uchicago.edu/~laci/`.

Bayesian network that expresses a's beliefs about a given event domain. Take BN_R to be the Bayesian network that codifies the actual probabilities and causal dependencies that hold for these events.

We can identify a's knowledge for this domain as the maximal subgraph BN_K of BN_B that matches a subgraph in BN_R, and which satisfies additional conditions C. These conditions will enforce constraints like the requirement that the beliefs encoded in BN_B are warranted by appropriate evidence. Notice that on this characterisation of knowledge, if a knows ϕ, then a believes ϕ, but of course the converse does not hold. C can be formulated to permit justified true belief to count as knowledge, or it can be strengthened to block this implication.[21]

By characterising knowledge and belief in terms of Bayesian networks we avoid the representability problem that traditional analyses inherit from possible worlds. The proposed account offers two additional advantages. First, it exhibits the acquisition of beliefs as a dynamic process driven by continual updates in an epistemic agent's observations. This flow of new information generates revised probability distributions over the instances of the random variables in a network. Belief revision has to be handled by a task specific update function in a classical worlds based model of belief. It is intrinsic to Bayesian networks.

Second, a Bayesian network generates causal inferences directly, through the dependencies that it encodes in its paths. In a traditional worlds model of epistemic states, inference depends on an epistemic logic, whose rules are added to the model. By contrast, in a Bayesian network BN inference follows from the probability theory that BN instantiates. The network is both a dynamic model of belief, and a system that supports epistemic inference.

5 Related Work

[47] propose a theory in which probability is distributed over the set of possible worlds. The probability of a sentence is the sum of the probability values of the worlds in which it is true. If these worlds are construed as maximal in the sense discussed here, then this proposal runs into the representability problem for worlds.

[9, 10] develop a compositional semantics within a probabilistic type theory (ProbTTR). On their approach the probability of a sentence is a judgment on the likelihood that a given situation is of a particular type, specified in terms of ProbTTR. They also sketch a Bayesian treatment of semantic learning. It is not entirely clear how probabilities for sentences are computed in their system. They do not offer an explicit treatment of vagueness or probabilistic inference. It is also not obvious that their type theory is relevant to a viable compositional probabilistic semantics.

[16, 32] propose a probabilistic view of natural language semantics and pragmatics. They take probability to be distributed over partial worlds. They do not make entirely clear the relationship between partial and complete worlds.

21 The claim that knowledge is justified true belief has been controversial at least since [15].

They also do not address the complexity issues involved in specifying worlds, partial or complete, as well as probability models. They implement probabilistic treatments of a scalar adjective, *tall*, and the sorities paradox for nouns like *heap* in the functional probabilistic programming language Church. Their analyses require a considerable amount of lexically specified content, and detailed information concerning speakers' and hearers' contextual knowledge. While their analyses offer thoughtful and promising suggestions on how to treat meaning in probabilistic terms, It is not obvious how their approach can be generalised to a robustly wide coverage model of combinatorial semantics and interpretation for natural language.

In addition, the Goodman-Lassiter account models vagueness by positing the existence of a univocal speaker's meaning that hearers estimate through distributing probability among alternative possible readings. They posit a boundary cut off point parameter for graded modifiers, where the value of this parameter is determined in context.

The approach that I am suggesting here is not forced to assume such an inaccessible boundary point for predicates. It allows us to interpret the probability value of a sentence as the likelihood that a competent speaker would endorse an assertion, given certain conditions (hypotheses). Therefore, predication remains intrinsically vague. It consists in applying a classifier to new instances on the basis of supervised training. We are not obliged to posit a contextually dependent cut off boundary for graded predicates.

[3] propose a compositional Bayesian semantics of natural language that implements this approach in a functional probabilistic programming language. It generates probability models that satisfy a set of specified constraints, and it uses Markov Chain Monte Carlo sampling to estimate the likelihood of a sentence being true in these models. It also sketches an account of semantic learning.

6 Conclusions and Future Work

I have argued that the tradition of formal semantics which uses possible worlds to model intensions, modality, and epistemic states is not built on cognitively viable foundations. Possible worlds of the kind posited in Kripke frame semantics are not tractably representable. Therefore, theories that rely on such a framework cannot explain the processes through which speakers actually interpret the expressions of a natural language. They also do not provide computationally manageable accounts of the ways in which epistemic agents reason about modality, knowledge and beliefs.

We have seen that by adapting the distinction between operational and denotation semantics from programming languages to natural language it is possible to develop a fine-grained treatment of intensions that dispenses with possible worlds. The intension of an expression is its operational meaning. Two expressions can have different intensions but provably equivalent denotations.

We replace Kripke frame semantics with probability models in order to interpret modal expressions, and we use Bayesian networks to encode knowledge, belief, and inference. While probability distributions, and Bayesian networks in particular, pose tractability problems, stratification, estimation, and approximation techniques allow us to effectively represent significant subclasses of these models. Therefore they offer a computationally realistic basis for handling epistemic states and inference.

If the approach that I have suggested here is to offer an interesting alternative to possible worlds semantics, then it will have to integrate the operational view of intensions into the probabilistic treatment of knowledge and belief. Specifically, it must explain how intensions are acquired by the sort of learning processes that are expressed in Bayesian networks.

In addition, it must develop a wide coverage system that combines a compositional semantics with a procedure for generating probability models in which it is possible to sample a large number of predicates. [3] provide an initial prototype for this system. Much work remains to be done on both the compositional semantics and the model testing components in order to create a robust Bayesian framework for natural language interpretation.

References

1. Babai, L.: Graph isomorphism in quasipolynomial time (2016), arXiv:1512.03547
2. Barwise, J., Perry, J.: Situations and Attitudes. MIT Press (Bradford Books), Cambridge, MA (1983)
3. Bernardy, J.P., Blanck, R., Chatzikyriakidis, S., Lappin, S.: A compositional bayesian semantics for natural language. In: Proceedings of the International Workshop on Language, Cognition and Computational Models, COLING 2018, Santa Fe, New Mexico (2018)
4. Blackburn, P., Bos, J.: Representation and Inference for Natural Language. CSLI, Stanford (2003)
5. Carnap, R.: Meaning and Necessity. University of Chicago Press, Chicago (1947)
6. Ciardelli, I., Groenendijk, J., Roelofsen, F.: Inquisitive semantics: A new notion of meaning. Compass **7(9)**, 459–476 (2013)
7. Ciardelli, I., Roelofsen, F., Theiler, N.: Composing alternatives. Linguistics and Philosophy **40**, 1–36 (2017)
8. Clark, A., Lappin, S.: Linguistic Nativism and the Poverty of the Stimulus. Wiley-Blackwell, Chichester, West Sussex, and Malden, MA (2011)
9. Cooper, R., Dobnik, S., Lappin, S., Larsson, S.: A probabilistic rich type theory for semantic interpretation. In: Proceedings of the EACL 2014 Workshop on Type Theory and Natural Language Semantics (TTNLS), pp. 72–79. Association of Computational Linguistics, Gothenburg, Sweden (2014)
10. Cooper, R., Dobnik, S., Lappin, S., Larsson, S.: Probabilistic type theory and natural language semantics. Linguistic Issues in Language Technology **10**, 1–43 (2015)
11. Cooper, R.: Type Theory and Language: From perception to linguistic communication. Unpublished ms, November 2016, University of Gothenburg (2016)
12. Fox, C., Lappin, S.: Type-theoretic logic with an operational account of intensionality. Synthèse **191(3)**, 563–584 (2015)
13. Fox, C., Lappin, S., Pollard, C.: A higher-order, fine-grained logic for intensional semantics. In: Alberti, G., Balough, K., Dekker, P. (eds.) Proceedings of the Seventh Symposium for Logic and Language. pp. 37–46. Pecs, Hungary (2002)

14. Fox, C., Lappin, S.: Formal Foundations of Intensional Semantics. Blackwell, Oxford (2005)
15. Gettier, E.L.: Is justified true belief knowledge? Analysis **23(6)**, 121–123 (1963)
16. Goodman, N., Lassiter, D.: Probabilistic semantics and pragmatics: Uncertainty in language and thought. In: Lappin, S., Fox, C. (eds.) The Handbook of Contemporary Semantic Theory, Second Edition, pp. 655–686. Wiley-Blackwell, Oxford (2015)
17. Groenendijk, J., Stokhof, M.: Dynamic Montague grammar. In: Kálmán, L., Polós, L. (eds.) Papers from the Second Symposium on Logic and Language. pp. 3–48. Akadémiai Kiadó, Budapest, Hungary (1990), also available as ITLI Prepublication Series LP–90–02
18. Groenendijk, J., Stokhof, M.: Dynamic predicate logic. Linguistics and Philosophy **14**(1), 39–100 (Feb 1991)
19. Halpern, J.: Should knowledge entail belief? Journal of Philosophical Logic **25**, 483–494 (1995)
20. Halpern, J.: Reasoning About Uncertainty. MIT Press, Cambridge MA (2003)
21. Heim, I.: E-type pronouns and donkey anaphora. Linguistics and Philosophy **13**, 137–177 (1990)
22. Holliday, W., Icard, T.: Measure semantics and qualitative semantics for epistemic modals. In: Proceedings of SALT 23. pp. 514–534 (2013)
23. Jonsson, B., Tarski, A.: Boolean algebras with operators. American Journal of Mathematics **73**, 891–939 (1951)
24. Koski, T., Noble, J.: Bayesian Networks: An Introductions. Wiley, Hoboken, NJ (2009)
25. Kratzer, A.: Modality. In: von Stechow, A., Wunderlich, D. (eds.) Semantics: An International Handbook of Contemporary Research, pp. 639–650. de Gruyter, Berlin (1991)
26. Kratzer, A.: Situations in natural language semantics. In: Stanford Encylopedia of Philosophy. Stanford University (2014)
27. Kripke, S.: A completeness theorem in modal logic. Journal of Symbolic Logic **24**, 1–14 (1959)
28. Kripke, S.: Semantical considerations on modal logic. Acta Philosophica Fennica **16**, 83–89 (1963)
29. Lappin, S.: An intensional parametric semantics for vague quantifiers. Linguistics and Philosophy **23**, 599–620 (2000)
30. Lappin, S.: Intensions as computable functions. Linguistic Issues in Language Technology **9**, 1–12 (2013)
31. Lappin, S.: Curry typing, polymorphism, and fine-grained intensionality. In: Lappin, S., Fox, C. (eds.) The Handbook of Contemporary Semantic Theory, Second Edition, pp. 408–428. Wiley-Blackwell, Malden, MA and Oxford (2015)
32. Lassiter, D., Goodman, N.: Adjectival vagueness in a bayesian model of interpretation. Synthese **194**, 3801–3836 (2017)
33. Lau, J.H., Clark, A., Lappin, S.: Grammaticality, acceptability, and probability: A probabilistic view of linguistic knowledge. Cognitive Science pp. 1–40 (2016)
34. Lewis, D.: Probabilities of conditionals and conditionals of probabilities II. The Philosophical Review Review **95**(4), 581–589 (1986)
35. Montague, R.: Formal Philosophy: Selected Papers of Richard Montague. Yale University Press, New Haven, CT/London, UK (1974), edited with an introduction by R. H. Thomason
36. Moschovakis, Y.N.: A logical calculus of meaning and synonymy. Linguistics and Philosophy **29**, 27–89 (2006)

37. Murphy, K.: An introduction to graphical models (2001), unpublished ms, University of British Columbia
38. Muskens, R.A.: Sense and the computation of reference. Linguistics and Philosophy **28**, 473–504 (2005)
39. Papadimitriou, C.: Computational Complexity. Addison-Wesley Publishing Co., Reading, MA (1995)
40. Partee, B.: Semantic facts and psychological facts. Mind and Language **2**(1), 43–52 (1988)
41. Pearl, J.: Bayesian decision methods. In: Shafer, G., Pearl, J. (eds.) Readings in Uncertain Reasoning, pp. 345–352. Morgan Kaufmann (1990)
42. Pollard, C.: Hyperintensions. Journal of Logic and Computation **18**, 257–282 (2008)
43. Russell, S., Norvig, P.: Artificial Intelligence: A Modern Approach. Prentice Hall Series in Artificial Intelligence, Englewood Cliffs, NJ (1995)
44. Stalnaker, R.: Indexical belief. Synthèse **49**(1), 129–151 (1981)
45. Stalnaker, R.: Inquiry. MIT Press/Bradford Books, Cambridge MA (1984)
46. Stump, A.: Programming Language Foundations. Wiley, Hoboken, NJ (2013)
47. van Eijck, J., Lappin, S.: Probabilistic semantics for natural language. In: Christoff, Z., Galeazzi, P., Gierasimszuk, N., Marcoci, A., Smets, S. (eds.) Logic and Interactive Rationality (LIRA), Volume 2. ILLC, University of Amsterdam (2012)

Complexity, Natural Language and Machine Learning

M. Dolores Jiménez-López

Research Group on Mathematical Linguistics, Universitat Rovira i Virgili,
Av. Catalunya 35, Tarragona, Spain
mariadolores.jimenez@urv.cat

1 Introduction: Overview of the Problem

In our work presented by this talk, we focus on linguistic complexity, from several perspectives. Are all languages equally complex? Can languages differ in complexity? Does it make sense to compare the complexity of languages? Complexity is a controversial concept in linguistics. Until recently, natural language complexity has not been widely researched, and it is still not clear how complexity has to be defined and measured. It is necessary to provide an objective and meaningful method to calculate linguistic complexity. In order to reach this goal, an interdisciplinary solution — where computational models should be taken into account — is needed. Studies in theoretical linguistics have to propose methods and techniques for the analysis of natural language complexity, since the results obtained from these studies may have important implications both from a theoretical and from a practical point of view.

2 Complexity and Natural Languages

Complexity is a controversial concept in linguistics. If we review how the concept has been treated within the discipline, we can clearly distinguish two different moments: the 20th century followed by the period starting from 2001. 20th century linguistics defended a view that the linguistic complexity is invariant and that languages are not measurable in terms of complexity. Those ideas have been dubbed the *ALEC statement* ('All Language are Equally Complex'), see [2], or the *linguistic equi-complexity dogma*, see [5]. From McWhorther's pioneering work [6], the equi-complexity dogma — which stated that the total complexity of a natural language is fixed because sub-complexities in linguistic sub-systems trade off — has been almost completely debunked. With the arrival of the 21st century, we have seen what Joseph and Newmeyer [4] call the "decline in popularity of the equal complexity principle." There have been many seminars, conferences, articles, monographs and collective volumes that have dealt with linguistic complexity and have challenged the *equi-complexity dogma*. In fact, we can say that, nowadays, the topic of complexity figures prominently in linguistics. However, even though, natural language complexity has been extensively studied for almost two decades, it is still not clear how complexity has to be defined and measured. This situation has led to the proposal of many methods and criteria to quantify the level of complexity of languages [1, 5, 10, 14, 13]. Currently, there is not any unanimously accepted solution to quantify linguistic complexity.

In the literature, there is no agreement about how to define a notion of complexity. As noted by Mufwene et. al. [12], it is surprising to see the scarcity of works that explain what complexity is when referring to language. Instead, we can find a variety of approaches that has led to a linguistic complexity taxonomy: absolute complexity vs. relative complexity; global complexity vs. local complexity; system complexity vs. structural complexity, etc.

With this diversity of definitions, measures and criteria to calculate or estimate complexity vary and depend on the specific research interests and on the definition of complexity adopted. There is no conventionally agreed metric for measuring the complexity of natural languages. The measures proposed could be grouped into two blocs: measures of absolute complexity and measures of relative complexity. Some researchers have attempted to apply the concept of complexity used in other disciplines (information theory, computational models, theory of complex systems, etc.) in order to find useful methods to calculate linguistic complexity.

3 Why Machine Learning for Measuring Linguistic Complexity?

One of the most used typology of complexity in linguistics is the one that distinguish between *absolute* complexity and *relative* complexity [7–9]. The *absolute — objective — complexity* approach defines complexity as an objective property of any given system, and it is measured in terms of the number of parts of the system, the interrelations between the parts, or the length of the description of the phenomenon [6, 1]. The *relative — agent-related — complexity* approach takes into account the users of language and identifies complexity with difficulty/cost of processing, learning or acquisition [5].

In general, researchers agree that it is more feasible to approach linguistic complexity from an objective or theory-oriented viewpoint than from a subjective or user-related perspective. On the other hand, generally, studies that have adopted a relative complexity approach have showed some preferences for L2 (i.e. second language) learners, see [5]. However, as pointed out by Miestamo [7], if we aim to reach a general definition of relative complexity, the primary relevance of L2 learners is not obvious. In fact, they could be considered the least important of the four possible groups that may be considered — speakers, hearers, L1 (i.e. first language) learners, L2 learners.

Taking into account the centrality of L1 learners, we defend that studies on relative complexity may check differences among languages by considering child first language acquisition. Due to the problems that methods for studying language acquisition (observational and experimental) may set out to the study of linguistic complexity, we defend that machine learning models [11] may be considered as important complementary tools that — by avoiding practical problems of analyzing authentic learner productions data — will make possible to consider children (or their simulation) as suitable candidates for evaluating the complexity of languages.

Machine learning may provide computational models for natural language acquisition. We take the stand that the use of formal or computational methods — and tools developed by such methods — to give a description of the machinery necessary to acquire a language is an important strategy within the field of language acquisition [3]. In general, it is recognized that computational methods can shed new light on processes of language acquisition, see [15].

We take up as important tasks the development of computational models in general, and machine learning in particular. They can provide powerful methods and techniques, which adequately represent the processes of natural language acquisition. Next important stages are to build potentially good tools based on such methods to deal with relative linguistic complexity.

Acknowledgments

This research has been supported by the Ministerio de Economía y Competitividad under the project number FFI2015-69978-P (MINECO/FEDER) of the Programa Estatal de Fomento de la Investigación Científica y Técnica de Excelencia, Subprograma Estatal de Generación de Conocimiento.

References

1. Dahl, O.: The Growth and Maintenance of Linguistic Complexity. John Benjamins, Amsterdam (2004)
2. Deutscher, G.: "Overall complexity": A wild goose chase? In: Sampson, G., Gil, D., Trudgill, P. (eds.) Language Complexity as an Evolving Variable, pp. 243–251. Oxford University Press, Oxford (2009)
3. Frank, M.C.: Computational models of early language acquisition (2011), http://langcog.stanford.edu/papers/F-underreview-b.pdf
4. Joseph, J., Newmeyer, F.: "All Languages Are Equally Complex": The rise and fall of a consensus. Historiographia Linguistica **39:2/3**, 341–368 (2012)
5. Kusters, W.: Linguistic Complexity: The Influence of Social Change on Verbal Inflection. LOT, Utrecht (2003)
6. McWhorter, J.: The world's simplest grammars are creole grammars. Linguistic Typology **6**, 125–166 (2001)
7. Miestamo, M.: On the Feasibility of Complexity Metrics. In: Krista, K., Sepper, M. (eds.) Finest linguistics. Proceedings of the Annual Finish and Estonian Conference of Linguistics, pp. 11–26. Tallinna Ülikooli Kirjastus, Tallinn (2006)
8. Miestamo, M.: Grammatical complexity in a cross-linguistic perspective. In: Miestamo, M., Sinnemäki, K., Karlsson, F. (eds.) Language Complexity: Typology, contact, change, pp. 23–42. John Benjamins, Amsterdam (2009)
9. Miestamo, M.: Implicational hierarchies and grammatical complexity. In: Sampson, G., Gil, D., Trudgill, P. (eds.) Language Complexity as an Evolving Variable, pp. 80–97. Oxford University Press, Oxford (2009)

10. Miestamo, M., Sinnemäki, K., Karlsson, F.: Language Complexity: Typology, contact, change. John Benjamins, Amsterdam (2008)
11. Mitchell, T.: Machine Learning. McGraw Hill series in computer science, McGraw-Hill (1997)
12. Mufwene, S., Coupé, C., Pellegrino, F.: Complexity in Language. Developmental and Evolutionary Perspectives. Cambridge University Press, New York (2017)
13. Newmeyer, F., Preston, L.: Measuring Grammatical Complexity. Oxford University Press, Oxford (2014)
14. Sampson, G., Gil, D., Trudgill, P.: Language Complexity as an Evolving Variable. Oxford University Press, Oxford (2009)
15. Wintner, S.: Computational Models of Language Acquisition. In: Gelbukh, A. (ed.) Computational Linguistics and Intelligent Text Processing, LNCS 6008, pp. 86–99. Springer, Berlin (2010)

MTT-semantics in Martin-Löf's Type Theory with HoTT's Logic*

Zhaohui Luo**

Dept of Computer Science, Royal Holloway, Univ of London
zhaohui.luo@hotmail.co.uk

Formal semantics in modern type theories (MTT-semantics for short) [7, 1] is a semantic framework for natural language, in the tradition of Montague's semantics [10]. While Montague's semantics is based on Church's simple type theory [2, 4] (and its models in set theory), MTT-semantics is based on dependent type theories, which we call modern type theories (MTTs), to distinguish them from the simple type theory.

Usually, we say that MTTs include predicative type theories such as Martin-Löf's (intensional) type theory (MLTT) [11] and impredicative type theories such as UTT [5] and pCIC [3]. However, so far, we have mainly developed MTT-semantics in the impredicative type theory UTT in which there is a totality *Prop* of all logical propositions. In contrast, Martin-Löf's MLTT, as employed in the work by Sundholm [13], Ranta [12] and others, is predicative and in it there is no such a type of all propositions. In fact, Martin-Löf has identified types with propositions [8, 9] and this gives rise to a logic based on the principle of propositions as types – the usual logic in MLTT – let's call it the PaT logic.

Unfortunately, unlike UTT, MLTT with PaT logic is *inadequate* to be used for MTT-semantics (this has been pointed out and discussed in [6]). This paper, besides describing the problem briefly, proposes the idea that MLTT, when extended with the h-logic developed in the HoTT project [14], can be used adequately as a foundational language for MTT-semantics.[1] This also justifies the inclusion of MLTT as one of the MTTs for MTT-semantics, as we have always done in previous writings.[2]

---

* This short paper accompanies the author's invited talk at LACompLing18: it gives a concise description of a part of the talk that describes unpublished work.

** Partially supported by EU COST Action CA15123 and CAS/SAFEA International Partnership Program.

[1] I should emphasise that further study is needed to demonstrate whether MLTT extended with HoTT's logic can adequately deal with all the semantic matters as studied based on UTT, although intuitively I do not see any serious problems. To mention a potential issue: in a predicative type theory, formally there is no totality of all propositions (and hence no totality of predicates) – one can only have relative totalities of propositions or predicates using predicative universes (cf., Prop in §2). This is not ideal but it is to be seen whether it causes any serious problems.

[2] Although the current work has not been published, its idea, i.e., using HoTT's logic instead of the PaT logic, has been in the author's mind for a long time. This has

1 Background: Problem and Proposal

As I explained in [6], Martin-Löf's type theory with PaT logic is inadequate for MTT-semantics. The reason is that, in order to employ types to represent collections such as those for CNs, some principle of proof irrelevance is needed and such a principle is incompatible with the PaT logic where types and propositions are identified.

For example, one may use Σ-types to represent CNs modified by intersective adjectives [12]: *handsome man* can be interpreted as $\Sigma(Man, handsome)$ where Man is a type and $handsome : Man \to U$ with U being a predicative universe. Then, one can ask: what is the identity criterion for *handsome man*? An obvious answer should be that it is the same as that for *man*: two handsome men are the same if, and only if, they are the same man. This implies that, for any man m, any two proofs of $handsome(m)$ should be the same – proof irrelevance comes into play here.

A principle of proof irrelevance stipulates that any two proofs of the same logical proposition be the same. However, in order to state this principle, there must be a clear distinction between logical propositions and other types so that proof irrelevance can be imposed for the former (and not for the latter). In Martin-Löf's type theory with PaT logic, however, propositions and types are identified and, therefore, proof irrelevance would have implied the collapse of all types into singleton or empty types: this is obviously absurd and unacceptable. In contrast, in an impredicative type theory such as UTT, the distinction between propositions and types is clear – one has a type *Prop* of propositions and, therefore, a principle of proof irrelevance can be stated and imposed in a straightforward way. For instance, proof irrelevance for computational equality can be imposed in UTT by means of the following rule [6]:

$$\frac{\Gamma \vdash P : Prop \quad \Gamma \vdash p : P \quad \Gamma \vdash q : P}{\Gamma \vdash p = q : P}$$

But, such rules would not be possible for MLTT with PaT logic.

Recently, based on Martin-Löf's type theory, researchers have developed Homotopy Type Theory (HoTT) [14] for formalisation of mathematics. One of the developments in the HoTT project is its logic (sometimes called h-logic) based on the idea that a logical proposition is a type that is either a singleton or empty. This, among other things, has given rise to a logic with a type of all (small) propositions. Our proposal is to use MLTT with HoTT's logic (or, more precisely, MLTT extended with h-logic) for MTT-semantics – let's call this type theory MLTT_h. We believe that, like UTT, MLTT_h serves as an adequate foundational semantic language as well.

partly contributed to the decision of including MLTT as one of the MTTs for MTT-semantics.

2 Martin-Löf's Type Theory with H-logic and Its Use for MTT-Semantics

We describe MLTT_h, MLTT with HoTT's logic, sometimes called h-logic. We shall assume the knowledge of MLTT (see Part III of [11] for its formal description) and describe, albeit concisely, the h-logic developed in the HoTT project [14].

Remark 1. MLTT_h only extends MLTT with the h-logic. It does not include the other extensions of MLTT in the HoTT project: in particular, we do not use the univalence axiom or any other higher inductive types except those in h-logic.

2.1 H-logic

In HoTT, a proposition is a type whose objects are all propositionally equal to each other. Formally, let U be the smallest universe in MLTT and $A : U$. Then A is a proposition in h-logic if the following is true/inhabited:

$$\mathsf{isProp}(A) = \Pi x, y{:}A.\ Id_A(x, y),$$

where Id is the propositional equality (called Id-type) in MLTT. We can then define the type of propositions in U to be the following Σ-type:

$$Prop_U = \Sigma X{:}U.\ \mathsf{isProp}(X).$$

In the following, we shall omit U and write PROP for $Prop_U$. Note that PROP is different from $Prop$ in an impredicative type theory like UTT, which is impredicative and contains all logical propositions. PROP does not – it only contains the propositions in the predicative universe U; sometimes, we say that PROP is the type of *small* propositions. Another thing to note is that an object of PROP is not just a proposition – it is a pair (A, p) such that A is a proposition in U and p is a proof of $\mathsf{isProp}(A)$.

The traditional logical operators can be defined and some of these definitions (e.g., disjunction and existential quantifier) use the following truncation operation that turns a type into a proposition.

– *Propositional Truncation.* Let A be a type. Then, there is a higher inductive type $\|A\|$ with the following rules:

$$\frac{\Gamma \vdash a : A}{\Gamma \vdash |a| : \|A\|} \qquad \frac{\Gamma\ valid}{\Gamma \vdash \mathsf{isProp}(\|A\|)\ true} \qquad \frac{\Gamma \vdash \mathsf{isProp}(B) \quad \Gamma \vdash f : A \to B}{\Gamma \vdash \kappa_A(f) : \|A\| \to B}$$

such that the elimination operator κ_A satisfies the definitional equality $\kappa_A(f, |a|) = f(a)$.

Note that $\|A\|$ is a higher inductive type and, in particular, in turning a non-propositional type A into a proposition $\|A\|$, one imposes that there is a proof of $\mathsf{isProp}(\|A\|)$, i.e., $\|A\|$ is a proposition – in other words, every two proofs of $\|A\|$ are equal (propositionally).[3]

The traditional logical operators can be defined as follows.

- $true = \mathbf{1}$ (the unit type) and $false = \emptyset$ (the empty type).
- $P \wedge Q = P \times Q$, $P \supset Q = P \rightarrow Q$, $\neg P = P \rightarrow \emptyset$ and $\forall x{:}A.P(x) = \Pi x{:}A.P(x)$.
- $P \vee Q = \|P + Q\|$ and $\exists x{:}A.P(x) = \|\Sigma x{:}A.P(x)\|$.

2.2 MTT-semantics in $\mathbf{MLTT}_h$

MTT-semantics can be done in MLTT_h,[4] including the following examples.

Predicates. We can approximate the notion of predicate by means of the relative totality PROP of small propositions – i.e., a predicate over type A is a function of type $A \rightarrow$ PROP. Therefore, we can interpret linguistic entities such as verb phrases, modifications by intersective adjectives, etc. as we have done before based on UTT.

Proof Irrelevance. In h-logic as described above, every two proofs of a proposition in PROP are equal (by definition, for the propositional equality Id) and, in particular, this is imposed for $\|A\|$ when a non-propositional type A is turned into a proposition $\|A\|$. Therefore, the problem described in §1 is resolved satisfactorily in MLTT_h.

References

1. Chatzikyriakidis, S., Luo, Z.: Formal Semantics in Modern Type Theories. Wiley & ISTE Science Publishing Ltd. (2018), (to appear)
2. Church, A.: A formulation of the simple theory of types. J. Symbolic Logic 5(1) (1940)
3. The Coq Development Team: The Coq Proof Assistant Reference Manual (Version 8.3), INRIA (2010)
4. Gallin, D.: Intensional and higher-order modal logic: with applications to Montague semantics (1975)
5. Luo, Z.: Computation and Reasoning: A Type Theory for Computer Science. Oxford University Press (1994)
6. Luo, Z.: Common nouns as types. In: Bechet, D., Dikovsky, A. (eds.) Logical Aspects of Computational Linguistics (LACL'2012). LNCS 7351 (2012)
7. Luo, Z.: Formal semantics in modern type theories with coercive subtyping. Linguistics and Philosophy 35(6), 491–513 (2012)
8. Martin-Löf, P.: An intuitionistic theory of types: predicative part. In: H.Rose, J.C.Shepherdson (eds.) Logic Colloquium'73 (1975)

[3] For people who are familiar with type theory, this implies that canonicity fails to hold for the resulting type theory.

[4] See Footnote 1.

9. Martin-Löf, P.: Intuitionistic Type Theory. Bibliopolis (1984)
10. Montague, R.: Formal Philosophy. Yale University Press (1974), collected papers edited by R. Thomason
11. Nordström, B., Petersson, K., Smith, J.: Programming in Martin-Löf's Type Theory: An Introduction. Oxford University Press (1990)
12. Ranta, A.: Type-Theoretical Grammar. Oxford University Press (1994)
13. Sundholm, G.: Constructive generalized quantifiers. Synthese 79(1), 1–12 (1989)
14. The Univalent Foundations Program: Homotopy type theory: Univalent foundations of mathematics. Tech. rep., Institute for Advanced Study (2013)

Categorial Proof Nets and Dependency Locality: A New Metric for Linguistic Complexity

Mehdi Mirzapour, Jean-Philippe Prost, and Christian Retoré

LIRMM, Montpellier University CNRS, 161 Rue Ada, France,
email: {mehdi.mirzapour — jean-philippe.prost — christian.retore}@lirmm.fr

Abstract. This paper provides a quantitative computational account of why a sentence has harder parse than some other one, or that one analysis of a sentence is simpler than another one. We take for granted Gibson's results on human processing complexity, and we provide a new metric which uses (Lambek) Categorial Proof Nets. In particular, we correctly model Gibson's account in his Dependency Locality Theory. The proposed metric correctly predicts some performance phenomena such as structures with embedded pronouns, garden pathing, unacceptability of center embedding, preference for lower attachment and passive paraphrases acceptability. Our proposal extends existing distance-based proposals on Categorial Proof Nets for complexity measurement while it opens the door to include semantic complexity, because of the syntax-semantics interface in categorial grammars.

Keywords: Computational Linguistics · Psycholinguistics · Human Processing · Categorial Grammar · Linear Logic · Lambek Calculus

1 Introduction

Linguistics and especially generative grammar à la Chomsky makes a distinction between *competence* and *performance* in the human processing of natural language [5]. The competence is, roughly speaking, our ideal ability without time and resource constraints to parse a sentence, i.e. to decide that it is grammatical or not. Competence is formally described by a formal grammar. The performance is how we actually parse a sentence; whether we succeed in achieving that and how much the sentence resists to our attempt to analyze it. Computing the space and time algorithmic complexity is a fake solution because no one knows the algorithm being used by human if it depends on the individual and on the kind of conversation; even if it were so, nothing guarantees that space and time algorithmic complexity matches the degree of difficulty we experience when processing sentences. So this paper, as well as some earlier work by others[14, 21], try to provide a formal and computable account of the results of psycholinguistics experiences regarding linguistic complexity. We focus on syntactic complexity as studied in a number of linguistic processing phenomena such as garden paths, unacceptability of center embedding, preference for lower attachment, passive paraphrases acceptability, and structures with embedded pronouns.

Regarding the psycholinguistics aspects, we mainly follow the studies by Gibson of linguistic complexity of human parsing. Gibson first studied the notion of the linguistic difficulty [10] through the maximal number of incomplete syntactic dependencies that the processor has to keep track of during the course of processing a sentence. We refer to this theory as Incomplete Dependence Theory (IDT) as coined by Gibson. IDT had some limitations for referent-sensitive linguistic phenomena, which justified the later introduction of the Syntactic Prediction Locality Theory [8]. A variant of this theory, namely Dependency Locality Theory (DLT), was introduced later [9] to overcome the limitations with respect to the new linguistic performance phenomena. In the original works, both IDT and DLT use properties of linguistic representations provided in Government-Binding Theory [6].

On the formal side, in order to compute the complexity of a sentence — in a way that matches Gibson's results — we use Lambek Categorial Grammar [16] by means of proof nets construction [19, Chap 6]. Proof nets were originally introduced by Girard [12] as the mathematical structures of proof in linear logic. Categorial proof nets are to categorial grammar what parse trees are to phrase structure grammar. This kind of approach was initiated by Johnson [14], who defines a measure of the instantaneous complexity when moving from a word to the next one (in particular for center embedded relative clauses) in a way that matches Gibson's and Thomas' analysis [11]. To define the complexity of a sentence, Johnson considers the maximum complexity between the words in a given sentence. This approach was refined by Morrill [21], who re-interprets axiom links in categorial proof nets as incomplete (or unresolved) dependencies. We rename this technique as *IDT-based complexity profiling* since it clearly inherits many aspects of Gibson's IDT, plus the new notion of profiling that exists in some psycholinguistic theories. This technique is quite successful at predicting linguistic performance phenomena such as garden paths, unacceptability of center embedding, preference for lower attachment and heavy noun phrase shift. Nevertheless, there is some predictive limitation for referent-sensitive phenomena such as structures with embedded pronouns. Our strategy to overcome this issue is to apply DLT instead of IDT on proof nets constructions which would lead to introduction of *DLT-based complexity profiling*. We will show how this reformulation can improve the predictive power of the existing models in favor of the referent-sensitive linguistic phenomena.

The purpose of developing our computational psycholinguistic model is not solely limited to measuring linguistic complexity. It is potentially applicable to some specific tasks in the domain of the formal compositional semantics. For instance, ranking different possible readings of a given ambiguous utterance, or more generally translating natural language sentences into weighted logical formulas. The rest of the paper is organized as follows: Section 2 summarizes Gibson's ideas on modeling the linguistic complexity of human sentence comprehension, namely IDT and DLT. In section 3 we then define proof nets, and recall the success and limitation of IDT-based complexity profiling. In section 4 we define our DLT-inspired measure, we show how it fixes some problems in previous work and how it gives a correct account of those phenomena. In section 5, we would see a limitation of our approach and a possible future study for solving that limitation. In the last section we conclude our paper and discuss possible future works.

2 Gibson's Theories on Linguistic Complexity

We provide a very quick review of Gibson's IDT and DLT in order to make the readers familiar with their underlying concepts. The question of how to automatically compute linguistic complexity based on both theories with categorial proof nets will be covered in the sections (3.2) and (4).

Incomplete dependency theory is based on the idea of counting missing incomplete dependencies during the incremental processing of a sentence when a new word attaches to the current linguistic structure. The main parameter in IDT is the number of incomplete dependencies when the new word integrates to the existing structure. This gives an explanation for the increasing complexity of the examples (1a)-(1c) which have nested relative clauses. In (1a), *the reporter* has one incomplete dependency; in (1b), *the senator* has three incomplete dependencies; in (1c) *John* has five incomplete dependencies at the point of processing. For the sake of space, we only explain the most complex case, i.e. (1c) in which the incomplete dependencies at the moment of processing *John* are: (i) the NP *the reporter* is dependent on a verb that should follow it; (ii) the NP *the senator* is dependent on a different verb to follow; and (iii) the pronoun *who* (before *the senator*) is dependent on a verb to follow; (iv) the NP *John* is dependent on another verb to follow; and (v) the pronoun *who* (before *John*) is dependent on a verb to follow. These are five unsaturated or incomplete or unresolved dependencies. IDT in its original form suggests to calculate the maximum number of incomplete dependencies of the words in a sentence. One can observe that the complexity is proportional to the number of incomplete dependencies.

(1a) The reporter disliked the editor.

(1b) The reporter [who the senator attacked] disliked the editor.

(1c) The reporter [who the senator [who John met] attacked] disliked the editor].

(1d) The reporter [who the senator [who I met] attacked] disliked the editor].

Dependency Locality Theory is a distance-based referent-sensitive linguistic complexity measurement put forward by Gibson to supersede the predictive limitations of the incomplete dependency theory. DLT posits two integration and storage costs. In this paper, we have only focused on the integration cost. The linguistic complexity is interpreted as the locality-based cost of the integration of a new word to the dependent word in the current linguistic structure which is relied on the number of the intervened new discourse-referents. By performing a measurement on these referents, we can predict the relative complexity, such as structures with embedded pronouns, illustrated in example (1d). The experiments [24] support the acceptability of (1d) over (1c). According to the discourse-based DLT structural integration cost hypothesis, referents for the first-person pronoun *I* is already present in the current discourse, so, integrating across them consumes fewer cognitive resources than integrating across the new discourse referents before *John*. By means of just two aspects of DLT, namely the structural integration and the discourse processing cost we would be capable to predict a number of linguistic phenomena as we will see in details with some examples.

3 Complexity Profiling in Categorial Grammars

3.1 Proof-nets as parse structures

Our exposition of the henceforth classical material on proof nets for categorial grammar follows [20] — the main original papers on this topic are [16, 12, 22, 23]. Categorial grammars are defined from a set $\mathcal{C}$ of grammatical categories, defined from base categories (for instance $\mathcal{B} = \{np, n, S\}$) including a special symbol S (for sentence) and operators, for instance :

$$\mathcal{C} ::= \mathcal{B} \mid \mathcal{C}\backslash\mathcal{C} \mid \mathcal{C}/\mathcal{C}$$

The symbols $\backslash$ and $/$ can be viewed as logical connectives, namely implication(s) of a logic, namely intuitionistic non-commutative multiplicative linear logic better known as the Lambek calculus. Such formulas can be viewed as formulas of linear logic, with conjunction $\otimes$ disjunction $\parr$ and negation $(\_)^\perp$ because implications can be defined from negation and disjunction:

Definition of $\backslash$ and $/$: $A \setminus B \equiv A^\perp \parr B \qquad B \,/\, A \equiv B \parr A^\perp$
De Morgan equivalences $(A^\perp)^\perp \equiv A \qquad (A \parr B)^\perp \equiv B^\perp \otimes A^\perp$
$(A \otimes B)^\perp \equiv B^\perp \parr A^\perp$

Some formulas have a polarity. Formulas are said to be positive (output) $\circ$ or negative (input) $\bullet$ as follows:[1]

$a : \circ, \qquad a^\perp : \bullet$

$\otimes$	$\bullet$	$\circ$
$\bullet$	*undefined*	$\bullet$
$\circ$	$\bullet$	*undefined*

$\parr$	$\bullet$	$\circ$
$\bullet$	*undefined*	$\circ$
$\circ$	$\circ$	*undefined*

So $a \parr a$ has no polarity, $a^\perp \parr b$ is positive, it is $a \setminus b$ while $b^\perp \otimes a$ is negative, it is the negation of $a \setminus b$. Categories are, roughly speaking, analogous to non-terminal categories in phrase structure grammars. But observe that they are endowed with an internal structure, i.e. $(np \setminus S) \,/\, np$ is a compound category and the rules make use of this internal structure, connectors $\backslash$ and $/$ and subcategories n, np and S. The rules (of the logic) do not depend on the language generated (or analyzed) by the grammar. They are the same for every language, and the lexicon makes the difference. The lexicon maps every word to a finite set of possible categories. A parse structure in a categorial grammar defined by a lexicon $\mathcal{L}$ for a sequence of words $w_1, \ldots, w_n$ simply is a proof of $c_1, \ldots, c_n \vdash S$ with $c_i \in \mathcal{L}(w_i)$ in some variant of the Lambek calculus. The rules for the basic (associative) Lambek calculus are:

$$\overline{A \vdash A}$$

[1] Here we are stricter than in other articles, i.e. we neither allow $\otimes$ of positive formulas nor $\parr$ of negative formulas, because we only use the $\backslash$ and $/$ symbols in categories (and not $\otimes$): only combining heterogeneous polarities guarantees that a positive formula is a category, and that a negative formula is the negation of a category.

$$\frac{H_1, H_2 \ldots, H_{n-1}, H_n \vdash C}{H_1, H_2 \ldots, H_{n-1} \vdash C \,/\, H_n}$$

$$\frac{H_1, H_2 \ldots, H_{n-1}, H_n \vdash C}{H_2 \ldots, H_{n-1}, H_n \vdash H_1 \setminus C}$$

$$\frac{H_1, \ldots, H_i, \ldots, H_n \vdash C \qquad G_1, \ldots, G_n \vdash A}{H_1, \ldots, H_{i-1}, G_1, \ldots, G_n, A \setminus H_i, H_{i+1}, \ldots, H_n \vdash C}$$

$$\frac{H_1, \ldots, H_{i-1}, H_i, H_{i+1} \ldots, H_n \vdash C \qquad G_1, \ldots, G_n \vdash A}{H_1, \ldots, H_{i-1}, H_i \,/\, A, G_1, \ldots, G_n, H_{i+1}, \ldots, H_n \vdash C}$$

Since the Lambek sequent calculus enjoys the cut-elimination property whereby a sequent is provable if and only if it is provable without the cut rule, we do not mention the cut rule. Categorial grammars are known for providing a transparent and computable interface between syntax and semantics. The reason is that the categorial parse structure is a *proof* in some variant of the Lambek calculus, and that this *proof* gives a way to combine semantic lambda terms from the lexicon into a lambda term which encodes a formula expressing the meaning of the sentence. We cannot provide more details herein, the reader is referred e.g. to [20, Chapter 3]. For instance, the categorial analysis of *Every barber shaves himself.* with the proper semantic lambda terms for each word in the sentence yields the logical form $\forall x.barber(x) \Rightarrow shave(x, x)$.

It has been known for many years that categorial parse structures, i.e. proof in some substructural logic, are better described as proof nets [23, 22, 18, 20]. Indeed, categorial grammars following the parsing-as-deduction paradigm, an analysis of a c phrase $w_1, \ldots, w_n$ is a proof of c under the hypotheses $c_1, ..., c_n$ where c_i is a possible category for the word w_i; and proofs in those systems are better denoted by graphs called proof nets. The reason is that different proofs in the Lambek calculus may represent the same syntactic structure (constituents and dependencies), but these essentially similar sequent calculus proofs correspond to a unique proof net. A proof net is a graph, whose nodes are formulas, and it consists of two parts:

subformula trees of the conclusions, in the right order, whose leaves are the base categories, and branching are two connectives $⅋$ and $\otimes$ —– as we have seen formulas with $\setminus$ and $/$ can be expressed from base categories and their negations with $⅋$ and $\otimes$ — for nodes that are not leaves the label can be limited to the main connective of the subformula instead of the whole formula, without loss of information;

axioms that are a set of pairwise disjoint edges connecting a leaf z to a leaf $z^\perp$, in such a way that every leaf is incident to some axiom link.

However not all such graphs are proof nets, only the one satisfying:[2]

Acyclicity Every cycle contains the two edges of the same $⅋$ branching.

Connectedness There is a path not involving the two edges of the same $⅋$ branching between any two vertices.

[2] This list is redundant: for instance intuitionism plus acyclicity implies connectedness.

Intuitionism Every conclusion can be assigned some polarity.
Non commutativity The axioms do not cross (are well bracketed).

The advantage of proof-nets over sequent calculus is that they avoid the phenomenon known as spurious ambiguities— that is when different parse structures correspond to the same syntactic structure (same constituent and dependencies). Indeed proofs (parse structures) with unessential differences are mapped to the same proof net. A (normal) deduction of $c_1, ..., c_n \vdash c$ (i.e. a syntactic analysis of a sequence of words as a constituent of category c) maps to a (normal) proof net with conclusions $(c_n)^{\perp}, ..., (c_1)^{\perp}, c$ [23, 20]. Conversely, every normal proof net corresponds to at least one normal sequent calculus proof [22, 20].

3.2 Incomplete Dependency-Based Complexity Profiling and its Limitation

In this subsection we recall the IDT-based measure of the linguistic complexity by Morrill [21] which itself improves over a first attempt by Johnson [14]. Both measures are based on the categorial proof nets. The general idea is simple: to re-interpret the axiom links as dependencies and to calculate the incomplete dependencies during the incremental processing by counting the incomplete axiom links for each word in a given sentence. This is almost the same as Gibson's idea in his IDT, except the fact that he uses some principles of Chomsky Government-Binding theory [6] instead of the categorial proof nets. The notion of counting incomplete dependencies for each node, called complexity profiling, is more effective in terms of prediction than approaches that only measures maximum number of the incomplete dependencies or the maximum cuts [14].

We can rewrite IDT-based complexity profiling [21] by the following definitions:

Definition 1: Let π be a a syntactic analysis of $w_1, \cdots, w_n$ with categories $C_1, \ldots,$ C_n — that is a categorial proof net with conclusions $(C_n)^{\perp}, ..., (C_1)^{\perp}, S$. Let C_{i_0} be one of the C_i ($i \in [1, n]$). The incomplete dependency number of C_{i_0} in π, written as $ID_\pi(C_{i_0})$, is the count of axioms $c - c'$ in π such that $c \in (C_{i_0-m} \cup S)$ ($m \geq 0$) and $c' \in C_{i_0+n+1}$ ($n \geq 0$).

Definition 2: Let π be a a syntactic analysis of $w_1, \cdots, w_n$ with categories $C_1, \ldots,$ C_n — that is a categorial proof net with conclusions $(C_n)^{\perp}, ..., (C_1)^{\perp}, S$. We define the IDT-based linguistic complexity of π, written $f_{idt}(\pi)$ by $(1 + \sum_{i=1}^{n} ID_\pi(C_i))^{-1}$.

Definition 3: Given two syntactic analyses π_i and π_j, not necessarily of the same words and categories, we say that π_i is IDT-preferred to π_j whenever $f_{idt}(\pi_i) >$ $f_{idt}(\pi_j)$.

Example: Figure (1) shows the two relevant proof nets for examples (2a) with subject-extracted relative clause and (2b) with object-extracted relative clause (examples from [9]). The relevant complexity profiles for (2a) and (2b) are illustrated in the figure (2). As it can be seen, the total sum of the complexity for (2b) is greater than (2a), thus, it can predict correctly the preference of (2a) over (2b) which is supported

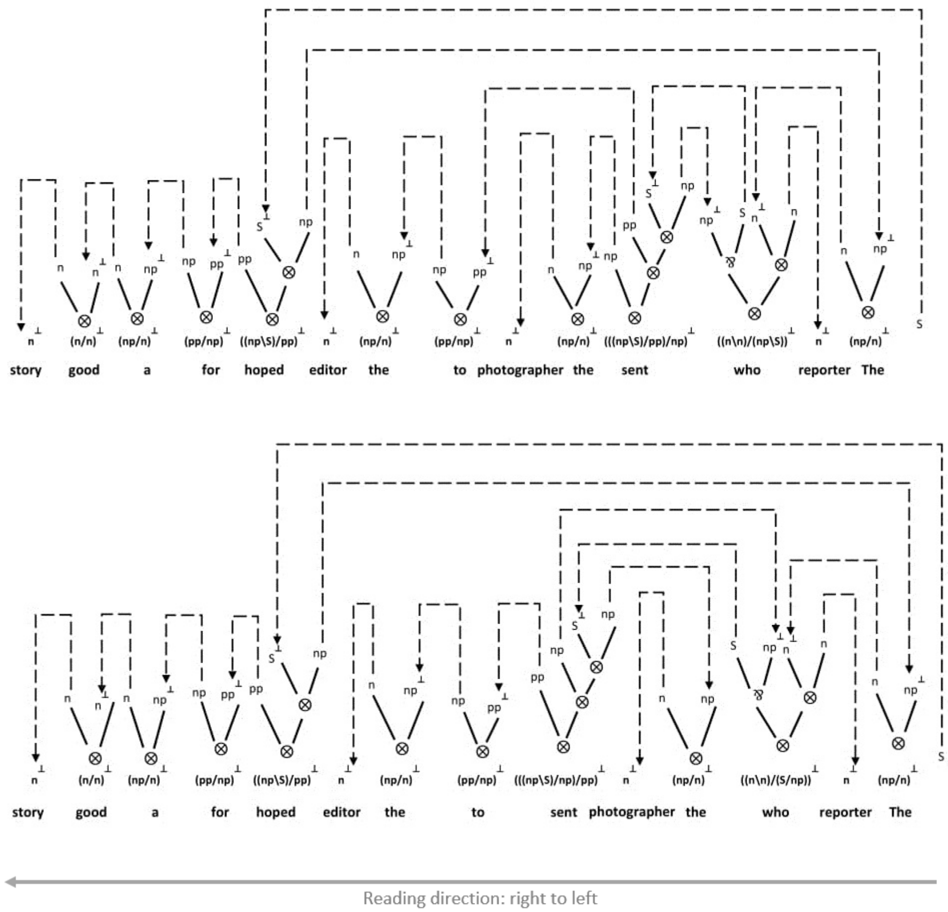


Fig. 1. Proof net analyses for (2a) located in top (subject-extracted relative clause) and (2b) in bottom (object-extracted relative clause).

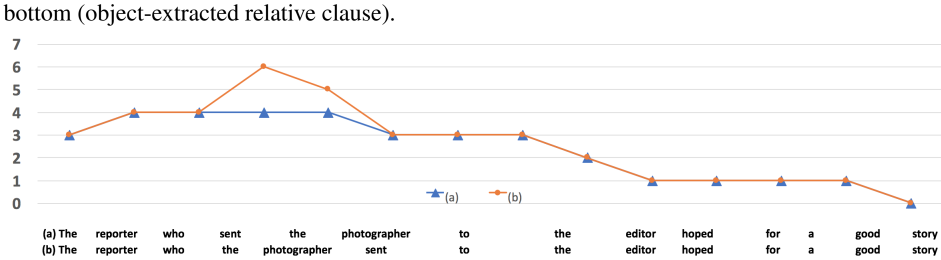


Fig. 2. IDT-based Complexity Profiles for (2a) and (2b).

by measuring reading time experiments [7].[3]

(2a) The reporter who sent the photographer to the editor hoped for a good story.

(2b) The reporter who the photographer sent to the editor hoped for a good story.

Obviously, IDT-based account does not use DLT as its underlying theory. Not surprisingly, the linguistic phenomena that can only be supported by DLT would not be supported by IDT-based complexity profiling. Figure (3) shows this failure. We can verify this by applying the definitions on the relevant proof nets as it is illustrated in the in the figure (4). As one may notice, the corresponding proof nets for the examples (1c) and (1d) are almost the same. Consequently, IDT-based complexity profiling cannot discriminate both examples, i.e. it generates the same number for both sentences in contrast to the experiments [24] as it is shown in the figure (3). This shows the importance of introducing DLT-based complexity profiling for proof nets in order to make more predictive coverage—as we will do so.

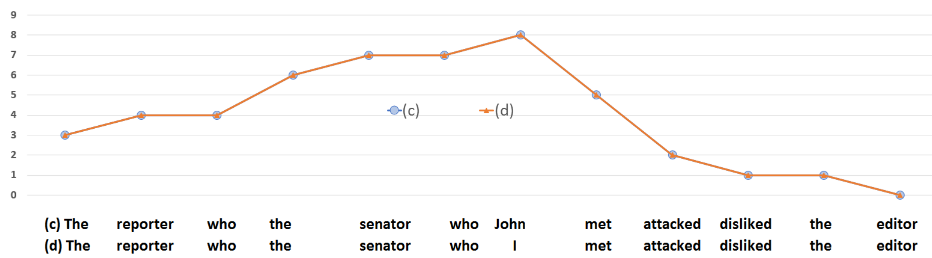


Fig. 3. IDT-based Complexity Profiles for (1c) and (1d).

4 A New Proposal: Distance Locality-Based Complexity Profiling

As we discussed, IDT-based complexity profiling is a distance-based measurement. However, it is not a referent-sensitive criterion and due to this fact, it cannot support some of the linguistic phenomena such as structures with embedded pronouns. One plausible strategy to overcome this lack is introducing DLT-based complexity profiling. This will allow us to have a referent-sensitive measurement. In this section, we provide the precise definitions of our DLT-based proposal on the basis of the categorial proof nets. Here they are:

Definition 4: A word w is said to be a discourse referent whenever it is a *proper noun*, *common noun* or *verb*.

[3] The same procedure, would show the the increasing complexity of the examples (1a)-(1c) by drawing the relevant proof-nets. This practice is avoided in this paper due the space limitation and its simplicity comparing to the running examples here.

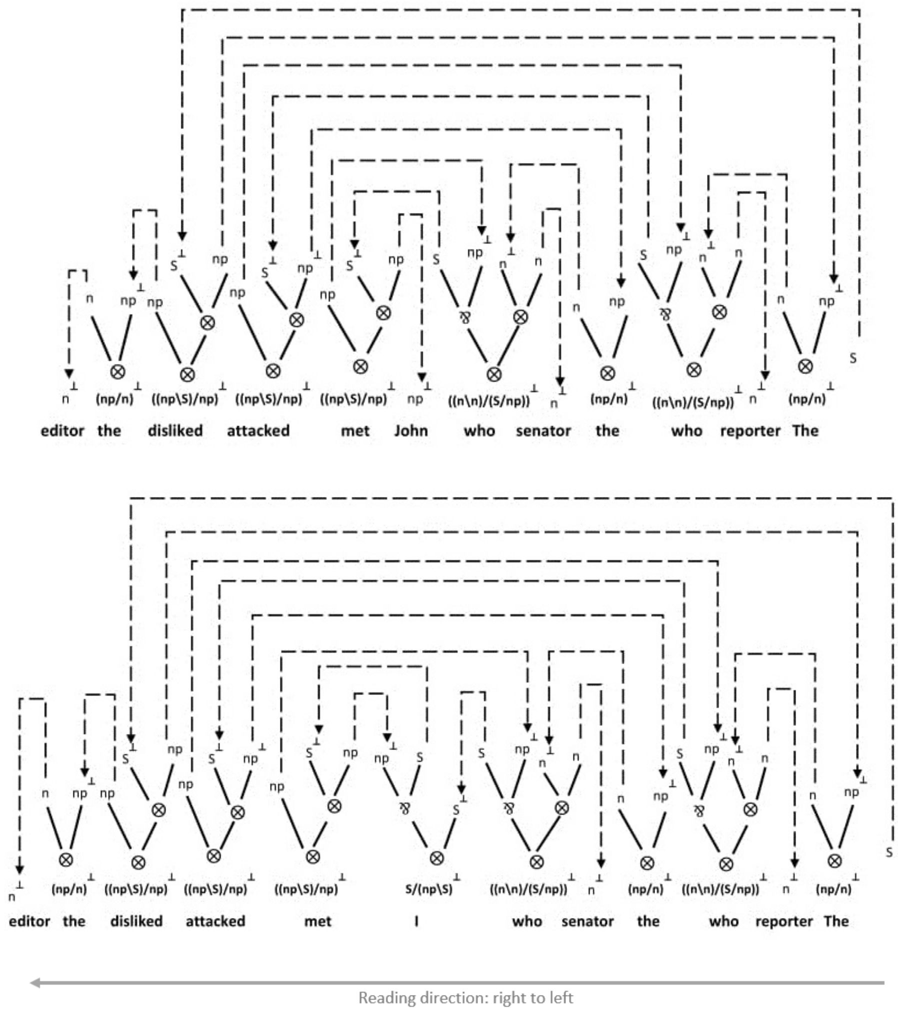


Fig. 4. Proof net analyses for both examples (1c) and (1d).

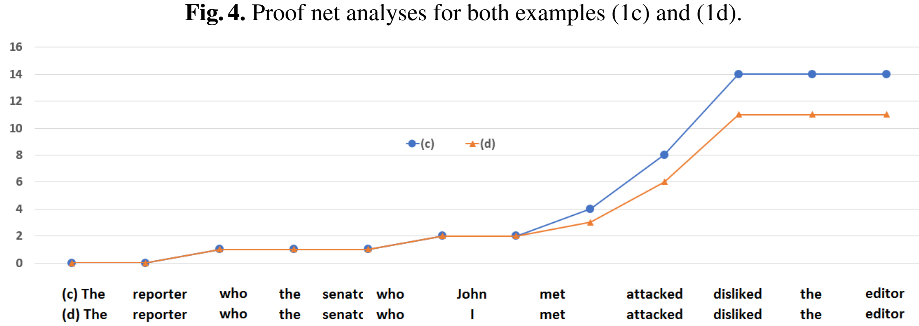


Fig. 5. Accumulative DLT-based Complexity Profiles for (1c) and (1d).

Definition 5: Let π be a a syntactic analysis of $w_1, \cdots, w_n$ with categories $C_1, \ldots,$ C_n — that is a categorial proof net with conclusions $(C_n)^\perp, ..., (C_1)^\perp, S$. Let $c - c'$ be an axiom in π such that $c \in C_i$ and $c' \in C_j$ ($i, j \in [1, n]$). We define the **length** of axiom $c - c'$ as the integer $i + 1 - j$.

Definition 6: Let π be a a syntactic analysis of $w_1, \cdots, w_n$ with categories $C_1, \ldots,$ C_n — that is a categorial proof net with conclusions $(C_n)^\perp, ..., (C_1)^\perp, S$. Let C_{i_0} be one of the C_i, and let consider axioms $c - c'$ with c in C_{i_0} and c' in some C_{i_0-k}. Let us consider the largest k for which such an axiom exists — this is the longest axiom starting from C_{i_0} with the previous definition. The dependency locality number of C_{i_0} in π, written $DL_\pi(C_{i_0})$ is the number of discourse referent words between $w_{i_0} : C_{i_0}$ and $w_{i_0-k} : C_{i_0-k}$. The boundary words, i.e. $w_{i_0} : C_{i_0}$ and $w_{i_0-k} : C_{i_0-k}$ should also be counted. Alternatively, it may be viewed as $k+1$ minus the number of non-discourse references among those $k + 1$ words.

Definition 7: Let π be a a syntactic analysis of $w_1, \cdots, w_n$ with categories $C_1, \ldots,$ C_n — that is a categorial proof net with conclusions $(C_n)^\perp, ..., (C_1)^\perp, S$. We define the DLT-based linguistic complexity of π, written $f_{dlt}(\pi)$ by $(1 + \sum_{i=1}^{n} DL_\pi(C_i))^{-1}$.

Definition 8: Given two syntactic analyses π_i and π_j, not necessarily of the same words and categories, we say that π_i is DLT-preferred to π_j whenever $f_{dlt}(\pi_i) > f_{dlt}(\pi_j)$.

Examples: We apply our new metric on examples (1c) and (1d). Figure (4) shows the relevant proof net for (1c) and (1d). The proof nets for both examples are the same except a difference in one of the lexicons in each example, i.e. *John* and *I*.[4] Figure (5) shows the accumulative chart-based representation of our measurement for each example. The axis Y shows the accumulative sum of dependency locality function applied to each category in axis X. The quick analysis of the profiles shows the total complexity numbers 14 and 11 for (1c) and (1d), respectively. This correctly predicts the preference of example (1d) over (1c) which was not possible in the IDT-based approaches.

The measurement for dependency locality number is quite straightforward. As an example, we calculate the dependency locality number for the word *attacked* in figure (4) for (1d). We can find the longest axiom link starting from *attacked* and ended to its right most category, namely, *who*. Then, we count the number of discourse referents intervened in the axiom link, which is actually three; namely, *attacked*, *met* and *senator*.

We can evaluate our proposal for measuring the linguistic complexity against other linguistic phenomena. Our experiment shows that the new metric supports both referent-sensitive and some of the non-referent-sensitive phenomena such as garden pathing, unacceptability of center embedding, preference for lower attachment and passive paraphrases acceptability. For saving space, we just illustrate Passive Paraphrases Acceptability [21] in this paper. This linguistic phenomenon is illustrated by examples (3a)

[4] Following Lambek [16], we have assigned the category $S/(np\backslash S)$ to relative pronoun *I*. Note that even assigning np, which is not a type-shifted category, would not change our numeric analysis at all.

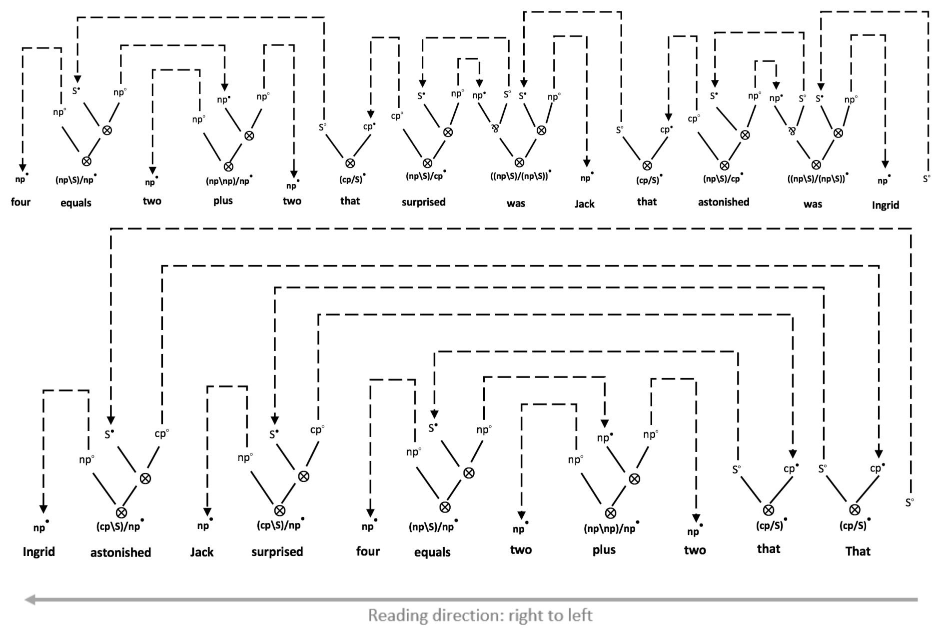


Fig. 6. Proof net analyses for (3a) in the top and (3b) in the bottom.

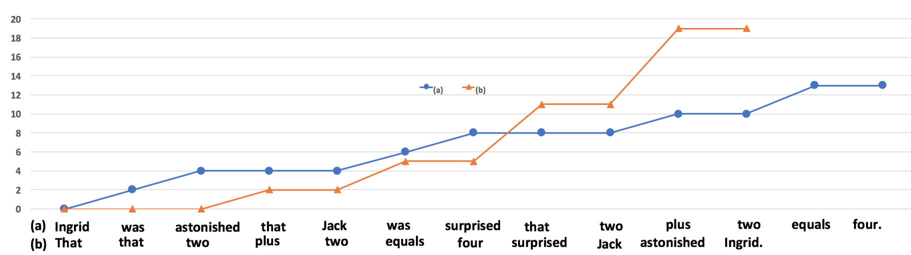


Fig. 7. Accumulative DLT-based complexity profiles for (3a) and (3b)

and (3b). Notice that the DLT-based complexity profile of the (3a) is lower even though the number of the sentences and the axiom links are more comparing to (3b). The real preference is on the syntactic forms in which (3a) is preferred to (3b). The relevant proof nets and the accumulated complexity profiles are illustrated in the figures (6) and (7), respectively.

Example 3a: Ingrid was astonished that Jack was surprised that two plus two equals four.

Example 3b: ?That that two plus two equals four surprised Jack astonished Ingrid.

5 Limitation

There is a limitation in our approach and it is the problem of ranking valid semantic meanings of a given multiple-quantifier sentence which cannot be supported by our proposal. A study [3] has shown the same problem in the IDT-based approach when dealing with some type of the expressions such as sentence-modifier adverbials and nested sentences. Thus, both IDT-based and DLT-based complexity profiling cannot correctly predict ranking the quantifier scoping problem. Hopefully, this can be treated with the hybrid models [3] in which Hilbert's epsilon and tau [13, 4] are exploited.

6 Conclusion and Possible Extensions

In this paper we explored how our DLT-based complexity profiling on proof nets can give a proper account of the complexity of a wide range of linguistic phenomena. We have also shown that IDT-based method could not support referent-sensitive linguistic performance phenomena. This was one of the main reasons for introducing the DLT-based complexity profiling technique within the framework of Lambek calculus. There are some extensions for our study and research:

- As we mentioned it is possible to bridge our model with other study [3] to overcome the problem of ranking quantifier scoping, which our proposal already has. As we discussed, we can exploit Hilbert's epsilon and tau operators [13, 4] for neutralizing the quantifier effect and making possible the complexity measurement by the penalty cost of the quantifiers re-ordering.
- Another important direction is to take into account not only the axioms of the proof-nets but also the logical structure, i.e., par-links, tensor-links and the correctness criterion. This is important indeed, because this structure is needed to compute the logical form (semantics) from the syntactic structure given by proof nets. For instance, nesting Lambek slashes (that are linear implications, and therefore par-links in the proof net) corresponds to higher order semantic constructions (e.g. predicates of predicates) and consequently this nesting of par-links increases the complexity of the syntactic and semantic human processing.
- It is possible to combine our method with studies in other directions: One potential candidate is the task of sentence correction/completion in Lambek Calculus [17]. The other task is measuring semantic gradience in natural language. Some line of research suggests this feature within lexical/compositional frameworks by creating and enrichment of the wide-coverage weighted lexical resources from crowd-sourced data [15].

Acknowledgement We would like to show our gratitude to Philippe Blache for his insightful discussion at our lab and also for inspiration that we got from his papers [1, 2]. We would like to thank our colleague Richard Moot as well for his numerous valuable comments on this work.

References

1. Blache, P.: A computational model for linguistic complexity. In: Proceedings of the first International Conference on Linguistics, Biology and Computer Science (2011)
2. Blache, P.: Evaluating language complexity in context: New parameters for a constraint-based model. In: CSLP-11, Workshop on Constraint Solving and Language Processing (2011)
3. Catta, D., Mirzapour, M.: Quantifier scoping and semantic preferences. In: Proceedings of the Computing Natural Language Inference Workshop (2017)
4. Chatzikyriakidis, S., Pasquali, F., Retoré, C.: From logical and linguistic generics to hilbert's tau and epsilon quantifiers. IfCoLog Journal of Logics and their Applications **4**(2), 231–255 (2017)
5. Chomsky, N.: Aspects of the Theory of Syntax. MIT Press, Cambridge (1965)
6. Chomsky, N.: Some concepts and consequences of the theory of government and binding, vol. 6. MIT press (1982)
7. Gibson, E., Ko, K.: An integration-based theory of computational resources in sentence comprehension. In: Fourth Architectures and Mechanisms in Language Processing Conference, University of Freiburg, Germany (1998)
8. Gibson, E.: Linguistic complexity: Locality of syntactic dependencies. Cognition **68**(1), 1–76 (1998)
9. Gibson, E.: The dependency locality theory: A distance-based theory of linguistic complexity. Image, language, brain pp. 95–126 (2000)
10. Gibson, E.A.F.: A computational theory of human linguistic processing: Memory limitations and processing breakdown. Ph.D. thesis, Carnegie Mellon University Pittsburgh, PA (1991)
11. Gibson, E., Thomas, J.: The processing complexity of english center-embedded and self-embedded structures. In: of Massachusetts, U. (ed.) The proceedings of the North-Eastern Linguistic Society 1996 (1996)
12. Girard, J.Y.: Linear logic. Theoretical Ccomputer Science **50**, 1–102 (1987)
13. Hilbert, D.: Die logischen grundlagen der mathematik. Mathematische Annalen **88**(1), 151–165 (1922)
14. Johnson, M.E.: Proof nets and the complexity of processing center-embedded constructions. In: Retoré, C. (ed.) Special Issue on Recent Advances in Logical and Algebraic Approaches to Grammar. Journal of Logic Language and Information, vol. 7(4), pp. 433–447. Kluwer (1998)
15. Lafourcade, M., Mery, B., Mirzapour, M., Moot, R., Retoré, C.: Collecting weighted coercions from crowd-sourced lexical data for compositional semantic analysis. In: New Frontiers in Artificial Intelligence, JSAI-isAI 2017. Lecture Notes in Computer Science. vol. 10838, pp. 214–230. Springer (2018)
16. Lambek, J.: The mathematics of sentence structure. The American Mathematical Monthly **65**(3), 154–170 (1958)
17. Mirzapour, M.: Finding missing categories in incomplete utterances. In: 24e Conférence sur le Traitement Automatique des Langues Naturelles (TALN). p. 149
18. Moot, R.: Proof nets for linguistic analysis. Ph.D. thesis, UIL-OTS, Universiteit Utrecht (2002)
19. Moot, R., Retoré, C.: The logic of categorial grammars: a deductive account of natural language syntax and semantics, vol. 6850. Springer (2012)
20. Moot, R., Retoré, C.: The logic of categorial grammars: a deductive account of natural language syntax and semantics, LNCS, vol. 6850. Springer (2012), http://www.springer.com/computer/theoretical+computer+science/book/978-3-642-31554-1

21. Morrill, G.: Incremental processing and acceptability. Computational linguistics **26**(3), 319–338 (2000)
22. Retoré, C.: Calcul de Lambek et logique linéaire. Traitement Automatique des Langues **37**(2), 39–70 (1996)
23. Roorda, D.: Proof nets for Lambek calculus. Logic and Computation **2**(2), 211–233 (1992)
24. Warren, T., Gibson, E.: The effects of discourse status on intuitive complexity: Implications for quantifying distance in a locality-based theory of linguistic complexity. In: Poster presented at the Twelfth CUNY Sentence Processing Conference, New York (1999)

A Proposal to Describe Fuzziness in Natural Language

Adrià Torrens Urrutia

Universitat Rovira i Virgili, Tarragona, Spain
adria.torrens@urv.cat

In this presentation, we highlight the need to propose formal models that consider grammaticality as a gradient property instead of the categorical view of grammaticality defended in theoretical linguistics. Given that deviations from the norm are inherent to the spontaneous use of language, linguistic analysis tools should account for different levels of grammaticality.

When people use natural language in natural conversation, it is very common to hesitate over what they are going to say, to abandon the discursive thread or to repeat words and phrases. Natural language is thus described as spontaneous, immediate and ambiguous, and it is often produced with grammar violations. These features mean a problem for traditional language analysis systems that, in general, reject sentences with errors. In fact, grammar has traditionally been defined in linguistics from a categorical point of view (that is to say, an input is either grammatical or ungrammatical). However, throughout the history of linguistics, many authors have considered the possibility that a grammar accounts for non-canonical or non-grammatical productions, yielding to a gradual or "fuzzy" conception of language.

While some authors such as Bouchard and Joos have claimed that "fuzziness is not present in grammar in any way" [9], or that "nothing in language has degrees" [12], other linguists postulate that a grammar should be able to account for any type of linguistic construction. In this sense, a grammar must accept that gradient phenomena are an inherent reality in natural language. In this regard, gradience is a well-known linguistic term that designates this conception. Aarts [1] defines gradience as a term to designate the spectrum of continuous phenomena in language, from categories at the level of grammar to sounds at the level of phonetics. The most studied gradual phenomenon in the processing of natural language is given by gradient acceptability and grammaticality regarding natural language inputs. Examples of this are: Aarts [1–3], Chomsky [10, 11], Bolinger [8], Ross [20], Prince and Smolensky [19], Keller [13, 14], Blache [4, 6], Manning [16], Smolensky and Legendre [15].

Some linguists have endeavored to demonstrate and formalize both the gradual and fuzzy aspects of language through formal systems that fit these conceptions. Examples of these approaches are: Optimality Theory in Prince and Smolensky [19], Linear Optimality Theory in Keller [13], Property Grammars in Blache [5, 6], Probability Theory in Manning [16], Harmonic Grammars in Smolensky and Legendre [15]. These approaches tried to explain the gradient phenomena in the domain of the phonology and syntax. They have usually been focused on the performance level of the language, illustrating gradience within the acceptability judgments. They also tried to justify finding gradient grammaticality in linguistic inputs by means of specific methods combining the optimality theory and the probability theory with a grammar of constraints. Even though the results, some generative approaches, such as in Newmeyer [17], still reject gradient grammaticality since this phenomenon is extracted from sources from the performance domain such as corpus or judgments. This criticism shows that the problem in accepting grammaticality as a gradient phenomenon is merely theoretical. In this sense, if a linguistic model, such as a grammar, defines gradient phenomena within the linguistic competence, that model would be able to demonstrate gradient grammaticality in natural language.

In this presentation, we claim that a model based on a formal grammar with constraints (or rules) combined with fuzzy logic can represent certain gradual phenomena of language, such as the levels of grammaticality that are found in the different constructions, regarding linguistic competence. This method provides a new perspective since a grammar has never been able to define their bases from a gradient perspective before.

We have applied this new interdisciplinary approach to the description of Spanish syntax. A property grammar, following Blache [6], has been used in our work to define the different constructions and linguistic elements of Spanish. Our property grammar has been modified in order to bring up descriptions with fuzzy logic. In this way, we have defined a fuzzy grammar that can represent the different gradual phenomena and variability that take place in Spanish.

The syntactic properties have been extracted automatically by applying the MarsaGram tool by Blache et al. [7] to the Corpus Universal Dependency Spanish Treebank. This corpus is obtained from the Universal data set Google dataset (version 2.0). It consists of 16,006 tree structures and 430,764 tokens. It is built from newspaper articles, blogs, and consumer reviews. The Spanish Universal Dependency Treebank provides dependency relationships, while MarsaGram classifies each group of constituents in each dependency by frequency, automatically deducting the properties that can be reviewed by linguists. This allows us to define and characterize Spanish constructions and their properties according to an objective data criterion. Once an adequate linguistic revision has been made, this linguistic information is used to define both the gradual relationships of the language and the diffuse phenomena in syntax. Frequency is the criteria for ranking the properties in the grammar. Every property takes place in a construction. We define all the different kinds of constructions in the language by means of the most frequent properties. This corpus methodology by means of frequency allows us to establish the gold standard in our grammar (the rules that are meant to have a value of 1) and, consequently, ranking all the rest of the rules in terms of values between 1 and 0.

A written corpus with utterances is a better option than an oral corpus. In order to extract the properties of the syntax, an oral corpus would present too much variability since oral outputs have more linguistic modules playing an important role in the linguistic inputs, such as prosody or pragmatics. In a written corpus, these modules are more constrained, and syntax and semantics take a more important role in written speech. A violation of a syntactic property would be less likely in a written corpus because pragmatics and prosody cannot soften the violation. This fact allows to extract, in a safer way, the gold standard of the syntactic rules of our grammar by means of frequency.

The application of fuzzy logic to the property grammar has been supervised by members of the Institute for Research and Applications of Fuzzy Modeling (IRAFM) Center of Excellence IT4 Innovations of Ostrava (Czech Republic). The fuzzy logic model used in our formal grammar is Fuzzy Natural Logic by Novák [18], which is a variation of a fuzzy type theory.

Many future benefits may come from this new approach. The first advantage is theoretical. Our proposal is able to define a grammar taking into account both a mathematical method, which represents objects in terms of degrees, and a grammar with constraints, which can define any kind of linguistic input. Thus, this combination is useful for representing the concept of the degrees of grammaticality in a grammar with a gradient approach. The second advantage is related to language technologies and its computational applications for users. This approach might improve human-machine interfaces since the machine would be able to process any kind of inputs in terms of degrees. It would classify any linguistic input in a scale of degrees of grammaticality. In consequence, this could have an impact to the development of more flexible computational tools that facilitate our interaction with machines. In addition, such a combined method could be the base for applications (Apps) for second language acquisition, in which, starting with a simple writing task, the learner would learn from her/his mistakes. The application could rank all the linguistic rules that have been found in a construction, present them for the learner, and provide the degrees of grammaticality and its violations according to a chosen grammar.

Acknowledgments

This research has been supported by the Ministerio de Economía y Competitividad under the project number FFI2015-69978-P (MINECO/FEDER) of the Programa Estatal de Fomento de la Investigación Científica y Técnica de Excelencia, Subprograma Estatal de Generación de Conocimiento.

References

1. Aarts, B.: Conceptions of gradience in the history of linguistics. In: Language Sciences. vol. 26, pp. 343–389. Elsevier (2004)
2. Aarts, B.: Fuzzy Grammar: A Reader. Oxford University Press on Demand (2004)
3. Aarts, B.: Modelling linguistic gradience. In: Studies in Language. vol. 28, pp. 1–49. John Benjamins Publishing Company (2004)
4. Blache, P.: Property Grammars and the Problem of Constraint Satisfaction. In: Erhard Hinrichs, Detmar Meurers, S.W. (ed.) Proceedings of ESSLLI 2000 workshop on Linguistic Theory and Grammar Implementation. pp. 47–56. Springer (2000)

5. Blache, P.: Property Grammars: A Fully Constraint-Based Theory. In: Christiansen, H., Skadhauge, P.R., Villadsen, J. (eds.) Constraint Solving and Language Processing. pp. 1–16. Springer Berlin Heidelberg, Berlin, Heidelberg (2005)
6. Blache, P.: Representing syntax by means of properties: a formal framework for descriptive approaches. In: Journal of Language Modelling. vol. 4, pp. 183–224. Polish Academy of Sciences (2016)
7. Blache, P., Rauzy, S., Montcheuil, G.: MarsaGram: an Excursion in the Forests of Parsing Trees. In: Calzolari, N., Choukri, K., Declerck, T., Goggi, S., Grobelnik, M., Maegaard, B., Mariani, J., Mazo, H., Moreno, A., Odijk, J., Piperidis, S. (eds.) Proceedings of the Tenth International Conference on Language Resources and Evaluation (LREC 2016). European Language Resources Association (ELRA), Paris, France (2016)
8. Bolinger, D.L.M.: Generality, gradience and the all-or-none. Mouton & Company, 14 edn. (1961)
9. Bouchard, D.: The Semantics of Syntax: A Minimalist Approach to Grammar. University of Chicago Press (1995)
10. Chomsky, N.: Aspects of the Theory of Syntax. Cambridge: MIT Press (1965)
11. Chomsky, N.: The Logical Structure of Linguistic Theory. Aarhus: Plenum Press. (1975)
12. Joos, M.: Readings in Linguistics: The Development of Descriptive Linguistics in America Since 1925. American Council of Learned Societies (1957)
13. Keller, F.: Gradience in Grammar: Experimental and Computational Aspects of Degrees of Grammaticality. Ph.D. thesis, Edinburgh: University of Edinburgh (2000)
14. Keller, F.: Linear Optimality Theory as a Model of Gradience in Grammar. In: Gisbert Fanselow, Caroline Fry, R.V., Schlesewsky, M. (eds.) In Gradience in Grammar: Generative Perspectives. pp. 270–287. Oxford University Press (2006)
15. Legendre, G., Miyata, Y., Smolensky, P.: Harmonic Grammar: A formal multi-level connectionist theory of linguistic well-Formedness: An application. In: Computer Science Technical Reports. pp. 884–891. Institute of Cognitive Science, University of Colorado at Boulder (1990)
16. Manning, C.: Probabilistic Syntax. In: Probabilistic Linguistics. pp. 1–40. MIT Press (2003)
17. Newmeyer, F.J.: Grammar is Grammar and Usage is Usage. In: Language. vol. 79, pp. 682–707. JSTOR (2003)
18. Novák, V.: Fuzzy Natural Logic: Towards Mathematical Logic of Human Reasoning. In: Seising, R., Trillas, E., Kacprzyk, J. (eds.) Towards the Future of Fuzzy Logic. pp. 137–165. Springer, Cham (2015)
19. Prince, A., Smolensky, P.: Optimality Theory: Constraint Interaction in Generative Grammar. New Brunswick: Rutgers University (1993)
20. Ross, J.R.: The Category Squish: Endstation Hauptwort. In: Peranteau, P.M., Levi, J.N., Phares, G.C. (eds.) Papers from the 8th Regional Meeting of the Chicago Linguistic Society. vol. 8, pp. 316–328. Chicago Linguistic Society (1972)

Referential Dependencies in Chinese: A Syntax-Discourse Processing Model

Yuan Xie[1]
Utrecht Institute of Linguistics OTS, Utrecht University, Trans 10, The Netherlands
y.xie@uu.nl

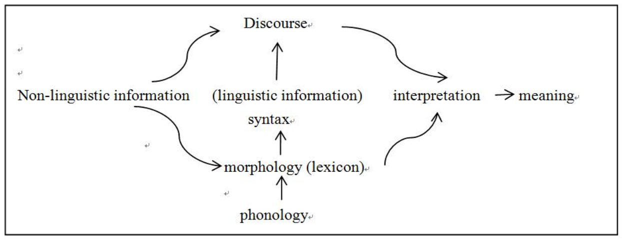
Abstract: In this paper, I am proposing a syntax-discourse processing model for the representation and interpretation of referential dependencies in Chinese. Chinese referentially dependent expressions (e.g. pronouns, reflexives, certain full noun phrases) are different from those in many indo-European languages and rely more on discourse (e.g. using bare noun phrases to express definiteness--lacking overt article *the*; sentence-free reflexive *ziji (self-N)*--referring to the speaker), for this reason, this model, taking both the morphosyntactic and discourse features of the referentially dependent expressions into consideration, reflects the view that referentially dependent nominal expressions and their antecedents are information units that are stored in our working memory system and the referential dependencies are established through the interactions of those information units in our working memory system.

Keywords: referential dependencies; syntax-discourse processing model; working memory; information units

1. Introduction

One of the key problems, the modeling of language processing, is how we can match structured representations of language with properties of the brain that allows human to process information generally. In other words: how is language represented and processed in our brain? First, we generally assume that language is a hierarchical system composed of different levels (e.g. phonology, morphology, syntax, etc). The interpretation of language parts, however, happens on the level of discourse--a level where both linguistic knowledge (e.g. phonological, morphological, syntactic information, etc) as well as non-linguistic knowledge (e.g. world knowledge, vision information, etc) converge:

1. Here I would like to give my sincere thanks to my two supervisors, Prof.dr. Sergey Avrutin and Prof. dr. Peter Coopmans. Thanks for their guidance, comments and corrections in the whole process of writing and revising this paper.



(Avrutin, 1999: 45, modified)

Fig.1 Representations of language in general

In Figure 1, two knowledge modules (e.g. linguistic and non-linguistic) converge on interpretation: non-linguistic information such as world knowledge, vision information, etc; linguistic information that comes from lexical properties, syntactic structures, etc.

Under this framework, I will show the representation of referential dependencies in Chinese, which also involves those two knowledge modules. Firstly, in terms of the linguistic information--the morphosyntactic and discourse distributions of the referentially dependent expressions in Chinese--is different from those in many other languages. For example, English has an article system which allows for a *bridging* relationship to be established between an indefinite DP and an definite DP (e.g. *John bought a book. The author is famous. "the author" is bridged into "a book"*). Chinese, in contrast, does not have such overt articles. The question is how Chinese encodes such bridging relations between DPs in the absence of overt articles. Another referential dependency is the one established between a reflexive element and its antecedent. This is also a *bridging* relation in many languages, whereby the reflexivizing morpheme -SELF typically has semantic relations historically with forms of inalienable possession (e.g. a body part, nose, rib, soul, etc), which shows a hidden connection with its antecedent. Moreover, Chinese, with two types of reflexives (e.g. *ziji* (self-N) and *taziji* (pro-self-N)), differ from many other languages in the number of morphosyntactic features encoded in the reflexives. For example, Chinese *ziji* (self-N) does not have person, gender, and number features, which English reflexive *himself/herself* has). The question then is how Chinese, a language with two reflexive expressions with different combinations of morphosyntactic features and distinctive discourse features, encodes bridging connections with their antecedents. Secondly, in terms of the non-linguistic information (e.g. memory, vision,etc), here in this paper I mainly focus on distinctive linguistic part (e.g. morphosyntactic/discourse feature) and the cognitive part will be explored in my future work.

Basing myself on on Heim's (1982) file change semantics, Avrutin's (1999) and Schumacher, Pinango and Avrutin's (2010) syntax-discourse model, I am proposing a model here that reflects the hypothesis that referential dependencies are established through a series of linguistic operations (e.g. morphosyntactic/discourse features), which finally is relating with non linguistic module (e.g. memory activation level). I will show what this model can account for and which language-related domains it can be applied to.

1.1 What can this model account for?

Previous linguistic proposals concerning referential dependencies have been proposed from either purely syntactic theoretical perspectives (e.g. Chomsky 1981; Huang & Tang, 1991; Reinhart & Reuland, 1993) or purely discourse-theoretical perspective (e.g. Ariel, 1990; Walker, Joshi & Prince, 1998; Kamp, Van Genabith, & Reyle, 2011). With the model I intend to propose, I am taking both morphosyntactic and discourse features into consideration and aim to account for the representation and processing of referential dependencies in Chinese. First, I will outline the representation part --how nominal elements like reflexives, pronouns and (in)definite NPs are represented in terms of memory cards (information units). Secondly, I will outline the processing part --what discourse operations Chinese referential dependencies have, what possible error patterns may occur during language processing and what possible explanations there could be from the perspective of processing resources (e.g. memory).

1.2 Which language domains can this model be applied to?

This model can be applied to such language-related domains as language processing, language acquisition, language pathology. For example, for language acquisition, using the rules established within this model, we can test children's knowledge of the referential dependencies and explain when and why children will obey or violate those rules, shedding light on what they already know and how they apply this knowledge. Similarly, we can also test the corresponding knowledge of aphasics and their (dis)ability to use this knowledge since these two populations show similarities.

2. The Representation of DPs: from syntax to discourse

2.1 Syntactic structure of DPs

The representation of nominal phrases in natural language, DPs, involves a translation process: from the syntactic structure into discourse for interpretation. The latter, in turn, is affected by memory activation level. The typical syntactic structure of DP is:

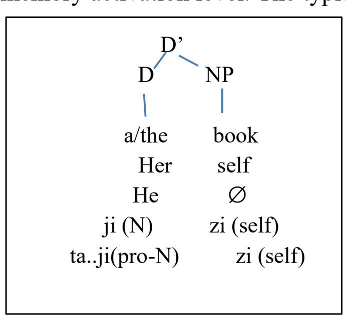


Fig.2 syntactic structure of DP

The distribution pattern of the full noun phrase DP *a/the book*, the pronoun *he* and the reflexive herself *ziji (self) taziji (pro-self-N)* is similar, with D occupied by *a/the he, her,* and NP occupied by *book, self, ,zi(self)*.

2.2 Translation from syntax to discourse

The structure of DP is composed of two parts: D and NP, as was shown in Figure 2. The structure of each information unit (e.g. memory card) is also composed of two parts: a frame (a memory place holder), introduced by the functional category D; a heading (an identifier), introduced by the lexical category NP:

NP: heading	**D:frame**

The translation from a syntactic DP onto a discourse memory card is as follows:

a. DP: [D° [NP]]	e.g. [her[self]]
b. D°----frame	e.g. D: her
c. NP---heading	e.g. NP: self

(Schumacher, Pinango, Avrutin, 2010:1743)

With Schumacher, Pinango and Avrutin (2010), I assume that the D, head of the DP, is the functional category specifying its categorical nature as the head of a nominal phrase [+N], and further consisting of a set of phi-features such as person, gender and number. Further specifications may include case, [+/- definiteness] and [+/-specificity]. These are translated into the frame of the memory card. NP is the lexical category with specific lexical features like dog [+animacy, -human, + hairy], boy [-adult, +human] etc, which can be translated into the heading part of the memory card. We define the frame and heading as follows:

Definition 1: Frames and Headings

A **frame** is a translation of features of the functional category (e.g. person, number, gender, +N). A **heading** is a translation of features of the lexical category.

Those features (either functional or lexical), are not always all present. For example, in the Dutch simplex expression *zich,* the frame *zich* only has the third person feature. The English pronouns *he* and *she* have all the three phi features (person, gender, number) and categorical feature [+N]. Likewise, we can determine that neither the English reflexive *himself/herself* nor the Chinese reflexives *ziji(self-N)* and *taziji (pro-self-N)* have an independent lexical feature (*self* is a variable). Given the determination of their presence or absence, we can formulate featural make-up in terms of sufficiency or insufficiency. This is defined as follows:

Definition 2: The Features of D and NP

a. For all the Ds in pronouns/reflexives, D has Sufficient Functional Features (SFF) if it has at least the following three features[+person] [+gender] [+number]; Otherwise, D is called Insufficient (IFF).

b. For all the NPs in pronouns/reflexives, NP has Sufficient Lexical Features (SLF) if it can be interpreted independently; If not, it is called Insufficient (ILF).

c. For all the Ds in full noun phrases, D has SFF if it has the [+definiteness] feature; if not, it is called ILF.

d. All the DPs that do not have an overt Ds or overt NP, they are labelled as containing Null Functional Feature (NFF) and Null Lexical Feature (NLF) respectively.

Since features (functional and lexical) can either be sufficient or insufficient, the corresponding memory cards (with frames and headings) also have two associated conditions. These are labelled as weak and strong.

Definition 3 The Strength of Frames and Prominence of Headings

a. A Strong Frame (SF) is a translation of sufficient functional features; A Weak Frame (WF) is a translation of insufficient functional features; An Empty Frame (EF) is a translation of null functional features.

b. A Strong Heading (SH) is a translation of sufficient lexical features; A Weak Heading (WH) is a translation of insufficient lexical features; An Empty Heading (EH) is a translation of null lexical features.

Given the above definitions (2 and 3), we can determines 9 (3*3) combinations of frames and headings. In (1-9) I have listed the possible mapping mechanisms and corresponding examples (drawing on Chinese, Dutch and English):

(1) $D_{SFF}NP_{SLF} \rightarrow F_{SF}H_{SH}$ (e.g. a dog) [1]

(2) $D_{NFF}NP_{ILF} \rightarrow F_{EF}H_{WH}$ (e.g. classical Chinese reflexive 'zi' (self))

(3) $D_{NFF}NP_{SLF} \rightarrow F_{EF}H_{SH}$ (e.g. Chinese bare NP : gou (dog))

(4) $D_{IFF}NP_{NLF} \rightarrow F_{WF}H_{EH}$ (e.g. zich, ancient Chinese `ji`(N))

(5) $D_{IFF}\ NP_{ILF} \rightarrow F_{WF}H_{WH}$ (e.g. zichzelf, ziji (self-N))

(6) $D_{IFF}\ NP_{SLF} \rightarrow F_{WF}H_{SH}$ (e.g. the dog, the bride)

(7) $D_{SFF}\ NP_{NLF} \rightarrow F_{SF}H_{EH}$ (e.g. him, her, it)

(8) $D_{SFF}\ NP_{ILF} \rightarrow F_{SF}H_{WH}$ (e.g. himself, herself)

(9) $D_{NFF}\ NP_{NLF} \rightarrow F_{EF}H_{\ EH}$ (e.g. null topic sentence in Chinese) ∅

Among all the nine conditions, we can see that only the card in condition (1) is complete, with a complete (strong) frame and a complete (strong) heading. This

[1] In all the nine conditions, we use nine graphs to show the mapping between a DP and a card: a box with full lines representing a strong frame; a box with dotted lines representing a weak frame; two full lines within the box representing a strong heading; two dotted lines within the box represent a weak heading; and an empty box/lines representing an empty frame/heading.

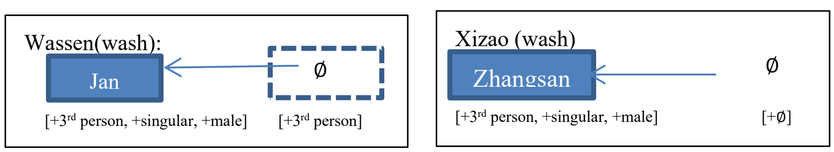
complete card is independent because it does not rely on other cards to be interpreted; all the other cards are incomplete, with either incomplete (weak/empty) frame or incomplete (weak/empty) heading. Those incomplete cards are dependent cards because they rely on other cards to be interpreted. In my model, there are three typical discourse operations between the dependent cards and independent cards: copy-and-paste, cut-and-paste and bridging. These will be illustrated in the next section.

3. The processing of DPs: three types of operations

3.1 Cut-and-Paste

Referential dependencies between the simplex expressions and their antecedents can be established through a cut-and-paste process on the level of discourse, for example:

(10) a. Jan$_i$ waste zich$_i$.
Jan washed zich
"John washed himself"
b. Zhangsan xizao le.
Zhangsan wash LE[2].
"Zhangsan washes himself



In (10a) the morphological feature of *zich* is third person, so it matches with the singular antecedent *Jan*; Also, *zich* cannot be stressed, fronted or have a new guise in (10a). Here *guise* is to be interpreted as similar with *sense*, which is in contrast with *reference*. For example, *morning star* and *evening star* have the same *reference* but they are with different *guises*. In other words, *guises* are the representations of the referents in discourse. Here in (10a), *zich* does not have the above discourse features (e.g. fronted, stressed or new guise), and that is why it does not exist independently in discourse. In other words, the memory card triggered by *zich* does not exist in discourse, therefore – in order for *zich* to get an interpretation, its card should be cut and pasted onto another card. In addition, the predicate *wash* provides a context for *zich* to be cut (giving rise to a so called inherently reflexive interpretation, where only one participant is acceptable). In (10b), although Chinese does not have an overt form like Dutch *zich*, the empty position introduced by the empty card does not exist in discourse either. We can now formulate the following rule

[2] About the markers in this paper: *LE* is an aspect marker, representing finishing; *DE* represents a modifier auxiliary, it usually occurs between a adjective and a noun, or a possessive relation between two nouns. AP represents aspect marker (present)

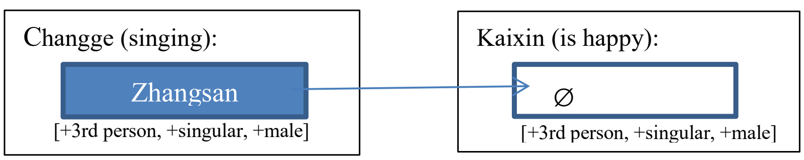
Cut-and-Paste Rule:
Information can be cut-and-pasted iff

a. the referentially dependent card has a weak (or empty) frame and an empty heading;
b. the morphosyntactic features of the frames do not give rise to a conflict between the referentially dependent card and the referentially independent card;
c. the discourse features of the heading cannot introduce an independent card in discourse and the predicate provides a context for it.

3.2 Copy-and-paste: Pronouns

Referential dependencies between strong pronouns (e.g. pronouns that have fully specified phi-features and different from clitics and simplex expressions such as Dutch *zich*, see Cardinaletti and Starke, 1999) and their antecedents are established through a copy-and-paste process on the level of discourse, for example:

(11) Zhangsan$_i$ zai changge. Ta$_i$ feichang kaixin.
Zhangsan AP sing. He very happy.
"Zhangsan is singing. He is very happy"



Because the card triggered by *ta* has an empty heading, it needs to copy information from other cards to establish the completeness of the information. In (11), *ta* copies information from *Zhangsan* under the instructions of the frame features (e.g. third person, singular, male). However, copy-and-paste is forbidden in the following cases:

(12) a. *Zhangsan$_i$ hen lei le. Tamen$_i$ shuizhao-le.
Zhangsan very tired LE. They fall asleep LE.
"Zhangsan was very tired. They fell asleep"
b. *Zhangsan$_i$ xihuan ta$_i$.
Zhangsan like him
"Zhangsan likes him"
c. Zhangsan$_i$ xihuan ta$_i$-de didi.
Zhangsan like he-DE brother
"Zhangsan likes his brother"
d. Zhangsan$_i$ zai ta$_i$ pangbian fang-le yi-ben shu

Zhangsan Prep him next to put- LE one-CL book
"Zhangsan put a book next to him"

f. *Ta$_i$ pashu de shihou, Zhangsan$_i$ shouli nazhe yige pingguo.
He climb DE time, Zhangsan hand-in hold one-CL apple
"When he climbs the trees, Zhangsan hold an apple in his hand"

In (12a), the morphological features of *tamen(they)* are inconsistent with those of *Zhangsan*, therefore, *tamen (they)* cannot copy information from *Zhangsan*. In (12b), information transfer via copy-and-paste is disallowed because *Zhangsan* and *ta (him)* are in the same information chunk. By information chunk we mean the smallest information unit triggered by the same event on the same level. For example, in (12c) and (12d), *ta* can copy and paste information from *Zhangsan* because *ta is* not in the same information chunk with the potential antecedent *Zhangsan (ta* is embedded in the possessive phrase and the prepositional phrase respectively), so copy-and-paste is possible in (12d) and (12c). The reason for impossibility of information transfer between *ta* and *Zhangsan* in (12b) is that the result of copy-and-paste would involve two identical guises (e.g. Zhangsan$_i$ likes Zhangsan$_i$), which is ungrammatical[3]. In (12f) transfer via copy-and-paste is not acceptable, either. The reason is that copy-and-paste is uni-directional, that is, information will copy from a more prominent card and paste it onto a less prominent card. In this paper, I propose a value system to roughly measure the prominence scale of cards:

Value System: prominence of cards:
Prominence value system is measured by values[+1]. The sources of prominence include:
(1) lexicon meanings (e.g. mental involvement verbs);
(2) syntactic positions (e.g. subject/object);
(3)discourse factors (e.g.context, topic, focus, discourse distance, perspectives, etc).

According to the value system, *ta* gets two values because it occurs in the subject position and carries the topic/focus[4] information in (12f) (value [+2]), therefore it is more prominent than *Zhangsan* (object, unfocused). In line with the observations spelled out here, we can formulate the copy-and-paste rule(s) as follows:

Copy-and-Paste Rule(s): Pronouns
Information can be copied-and-pasted iff
a. the referentially dependent card has a strong frame and an empty heading;
b. the morphosyntactic features of the frames are matched between the referentially dependent card and the referentially independent card.
c. the referentially dependent card and the referentially independent card that provides information for the former are not in the same information chunk.
d. the referentially dependent card is at least equal in prominence with the referentially independent card.

[3] In the example *Zhangsan likes Zhangsan,* the two *Zhangsan* may end up with one reference (e.g. two people whose names are *Zhangsan* or two situations that one *Zhangsan* involves in), but two identical guises are forbidden.

[4] For details of focus information on Chinese pronoun *ta*, see Lust, Chien, Chiang & Eisele (1996: 30).

3.3 Bridging

There are two types of bridging in this model. One is the bridging process between an indefinite DP and a definite DP. We label this as a form of *loose bridging*, to draw a contrast with another type of bridging, called *tight bridging*. In the latter case the referentially dependent card contains a heading (e.g. the reflexiving element SELF) denoting an inalienable possession relation with its antecedent (e.g. a body part). The following examples illustrate this.

Tight Bridging

In Chinese, illustrations of tight bridging can be divided into the following types:

(13) a. Zhangsan$_i$ renwei Lisi$_j$ ti-le ziji$_{i/j}$.
Zhangsan think Lisi kick-LE self-N
"Zhangsan thought that Lisi kicked him/himself "

b. Zhangsan$_i$ gaosu-le Lisi$_j$ ziji$_{i/*j}$-de fenshu.
Zhangsan tell-LE Lisi self-N-DE scores
"Zhangsan told Lisi about his own scores"

c. Wangwu$_i$ renwei wo$_j$ nao-le ziji$_{*i/j}$.
Wangwu think I scratch-LE self-N
"Wangwu thought that I scratched myself"

d. Zhangsan$_i$ renwei Lisi$_j$ kanjian-le taziji$_{*i/j}$
Zhangsan think Lisi see LE pro-self-N
"Zhangsan thought that Lisi saw himself"

f. Zhangsan$_i$ gaosu-le Lisi$_j$ taziji$_{i/j}$-de fenshu.
Zhangsan tell-LE Lisi pro-self-N-DE scores
"Zhangsan told Lisi about his scores"

In (13a), *ziji (self-N)* can be bridged either to *Zhangsan* or to *Lisi*. We formulate it as a continuing rule because the two cards *Zhangsan* and *Lisi* have equal prominence. According to the value system above, *Zhangsan* gets one value [+1] from the lexical meaning of the mental verb *thought* while *Lisi* gets one value [+1] from the discourse distance (closer to the the reflexive). Therefore, they are of equal prominence; in (13b), *ziji (self-N)* can only bridge to the distant *Zhangsan* because *Zhangsan* is more prominent than *Lisi*. *Zhangsan* is more prominent than *Lisi* because of the matrix verb *told* [+1] and the subject position of *Zhangsan*[+1], although *Lisi* also gets one value from distance [+1].

In (13c) and (13d), *ziji (self-N)* can only bridge into local *wo (I).* This is not because *wo (I)* is more prominent than *Zhangsan* but because of the hidden frame and heading feature-matching: Chinese *ziji (self-N)* contains hidden discourse information (e.g.the speaker) and hidden semantic information (e.g. *zi* means nose), and once *ziji* encounters the first person pronoun *wo(I),* the hidden discourse feature (e.g. the speaker) should be realized as early as possible. This is the same in (13d), in which *taziji (pro-self-N)* can only bridge to *Lisi* because of the combination features of *ta* and *ziji* should be realized as early as possible. Thus, we formulate the following rules:

Bridging Rule I: Continuing, Jumping and Principle of Earlier Realization
Information can be tightly bridged iff (i) the referentially dependent card has a strong (or weak) frame and a weak heading; (ii) the morphological features of the referentially dependent card and the referentially independent card are not in conflict, and:

a. bridging will continue if the referentially independent cards have equal prominence.
b. bridging will jump over the less prominent cards if the referentially independent cards have unequal prominence.
c. bridging should be realized as early as possible once the morphosyntactic features of the frames and the lexical features of the headings are matched.

As to the complex reflexive *taziji (pro-self-N)*, another property that needs to be mentioned is that when *taziji* and its antecedent card are in the same information chunk (e.g. *Lisi* and *taziji* in (13d)), it behaves like English reflexive *himself/herself*. When it is outside of this specific information chunk (e.g. in (13f)), it functions like a stressed pronoun, with the structure: [the pronoun *ta* + emphatic use].

Loose Bridging

Chinese does not have the determiner article *the*. Instead, the definiteness of a DP is usually encoded by other devices, such as syntactic position (e.g. preverbal position Chao, 1968; Xu, 1995; Cheng & Sybesma, 1999), lexical devices such as demonstratives, classifiers (e.g. Sybesma & Sio, 2008), aspects of world knowledge and visual information such as pointing (e.g. Avrutin, 1999). Here are some examples:

(14) a. Zhangsan mai-le (yi)-ben shu. Zuozhe hen youming.
Zhangsan buy-LE (one)-CL book. Author very famous
"Zhangsan bought an book. The author is very famous"
b. Zhangsan$_i$ mai-le-ben shu. Ta$_i$ hen xihuan na-wei zuozhe.
Zhangsan buy-LE-CL book. He very like that-CL author.
"Zhangsan bought a book. He likes that author very much"
c. Taiyang hen da.
Sun very big
"The sun is very big"
d. Che feichang xuanku. (pointing)
Car very fancy
"The car is very fancy"

In (14a), the definiteness of *zuozhe (author)* is encoded by its preverbal position; in (14b), although *zuozhe* occurs in postverbal position, the demonstrative *na-ge (that)* turns *zuozhe* into a definite DP; in (14c), the definiteness of the DP *taiyang (sun)* is encoded through the common world knowledge shared by human beings; in (14d), the definiteness of the DP *che (car)* is encoded through pointing (visual information). On the other hand, indefiniteness in Chinese is usually encoded by the construction "(numerals) + classifiers + NP", such as *yi-tiao-yu* (one-CL^{shape}-fish: a fish). For further details and discussion see Cheng & Sybesma (1999).

Bridging Rule II: Source of Definiteness
Information can be loosely bridged iff
a. the referentially dependent card has a weak frame and a weak heading;
b. the morphosyntactic feature of the referentially dependent card is [+definite] and that of the referentially independent card is [-definite]. The source of definiteness in Chinese is encoded through lexicon devices (e.g. *na-ge* (that-CL)), syntactic position (preverbal); visual information (e.g.pointing), and world knowledge (e.g.presupposition), etc.

Until now, I have illustrated how different kinds of DPs are represented and how different types of referential dependencies are interpreted in Chinese. The next step is to explain why such referential dependencies are in fact established. In this paper, I suggest that the morphosyntactic features and discourse features together help us establishing the dependencies between cards. For example, in terms of discourse prominence, according to Ariel (1990), the prominence of antecedents is related with the degree of memory activation. The activation level of memory can be correlated with the levels of processing (e.g. Craik& Lockhart, 1972; Anderson, 2005; Baddeley & Hitch, 2017). According to Craik & Lockhart (1972), there are two levels of processing in the memory system--deep and shallow. Deep or shallow processing will result in different memory traces, which can be mapped to the prominence scale of the cards. In other words, the prominence of the cards may reflect the level of memory activation. Since this model is an initial trial for the interpretation of referential dependencies in Chinese, especially for the memory activation part, a more detailed framework is needed in the future model modification process.

4. Conclusion and Future Work

In this paper I have introduced a syntax-discourse interface processing model for the representation and interpretation of referential dependencies in Chinese. Under the hypothesis that referential dependencies are highly related with the level of memory activation, this model suggests that referential dependencies can be established through interactions among memory cards by means of such rules as copy-and-paste, cut-and-paste and bridging. These memory cards (information units) themselves are composed of frames (e.g. projected by D, with functional categorical features) of different strengths, and headings (e.g. projected by NP, with lexical categorical features) of different degrees of prominence.

There are three points that need extra attention. The first one is about the modules of language processing. The model I am proposing in this paper involves two knowledge modules:linguistic and non-linguistic. In this paper, I mainly illustrate the linguistic module. As to the non-linguistic module (e.g. the memory/cognitive part), the present model is an embryonic outline of the role of memory and it lacks survey on the previous cognitive models. All these need further specification in the future work. The second point is about the adequacy of the data, although I have illustrated the working of the model by just a few examples, they are enough to deduce the basic rules above, for I have selected the most prototypical examples from different constructions in Chinese. Due to the limitation of the length of the paper, I have not been able to explain different constructions in greater detail. The final point is about the application of the model. This model is expected to be applied to language acquisition and language pathology, for example, it will be tested against obtained empirical data from experiments testing Chinese children's knowledge of referential dependencies at different ages and the corresponding knowledge of Chinese different types of aphasics in the future work.

References

[1] Anderson, J. R. Cognitive psychology and its implications. Macmillan (2005).

[2] Ariel, M. Accessing noun-phrase antecedents. Routledge (1990).

[3] Avrutin, S. Development of the syntax-discourse interface. Springer Science & Business Media (1999).

[4] Baddeley, A. D., & Hitch, G. J. Is the levels of processing effect language limited? Journal of Memory and Language, 92, 1-13 (2017).

[5] Cardinaletti, A., & Starke, M. The typology of structural deficiency: A case study of the three classes of pronouns. Clitics in the languages of Europe, Mouton de Gruyter, Berlin, 145-233 (1999).

[6] Chao, Y. J. A grammar of spoken Chinese. University of California. Press (1968).

[7] Cheng, L. S., & Sybesma, R. Bare and not-so-bare nouns and the structure of NP. Linguistic Inquiry, 30(4), 509-542 (1999).

[8] Chomsky, N., Lectures on government and binding, Foris Publications, Dordrecht, (1981).

[9] Craik, F. I., & Lockhart, R. S. Levels of processing: A framework for memory research. Journal of verbal learning and verbal behavior, 11(6), 671-684 (1972).

[10] Heim, I. The semantics of definite and indefinite NPs. University of Massachusetts at Amherst dissertation (1982).

[11] Kamp, H., Van Genabith, J., & Reyle, U. Discourse representation theory. In Handbook of philosophical logic (pp. 125-394). Springer, Dordrecht (2011).

[12] Lust, B., Chien, Y. C., Chiang, C. P., & Eisele, J. Chinese pronominals in universal grammar: A study of linear precedence and command in Chinese and English children's first language acquisition. Journal of East Asian Linguistics, *5*(1), 1-47 (1996).

[13] Reinhart, T., & Reuland, E. Reflexivity. Linguistic Inquiry, 24(4), 657-720 (1993).

[14] Schumacher, P. B., Piñango, M. M., Ruigendijk, E., & Avrutin, S. Reference assignment in dutch: evidence for the syntax–discourse divide. Lingua, 120(7), 1738-1763 (2010).

[15] Sybesma, R., & Sio, J. U. S., D is for demonstrative – investigating the position of the demonstrative in Chinese and Zhuang. The Linguistic Review, 25(3-4), 453-478 (2008).

[16] Walker, M. A., Joshi, A. K., & Prince, E. F. (Eds.). Centering theory in discourse. Oxford University Press (1998).

[17] Xu, L. Definiteness effects on Chinese word order. Cahiers de linguistique-Asie orientale, 24(1), 29-48 (1995).