Incremental Graph-Based Semantics & Reasoning for Conversational AI

Angus Addlesee
Heriot-Watt University
Edinburgh
United Kingdom
a.addlesee@hw.ac.uk

Arash Eshghi
Heriot-Watt University
Edinburgh
United Kingdom
a.eshghi@hw.ac.uk

Abstract

The next generation of conversational AI systems need to: (1) process language *incrementally*, token-by-token to be more responsive and enable handling of conversational phenomena such as pauses, restarts and self-corrections; (2) *reason incrementally* allowing meaning to be established beyond what is said; (3) be *transparent* and *controllable*, allowing designers as well as the system itself to easily establish reasons for particular behaviour and tailor to particular user groups, or domains. In this short paper we present ongoing preliminary work combining Dynamic Syntax (DS) - an incremental, semantic grammar framework - with the Resource Description Framework (RDF). This paves the way for the creation of incremental semantic parsers that progressively output semantic RDF graphs as an utterance unfolds in real-time. We also outline how the parser can be integrated with an incremental reasoning engine through RDF. We argue that this DS-RDF hybrid satisfies the desiderata listed above, yielding semantic infrastructure that can be used to build responsive, real-time, interpretable Conversational AI that can be rapidly customised for specific user groups such as people with dementia.

1 Introduction

Humans process language *in real-time* (i.e. incrementally word by word) (Ferreira et al., 2004; Purver et al., 2009; Howes et al., 2011) which gives rise to many conversational phenomena such as interruptions, disfluencies, restarts, corrections and split utterances (Healey et al., 2011; Hough, 2015; Howes and Eshghi, 2017). These phenomena support normal conversation (Goodwin, 1981; Bavelas and Gerwing, 2011), and are likely to be more common in specific groups such as people with dementia (PwD) who have to frequently restart and reformulate entire utterances (Boschi et al., 2017). Yet all commercial voice assistants, and most research systems, are turn-based (Enomoto et al., 2020), tending to ignore these phenomena altogether (Addlesee et al., 2020). Dialogue also involves *reasoning* with ad-hoc, domain-specific rules (Breitholtz, 2020) learned from an early age (Breitholtz and Howes, 2020). This allows participants to infer meaning beyond what is said on the surface of the conversation, and constitutes knowledge of a domain or topic of discourse. Like language processing, reasoning also proceeds on an incremental basis allowing e.g. a hearer to predict what the speaker is going to say before they have said it (Howes et al., 2012). In conversational AI, this *incremental reasoning* capability becomes even more important for specific user groups such as PwD who often need assis-

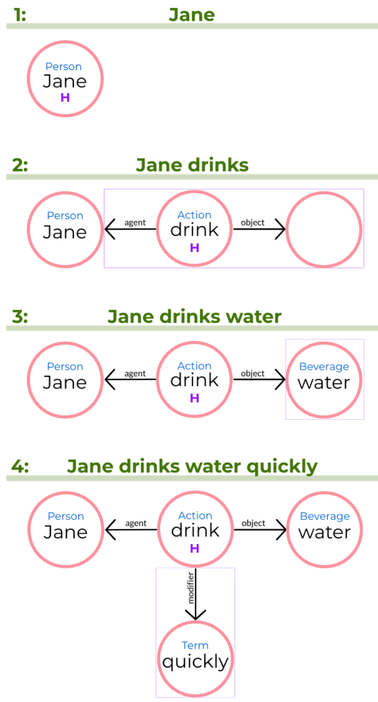


Figure 1: A word-by-word DS-RDF parse of "Jane drinks water quickly"

Proceedings of the Conference on Reasoning and Interaction (ReInAct), pages 1–7
October 4-6, 2021. ©2021 Association for Computational Linguistics

tance from a hearer to complete their own turn, and carry the conversation forward (Ash et al., 2006).

In this short paper, we extend previous work on incremental semantic processing in dialogue by combining an inherently incremental grammar framework, Dynamic Syntax (DS, (Kempson et al., 2001; Cann et al., 2005)), with the Resource Description Framework (RDF, (Lassila et al., 1998)) - dubbed DS-RDF[1] - paving the way for incremental semantic parsers that output RDF semantic graphs word by word as utterances unfold in dialogue (see Fig. 1). Such a parser can then act as a *transparent* and *controllable* language processing layer in incremental conversational AI[2]. While DS already enjoys extensive research applying it to computational dialogue processing (see Purver et al. (2011); Hough (2015); Eshghi et al. (2017) among many others), DS-RDF has the important added benefit, through RDF, that it can hook seamlessly into the rich semantic resources that exist for RDF (Auer et al., 2007; Chiarcos et al., 2013; Vrandečić and Krötzsch, 2014; Zhang et al., 2014). Further, as we outline below, it can also integrate easily with existing incremental RDF reasoning engines such as RDFox (Nenov et al., 2015).

2 Dynamic Syntax

Dynamic Syntax (DS, Kempson et al., 2001; Cann et al., 2005; Kempson et al., 2016) is an action-based grammar framework that directly captures the time-linear, incremental nature of language processing (i.e. both understanding and generation), on a word by word or token by token basis. It models the linear construction of *semantic* representations (i.e. *interpretations*) as progressively more linguistic input is parsed or generated. DS is idiosyncratic in that it does not recognise an independent level of syntactic representation over words: syntax on this view is sets of constraints on the incremental processing of semantic information in potentially multiple modalities (e.g. language and vision); and *grammaticality* is defined in purely procedural terms, i.e. as *parsability in a context*: the successful incremental construction of a semantic representation using all information given by the words in a string.

The output of parsing any given string of words, or non-verbal tokens, is thus a *semantic* tree representing its predicate-argument structure - see Fig. 2. DS trees are always binary branching, with argument nodes conventionally on the right and functor nodes to the left. In DS's original form, tree nodes correspond to terms in the lambda calculus, decorated with labels expressing their semantic type (e.g. $Ty(e)$) and FOL formulae; and beta-reduction determines the type and formula at a mother node from those at its daughters (Fig. 2). These trees can be *partial*, containing unsatisfied *requirements* potentially for any element (e.g. $?Ty(e)$, a requirement for future development to $Ty(e)$), and contain a *pointer*, $\diamondsuit$, labelling the node currently under development.

Actions in DS The parsing process in DS is defined in terms of conditional *actions*: procedural specifications for monotonic semantic tree growth. *Computational Actions* are language-general structure-building principles which apply whenever their preconditions are met; and *Lexical Actions* are language-specific actions corresponding to and triggered by specific lexical tokens (words or gestures). All actions take the form of 'macros' to provide update operations on semantic trees, instantiated as IF…THEN…ELSE rules which yield semantically transparent structures when applied. Fig. 2 is a simplified illustration of the parsing process for "John upset Mary" where the application of Computational Actions is omitted for simplicity. For more detail on lexical specification in DS, see Eshghi et al. (2011), pages 4-5.

3 Knowledge Graphs and RDF

Knowledge graphs formally represent semantics by describing entities and the relations between them, usually represented in RDF (resource description framework); a data representation model for knowledge graphs (Lassila et al., 1998). These RDF models are both human and machine-readable, enabling the collaborative development of interoperable resources and tools. One pertinent example, for our use case, is the use of ontologies as a schema layer. These formal descriptions of data structures are curated and maintained by communities of experts in the ontology's respective field (e.g. linguistics (Chiarcos et al., 2013)).

[1] This work is ongoing so only examples work currently. DS-RDF will be released open-source to the community as the full implementation is developed.

[2] This contrasts with end-to-end trained, neural dialogue systems originating in Vinyals and Le (2015) which are more difficult to control, and whose internal neural representations are much harder to interpret

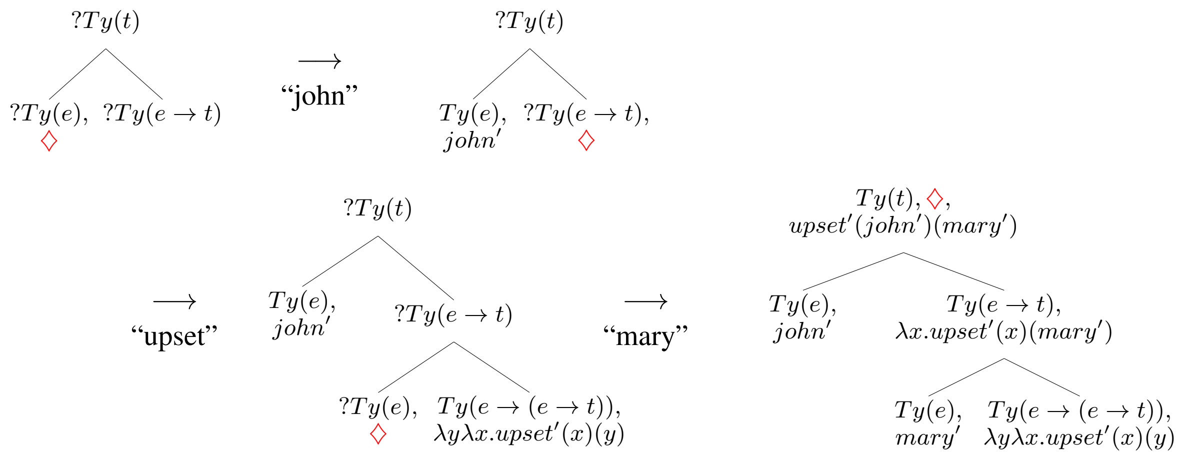


Figure 2: Incremental parsing in DS producing semantic trees: *"John upset Mary"*

Using an ontology enables a shared understanding between the subject experts, users, and applications. For example, schema.org is founded and still actively developed by Google, Microsoft, Yahoo!, and Yandex to provide a shared structure for data on the internet - underlying today's search engines (Mika, 2015).

The WikibaseLexeme data model (Nielsen, 2020) aims to align with the lemon model (McCrae et al., 2011) (which provides rich linguistic grounding for ontologies and the syntax-semantics interface) and is a sibling project of many other huge collaborative projects like Wikipedia, Wikidata, and Wikifunctions (Vrandečić, 2021). The creators plan to create a language-independent version of Wikipedia - encouraging the creation of structured linguistic Wikifunctions. By using a common data model, we can make use of future Wikifunctions as well as current resources.

In order to tweak speech processing with all of this structured linguistic information, we need a point at which we can control the system's behaviour and tailor it for a specific user-group or domain. Using RDFox (Nenov et al., 2015), we have implemented an RDF reasoning layer to enable logical inference for deducing implicit knowledge from known explicit knowledge. We can add and modify logical rules at this layer, providing us with our required ability to control the processing of language. These rules are modelled with the Datalog language (Abiteboul et al., 1995) and can operate efficiently incrementally (Motik et al., 2015), not reasoning across the entire graph every time a new token is uttered. We can therefore define new rules as we learn more about speech impairment, or speech within a particular domain, and deduce implicit knowledge as a user speaks.

4 DS-RDF: Combining Dynamic Syntax with RDF

Dynamic Syntax was originally conceived (Kempson et al., 2001) with the Epsilon Calculus (an extension of FOL) as the formalism in which semantic representations were couched (see e.g. the node decorations in Fig. 2). However, as Kempson et al. (2001) themselves note in chapter 19, what DS models is the real-time parsing *process*, i.e. the *compositional dynamics* of language processing in terms of the twin concepts of underspecification and subsequent update. DS is thus able to remain entirely *agnostic* about the choice of semantic representation. This generality has indeed been exploited in the past: Purver et al. (2010, 2011); Eshghi et al. (2012) used it to combine DS with Type Theory with Records (TTR, Cooper (2005); Cooper and Ginzburg (2015)) with Record Types and functions over these decorating tree nodes, allowing, among other things, the maximal semantics of partial, as well as complete, trees to be computed via type inference at every step (Hough and Purver, 2012). Sadrzadeh et al. (2018); Purver et al. (2021) later showed how DS can be combined with a wholly different kind of semantic representation: that of distributional, or vector-space semantics (VSS, see Clark (2015) for an overview) captured via tensors and vectors, thus enabling VSS to be derived incrementally.

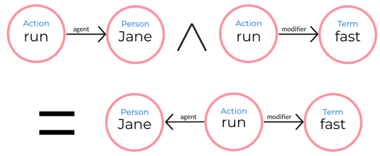
To interface DS with RDF - or any other semantic formalism - there are two key operations that need to be defined over the target (RDF) representations:

Semantic Composition For symbolic representations like RDF graphs here, this operation is standardly some version of the *beta-reduction* operation from the lambda calculus. Here, we follow

Purver et al. (2010, 2011) in taking the semantics of verbs (as well as nouns and adjectives) to be functions from RDF graphs to RDF graphs which *conjoin* ($\bigwedge$) their argument graph with their body – more on this below. The result is a graph that connects the argument graph with the body graph at an argument node designated as `head`[3] and illustrated as **H** - what follows is a simple example where the functional semantics of the intransitive verb `run` is applied to its subject RDF graph representing `Jane`[4]:

Notice that the resulting graph has only 2 nodes, not 3, with the argument graph `head` node collapsing on the function body's node with the same URI, namely that designated as `G.head`.

Conjunction Relative clauses, adverbials as well as some elliptical dialogue phenomena are modelled in DS via LINKed trees: these structures have the effect of temporarily shifting processing to the LINKed tree, whose root content is required to share a term with the node from which they link off (for details, see Kempson et al. (2015), Sec. 1.2.3). When evaluated, the root content of the LINKed tree is *conjoined* with that of the node from which they had linked off; in FOL this operation is simply logical conjunction; in TTR, it is the meet operation (Hough and Purver, 2014); in RDF we similarly define it to be that of *asymmetric merge* of two graphs whereby nodes with the same URI – namely the shared term – collapse while the decorations/contents of this collapsed node come from the right hand side argument of the merge operation ($\bigwedge$). We illustrate with an example showing how the semantics of the sentence "Jane runs fast" is computed by conjoining the content of the adverbial, 'fast' (on a DS LINKed tree not shown here) as a modifier of the 'running' action, is conjoined with the the content of the matrix clause, 'Jane runs', with the shared term/node collapsing:



Two more auxiliary mechanisms are needed for complete integration: (a) Inferring the maximal semantics of DS partial trees involves decorating the nodes not yet developed with *underspecified RDF graphs* of the right type - this can be done by excluding node annotations in these graphs - see e.g. Fig. 1, Step 2; (b) *Subsumption*: roughly, A subsumes B if A is monotonically extensible to B – subsumption checking is crucial in both generation (Hough and Purver, 2012; Eshghi et al., 2012) and grammar induction (Eshghi et al., 2013); in TTR, this is the inverse of the subtype ($\sqsubseteq$) operation (supertype). In RDF, we can define this relation to be that of the *subgraph* with appropriate node subsumption operations.

With the above operations defined, we now have the interface between DS and RDF spelled out, allowing DS trees (see Fig. 2) to be decorated with RDF representations, and for incremental semantics to be derived in RDF. This allows us in turn to integrate RDF easily with DyLan (Eshghi et al., 2011; Eshghi, 2015), the existing DS parser implementation. DS-RDF can therefore progressively enrich an RDF semantic graph as a sentence or utterance unfolds in time. We have illustrated this in Fig. 1, showing a parse of "Jane drinks water quickly", but abstracting away from the underlying DS machinery for simplicity, only showing the RDF semantic graphs at each step.

5 Next Steps

We have formally defined the operations required to create DS-RDF: an incremental graph-based semantic parser that can be integrated with an incremental reasoning engine (RDFox). We are currently implementing DS-RDF and have example parses working with RDFox. Our next step is to bootstrap a wide-coverage lexicon from existing resources and evaluate it on meaning banks (e.g. the Groningen meaning bank (Bos et al., 2017)). We also plan to edit the output structure to align with the WikibaseLexemes model, enabling seamless integration with Wikifunctions in the future. Finally, we plan to evaluate DS-RDF using domain specific dialogues and a corpus that we are currently collecting with people that have dementia.

[3]We follow Eshghi et al. (2013) in assuming a *semantic head* node in all RDF graphs that corresponds to the DS node type: this is different from the notion of a syntactic head used in other grammar frameworks

[4]To illustrate our RDF graphs, we are representing a few common triples with each node. For example, the 'Jane' node in Fig. 1 illustrates three triples: (1) this node (with a unique identifier) is labelled "Jane", (2) this node is an instance of the "schema:Person" class, and (3) this node is currently the *head node* during a parse (denoted by the purple "H").

Acknowledgments

The first author is funded by Wallscope and The Data Lab. We thank them for their support. We would also like to thank Oxford Semantic Technologies for providing us with full access to RDFox throughout this project, and as we continue to develop DS-RDF.

References

Serge Abiteboul, Richard Hull, and Victor Vianu. 1995. *Foundations of databases*, volume 8. Addison-Wesley Reading.

Angus Addlesee, Yanchao Yu, and Arash Eshghi. 2020. A comprehensive evaluation of incremental speech recognition and diarization for conversational ai. In *Proceedings of the 28th International Conference on Computational Linguistics*, pages 3492–3503.

Sharon Ash, Peachie Moore, S Antani, G McCawley, Melissa Work, and Murray Grossman. 2006. Trying to tell a tale: Discourse impairments in progressive aphasia and frontotemporal dementia. *Neurology*, 66(9):1405–1413.

Sören Auer, Christian Bizer, Georgi Kobilarov, Jens Lehmann, Richard Cyganiak, and Zachary Ives. 2007. Dbpedia: A nucleus for a web of open data. In *The semantic web*, pages 722–735. Springer.

Janet Beavin Bavelas and Jennifer Gerwing. 2011. The listener as addressee in face-to-face dialogue. *International Journal of Listening*, 25(3):178–198.

Johan Bos, Valerio Basile, Kilian Evang, Noortje Venhuizen, and Johannes Bjerva. 2017. The groningen meaning bank. In Nancy Ide and James Pustejovsky, editors, *Handbook of Linguistic Annotation*, volume 2, pages 463–496. Springer.

Veronica Boschi, Eleonora Catricala, Monica Consonni, Cristiano Chesi, Andrea Moro, and Stefano F Cappa. 2017. Connected speech in neurodegenerative language disorders: a review. *Frontiers in psychology*, 8:269.

E. Breitholtz. 2020. *Enthymemes and Topoi in Dialogue: The Use of Common Sense Reasoning in Conversation*. Current Research in the Semantics / Pragmatics Interface. Brill.

Ellen Breitholtz and Christine Howes. 2020. Communicable reasons: How children learn topoi through dialogue. In *Proceedings of the 24th Workshop on the Semantics and Pragmatics of Dialogue*, Waltham, MA. SEMDIAL.

Ronnie Cann, Ruth Kempson, and Lutz Marten. 2005. *The Dynamics of Language*. Elsevier, Oxford.

Christian Chiarcos, Philipp Cimiano, Thierry Declerck, and John Philip McCrae. 2013. Linguistic linked open data (llod). introduction and overview. In *Proceedings of the 2nd Workshop on Linked Data in Linguistics (LDL-2013): Representing and linking lexicons, terminologies and other language data*, pages i–xi.

Stephen Clark. 2015. Vector space models of lexical meaning. In Shalom Lappin and Chris Fox, editors, *Handbook of Contemporary Semantics*, 2nd edition. Wiley-Blackwell.

Robin Cooper. 2005. Records and record types in semantic theory. *Journal of Logic and Computation*, 15(2):99–112.

Robin Cooper and Jonathan Ginzburg. 2015. Type theory with records for natural language semantics. *The Handbook of Contemporary Semantic Theory*, pages 375–407.

Mika Enomoto, Yasuharu Den, and Yuichi Ishimoto. 2020. A conversation-analytic annotation of turn-taking behavior in japanese multi-party conversation and its preliminary analysis. In *Proceedings of The 12th Language Resources and Evaluation Conference*, pages 644–652.

A. Eshghi, M. Purver, and Julian Hough. 2011. Dylan: Parser for dynamic syntax. Technical report, Queen Mary University of London.

Arash Eshghi. 2015. DS-TTR: An incremental, semantic, contextual parser for dialogue. In *Proceedings of Semdial 2015 (goDial), the 19th workshop on the semantics and pragmatics of dialogue*.

Arash Eshghi, Julian Hough, and Matthew Purver. 2013. Incremental grammar induction from child-directed dialogue utterances. In *Proceedings of the 4th Annual Workshop on Cognitive Modeling and Computational Linguistics (CMCL)*, pages 94–103, Sofia, Bulgaria. Association for Computational Linguistics.

Arash Eshghi, Julian Hough, Matthew Purver, Ruth Kempson, and Eleni Gregoromichelaki. 2012. Conversational interactions: Capturing dialogue dynamics. In S. Larsson and L. Borin, editors, *From Quantification to Conversation: Festschrift for Robin Cooper on the occasion of his 65th birthday*, volume 19 of *Tributes*, pages 325–349. College Publications, London.

Arash Eshghi, Igor Shalyminov, and Oliver Lemon. 2017. Bootstrapping incremental dialogue systems from minimal data: the generalisation power of dialogue grammars. In *Proceedings of the 2017 Conference on Empirical Methods in Natural Language Processing (EMNLP)*.

Fernanda Ferreira, Ellen F. Lau, and Karl G. D. Bailey. 2004. Disfluencies, language comprehension, and tree adjoining grammars. *Cognitive Science*, 28(5):721ñ749.

Charles. Goodwin. 1981. *Conversational organization: interaction between speakers and hearers*. Academic Press.

P. G. T. Healey, Arash Eshghi, Christine Howes, and Matthew Purver. 2011. Making a contribution: Processing clarification requests in dialogue. In *Proceedings of the 21st Annual Meeting of the Society for Text and Discourse*, Poitiers.

Julian Hough. 2015. *Modelling Incremental Self-Repair Processing in Dialogue*. Ph.D. thesis, Queen Mary University of London.

Julian Hough and Matthew Purver. 2012. Processing self-repairs in an incremental type-theoretic dialogue system. In *Proceedings of the 16th SemDial Workshop on the Semantics and Pragmatics of Dialogue (SeineDial)*, pages 136–144, Paris, France.

Julian Hough and Matthew Purver. 2014. Probabilistic type theory for incremental dialogue processing. In *Proceedings of the EACL 2014 Workshop on Type Theory and Natural Language Semantics (TTNLS)*, pages 80–88, Gothenburg, Sweden. Association for Computational Linguistics.

Christine Howes and Arash Eshghi. 2017. Feedback relevance spaces: The organisation of increments in conversation. In *IWCS 201712th International Conference on Computational SemanticsShort papers*.

Christine Howes, Patrick G. T. Healey, Matthew Purver, and Arash Eshghi. 2012. Finishing each other's ... responding to incomplete contributions in dialogue. In *Proceedings of the 34th Annual Meeting of the Cognitive Science Society (CogSci 2012)*, pages 479–484, Sapporo, Japan.

Christine Howes, Matthew Purver, Patrick G. T. Healey, Gregory J. Mills, and Eleni Gregoromichelaki. 2011. On incrementality in dialogue: Evidence from compound contributions. *Dialogue and Discourse*, 2(1):279–311.

Ruth Kempson, Ronnie Cann, Arash Eshghi, Eleni Gregoromichelaki, and Matthew Purver. 2015. Ellipsis. In Shalom Lappin and Chris Fox, editors, *The Handbook of Contemporary Semantics*. Wiley-Blackwell.

Ruth Kempson, Ronnie Cann, Eleni Gregoromichelaki, and Stergios Chatzikiriakidis. 2016. Language as mechanisms for interaction. *Theoretical Linguistics*, 42(3-4):203–275.

Ruth Kempson, Wilfried Meyer-Viol, and Dov Gabbay. 2001. *Dynamic Syntax: The Flow of Language Understanding*. Wiley-Blackwell.

Ora Lassila, Ralph R. Swick, World Wide, and Web Consortium. 1998. Resource description framework (rdf) model and syntax specification.

John McCrae, Dennis Spohr, and Philipp Cimiano. 2011. Linking lexical resources and ontologies on the semantic web with lemon. In *Extended Semantic Web Conference*, pages 245–259. Springer.

Peter Mika. 2015. On schema. org and why it matters for the web. *IEEE Internet Computing*, 19(4):52–55.

Boris Motik, Yavor Nenov, Robert Piro, and Ian Horrocks. 2015. Incremental update of datalog materialisation: the backward/forward algorithm. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 29.

Yavor Nenov, Robert Piro, Boris Motik, Ian Horrocks, Zhe Wu, and Jay Banerjee. 2015. Rdfox: A highly-scalable rdf store. In *International Semantic Web Conference*, pages 3–20. Springer.

Finn Nielsen. 2020. Lexemes in wikidata: 2020 status. In *Proceedings of the 7th Workshop on Linked Data in Linguistics (LDL-2020)*, pages 82–86.

Matthew Purver, Arash Eshghi, and Julian Hough. 2011. Incremental semantic construction in a dialogue system. In *Proceedings of the 9th International Conference on Computational Semantics*, pages 365–369, Oxford, UK.

Matthew Purver, Eleni Gregoromichelaki, Wilfried Meyer-Viol, and Ronnie Cann. 2010. Splitting the 'I's and crossing the 'You's: Context, speech acts and grammar. In *Aspects of Semantics and Pragmatics of Dialogue. SemDial 2010, 14th Workshop on the Semantics and Pragmatics of Dialogue*, pages 43–50, Poznań. Polish Society for Cognitive Science.

Matthew Purver, Christine Howes, Eleni Gregoromichelaki, and Patrick G. T. Healey. 2009. Split utterances in dialogue: A corpus study. In *Proceedings of the 10th Annual SIGDIAL Meeting on Discourse and Dialogue (SIGDIAL 2009 Conference)*, pages 262–271, London, UK. Association for Computational Linguistics.

Matthew Purver, Mehrnoosh Sadrzadeh, Gijs Wijnholds, Ruth Kempson, and Julian Hough. 2021. Incremental Composition in Distributional Semantics. *Journal of Logic, Language and Information*.

Mehrnoosh Sadrzadeh, Matthew Purver, Julian Hough, and Ruth Kempson. 2018. Exploring semantic incrementality with dynamic syntax and vector space semantics. In *Proceedings of the 22nd Workshop on the Semantics and Pragmatics of Dialogue (SemDial 2018 - AixDial)*, Aix-en-Provence.

Oriol Vinyals and Quoc Le. 2015. A neural conversational model. In *Proceedings of ICML Deep Learning Workshop 2015*.

Denny Vrandečić. 2021. Building a multilingual wikipedia. *Communications of the ACM*, 64(4):38–41.

Denny Vrandečić and Markus Krötzsch. 2014. Wikidata: a free collaborative knowledgebase. *Communications of the ACM*, 57(10):78–85.

Lei Zhang, Michael Färber, and Achim Rettinger. 2014. xlid-lexica: Cross-lingual linked data lexica. In *LREC*, pages 2101–2105.