Functional Distributional Semantics: Learning Linguistically Informed Representations from a Precisely Annotated Corpus

Guy Emerson

Artificial intelligence has advanced to the point that the best AI systems can now beat the best human players at the game of Go, a game which takes years for a person to master. On the other hand, I am sure we have all seen examples of how machine translation can go wrong. In an extreme case in October 2017, a person was mistakenly arrested because Facebook incorrectly translated an Arabic phrase meaning "good morning" to a Hebrew phrase meaning "attack them". What makes language so difficult to model computationally?

In recent years, the field of computational linguistics has become dominated by *neural network* models. Such models are good at solving tasks that can be defined in narrow terms, but much worse at tasks requiring problem solving, reasoning, or dealing with unexpected input – in the mistranslation mentioned above, an unusual phrase for "good morning" was used, which the system had presumably never seen before.

Fundamentally, neural network models represent information as vectors – that is, as lists of numbers. Vectors are computationally convenient, enabling efficient algorithms, but they are not naturally suited for representing *semantics*. The aim of my dissertation is to develop a semantic framework which is both compatible with formal linguistic theory, and also empirically testable using real-world data. My motivations are twofold: to shed light on what it means to know a language, and to push forward the limits of machine learning and artificial intelligence. This work is necessarily inter-disciplinary, requiring concepts from linguistics, philosophy of language, mathematics, and computer science. The core ideas have been published in a series of papers (Emerson and Copestake, 2016, 2017a, 2017b).

The Dissertation in a Nutshell

In my dissertation, I focus on *distributional semantics*, which has the goal of learning the meanings of words from a *corpus* (a body of text). This plays a central role in modern computational linguistics. The core idea is that the contexts in which a word appears give us information about its meaning. For example, from the contexts shown in Figure 1, we might learn that horses are animals, and are involved in racing and agriculture. Learning such information automatically is the goal of distributional semantics.

… being hurt by another	horse	especially if some rider …
… beaten by a better	horse	at the distance on …
… from these studies that	horses	reared with other horses …
… horses reared with other	horses	in a free and …
… 'Is that all your	horse	gets to eat?' in …
… cache of cattle and	horse	bones, while from the …
… was a sterling good	horse,	especially at Ascot, but …
… way as a domestic	horse	that it is stabled …
… 1790 – that is, one	horse	or two cows for …
… as coarse as a	horse	's tail straying from …

Figure 1: Ten instances of "horse" in the British National Corpus.

The twin challenges are: how do we represent meaning, and how do we learn these representations? The current state of the art is to represent meanings as vectors (Turney and Pantel, 2010; Mikolov et al., 2013) – but vectors do not correspond to any traditional notion of meaning. In particular, there is no way to talk about *truth*, a crucial concept in logic and formal semantics.

I have developed a framework for distributional semantics that answers this challenge. The meanings of words are not represented as vectors, but as mathematical *functions*, which map from entities (intuitively, objects in the world), to probabilities of truth. For example, the function for "cup" would return high probabilities for typical cups, middling values for entities near the boundary of the concept (such as mugs, glasses, and bowls), and low values for other entities. Such a function can be interpreted both in the machine learning sense of a *classifier*, and in the formal semantic sense of a *truth-conditional function*. This simultaneously allows both the use of machine learning techniques to exploit large datasets, and also the use of formal semantic techniques to manipulate the learnt representations. I have empirically demonstrated that this model can improve performance compared to vector space models. Furthermore, because the framework is linguistically interpretable, there is a clear and plausible path from my work to general-purpose linguistic representation and reasoning.

Semantic Functions

I represent the meaning of a word as a function from entities to probabilities of truth, which I call a *semantic function*. The use of probabilities has experimental support (Labov, 1973; Murphy, 2002), and connects with recent linguistic and philosophical work on vague predicates (Sutton, 2017; Lassiter and Goodman, 2015). For example, the word "expensive" is vague because there is no fixed price above which something should be considered expensive. Probabilities provide a way to characterise this uncertainty.

An illustration of a semantic function is given in Figure 2, for the word "pepper". Peppers come in many colours, most typically green, yellow, or red. The semantic function for "pepper" is given by the solid bars – it is high for all the peppers, but lower for the unusual colours, which a person might be hesitant to call a pepper. The function takes the value 0 for the carrot and the cucumber, which are definitely not peppers. Because this function produces probabilities, we can use it for *Bayesian inference*, a method for logical reasoning under uncertain knowledge. For example, suppose I have one of the vegetables in Figure 2, but you don't know which. Your uncertainty could be represented by the black shaded bars – blue peppers do not exist, so there is no probability of that, while purple peppers are rare, so there is a low probability of that. If I now tell you that what I have is a pepper, you could update your beliefs to the orange bars – the carrot and cucumber are ruled out, while the peppers are more likely. In Bayesian terminology, this is an update from a *prior* probability distribution, to a *posterior* probability distribution.

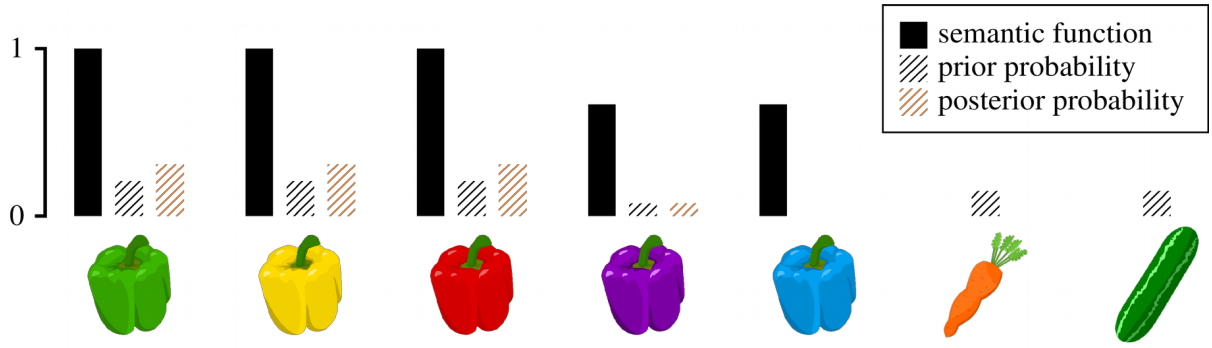

Figure 2: An illustration of a *semantic function* and its use in *Bayesian inference*. See text for details.

Probabilistic Model Theory

A standard approach to formal semantics is *model theory* (Cann, 1993; Allan, 2011; Kamp and Reyle, 2013). In model theory, we have a set of *entities*, and *predicates* which are true or false of each entity. Meanings of sentences are represented using logical formulae – given the entities in the model, we can work out whether a particular formula is true or false. In my dissertation, I develop a probabilistic version of model theory, which uses semantic functions.

From the machine learning perspective, probabilities make model theory easier to work with, as we can smoothly change between different entities, and between different truth values – for example, in the simple model in Figure 2, we can make any of the bars larger or smaller. Furthermore, the connection with formal semantics provides natural operations for *logical entailment* (determining if one sentence implies another) and *semantic composition* (determining the meaning of a phrase from the meanings of its parts). These are decisive advantages over vector space models.

From the formal semantic perspective, probabilities provide a new analytical tool. One problem much discussed in the literature is *context dependence*. For example, the word "cut" can refer to both cutting grass and cutting cake, despite the different tools and different physical actions – the interpretation of "cut" depends on the context (Searle, 1980). In my dissertation, I provide an account of context dependence that overcomes difficulties pointed out in the literature (Recanati, 2012). The meaning of a predicate is represented by a function, but the entity the predicate *refers* to (a particular event of cutting grass or cutting cake) depends on the context, in a way that can be formalised using Bayesian inference. This is a new mechanism for studying context dependence, which can leverage both formal semantic theory and machine learning techniques.

Graphical Models: A Link Between Formal Semantics and Machine Learning

Logical formulae are inconvenient for many machine learning algorithms. However, Copestake (2009) showed that logical formulae can be represented as *semantic dependency graphs*, as illustrated in Figures 3 and 4. A dependency graph consists of *nodes* (typically corresponding to words), and *links* between the nodes (specifying how the nodes are related).

The benefit of using graphs is that they are more convenient for machine learning models – in particular, for *probabilistic graphical models*. I have developed a probabilistic graphical model, which incorporates the probabilistic version of model theory described above, and which can generate semantic dependency graphs. The aim is to get the best of both formal semantics and machine learning: the overall structure comes from semantic theory (in the form of semantic dependency graphs), but the details are left for machine learning to fill in based on observed data (such as what kinds of entities tend to occur together, and whether one predicate implies another).

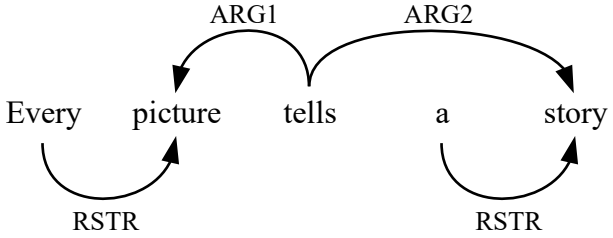

Figure 3: A semantic dependency graph for the sentence "Every picture tells a story"

$$\forall x\, \exists y\, \exists z\, \text{picture}(x) \rightarrow \text{story}(z) \wedge \text{tell}(y) \wedge \text{ARG}1(y, x) \wedge \text{ARG}2(y, z)$$

Figure 4: A logical formula for the sentence "Every picture tells a story"

Experimental Results

My framework is more expressive than vector space models, but this comes at a cost. The core problem is that when training a model on text, we do not observe the entities themselves, but only their textual descriptions. In machine learning terminology, the entities are *latent variables*. In principle, we can apply Bayesian inference to work out which entities are likely for a particular sentence. However, an exact calculation requires considering every possible combination of entities and checking how they match the sentence. This is intractable, because there are too many entities, and too many combinations of them. To use the framework in practice, we need simpler calculations that approximate Bayesian inference.

I have adapted two *approximate inference* techniques for my framework. The first is a *Markov Chain Monte Carlo* method, which considers a small set of likely entities, rather than all possible entities. Given enough time for the calculations, this technique is guaranteed to give a good approximation. The second is a *Variational Inference* method, in which I make simplifying assumptions, and find the exact solution under these assumptions. This second technique does not have the same theoretical guarantees as the first technique, but it can be calculated much faster.

Armed with these approximate inference techniques, I have verified my framework works in practice, by training a model on the English Wikipedia, and testing it on several evaluation datasets. The results have shown that it can learn information not captured by vector space models. The most exciting result is on the RELPRON dataset (Rimell et al., 2017), which tests how well a model can understand the meaning of phrases, rather than individual words. Rimell et al. concluded that they may have reached the limits of performance with vector space models. However, using my semantic function model, I have achieved better results. In particular, I improved performance on the "confounders" that Rimell et al. included – these are particularly challenging phrases which tripped up every vector space model they tested.

The Bigger Picture

Read narrowly, my dissertation introduces a linguistically interpretable and computationally tractable framework for learning the meanings of words from text. However, the dissertation also represents the basis of a larger research project. I plan to extend my framework to further levels of linguistic structure – this is necessary, because sentences do not exist in isolation, but must be understood in their full context, including both other sentences and the outside world.

Many phrases cannot be understood by simply combining their parts. For example, a "magic carpet" is both magic and a carpet, but there is the connotation that it is a flying carpet – rather than a carpet that magically cleans itself. Such semi-compositional expressions are poorly understood both theoretically and computationally (Sag et al., 2002; Reddy et al., 2011; Vincze, 2012), but their ubiquity means they must be properly accounted for. I plan to extend my model to allow phrases to carry additional meaning not associated with any of their parts.

A traditional distinction in linguistics is that, while *semantics* deals with literal meanings, *pragmatics* deals with meanings in context – for example, "some of the babies were smiling" might be taken to mean "some *but not all* of the babies were smiling". Current work in computational pragmatics assumes a hand-written semantic model for a small domain, and investigates how to automatically produce pragmatic inferences (Frank and Goodman, 2012). My framework is compatible with these pragmatic models, and could provide a semantic model across a large domain, allowing us to test these pragmatic models when scaled up to a realistic size.

Finally, all distributional semantic models come up against the *symbol grounding problem* – if meanings of words are defined in terms of other words, the definitions are circular (Harnad, 1990). Indeed, people do not learn language from text or speech alone, but also connect words with their sensory perception. With its connection to both machine learning and formal semantics, my framework provides a basis for exploring this problem, as state-of-the-art image processing techniques could be directly incorporated into the semantic functions.

A good model of language must be able to represent semantic structure, must be sensitive to the larger context, and must be able to learn from disparate data sources (including both text and grounded information such as images). Producing such a model would constitute a major step forward in computational linguistics and artificial intelligence.

References

Keith Allan. *Natural language semantics*. Blackwell Publishers, 2001.

Gemma Boleda and Aurélie Herbelot. Formal distributional semantics: Introduction to the special issue. *Computational Linguistics*, 42(4):619–635, 2017.

Ronnie Cann. *Formal semantics: an introduction*. Cambridge University Press, 1993.

Ann Copestake. Slacker semantics: Why superficiality, dependency and avoidance of commitment can be the right way to go. In *Proceedings of the 12th Conference of the European Chapter of the Association for Computational Linguistics*, pages 1–9, 2009.

Guy Emerson and Ann Copestake. Functional Distributional Semantics. In *Proceedings of the 1st Workshop on Representation Learning for Natural Language Processing*, pages 40–52, 2016.

Guy Emerson and Ann Copestake. Variational Inference for Logical Inference. In *Proceedings of the 2017 Conference on Logic and Machine Learning for Natural Language (LaML)*, 2017a.

Guy Emerson and Ann Copestake. Semantic Composition via Probabilistic Model Theory. In *Proceedings of the 12th International Conference on Computational Semantics (IWCS)*, 2017b.

Michael C Frank and Noah D Goodman. Predicting pragmatic reasoning in language games. *Science*, 336(6084):998–998, 2012.

Stevan Harnad. The symbol grounding problem. *Physica D: Nonlinear Phenomena*, 42(1-3): 335–346, 1990.

Hans Kamp and Uwe Reyle. *From discourse to logic: Introduction to modeltheoretic semantics of natural language, formal logic and discourse representation theory*, volume 42 of *Studies in Linguistics and Philosophy*. Springer Science & Business Media, 2013.

William Labov. The boundaries of words and their meanings. In Charles-James Bailey and Roger W. Shuy, editors, *New Ways of Analyzing Variation in English*, pages 340–371. Georgetown University Press, 1973.

Daniel Lassiter and Noah D Goodman. Adjectival vagueness in a Bayesian model of interpretation. *Synthese*, pages 1–36, 2015.

Tomas Mikolov, Kai Chen, Greg Corrado, and Jeffrey Dean. Efficient estimation of word representations in vector space. In *Proceedings of the 1st International Conference on Learning Representations*, 2013.

Gregory Murphy. *The big book of concepts*. MIT press, 2002.

François Recanati. Compositionality, flexibility, and context-dependence. In Wolfram Hinzen, Edouard Machery, and Markus Werning, editors, *Oxford Handbook of Compositionality*, chapter 8, pages 175–191. 2012.

Siva Reddy, Diana McCarthy, and Suresh Manandhar. An empirical study on compositionality in compound nouns. In *Proceedings of the 5th International Joint Conference on Natural Language Processing (IJCNLP)*, pages 210–218, 2011.

Laura Rimell, Jean Maillard, Tamara Polajnar, and Stephen Clark. RELPRON: A relative clause evaluation dataset for compositional distributional semantics. *Computational Linguistics*, 42 (4):661–701, 2016.

Ivan Sag, Timothy Baldwin, Francis Bond, Ann Copestake, and Dan Flickinger. Multiword expressions: A pain in the neck for NLP. In *Proceedings of the 3rd International Conference on Computational Linguistics and Intelligent Text Processing (CICLing)*, pages 1–15. Springer, 2002.

John R. Searle. The background of meaning. In John R. Searle, Ferenc Kiefer, and Manfred Bierwisch, editors, *Speech act theory and pragmatics*, pages 221–232. Reidel, 1980.

Peter R. Sutton. Probabilistic approaches to vagueness and semantic competency. *Erkenntnis*, 2017.

Peter D. Turney and Patrick Pantel. From frequency to meaning: Vector space models of semantics. *Journal of Artificial Intelligence Research*, 37:141–188, 2010.

Veronika Vincze. *Semi-compositional noun + verb constructions: Theoretical questions and computational linguistic analyses*. PhD thesis, University of Szeged, 2012.